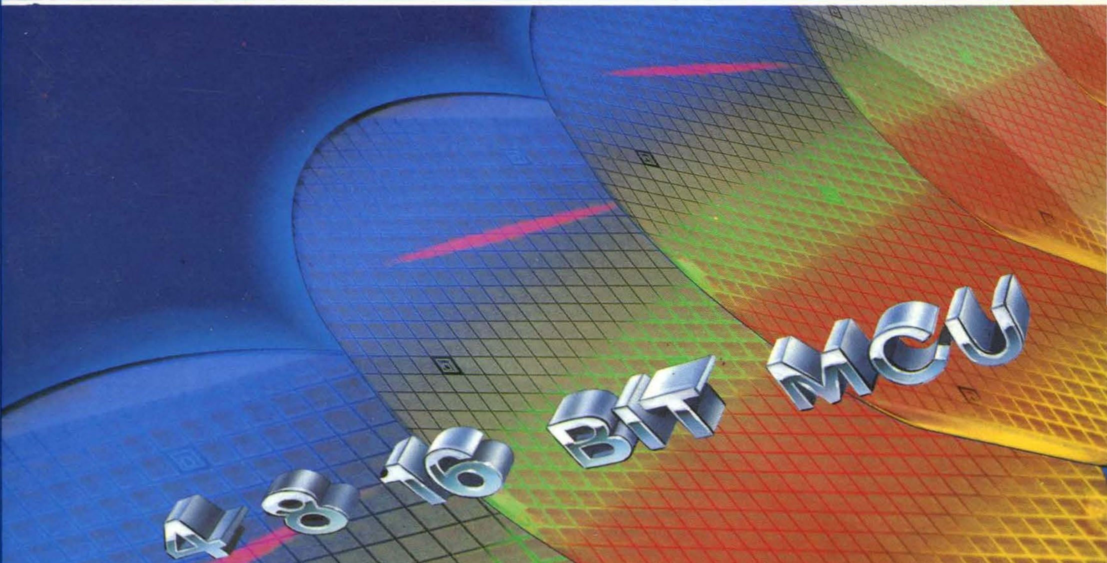
MICROCOMPUTERS

1987

MICROCOMPUTERS
DATA BOOK



THOMSON
COMPONENTS
SEMICONDUCTEURS

MICROCOMPUTERS

WARNING !

THE FOLLOWING PRODUCTS HAVE BEEN DISCONTINUED

- TS 94104 1-139
- TS 94120 TS 94144 .. 1-153
- MK 68200 FAMILY 3-5
- MK 68HC200 FAMILY . 3-77

CONTENTS

Page

GENERAL INFORMATION

CHAPTER 1 - 4 BIT MICROCOMPUTERS

CHAPTER 2 - 8-BIT MICROCOMPUTERS

EF6801 FAMILY

EF6804 FAMILY

EF6805 FAMILY

MK3870 FAMILY

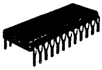
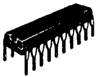
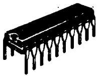
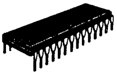
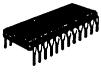
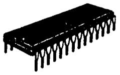
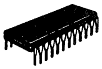
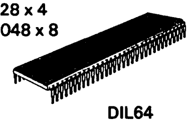
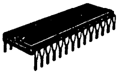
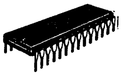
CHAPTER 3 - 16-BIT MICROCOMPUTERS

CONTENTS

ALPHANUMERICAL INDEX

4-bit microcomputers

4 BIT MICROCOMPUTERS

Function	Part number	Characteristic		Package	Page
LOW COST MCUs	ETC9410/9310 ETL9410/9310	With their efficient instruction set (1 byte instruction) and wide I/O option list, these MCUs are designed for home appliances and consumer applications. ● One 4-bit SIO ● Emulation and development on DEVICE	● RAM: 32 x 4 ● ROM: 512 x 8 ● 19 I/O lines	 DIL24	1-5 1-21
	ETC9411/9311 ETL9411/9311		● RAM: 32 x 4 ● ROM: 512 x 8 ● 16 I/O lines	 DIL20	1-5 1-21
	ETL9413 ETL9313		● RAM: 32 x 4 ● ROM: 512 x 8 ● 15 I/O lines	 DIL20	1-41
	ET9420/9320 ETC9420/9320 ETL9420/9320		● RAM: 64 x 4 ● ROM: 1024 x 8 ● 23 I/O lines	 DIL28	1-45 1-69 1-91
	ET9421/9321 ETC9421/9321 ETL9421/9321		● RAM: 64 x 4 ● ROM: 1024 x 8 ● 19 I/O lines	 DIL24	1-45 1-69 1-91
	ET9422/9322 ETC9422/9322 ETL9422/9322		● RAM: 64 x 4 ● ROM: 1024 x 8 ● 15 I/O lines	DIL20	1-45 1-69 1-91
	ETC9444/9344 ETL9444/9344		● RAM: 128 x 4 ● ROM: 2048 x 8 ● 23 I/O lines	 DIL28	1-69 1-115
	ETC9445/9345 ETL9445/9345		● RAM: 128 x 4 ● ROM: 2048 x 8 ● 23 I/O lines	 DIL24	1-69 1-115
MOTOR SPEED CONTROL LOW COST MCUs	TS94104	Well suited for speed measurement, direct triac drive in industrial environment (watchdog feature). Individual clock for MCU, speed measurement timer and triac drive timer. ● One 4-bit SIO ● Emulation and development on DEVICE	● RAM: 128 x 4 ● ROM: 2048 x 8	 DIL64	1-139
	TS94120		● RAM: 64 x 4 ● ROM: 1024 x 8 ● 23 I/O lines	 DIL28	1-153
	TS94144		● RAM: 128 x 4 ● ROM: 2048 x 8 ● 23 I/O lines	 DIL28	1-153

THOMSON COMPONENTS

SEMICONDUCTEURS

ETC9410 • ETC9411 • ETC9310 • ETC9311

SINGLE CHIP MICROCONTROLLERS

The ETC 9410, C 9411, C 9310, and C 9311, fully Static, Single-Chip CMOS Microcontrollers are fully compatible with the COPS® family, fabricated using double-poly, silicon gate CMOS technology. These Controller Oriented Processors are complete microcomputers containing all system timing, internal logic, ROM, RAM, and I/O necessary to implement dedicated control functions in a variety of applications. Features include single supply operation, a variety of output configuration options, with an instruction set, internal architecture and I/O scheme designed to facilitate keyboard input, display output and BCD data manipulation. The ETC 9411 is identical to the ETC 9410 but with 16 I/O lines instead of 20. They are an appropriate choice for use in numerous human interface control environments. Standard test procedures and reliable high-density fabrication techniques provide the medium to large volume customers with a customized Controller Oriented Processor at a low end-product cost.

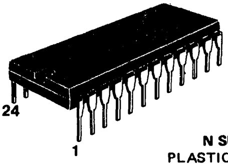
The ETC 9310/C9311 is the extended temperature range version of the ETC 9410/C9411.

- Lowest Power Dissipation (40μW typical)
- Low Cost
- Power saving HALT mode with Continue function
- Powerful Instruction Set
- 512 × 8 ROM, 32 × 4 RAM
- 20 I/O lines (ETC 9410)
- Two-level subroutine stack
- DC to 4 μs instruction time
- Single supply operation (2.4V to 5.5V)
- General purpose and TRI-STATE® outputs
- Internal binary/counter register with MICROWIRE® compatible serial I/O
- LSTTL/CMOS compatible in and out
- Software/hardware compatible with other members of the ET 9400 family
- Extended temperature (– 40°C to + 85°C).

CMOS

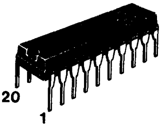
CASES

ETC9410 • ETC9310



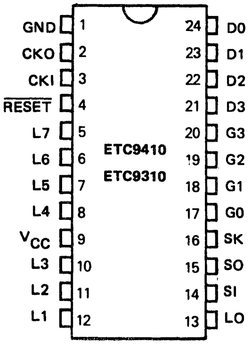
N SUFFIX
PLASTIC PACKAGE

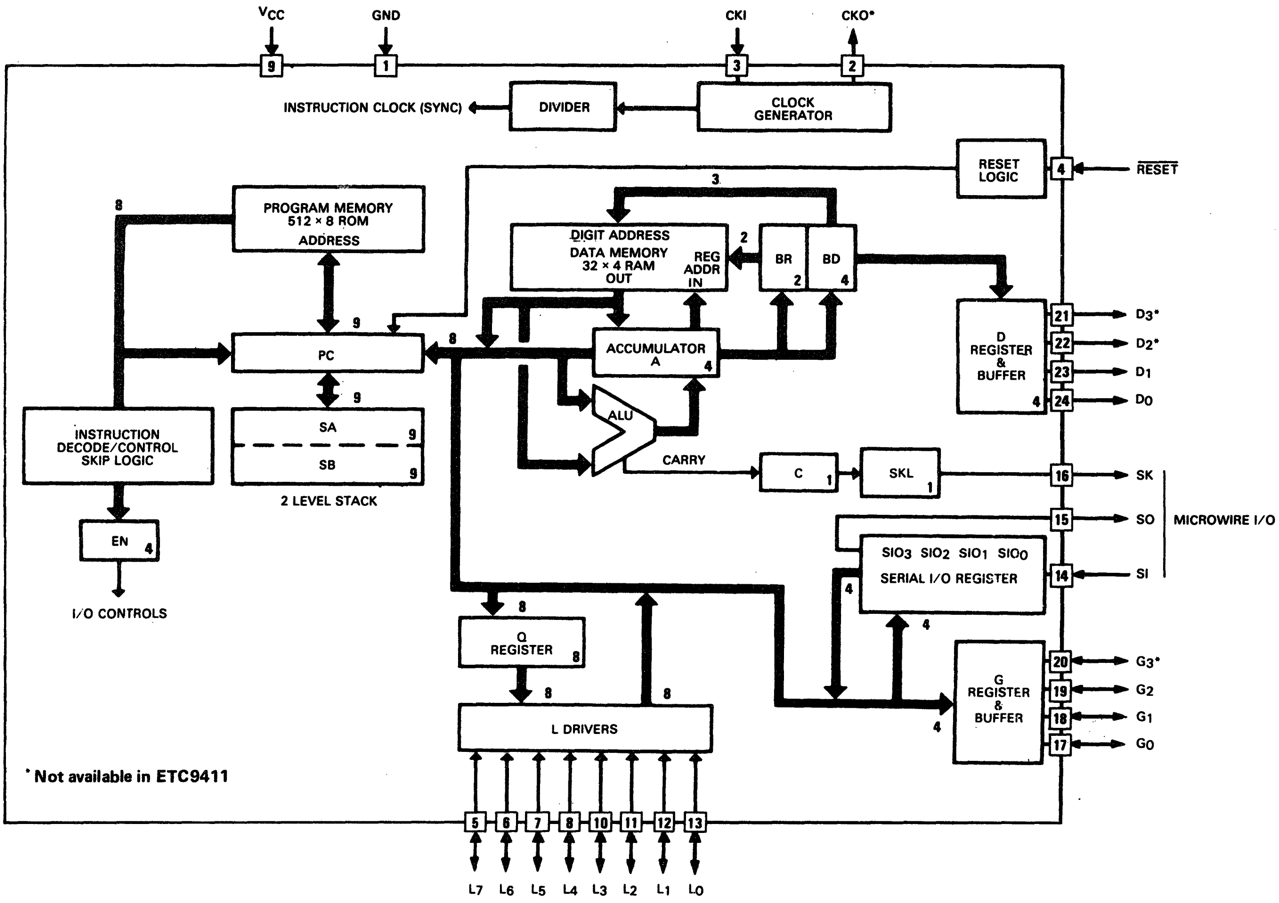
ETC9411 • ETC9311



N SUFFIX
PLASTIC PACKAGE

PIN ASSIGNMENT





* Not available in ETC9411

FIGURE 1 – BLOCK DIAGRAM (24-pin version)

ETC 9410 • ETC 9411

ABSOLUTE MAXIMUM RATINGS

Supply Voltage (V_{CC})	6V
Voltage at Any Pin Relative to GND	– 0.3V to V_{CC} + 0.3 V
Ambient Operating Temperature	0°C to + 70° C
Ambient Storage Temperature	– 65°C to + 150°C
Lead Temperature (Soldering, 10 seconds)	300° C
Total Source Current	25 mA
Total Sink Current	25 mA

Absolute maximum ratings indicate limits beyond which damage to the device may occur. DC and AC electrical specifications are not ensured when operating the device at absolute maximum ratings.

DC ELECTRICAL CHARACTERISTICS 0°C<T_A<+70°C (Unless otherwise specified)

Parameter	Conditions	Min.	Max.	Units
Operating Voltage		2.4	5.5	V
Power Supply Ripple (Note 5)	Peak to peak	–	0.1V_{CC}	
Supply Current (Note 1)	V_{CC} = 2.4V tc = 125 μs	-	40	μA
	V_{CC} = 5.0V tc = 16 μs	-	500	μA
	V_{CC} = 5.0V tc = 4 μs	-	2000	μA
	(tc is the instruction cycle time)			
HALT Mode Current (Note 2)	V_{CC} = 5.0V Fin = 0kHz	–	15	μA
	V_{CC} = 2.4V Fin = 0kHz	–	6	μA
Input Voltage Levels				
RESET, CKI				
Logic High		0.9V_{CC}	–	V
Logic Low		-	0.1V_{CC}	V
All other inputs				
Logic High		0.7V_{CC}	–	V
Logic Low		-	0.2V_{CC}	V
Hi – Z input leakage		– 1	+ 1	μA
Input capacitance		-	+ 7	pF
Output Voltage Levels	Standard outputs			
LSTTL Operation	V_{CC} = 5.0V ± 5 %			
Logic High	IOH =–25 μA	2.7	–	V
Logic Low	IOL = 400 μA	–	0.4	V
CMOS Operation				
Logic High	I_{OH} = –10 μA	V_{CC}-0.2	–	V
Logic Low	I_{OL}= 10μA	-	0.2	V
Output current levels	(except CKO)			
Sink (Note 6)	V_{CC} = 4.5V, Vout = V_{CC}	1.2	-	mA
	V_{CC} = 2.4V, Vout = V_{CC}	0.2	-	mA
Source (standard option)	V_{CC} = 4.5V, Vout = 0V	0.5	-	mA
	V_{CC} = 2.4V, Vout = 0V	0.1	-	mA
Source (low current option)	V_{CC} = 4.5V, Vout = 0V	30	330	μA
	V_{CC} = 2.4V, Vout = 0V	6	80	μA
CKO (as clock out) current levels				
Sink	V_{CC} = 4.5V, CKI = V_{CC}, Vout = V_{CC}			
divide by 4		0.3	–	mA
divide by 8		0.6	–	mA
divide by 16		1.2	–	mA
Source	V_{CC} = 4.5V, CKI = 0, Vout = 0			
divide by 4		0.3	-	mA
divide by 8		0.6	–	mA
divide by 16		1.2	–	mA

ETC 9410 • ETC 9411

DC ELECTRICAL CHARACTERISTICS (continued)

Parameter	Conditions	Min.	Max.	Units
Allowable Loading on CKO (as HALT)		–	100	pF
Current needed to over-ride HALT	(note 3)			
To continue	V_{CC} = 4.5V, ViN = .2V_{CC}	–	.6	mA
To halt	V_{CC} = 4.5V, ViN = .7V_{CC}	–	1.6	mA
TRI-STATE ® or open drain leakage current		−2	+ 2	µA

ETC 9410 • ETC 9411

AC ELECTRICAL CHARACTERISTICS (0°C<T_A<+70°C (Unless otherwise specified)

Parameter	Conditions	Min.	Max.	Units
Instruction Cycle	$V_{CC} \geqslant 4.5V$	4	DC	µs
Time = tc	$4.5V > V_{CC} \geqslant 2.4V$	16	DC	µs
Operating CKI	4 mode, $V_{CC} \geqslant 4.5V$	DC	1.0	MHz
Frequency	8 mode, $V_{CC} \geqslant 4.5V$	DC	2.0	MHz
	16 mode, $V_{CC} \geqslant 4.5V$	DC	4.0	MHz
	4 mode, $V_{CC} \geqslant 2.4V$	DC	250	kHz
	8 mode, $V_{CC} \geqslant 2.4V$	DC	500	kHz
	16 mode, $V_{CC} \geqslant 2.4V$	DC	1.0	MHz

Instruction Cycle Time - CKI (RC) **(Note 4)**	R = 30k ± 5 %, V_{CC} = 5V
	C = 82pF ± 5 % (÷ 4 Mode)
		8	16	µs

INPUTS : (See Fig. 3)				
t_{SETUP}	G Imputs, $V_{CC} \geqslant 4.5V$	tc/4 = + 0.7	–	µs
	SI Input, $V_{CC} \geqslant 4.5V$	0.3	–	µs
	All Others, $V_{CC} \geqslant 4.5V$	1.7	–	µs
t_{HOLD}	$V_{CC} \geqslant 4.5V$	0.25	–	µs
	$V_{CC} \geqslant 2.4V$	1.0	–	µs
OUTPUT				
Propagation delay (see fig. 3)	Vout = 1.5V, C_L = 100pF, R_L = 5K			
t_{PD1}, t_{PD0}	$V_{CC} \geqslant 4.5V$	–	1.0	µs
t_{PD1}, t_{PD0}	$V_{CC} \geqslant 2.4V$	–	4.0	µs

Note 1 : Supply current is measured after running for 2000 cycle times with a square -wave clock on CKI, CKO open, and all other pins pulled up to V_{CC} with 20k resistors.

Note 2 : The HALT mode will stop CKI from oscillating in the RC and crystal configurations.

Note 3 : When forcing HALT, current is only needed for a short time (approx. 200 ns) to flip the HALT flip-flop.

Note 4 : This parameter is only sampled and not 100 % tested.

Note 5 : Voltage change must be less than 0.5 volt in a 1 ms period.

Note 6 : SO output sink current must be limited to keep V_{OL} below 0.2 V_{CC} when port is running in order to prevent entering test mode.

ETC 9310 • ETC 9311

ABSOLUTE MAXIMUM RATINGS

Supply Voltage (V_{CC})	6V
Voltage at any pin	– 0.3V to V_{CC} + 0.3V
Total allowable source current	25mA
Total allowable sink current	25mA
Operating temperature range	– 40°C to + 85°C
Storage temperature range	– 65°C to + 150°C
Lead temperature (soldering, 10 sec.)	300°C

Absolute maximum ratings indicate limits beyond which damage to the device may occur. DC and AC electrical specifications are not ensured when operating the device at absolute maximum ratings.

DC ELECTRICAL CHARACTERISTICS – 40°C≤T_A≤+85°C (Unless otherwise specified).

Parameter	Conditions	Min.	Max.	Units
Operating Voltage		3.0	5.3	V
Power Supply Ripple(Note 5)	Peak to Peak	–	0.1V_{CC}	V
Supply Current (Note 1)	V_{CC} = 3.0V, t_c = 125µs V_{CC} = 5.0V, t_c = 16µs V_{CC} = 5.0V, t_c = 4µs (t_c is instruction cycle time)	– – –	60 600 2500	µA µA µA
HALT Mode Current (Note 2)	V_{CC} = 5.0V, F_{IN} = 0kHz V_{CC} = 3.0V, F_{IN} = 0kHz	– –	25 13	µA µA
Input Voltage Levels RESET, CKI Logic High InpuLogic Low All Other Inputs Logic High Logic Low		 0.9 V_{CC} – 0.7 V_{CC} –	 – 0.1 V_{CC} – 0.2 V_{CC}	 V V V V
Hi-Z Input Leakage		– 2	+ 2	µA
Input Capacitance		–	7	pF
Output Voltage Levels LSTTL Operation Logic High Logic Low CMOS Operation Logic High Logic Low	Standard Outputs V_{CC} = 5.0V ± 5 % I_{OH} = 25µA I_{OL} = 400µA I_{OH} = – 10µA I_{OL} = 10µA	 2.7 – V_{CC} – 0.2 –	 – 0.4 – 0.2	 V V V V
Output Current Levels Sink (Note 6) Source (Standard Option) Source (Low Current Option)	 V_{CC} = 4.5V, V_{OUT} = V_{CC} V_{CC} = 3.0V, V_{OUT} = V_{CC} V_{CC} = 4.5V, V_{OUT} = 0 V V_{CC} = 3.0V, V_{OUT} = 0 V V_{CC} = 4.5V, V_{OUT} = 0 V V_{CC} = 3.0V, V_{OUT} = 0 V	 1.2 0.2 0.5 0.1 30 8	 – – – – 440 200	 mA mA mA mA µA µA

ETC 9310 • ETC 9311

DC ELECTRICAL CHARACTERISTICS (continued)

Parameter	Conditions	Min.	Max.	Units
CKO Current Levels (As Clock Out)				
Sink ÷ 4	V_{CC} = 4.5V, CKI = V_{CC}, V_{OUT} = V_{CC}	0.3	–	mA
÷ 8		0.6	–	mA
÷16		1.2	–	mA
Source ÷ 4	V_{CC} = 4.5V, CKI = OV, V_{OUT} = OV	0.3	–	mA
÷ 8		0.6	–	mA
÷16		1.2	–	mA
Allowable Loading on CKO (as HALT I/O pin)		–	100	pF
Current Needed to Override HALT (Note 3)				
To Continue	V_{CC} = 4.5V, V_{IN} = 0.2 V_{CC}	–	0.8	mA
To Halt	V_{CC} = 4.5V, V_{IN} = 0.7 V_{CC}	–	2.0	mA
TRI-STATE or Open Drain Leakage Current		– 4	+ 4	µA

ETC 9310 • ETC 9311

AC ELECTRICAL CHARACTERISTICS —40°C$\leqslant T_A \leqslant$+85°C (Unless otherwise specified)

Parameter	Conditions	Min.	Max.	Units
Instruction Cycle Time (t_C)	$V_{CC} \leqslant 4.5V$	4	DC	µs
	$4.5V > V_{CC} \geqslant 3.0V$	16	DC	µs
Operating CKI ÷ 4 mode		DC	1.0	MHz
Frequency ÷ 8 mode	$V_{CC} \geqslant 4.5V$	DC	2.0	MHz
÷ 16 mode		DC	4.0	MHz
÷ 4 mode		DC	250	kHz
÷ 8 mode	$4.5V > V_{CC} \geqslant 3.0V$	DC	500	kHz
÷ 16 mode		DC	1.0	MHz

Instruction Cycle Time
RC Oscillator (Note 4)
	R = 30 k ± 5 %, V_{CC} = 5V
	C = 82pF ± 5 % (÷ 4 Mode)
		8	16	µs

Inputs (See Figure 3)				
t_{SETUP}	G Inputs	tc/4 + 0.7	-	µs
	SI Input $V_{CC} \geqslant 4.5V$	0.3	-	µs
	All Others	1.7	-	µs
t_{HOLD}	$V_{CC} \geqslant 4.5V$	0.25	-	µs
	$V_{CC} \geqslant 3.0V$	1.0	-	µs
Output Propagation Delay	V_{OUT} = 1.5V, C_L = 100pF, R_L = 5k			
t_{PD1}, t_{PD0}	$V_{CC} \geqslant 4.5V$	-	1.0	µs
t_{PD1}, t_{PD0}	$V_{CC} \geqslant 3.0V$	-	4.0	µs

Note 1 : Supply current is measured after running for 2000 cycle times with a square-ware clock on CKI, CKO open, and all other pins pulled up to V_{CC} with 20 k resistors.

Note 2 : The HALT mode will stop CKI from oscillating in the RC and crystal configurations.

Note 3 : When forcing HALT, current is only needed for a short time (approximately 200 ns) to flip the HALT flip-flop.

Note 4 : This parameter is only sampled and not 100 % tested.

Note 5 : Voltage change must be less than 0.5 volt in a 1 ms period.

Note 6 : SO output sink current must be limited to keep V_{OL} below $0.2V_{CC}$ when port is running in order to prevent entering test mode.

FUNCTIONAL DESCRIPTION

To ease reading of this description, only ETC 9410 and/or ETC 9411 are referenced ; however, all such references apply equally to ETC 9310 and/or ETC 9311,

A block diagram of the ETC 9410 is given in Figure 1. Data paths are illustrated in simplified form to depict how the various logic elements communicate with each other in implementing the instruction set of the device. Positive logic is used. When a bit is set, it is logic "1" when a bit is reset, it is a logic "0".

Program Memory

Program memory consists of a 512-byte ROM. As can be seen by an examination of the ETC 9410/C 9411 instruction set, these words may be program instructions, program data, or ROM addressing data. Because of the special characteristics associated with the JP, JSRP, JID, and LQID instructions, ROM must often be thought of being organized into 8 pages of 64 words (bytes) each.

ROM addressing

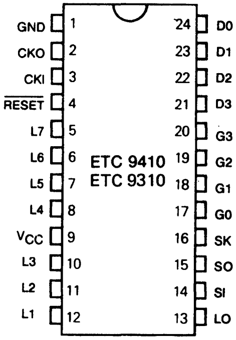
ROM addressing is accomplished by a 9-bit PC register. Its binary value selects one of the 512 8-bit words contained in ROM. A new address is loaded into the PC register during each instruction cycle. Unless the instruction is a transfer of control instruction, the PC register is loaded with the next sequential 9-bit binary count value. Two levels of subroutine nesting are implemented by two 9-bit subroutine save registers, SA and SB.

ROM instruction words are fetched, decoded, and executed by the Instruction Decode, Control and Skip Logic circuitry.

Data Memory

Data memory consists of a 128-bit RAM, organized as 4 data registers of 8 × 4-bit digits. RAM addressing is implemented by a 6-bit B register whose upper 2 bits (Br) select 1 of 4 data registers and lower 3 bits of the 4-bit Bd select 1 of eight 4-bit digits in the selected data register. While the 4-bit contents of the selected RAM digit (M) are usually loaded into or from, or exchanged with, the A register (accumulator), they may also be loaded into the Q latches or loaded from the L pprts. RAM addressing may also be performed directly by the XAD 3,15 instruction. The Bd register also serves as a source register for 4-bit data sent directly to the D outpouts.

The most significant bit of Bd is not used to select a RAM digit. Hence, each physical digit of RAM may be selected by two different values of Bd a shown in Figure 4 below. The skip condition for XIS and XDS instructions will be true if Bd changes between 0 and 15, but NOT between 7 and 8 (see Table 3).



PIN	DESCRIPTION
$L_7 - L_0$	8 bidirectional I/O ports with TRI-STATE®
G_3-G_0	4 bidirectional I/O ports (G_2-G_0 for 20 pin package)
D_3-D_0	4 genaral purpose outputs (D_1-D_0 for 20 pin package)
SI	Serial input (or counter input)
SO	Serial output (or general purpose output)
SK	Logic-controlled clock (or general purpose output)
CKI	System oscillator input
CKO	Crystal oscillator output (or HALT mode I/O port) (24 pin package only)
$\overline{\text{RESET}}$	System reset input
V_{CC}	System power supply
GND	System ground

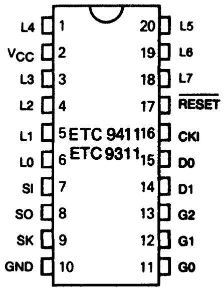


FIGURE 2 – CONNECTION DIAGRAMS

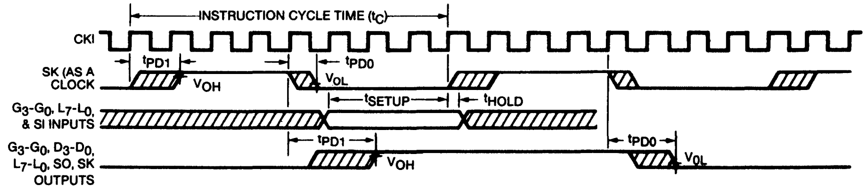

INSTRUCTION CYCLE TIME (t_C)
CKI
SK (AS A CLOCK
t_{PD1}
V_{OH}
t_{PD0}
V_{OL}
G_3-G_0, L_7-L_0, & SI INPUTS
t_{SETUP}
t_{HOLD}
G_3-G_0, D_3-D_0, L_7-L_0, SO, SK OUTPUTS
t_{PD1}
V_{OH}
t_{PD0}
V_{OL}

FIGURE 3 – INPUT/OUTPUT TIMING DIAGRAMS (DIVIDE-BY-8 MODE)

The D register provides 4 general purpose outputs and is used as the destination register for the 4-bit contents of Bd.

The XAS instruction copies C into the SKL Latch. In the counter mode, SK is the output of SKL ; in the shift register mode, SK is a sync clock, inhibited when SKL is a logic "0"

The EN register is an internal 4-bit register loaded under program control by the LEI instruction. The state of each bit of this register selects or deselects the particular feature associated with each bit of the EN register (EN3-EN0).

1. The least significant bit of the enable register, EN0, selects the SIO register as either a 4-bit shift register or as a 4-bit binary counter. With EN0 set, SIO is an asynchronous binary counter, DECREMENTING its value by one upon each low going pulse ("1" to "0") occurring on the SI input. Each pulse must be at least 2 (two) instruction cycles wide. SK outputs the value of SKL. The SO output is equal to the value of EN3. With EN0 reset. SIO is a serial shift register, shifting left each instruction cycle time. The data present at SI is shifted into the least significant bit of SIO. SO can be enabled to output the most significant bit of SIO each instruction cycle time. (See 4 below). The SK output becomes a logic controlled clock.
2. EN1 is not used, it has NO effect on the ETC 9410/ C 9411.
3. With EN2 set, the L drivers are enabled to output the data in Q to the L I/O ports. Resetting EN2 disables the L drivers, placing the L I/O ports in a hight impedance input state.
4. EN3, in conjunction with EN0, affects the SO output. With EN0 set (binary counter option selected), SO will output the value loaded into EN3. With EN0 reset (serial shift register option selected), setting EN3 enables SO as the output of the SIO shift register, outputting serial shifted data each instruction time. Resetting EN3 with the serial shift register option selected, disables SO as the shift register output ; data continues to be shifted through SIO and can be exchanged with A via an XAS instruction but SO remains reset to "0"

Enable Register Modes — Bits EN_3 and EN_0

EN_3	EN_0	SIO	SI	SO	SK
0	0	Shift Register	Input to Shift Register	0	If SKL = 1, SK = Clock If SKL = 0, SK = 0
1	0	Shift Register	Input to Shift Register	Serial Out	If SKL = 1, SK = Clock If SKL = 0, SK = 0
0	1	Binary Counter	Input to Binary Counter	0	If SKL = 1, SK = 1 If SKL = 0, SK = 0
1	1	Binary Counter	Input to Binary Counter	1	If SKL = 1, SK = 1 If SKL = 0, SK = 0

Internal Logic

The internal logic of the ETC 9410/C 9411 is designed to insure fully static operation of the device.

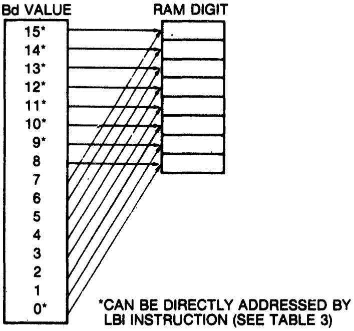
The 4-bit A register (accumulator) is the source and destination register for most I/O, arithmetic, logic and data memory access operations. It can also be used to load the Bd portion of the register, to load 4 bits of the 8-bit Q latch data and to perform data exchanges with the SIO register.

A 4-bit adder performs the arithmetic and logic functions of the ETC 9410/C 9411, storing its results in A. It also outputs the carry information to a 1-bit carry register, most often employed to indicate arithmetic overflow. The C registerin conjunction with the XAS instruction and the EN register, also serves to control the SK output. C can be outputted directly to SK or can enable SK to be a sync clock each instruction cycle time. (See XAS instruction and EN register description below).

The G register contents are outputs to 4 general-purpose bidirectional I/O ports.

The Q register is an internal, latched, 8-bit register, used to hold data loaded from RAM and A, as well as 8-bit data from ROM. Its contents are output to the L I/O ports when the L drivers are enabled under program control. (See LEI instruction).

The 8 L drivers, when enabled, output the contents of latched Q data to the L I/O ports. Also, the contents of L may be read directly into A and RAM.



*CAN BE DIRECTLY ADDRESSED BY LBI INSTRUCTION (SEE TABLE 3)

FIGURE 4 – RAM DIGIT ADDRESS TO PHYSICAL RAM DIGIT MAPPING

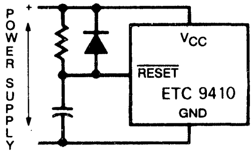
The SIO register functions as a 4-bit serial-in/serial-out shift register or as a binary counter depending upon the contents of the EN register. (See EN register description above). Its contents can be exchanged with A, allowing it to input or output continuous serial data stream. With SIO functioning as a serial-in/serial-out shift register and SK as a sync-clock, the ETC 9410/C 9411 is MICROWIRE® compatible.

Initialization

The internal reset logic will initialize the device upon power-up if the power supply rise time is less than 1 ms and if the operating frequency at CKI is greater than 32 kHz, otherwise the external RC network shown in Figure 5 must be connected to the RESET pin. The RESET pin is configured as a Schmitt trigger input. If not used, it should be connected to V_{CC}. Initialization will occur whenever a logic "0" is applied to the RESET input, providing it stays low for at least three instruction cycle times.

NOTE : If CKI clock is less than 32 kHz, the internal reset logic (option 25 = I) MUST be disabled and the external RC network must be present.

FIGURE 5 – POWER-UP CLEAR CIRCUIT



RC > 5 × Power Supply Rise Time
and RC > 100 × CKI period

Upon initialization, the PC register is cleared to 0 (ROM address 0) and the A, B, C, D, EN, and G registers are cleared. The SK output is enabled as a SYNC output, providing a pulse each instruction cycle time. Data Memory (RAM) is not cleared upon initialization. The first instruction at address 0 must be a CLRA (clear A register).

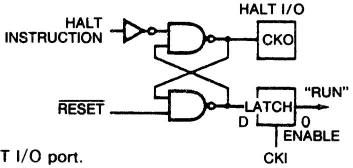
HALT Mode

The ETC 9410/C 9411 is a FULLY STATIC circuit ; therefore, the user may stop the system oscillator at any time to halt the chip. The chip may also be halted by the HALT instruction or by forcing CKO high when it is used as an HALT I/O port. Once in the HALT mode, the internal circuitry does not receive any clock signal, and is therefore frozen in the exact state it was in when halted. All information is retained until continuing. The HALT mode is the minimum power dissipation state.

The HALT mode has slight differences depending upon the type of oscillator used.

a) One-pin oscillator - (RC or External)
The HALT mode may be entered into by either program control (HALT instruction) or by forcing CKO to a logic "1" state.
The circuit may be awakened by one of two different methods :
1 - Continue function – by forcing CKO to a logic "0", the system clock is re-enabled and the circuit continues to operate from the point where it was stopped.
2 - Restart - forcing the RESET pin to a logic "0" will restart the chip regardless of HALT or CKO (see Initialization).

b) Two-pin oscillator - (Crystal)
The HALT mode may be entered into by program control (HALT instruction) which forces CKO to a logic "1" state. The circuit can be awakened only by the RESET function.



HALT I/O port.

ETC 9411

If the ETC 9410 is bonded as a 20-pin package, it becomes the ETC 9411 illustrated in Figure 2, ETC 9410/C 9411 Connection Diagrams. Note that the ETC 9411 does not contain D2, D3, G3, or CKO. Use of this option of course precludes use of D2, D3, G3, and CKO options. All other options are available for the ETC 9411.

CKO Pin Options

In a crystal controlled oscillator system, CKO is used as an output to the crystal network. CKO will be forced high during the execution of a HALT instruction, thus inhibiting the crystal network. If a one pin oscillator system is chosen (RC or external), CKO will be a selected as HALT and is an I/O flip-flop which is an indicator of the HALT status. An external signal can override this pin to start and stop the chip. By forcing a high level to CKO; the chip wil stop as soon as CKI is high and the CKO output will go high to keep the chip stopped. By forcing a low level to CKO, the chip will continue and CKO output will go low.

All features associated with the CKO I/O pin are available with the 24-pin package only.

Oscillator Options

There are three options available that define the use of CKI and CKO.

a) Crystal Controlled Oscillator. CKI and CKO are connected to an external crystal. The instruction cycle time equals the crystal frequency divided by 16 (optionally by 8 or 4).

b) External Oscillator. CKI is configured as a LSTTL compatible input accepting an external clock signal. The external frequency is divided by 16 (optionally by 8 or 4) to give the instruction cycle time. CKO is the HALT I/O port.

c) RC Controlled Oscillator. CKI is configured as a single pin RC controlled Schmitt trigger oscillator. The instruction cycle equals the oscillation frequency divided by 4. CKO is the HALT I/O port.

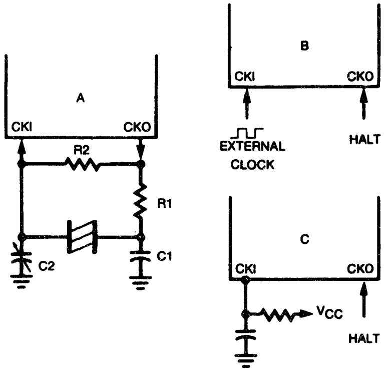


FIGURE 6 – ETC9410C OSCILLATOR

Crystal or resonator

Crystal value	R1	R2	C1 (pF)	C2 (pF)
		Component Values		
32 kHz	220 k	20 M	30	5 36
455 kHz	5 k	10 M	80	40
2.096 MHz	2 k	1 M	30	5 36
4 MHz	1 k	1 M	30	5 36

R C controlled Oscillator

R	C	Cycle time	V_{CC}
15 k	82 pF	4 to 9 µs	≥4.5V
30 k	82 pF	8 to 16 µs	≥4.5V
60 k	100 pF	16 to 32 µs	2.4 to 4.5V

Note : 15k ≤ R ≤ 150k ; 50pF ≤ C ≤ 150pF

This circuit and these values are for indication only. As the oscillator characteristics are not guaranteed, please consider and examine the circuit constants carefully on your application.

FIGURE 6 – ETC9410 OSCILLATOR

I/O Options

ETC 9410/C 9411 outputs have the following optional configurations, illustrated in Figure 7 :

a) Standard - A CMOS push-pull buffer with an N-channel device to ground in conjunction with a P-channel device to V_{CC}, compatible with CMOS and LSTTL.

b) Low Current - This is the same configuration as a) above except that the sourcing current much less.

c) Open Drain - An N-channel device to ground only, allowing external pull-up as required by the user's application.

d) Standard TRI-STATE® L Output - A CMOS output buffer similar to a) which may be disabled by program control.

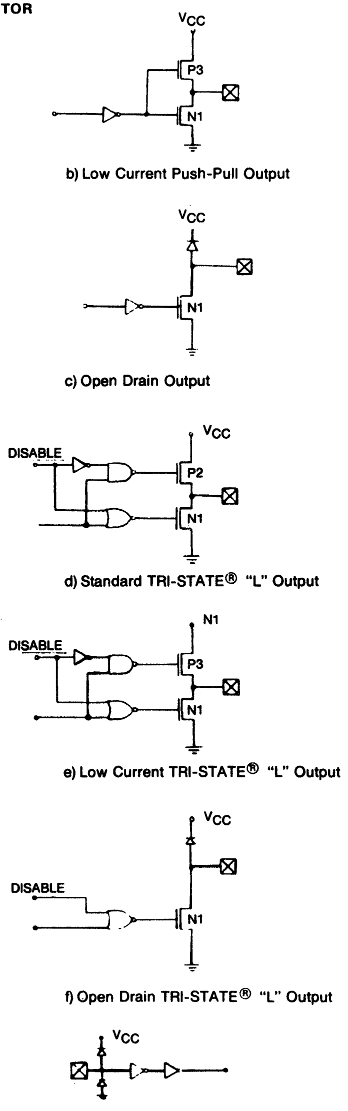
e) Low-Current TRI-STATE® L Output - This is the same as d) above except that the sourcing current is much less.

f) Open-Drain TRI-STATE® L Output - This has the N channel device to ground only.

The SI and RESET inputs are Hi-Z inputs (Fig. 7 g)

When using either the G or L I/O ports as inputs, a pull-up device is necessary. This can be an external device or the following alternative is available : Select the low-current output option. Now, by setting the output registers to a logic "I" level, the P-channel devices will act as the pull-up load. Note that when using the L ports in this fashion the Q registers must be set to a logic "I" level and the L drivers MUST BE ENABLED by an LEI instruction.

All output drivers use one or more of three common devices numbered 1 to 3. Minimum and maximum current (I_{out} V_{out} curves are given in Figure 8 for each of these devices to allow the designer to effectively use these I/O configurations.



b) Low Current Push-Pull Output

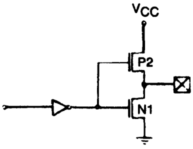
c) Open Drain Output

d) Standard TRI-STATE® "L" Output

e) Low Current TRI-STATE® "L" Output

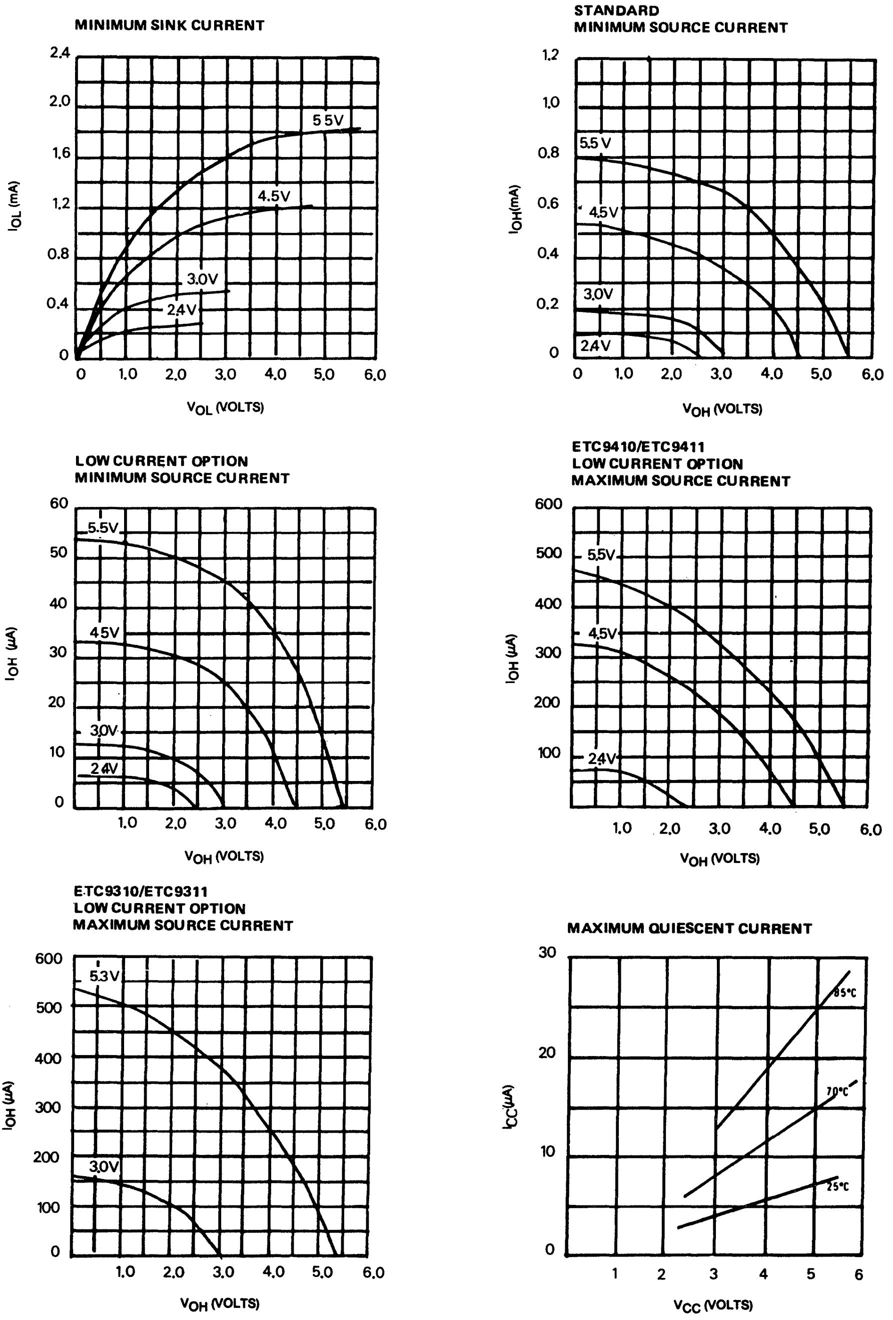
f) Open Drain TRI-STATE® "L" Output

g) Hi-Z Input



a) Standard Push Pull Output

FIGURE 7 – I/O CONFIGURATIONS


MINIMUM SINK CURRENT

STANDARD MINIMUM SOURCE CURRENT

LOW CURRENT OPTION MINIMUM SOURCE CURRENT

ETC9410/ETC9411 LOW CURRENT OPTION MAXIMUM SOURCE CURRENT

ETC9310/ETC9311 LOW CURRENT OPTION MAXIMUM SOURCE CURRENT

MAXIMUM QUIESCENT CURRENT

FIGURE 8 – INPUT/OUTPUT CHARACTERISTICS

ETC 9410/C 9411 INSTRUCTION SET

Table 2 is a symbol table providinq internal architecture, instruction operand and operational symbols used in the instruction set table.

Table 3 provides the mnemonic, operand, machine, code, data flow, skip conditions and description associated with each instruction in the ETC 9410/C 9411 instruction set.

TABLE 2 – ETC9410/C 9411 INSTRUCTION SET TABLE SYMBOLS

Symbol	Definition
INTERNAL	ARCHITECTURE SYMBOLS
A	4-bit Accumulator
B	6-bit RAM Address Register
Br	Upper 2 bits of B (register address)
Bd	Lower 4 bits of B (digit address)
C	1-bit Carry Register
D	4-bit Data Output Port
EN	4-bit Enable Register
G	4-bit Register to latch data for G I/O Port
L	8-bit TRI-STATE I/O Port
M	4-bit contents of RAM Memory pointed to by B Register
PC	9-bit ROM Address Register (program counter)
Q	8-bit Register to latch data for L I/O Port
SA	9-bit Subroutine Save Register A
SB	9-bit Subroutine Save Register B
SIO	4-bit Shift Register and Counter
SK	Logic-Controlled Clock Output

Symbol	Definition
INSTRUCTION OPERAND SYMBOLS	
d	4-bit Operand Fielo, 0 - 15 binary (RAM Digit Select)
r	2-bit Operand Field, 0 - 3 binary (RAM Register Select)
a	9-bit Operand Field, 0 - 511 binary (ROM Address)
y	4-bit Operand Field 0 - 15 binary (Immediate Data)
RAM(s)	Contents of RAM location addressed by s
ROM(t)	Contents of ROM location addressed by t
OPERATIONAL SYMBOLS	
+	Plus
—	Minus
→	Replaces
↔	Is exchanged with
=	Is equal to
A	The one's complement of A
⊕	Exclusive-OR
:	Range of values

TABLE 3 – ETC9410/C 9411 INSTRUCTION SET

Mnemonic	Operand	Hex Code	Machine Language Code (Binary)	Data Flow	Skip Conditions	Description
ARITHMETIC INSTRUCTIONS						
ASC		30	0 0 1 1 | 0 0 0 0	A + C + RAM(B) → A Carry → C	Carry	Add with Carry, Skip on Carry
ADD		31	0 0 1 1 | 0 0 0 1	A + RAM(B) → A	None	Add RAM to A
AISC	y	5-	0 1 0 1 | Y	A + y → A	Carry	Add Immediate, Skip on Carry (y ≠ 0)
CLRA		00	0 0 0 0 | 0 0 0 0	0 → A	None	Clear A
COMP		40	0 1 0 0 | 0 0 0 0	$\bar{A}$ → A	None	one's complement of A to A
NOP		44	0 1 0 0 | 0 1 0 0	None	None	No Operation
RC		32	0 0 1 1 | 0 0 1 0	"0" → C	None	Reset C
SC		22	0 0 1 0 | 0 0 1 0	"1" → C	None	Set C
XOR		02	0 0 0 0 | 0 0 1 0	A ⊕ RAM (B) → A	None	Exclusive-OR RAM with A

TABLE 3 – ETC9410/C 9411 INSTRUCTION SET (continued)

TRANSFER OF CONTROL INSTRUCTIONS

JID		FF	1 1 1 1 | 1 1 1 1	ROM (PC $_8$A,M)→PC$_{7:0}$	None	Jum Indirect
JMP	a	6- --	0 1 1 0 | 0 0 0 | a_8 $a_{7:0}$	a→PC	None	Jump
JP	a	-- --	1 | $a_{6:0}$ (pages 2.3 only) or 1 1 | $a_{5:0}$	a→PC$_{6:0}$ a→PC$_{5:0}$	None	Jump within Page (Note 1)
JSRP	a	--	1 0 | $a_{5:0}$	PC + 1→SA→SB 010→PC$_{8:6}$ a→PC$_{5:0}$	None	Jump to Subroutine Page (Note 2)
JSR	a	6-	0 1 1 0 | 1 0 0 | a $a_{7:0}$	PC + 1→SA→SB a→PC	None	Jump to Subroutine
RET		48	0 1 0 0 | 1 0 0 0	SB→SA→PC	None	Return from Subroutine
RETSK		49	0 1 0 0 | 1 0 0 1	SB→SA→PC	Always Skip on Return	Return from Subroutine then Skip
HALT		33 38	0 0 1 1 | 0 0 1 1 0 0 1 1 | 1 0 0 0		None	Halt processor

MEMORY REFERENCE INSTRUCTIONS

CAMQ		33 3C	0 0 1 1 | 0 0 1 1 0 0 1 1 | 1 1 0 0	A→Q$_{7:4}$ RAM (B)→Q$_{3:0}$	None	Copy A, RAM to Q
LD	r	-5	0 0 r | 0 1 0 1	RAM (B)→A Br ⊕ r→Br	None	Load RAM into A, Exclusive-OR Br with r
LQID		BF	1 0 1 1 | 1 1 1 1	ROM(PC$_8$A,M)→Q SA→SB	None	Load Q Indirect
RMB	0	4C	0 1 0 0 | 1 1 0 0	0→RAM (B)$_0$	None	Reset RAM Bit
	1	45	0 1 0 0 | 0 1 0 1	0→RAM (B)$_1$		
	2	42	0 1 0 0 | 0 0 1 0	0→RAM (B)$_2$		
	3	43	0 1 0 0 | 0 0 1 1	0→RAM (B)$_3$		
SMB	0	4D	0 1 0 0 | 1 1 0 1	1→RAM (B)$_0$	None	Set RAM Bit
	1	47	0 1 0 0 | 0 1 1 1	1→RAM (B)$_1$		
	2	46	0 1 0 0 | 0 1 1 0	1→RAM (B)$_2$		
	3	4B	0 1 0 0 | 1 0 1 1	1→RAM (B)$_3$		
STII	y	7-	0 1 1 1 | y	y→RAM (B) Bd + 1→Bd	None	Store Memory Immediate and Increment Bd
X	r	-6	0 0 | r | 0 1 1 0	RAM (B)↔A Br ⊕ r→Br	None	Exchange RAM with A, Exclusive-OR Br with r
XAD	3,15	23 BF	0 0 1 0 | 0 0 1 11 1 0 1 1 | 1 1 1 1	RAM (3,15)↔A	None	Exchange A with RAM (3,15)
XDS	r	-7	0 0 | r | 0 1 1 1	RAM (B)↔A Bd − 1→Bd Br ⊕ r→Br	Bd decrements past 0	Exchange RAM with A and Decrement Bd, Exclusive-OR Br with r
XIS	r	4	0 0 | r | 0 1 0 0	RAM (B)↔A Bd + 1→Bd Br ⊕ r→Br	Bd increments past 15	Exchange RAM with A and Increment Bd, Exclusive Or Br with r

TABLE 3 – ETC9410/C 9411 INSTRUCTION SET (continued)

Mnemonic	Operand	Hex Code	Machine Language Code (Binary)	Data Flow	Skip Conditions	Description
REGISTER REFERENCE INSTRUCTIONS						
CAB		50	0101 0000	A → Bd	None	Copy A to Bd
CBA		4E	0100 1110	Bd → A	None	Copy Bd to A
LBI	r.d		00 r (d - 1) (d = 0, 9:15)	r.d → B	Skip until not a LBI	Load B Immediate with r.d
LEI	y	33 6-	0011 0011 0110 y	y → EN	None	Load EN Immediate
TEST INSTRUCTIONS						
SKC		20	0010 0000		C = "1"	Skip if C is True
SKE		21	0010 0001		A = RAM(B)	Skip if A Equals RAM
SKGZ		33 21	0011 0011 0010 0001		$G_{3:0}$ = 0	Skip if G is Zero (all 4 bits)
SKGBZ		33	0011 0011	1st byte		Skip if G Bit is Zero
	0	01	0000 0001	2nd byte	G_0 = 0	
	1	11	0001 0001		G_1 = 0	
	2	03	0000 0011		G_2 = 0	
	3	13	0001 0011		G_3 = 0	
SKMBZ	0	01	0000 0001		$RAM(B)_0$ = 0	Skip if RAM Bit is Zero
	1	11	0001 0001		$RAM(B)_1$ = 0	
	2	03	0000 0011		$RAM(B)_2$ = 0	
	3	13	0001 0011		$RAM(B)_3$ = 0	
INPUT/OUTPUT INSTRUCTIONS						
ING		33 2A	0011 0011 0010 1010	G → A	None	Input G Ports to A
INL		33 2E	0011 0011 0010 1110	$L_{7:4}$ → RAM(B) $L_{3:0}$ → A	None	Input L Ports to RAM, A
OBD		33 3E	0011 0011 0011 1110	Bd → D	None	Output Bd to D Outputs
OMG		33 3A	0011 0011 0011 1010	RAM(B) → G	None	Output RAM to G Ports
XAS		4F	0100 1111	A ←→ SIO, C → SKL	None	Exchange A with SIO

Note 1 : The JP instruction allows a jump, while in subroutine pages 2 or 3, to any ROM location within the two-page boundary of pages 2 or 3. The JP instruction, otherwise, permits a jump to a ROM location within the current 64-word page. JP may not jump to the last word of a page.

Note 2 : A JSRP transfers program control to subroutine page 2 (0010 is loaded into the upper 4 bits of P). A JSRP may not be used when in pages 2 or 3. JSRP may not jump to the last word in page 2.

The following information is provided to assist the user in uderstanding the operation of several unique instructions and to provide notes useful to programmers in writing ETC 9410 C 9411.

XAS Instruction

XAS (Exchange A with SIO) exchanges the 4-bit contents of the accumulator with the 4-bit contents of the SIO register or binary counter data, depending on the value of the EN register. An XAS instruction will also affect the SK output. (See Functional Description, EN Register, above). If SIO is selected as a shift register, an XAS instruction must be performed once every 4 instruction cycles to effect a continuous data stream.

JID Instruction

JID (Jump Indirect) is an indirect addressing instruction, transferring program control to a new ROM location pointed to indirectlv by A and M. It loads the lower 8 bits of the ROM address register PC with the *contents* of ROM Addressed by the 9-bit word, PC_8, A, M. PC_8 is not affected by this instruction.

Note That JID requires 2 instruction cycles to execute.

LQID Instruction

LQID (Load Q Indirect) loads the 8-bit Q register with the contents of ROM pointed to by the 9-bit word PC_8, A, M. LQID can be used for table lookup or code conversion such as BCD to seven-segment. The LQID instruction "pushes" the stack (PC + 1 → SA → SB) and replaces the least significant 8 bits of PC as follows : A → $PC_{7:4}$ RAM (B) → $PC_{3:0}$ leaving PC_8 unchanged. The ROM data pointed to by the new address is fetched and loaded into the Q latches. Next, the stack is "popped" (SB → SA → PC), restoring the saved value of PC to continue sequential program execution. Since LQID pusches SA → SB, the previous contents of SB are lost. Also, when LQID pops the stack, the previously pushed contents of SA are left in SB. The net result is that the contents of SA are placed in SB (SA → SB). Note that LQID takes two instruction cycle times to execute.

Instruction Set Notes

a) The first word of a ETC 9410/C 9411 program (ROM address 0) must be a CLRA (Clear A) instruction.

b) Although skipped instructions are not executed, one instruction cycle time is devoted to skipping each byte of the skipped instruction. Thus all program paths except JID and LQID take the same number of cycle times whether instructions are skipped or executed. JID and LQID instructions take 2 cycles if executed and 1 cycle if skipped.

c) The ROM is organized into 8 pages of 64 words each. The Program Counter is a 9-bit binary counter, and will count through page boundaries. If a JP, JSRP, JID or LQID instruction is located in the last word of a page, the instruction operates as if it were in the next page. For example : a JP located in the last word of a page will jump to a location in the next page. Also, a LQID or JID located in the last word of page 3 or 7 will access data in the next group of 4 pages.

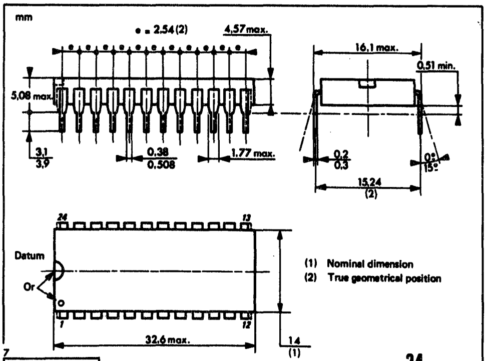
OPTION LIST

The ETC 9410/ETC 9411 mask-programmable options are assigned numbers which correspond with the ETC 9410 pins.

The following is a list of ETC9410 options. When specifying a ETC 9411chip, Options 20, 21, and 22 must be set to 0. The options are programmed at the same time as the ROM pattern to provide the user with the hardware flexibility to interface to various I/O components using little or no external circuitry.

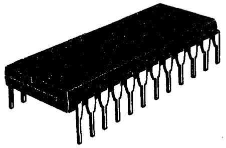
Option 1 : = 0 : Ground Pin. No options available.
Option 2 : CKO I/O Port. Determined by Option 3
Option 3 : CKI Input
= 0 : Crystal controlled oscillator input (÷ 4).
= 1 : Single-pin RC-controlled oscillator (÷ 4).
= 2 : External oscillator input (÷ 4).
= 3 : Crystal oscillator input (÷ 8).
= 4 : External oscillator input (÷ 8).
= 5 : Crystal oscillator input (÷ 16).
= 6 : External oscillator input (÷ 16).
Option 4 : RESET Input = 1 : Hi-Z input. No option available
Option 5 : L_7 Driver
= 0 : Standard TRI-STATE push-pull output
= 1 : Low-current TRI-STATE push-pull output
= 2 : Open-drain TRI-STATE output
Option 6 : L_6 Driver. (Same as Option 5.)
Option 7 : L_5 Driver. (Same as Option 5.)
Option 8 : L_4 Driver. (Same as Option 5.)
Option 9 : V_{CC} Pin.
Option 10 : L_3 Driver. (Same as Option 5.)
Option 11 : L_2 Driver. (Same as Option5.)
Option 12 : L_1 Driver. (Same as Option 5.)
Option 13 : L_0 Driver. (Same as Option 5.)
Option 14 : SI Input.
No option available.
= 1 : Hi-Z input
Option 15 : SO Output.
= 0 : Standard push-pull output.
= 1 : Low-current push-pull output.
= 2 : Open-drain output
Option 16 : SK Driver. (Same as Option 15.)
Option 17 : G_0 I/O Port. (Same as Option 15.)
Option 18 : G_1 I/O Port. (Same as Option 15.)
Option 19 : G_2 I/O Port. (Same as Option 15.)
Option 20 : G_3 Output. (Same as Option 15.)
Option 21 : D_3 Output. (Same as Option 15.)
Option 22 : D_2 Output. (Same as Option 15.)
Option 23 : D_1 Output. (Same as Option 15.)
Option 24 : D_0 Output. (Same as option 15.)
Option 25 : Internal Initialization logic.
= 0 : Normal operation.
= 1 : No internal initialization Logic
Option 26 : No option available.
Option 27 : Chip Bonding
= 0 : ETC 9410 (24-pin device).
= 1 : ETC 9411 (20-pin device).
= 2 : ETC 9410 and ETC 9411.

PHYSICAL DIMENSIONS

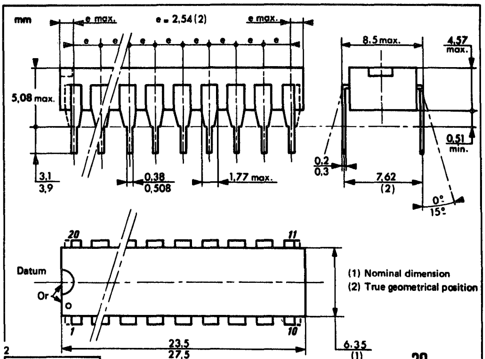


DIN				
A51D CEI	D.A.T.A.	JEDEC	F 118 SITELESC	CB-68

CB-68

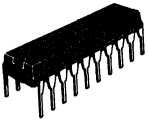


N SUFFIX
PLASTIC PACKAGE



DIN				
CEI	D.A.T.A.	JEDEC	SITELESC	CB-194

CB-194



N SUFFIX
PLASTIC PACKAGE

These specifications are subject to change without notice.
Please inquire with our sales offices about the availability of the different products.

Printed in France

Ref.DSETC93941011

HOMSON
COMPONENTS
EMICONDUCTEURS

ETL9410 • ETL9411 • ETL9310 • ETL9311

SINGLE CHIP MICROCONTROLLERS

The ETL9410 and ETL9411 Single-Chip N-Channel Microcontrollers are fully compatible with the COPS® family, fabricated using N-channel, silicon gate MOS technology. The Controller Oriented Processors are complete microcomputers containing all system timing, internal logic, ROM, RAM and I/O necessary to implement dedicated control functions in a variety of applications. Features include single supply operation, a variety of output configuration options, with an instruction set, internal architecture and I/O scheme designed to facilitate keyboard input, display output and BCD data manipulation. The ETL9411 is identical to the ETL9410, but with 16 I/O lines instead of 19. They are an appropriate choice for use in numerous human interface control environments. Standard test procedures and reliable high-density fabrication techniques provide the medium to large volume customers with a customized Controller Oriented Processor at a low end-product cost.

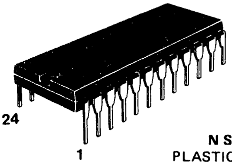
The ETL9310 and ETL9311 are exact functional equivalents but extended temperature versions of ETL9410 and ETL9411 respectively.

- ● Low cost
- ● Powerful instruction set
- ● 512 × 8 ROM. 32 × 4 RAM
- ● 19 I/O lines ETL9410 - COP
- ● Two-level subroutine stack
- ● 16μs instruction time
- ● Single supply operation (4.5 - 6.3V)
- ● Low current drain (6mA max.)
- ● Internal binary counter register with MICROWIRE® serial I/O capability
- ● General purpose and TRI-STATE® outputs
- ● LSTTL/CMOS compatible in and out
- ● Direct drive of LED digit and segment lines
- ● Software/hardware compatible with other members of ET9400 family
- ● Extended temperature range device ETL9310 ETL9311 (-40° C to + 85° C)
- ● Wider supply range (4.5 - 9.5V) optionally available

NMOS

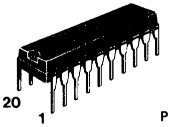
CASES

ETL9410 • ETL9310



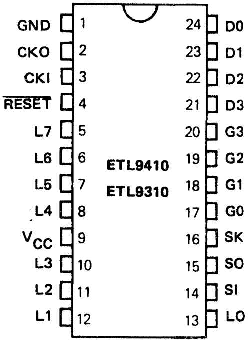
N SUFFIX
PLASTIC PACKAGE

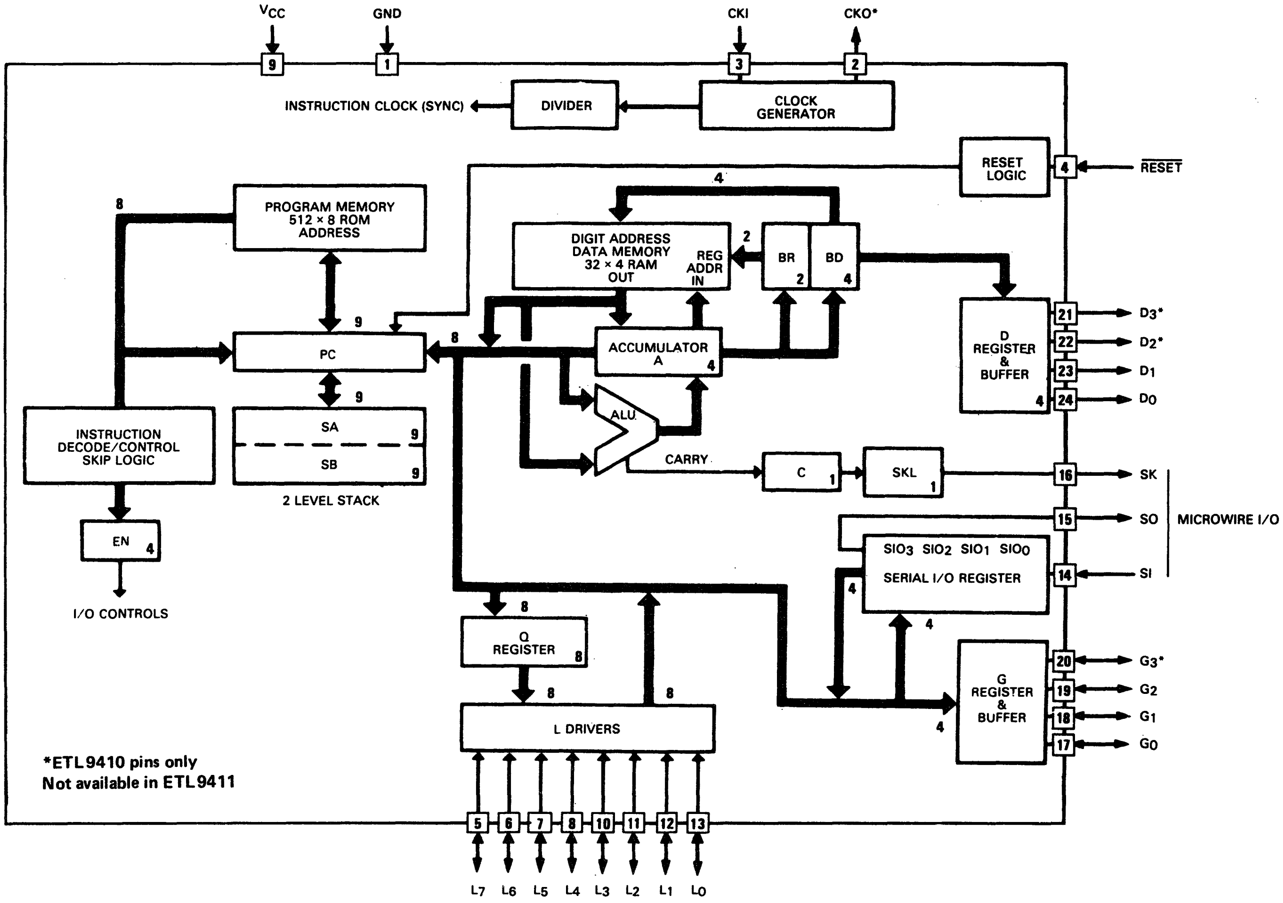
ETL9411 • ETL9311



N SUFFIX
PLASTIC PACKAGE

PIN ASSIGNMENT





*ETL9410 pins only
Not available in ETL9411

FIGURE 1 – BLOCK DIAGRAM (24-pin version)

ETL9410 • ETL9411

ABSOLUTE MAXIMUM RATINGS

Voltage at Any Pin Relative to GND	−0.5V to +10V
Ambient Operating Temperature	0°C to +70°C
Ambient Storage Temperature	−65°C to +150°C
Lead Temperature (Soldering, 10 seconds)	300°C
Power Dissipation	
ET L 9410	0.75 Watt at 25°C
	0.4 Watt at 70°C
ET L 9411	0.65 Watt at 25°C
	0.3 Watt at 70°C
Total Source Current	120 mA
Total Sink Current	100 mA

Absolute maximum ratings indicate limits beyond which damage to the device may occur. DC and AC electrical specifications are not ensured when operating the device at absolute maximum ratings.

DC ELECTRICAL CHARACTERISTICS $0°C \leqslant T_A \leqslant +70°C$, $4.5V \leqslant V_{CC} \leqslant 9.5V$ (Unless otherwise specified)

Parameter	Conditions	Min.	Max.	Units
Standard Operating Voltage (V_{CC})	Note 1	4.5	6.3	V
Optional Operating Voltage (V_{CC})		4.5	9.5	V
Power Supply Ripple	peak to peak	–	0.5	V
Operating Supply Current	all inputs and outputs open	–	6	mA
Input Voltage Levels				
CKI Input Levels				
Ceramic Resonator Input (÷8)				
Logic High (V_{IH})		2.0	–	V
Logic Low (V_{IL})		−0.3	0.4	V
Schmitt Trigger Input (÷4)				
Logic High (V_{IH})		0.7 V_{CC}	–	V
Logic Low (V_{IL})		−0.3	0.6	V
$\overline{\text{RESET}}$ Input Levels	(Schmitt Trigger Input)			
Logic High		0.7 V_{CC}	–	V
Logic Low		−0.3	0.6	V
SO Input Level (Test mode)	Note 2	2.0	2.5	V
All Other Inputs				
Logic High	V_{CC} = Max.	3.0	–	V
Logic High	with TTL trip level options	2.0	–	V
Logic Low	selected, V_{CC} = 5V ± 5%	−0.3	0.8	V
Logic High	with high trip level options	3.6	–	V
Logic Low	selected	−0.3	1.2	V
Input Capacitance		–	7	pF
Hi-Z Input Leakage		−1	+1	µA
Output Voltage Levels				
LSTTL Operation	V_{CC} = 5V ± 5%			
Logic High (V_{OH})	$I_{OH} = -25\mu A$	2.7	–	V
Logic Low (V_{OL})	$I_{OL} = 0.36mA$	–	0.4	V
CMOS Operation				
Logic High	$I_{OH} = -10\mu A$	V_{CC} − 1	–	V
Logic Low	$I_{OL} = +10\mu A$	–	0.2	V

Note 1 : V_{CC} voltage change must be less than 0.5V in a 1 ms period to maintain proper operation.
Note 2 : SO output "0" level must be less than 0,8V for normal operation.

ETL9410 • ETL9411

DC ELECTRICAL CHARACTERISTICS (continued) 0°C≤T_A≤+70°C, 4.5V≤V_{CC}≤9.5V (Unless otherwise specified)

Parameter	Conditions	Min.	Max.	Units
Output Current Levels				
Output Sink Current				
SO and SK Outputs (I_{OL})	V_{CC} = 9.5V, V_{OL} = 0.4V	1.8	–	mA
	V_{CC} = 6.3V, V_{OL} = 0.4V	1.2	–	mA
	V_{CC} = 4.5V, V_{OL} = 0.4V	0.9	–	mA
L_0-L_7 Outputs, G_0-G_3 and	V_{CC} = 9.5V, V_{OL} = 0.4V	0.8	–	mA
LSTTL D_0-D_3 Outputs (I_{OL})	V_{CC} = 6.3V, V_{OL} = 0.4V	0.5	–	mA
	V_{CC} = 4.5V, V_{OL} = 0.4V	0.4	–	mA
D_0-D_3 Outputs with High	V_{CC} = 9.5V, V_{OL} = 1.0V	15	–	mA
Current Options (I_{OL})	V_{CC} = 6.3V, V_{OL} = 1.0V	11	–	mA
	V_{CC} = 4.5V, V_{OL} = 1.0V	7.5	–	mA
D_0-D_3 Outputs with Very	V_{CC} = 9.5V, V_{OL} = 1.0V	30	–	mA
High Current Options (I_{OL})	V_{CC} = 6.3V, V_{OL} = 1.0V	22	–	mA
	V_{CC} = 4.5V, V_{OL} = 1.0V	15	–	mA
CKI (Single-pin RC oscillator)	V_{CC} = 4.5V, V_{IH} = 3.5V	2	–	mA
Output Source Current				
Standard Configuration,	V_{CC} = 9.5V, V_{OH} = 2.0V	−140	−800	μA
All Outputs (I_{OH})	V_{CC} = 6.3V, V_{OH} = 2.0V	−75	−480	μA
	V_{CC} = 4.5V, V_{OH} = 2.0V	−30	−250	μA
Push-Pull Configuration	V_{CC} = 9.5V, V_{OH} = 4.75V	−1.4	–	mA
SO and SK Outputs (I_{OH})	V_{CC} = 6.3V, V_{OH} = 2.4V	−1.4	–	mA
	V_{CC} = 4.5V, V_{OH} = 1.0V	−1.2	–	mA
LED Configuration, L_0-L_7				
Outputs, Low Current	V_{CC} = 9.5V, V_{OH} = 2.0V	−1.5	−18	mA
Driver Option (I_{OH})	V_{CC} = 6.0V, V_{OH} = 2.0V	−1.5	−13	mA
LED Configuration, L_0-L_7				
Outputs, High Current	V_{CC} = 9.5V, V_{OH} = 2.0V	−3.0	−35	mA
Driver Option (I_{OH})	V_{CC} = 6.0V, V_{OH} = 2.0V	−3.0	−25	mA
TRI-STATE® Configuration,	V_{CC} = 9.5V, V_{OH} = 5.5V	−0.75	–	mA
L_0-L_7 Outputs, Low	V_{CC} = 6.3V, V_{OH} = 3.2V	−0.8	–	mA
Current Driver Option (I_{OH})	V_{CC} = 4.5V, V_{OH} = 1.5V	−0.9	–	mA
TRI-STATE® Configuration,	V_{CC} = 9.5V, V_{OH} = 5.5V	−1.5	–	mA
L_0-L_7 Outputs, High	V_{CC} = 6.3V, V_{OH} = 3.2V	−1.6	–	mA
Current Driver Option (I_{OH})	V_{CC} = 4.5V, V_{OH} = 1.5V	−1.8	–	mA
Input Load Source Current	V_{CC} = 5.0V, V_{IL} = 0V	−10	−140	μA
CKO Output				
RAM Power Supply Option				
Power Requirement	V_R = 3.3V	–	1.5	mA
TRI-STATE® Output Leakage				
Current		−2.5	+2.5	μA
Total Sink Current Allowed				
All Outputs Combined		–	100	mA
D Port		–	100	mA
L_7-L_4, G Port		–	4	mA
L_3-L_0		–	4	mA
Any Other Pin		–	2.0	mA
Total Source Current Allowed				
All I/O Combined		–	120	mA
L_7-L_4		–	60	mA
L_3-L_0		–	60	mA
Each L Pin		–	25	mA
Any Other Pin		–	1.5	mA

ETL9310 • ETL9311

ABSOLUTE MAXIMUM RATINGS

Voltage at Any Pin Relative to GND	−0.5V to +10V
Ambient Operating Temperature	−40°C to +85°C
Ambient Storage Temperature	−65°C to +150°C
Lead Temperature (Soldering, 10 seconds)	300°C
Power Dissipation	
ET L 9310	0.75 Watt at 25°C
	0.25 Watt at 85°C
ET L 9311	0.65 Watt at 25°C
	0.20 Watt at 85°C
Total Source Current	120 mA
Total Sink Current	100 mA

Absolute maximum ratings indicate limits beyond which damage to the device may occur. DC and AC electrical specifications are not ensured when operating the device at absolute maximum ratings.

DC ELECTRICAL CHARACTERISTICS – 40°C≤T_A≤+85°C, 4.5V≤V_{CC}≤7.5V (Unless otherwise specified)

Parameter	Conditions	Min.	Max.	Units
Standard Operating Voltage (V_{CC})	Note 1	4.5	5.5	V
Optional Operating Voltage (V_{CC})		4.5	7.5	V
Power Supply Ripple	peak to peak	–	0.5	V
Operating Supply Current	all inputs and outputs open	–	8	mA
Input Voltage Levels				
Ceramic Resonator Input(÷8)				
Crystal Input				
Logic High (V_{IH})		2.2	–	V
Logic Low (V_{IL})		−0.3	0.3	V
Schmitt Trigger Input (÷4)				
Logic High (V_{IH})		0.7 V_{CC}	–	V
Logic Low (V_{IL})		−0.3	0.4	V
$\overline{\text{RESET}}$ Input Levels	(Schmitt Trigger Input)			
Logic High		0.7 V_{CC}	–	V
Logic Low		−0.3	0.4	V
SO Input Level (Test mode)	Note 2	2.2	2.5	V
All Other Inputs				
Logic High	V_{CC} = Max.	3.0	–	V
Logic High	with TTL trip level options	2.2	–	V
Logic Low	selected, V_{CC} = 5V ± 5%	−0.3	0.6	V
Logic High	with high trip level options	3.6	–	V
Logic Low	selected	−0.3	1.2	V
Input Capacitance		–	7	pF
Hi-Z Input Leakage		−2	+2	μA
Output Voltage Levels				
LSTTL Operation	V_{CC} = 5V ± 5%			
Logic High (V_{OH})	I_{OH} = −20μA	2.7	–	V
Logic Low (V_{OL})	I_{OL} = 0.36mA	–	0.4	V
CMOS Operation				
Logic High	I_{OH} = −10μA	V_{CC} − 1	–	V
Logic Low	I_{OL} = +10μA	–	0.2	V

Note 1 : V_{CC} voltage change must be less than 0.5V in a 1 ms period to maintain proper operation.
Note 2 : SO output "0" level must be less than 0.6V for normal operation.

ETL9310 • ETL9311

DC ELECTRICAL CHARACTERISTICS (continued) $-40°C \leq T_A \leq +85°C$, $4.5V \leq V_{CC} \leq 7.5V$ (Unless otherwise specified)

Parameter	Conditions	Min.	Max.	Units
Output Current Levels				
Output Sink Current				
SO and SK Outputs (I_{OL})	$V_{CC} = 7.5V$, $V_{OL} = 0.4V$	1.4	–	mA
	$V_{CC} = 5.5V$, $V_{OL} = 0.4V$	1.0	–	mA
	$V_{CC} = 4.5V$, $V_{OL} = 0.4V$	0.8	–	mA
L_0-L_7 Outputs, G_0-G_3 and LSTTL, D_0-D_3 Outputs (I_{OL})	$V_{CC} = 7.5V$, $V_{OL} = 0.4V$	0.6	–	mA
	$V_{CC} = 5.5V$, $V_{OL} = 0.4V$	0.5	–	mA
	$V_{CC} = 4.5V$, $V_{OL} = 0.4V$	0.4	–	mA
D_0-D_3 Outputs with High Current Options (I_{OL})	$V_{CC} = 7.5V$, $V_{OL} = 1.0V$	12	–	mA
	$V_{CC} = 5.5V$, $V_{OL} = 1.0V$	9	–	mA
	$V_{CC} = 4.5V$, $V_{OL} = 1.0V$	7	–	mA
D_0-D_3 Outputs with Very High Current Options (I_{OL})	$V_{CC} = 7.5V$, $V_{OL} = 1.0V$	24	–	mA
	$V_{CC} = 5.5V$, $V_{OL} = 1.0V$	18	–	mA
	$V_{CC} = 4.5V$, $V_{OL} = 1.0V$	14	–	mA
CKI (Single-pin RC oscillator)	$V_{CC} = 4.5V$, $V_{IH} = 3.5V$	2	–	mA
Output Source Current				
Standard Configuration, All Outputs (I_{OH})	$V_{CC} = 7.5V$, $V_{OH} = 2.0V$	−100	−900	μA
	$V_{CC} = 5.5V$, $V_{OH} = 2.0V$	−55	−600	μA
	$V_{CC} = 4.5V$, $V_{OH} = 2.0V$	−28	−350	μA
Push-Pull Configuration SO and SK Outputs (I_{OH})	$V_{CC} = 7.5V$, $V_{OH} = 3.75V$	−0.85	–	mA
	$V_{CC} = 5.5V$, $V_{OH} = 2.0V$	−1.1	–	mA
	$V_{CC} = 4.5V$, $V_{OH} = 1.0V$	−1.2	–	mA
LED Configuration, L_0-L_7 Outputs, Low Current Driver Option (I_{OH})	$V_{CC} = 7.5V$, $V_{OH} = 2.0V$	−1.4	−27	mA
LED Configuration, L_0-L_7 Outputs, High Current Driver Option (I_{OH})	$V_{CC} = 7.5V$, $V_{OH} = 2.0V$	−2.7	−54	mA
TRI-STATE® Configuration, L_0-L_7 Outputs, Low Current Driver Option (I_{OH})	$V_{CC} = 7.5V$, $V_{OH} = 4.0V$	−0.7	–	mA
	$V_{CC} = 5.5V$, $V_{OH} = 2.7V$	−0.6	–	mA
	$V_{CC} = 4.5V$, $V_{OH} = 1.5V$	−0.9	–	mA
TRI-STATE® Configuration, L_0-L_7 Outputs, High Current Driver Option (I_{OH})	$V_{CC} = 7.5V$, $V_{OH} = 4.0V$	−1.4	–	mA
	$V_{CC} = 5.5V$, $V_{OH} = 2.7V$	−1.2	–	mA
	$V_{CC} = 4.5V$, $V_{OH} = 1.5V$	−1.8	–	mA
Input Load Source Current	$V_{CC} = 5.0V$, $V_{IL} = 0V$	−10	−200	μA
CKO Output				
RAM Power Supply Option Power Requirement	$V_R = 3.3V$	–	2.0	mA
TRI-STATE® Output Leakage Current		−5	+5	μA
Total Sink Current Allowed				
All Outputs Combined		–	100	mA
D Port		–	100	mA
L_7-L_4, G Port		–	4	mA
L_3-L_0		–	4	mA
Any Other Pin		–	2.0	mA
Total Source Current Allowed				
All I/O Combined		–	120	mA
L_7-L_4		–	60	mA
L_3-L_0		–	60	mA
Each L Pin		–	25	mA
Any Other Pin		–	1.5	mA

AC ELECTRICAL CHARACTERISTICS

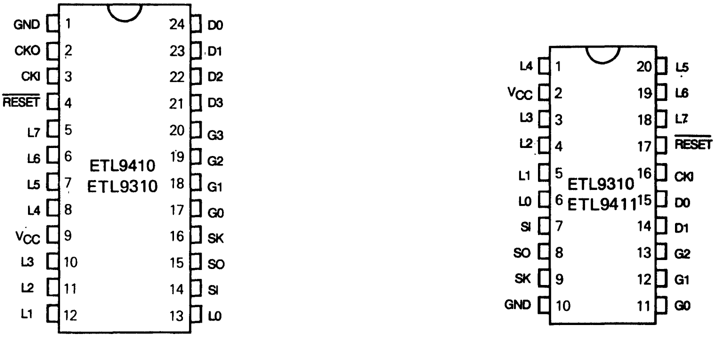
ETL 9410/L 9411 : 0°C ≤ T_A ≤ +70°C, 4.5V ≤ V_{CC} ≤ 9.5V (Unless otherwise specified)
ETL 9310/L 9311 : −40°C ≤ T_A ≤ +85°C, 4.5V ≤ V_{CC} ≤ 7.5V (Unless otherwise specified)

Parameter	Conditions	Min.	Max.	Units
Instruction Cycle Time — t_C		16	40	μs
CKI				
Input Frequency — f_I	÷8 mode ÷4 mode	0.2 0.1	0.5 0.25	MHz MHz
Duty Cycle		30	60	%
Rise Time	f_I = 0.5 MHz	–	500	ns
Fall Time		–	200	ns

CKI Using RC (÷4)	R = 56 kΩ ± 5%
	C = 100 pF ± 10%			

Instruction Cycle Time		16	28	μs
CKO as SYNC Input				
t_{SYNC}		400	–	ns
INPUTS:				
G_3-G_0, L_7-L_0				
t_{SETUP}		–	8.0	μs
t_{HOLD}		–	1.3	μs
SI				
t_{SETUP}		–	2.0	μs
t_{HOLD}		–	1.0	μs
OUTPUT PROPAGATION DELAY	Test condition: C_L = 50 pF, R_L = 20 kΩ, V_{OUT} = 1.5V			
SO, SK Outputs				
t_{pd1}, t_{pd0}		–	4.0	μs
All Other Outputs				
t_{pd1}, t_{pd0}		–	5.6	μs

FIGURE 2 – CONNECTION DIAGRAMS



Pin	Description
L_7-L_0	8 bidirectional I/O ports with TRI-STATE®
G_3-G_0	4 bidirectional I/O ports (G_2-G_0 for COP411L)
D_3-D_0	4 general purpose outputs (D_1-D_0 for ETL 9411)
SI	Serial input (or counter input)
SO	Serial output (or general purpose output)

Pin	Description
SK	Logic-controlled clock (or general purpose output)
CKI	System oscillator input
CKO	System oscillator output (or RAM power supply or SYNG input) (ETL 9410 only)
RESET	System reset input
V_{CC}	Power supply
GND	Ground

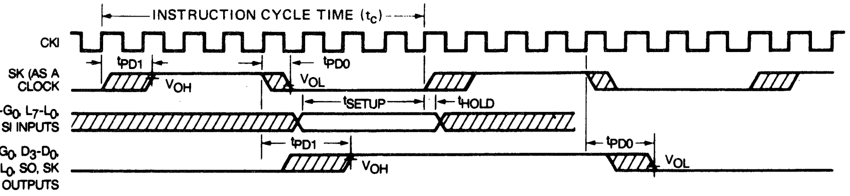

INSTRUCTION CYCLE TIME (t_C)
CKI
SK (AS A CLOCK
tPD1
tPD0
VOH
VOL
tSETUP
tHOLD
G3-G0, L7-L0, & SI INPUTS
tPD1
tPD0
VOH
VOL
G3-G0, D3-D0, L7-L0, SO, SK OUTPUTS

FIGURE 3 – INPUT/OUTPUT TIMING DIAGRAMS (CERAMIC RESONATOR DIVIDE-BY-8 MODE)

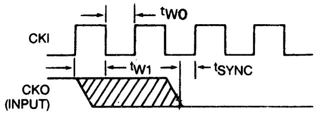


FIGURE 3a – SYNCHRONIZATION TIMING

FUNCTIONAL DESCRIPTION

A block diagram of the ET L 9410 is given in Figure 1. Data paths are illustrated in simplified form to depict how the various logic elements communicate with each other in implementing the instruction set of the device. Positive logic is used. When a bit is set, it is a logic "1" (greater than 2 volts). When a bit is reset, it is a logic "0" (less than 0.8 volts).

All functional references to the ET L 9410 / L 9411 - also apply to the ET L 9310 / L 9311..

Program Memory

Program Memory consists of a 512-byte ROM.
As can be seen by an examination of the ET L 9410 / L 9411 instruction set, these words may be program instructions, program data or ROM addressing data. Because of the special characteristics associated with the JP, JSRP, JID and LQID instructions, ROM must often be thought of as being organized into 8 pages of 64 words each.

ROM addressing is accomplished by a 9-bit PC register. Its binary value selects one of the 512 8-bit words contained in ROM. A new address is loaded into the PC register during each instruction cycle. Unless the instruction is a transfer of control instruction, the PC register is loaded with the next sequential 9-bit binary count value. Two levels of subroutine nesting are implemented by the 9-bit subroutine save registers, SA and SB, providing a last-in, first-out (LIFO) hardware subroutine stack.

ROM instruction words are fetched, decoded and executed by the instruction Decode, Control and Skip Logic circuitry.

Data Memory

Data memory consists of a 128-bit RAM, organized as 4 data registers of 8 4-bit digits. RAM addressing is implemented by a 6-bit B register whose upper 2 bits (Br) select 1 of 4 data registers and lower 3 bits of the 4-bit Bd select 1 of 8 4-bit digits in the selected data register. While the 4-bit contents of the selected RAM digit (M) is usually loaded into or from, or exchanged with, the A register (accumulator), it may also be loaded into the Q latches or loaded from the L ports. RAM addressing may also be performed directly by the XAD 3, 15 instruction. The Bd register also serves as a source register for 4-bit data sent directly to the D outputs.

The most significant bit of Bd is not used to select a RAM digit. Hence each physical digit of RAM may be selected by two different values of Bd as shown in Figure 4 below. The skip condition for XIS and XDS instructions will be true if Bd changes between 0 and 15, but NOT between 7 and 8 (see Table 3).

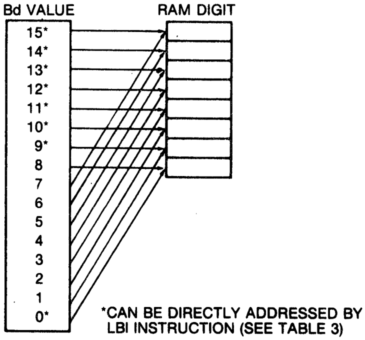


FIGURE 4 – RAM DIGIT ADDRESS TO PHYSICAL RAM DIGIT MAPPING

Internal Logic

The 4-bit A register (accumulator) is the source and destination register for most I/O, arithmetic, logic and data memory access operations. It can also be used to load the Bd portion of the B register, to load 4 bits of the 8-bit Q latch data, to input 4 bits of the 8-bit L I/O port data and to perform data exchanges with the SIO register.

A 4-bit adder performs the arithmetic and logic functions of the ET L 9410 / L 9411, storing its results in A. It also outputs a carry bit to the 1-bit C register, most often employed to indicate arithmetic overflow. The C register, in conjunction with the XAS instruction and the EN register, also serves to control the SK output. C can be outputted directly to SK or can enable SK to be a sync clock each instruction cycle time. (See XAS instruction and EN register description, below).

The G register contents are outputs to 4 general-purpose bidirectional I/O ports.

The Q register is an internal, latched, 8-bit register, used to hold data loaded from M and A, as well as 8-bit data from ROM. Its contents are output to the L I/O ports when the L drivers are enabled under program control. (See LEI instruction.)

The 8 L drivers, when enabled, output the contents of latched Q data to the L I/O ports. Also, the contents of L may be read directly into A and M. L I/O ports can be directly connected to the segments of a multiplexed LED display (using the LED Direct Drive output configuration option) with Q data being outputted to the Sa-Sg and decimal point segments of the display.

The SIO register functions as a 4-bit serial-in/serial-out shift register or as a binary counter depending on the contents of the EN register. (See EN register description, below.) Its contents can be exchanged with A, allowing it to input or output a continuous serial data stream. SIO may also be used to provide additional parallel I/O by connecting SO to external serial-in/parallel-out shift registers.

The XAS instruction copies C into the SKL Latch. In the counter mode, SK is the output of SKL in the shift register mode, SK outputs SKL ANDed with internal instruction cycle clock.

The EN register is an internal 4-bit register loaded under -program control by the LEI instruction. The state of each bit of this register selects or deselects the particular feature associated with each bit of the EN register (EN_3-EN_0).

1. The least significant bit of the enable register, EN_0, selects the SIO register as either a 4-bit shift register or a 4-bit binary counter. With EN_0 set, SIO is an asynchronous binary counter, *decrementing* its value by one upon each low-going pulse ("1" to "0") occurring on the SI input. Each pulse must be at least two instruction cycles wide. SK outputs the value of SKL. The SO output is equal to the value of EN_3. With EN_0 reset, SIO is a serial shift register shifting left each instruction cycle time. The data present at SI goes into the least significant bit of SIO. SO can be enabled to output the most significant bit of SIO each cycle time. (See 4 below.) The SK output becomes a logic-controlled clock.
2. EN_1 is not used. It has no effect on ET L 9410 / L 9411-operation.
3. With EN_2 set, the L drivers are enabled to output the data in Q to the L I/O ports. Resetting EN_2 disables the L drivers, placing the L I/O ports in a high-impedance input state.
4. EN_3, in conjunction with EN_0, affects the SO output. With EN_0 set (binary counter option selected) SO will output the value loaded into EN_3. With EN_0 reset (serial shift register option selected), setting EN_3 enables SO as the output of the SIO shift register, outputting serial shifted data each instruction time. Resetting EN_3 with the serial shift register option selected disables SO as the shift register output; data continues to be shifted through SIO and can be exchanged with A via an XAS instruction but SO remains reset to "0." Table I provides a summary of the modes associated with EN_3 and EN_0.

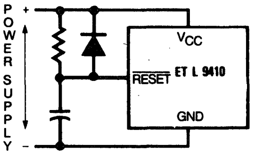
Initialization

The Reset Logic will initialize (clear) the device upon power-up if the power supply rise time is less than 1 ms and greater than 1 μs. If the power supply rise time is greater than 1 ms, the user must provide an external RC

Enable Register Modes – Bits EN_3 and EN_0

EN_3	EN_0	SIO	SI	SO	SK
0	0	Shift Register	Input to Shift Register	0	If SKL = 1, SK = Clock If SKL = 0, SK = 0
1	0	Shift Register	Input to Shift Register	Serial Out	If SKL = 1, SK = Clock If SKL = 0, SK = 0
0	1	Binary Counter	Input to Binary Counter	0	If SKL = 1, SK = 1 If SKL = 0, SK = 0
1	1	Binary Counter	Input to Binary Counter	1	If SKL = 1, SK = 1 If SKL = 0, SK = 0

network and diode to the $\overline{\text{RESET}}$ pin as shown below (Figure 5). The $\overline{\text{RESET}}$ pin is configured as a Schmitt trigger input. If not used it should be connected to V_{CC}. Initialization will occur whenever a logic "0" is applied to the $\overline{\text{RESET}}$ input, provided it stays low for at least three instruction cycle times.



RC≥5xPOWER SUPPLY RISE TIME

FIGURE 5 – POWER-UP CLEAR CIRCUIT

Upon initialization, the PC register is cleared to 0 (ROM address 0) and the A, B, C, D, EN, and G registers are cleared. The SK output is enabled as a SYNC output, providing a pulse each instruction cycle time. *Data Memory (RAM) is not cleared upon initialization.* The first instruction at address 0 must be a CLRA.

Oscillator

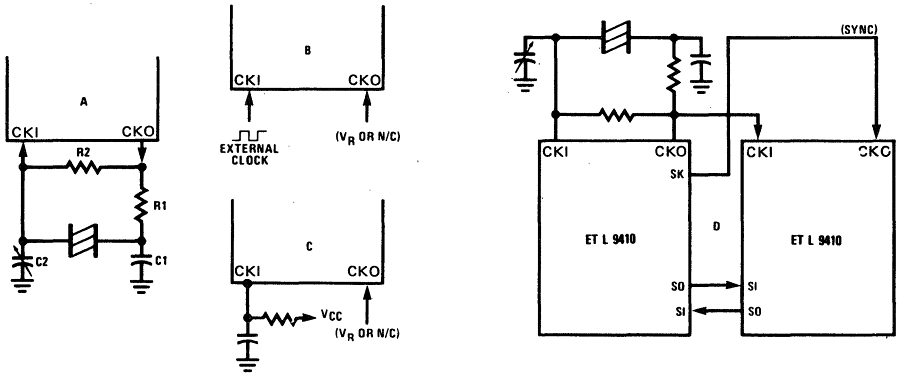
There are four basic clock oscillator configurations available as shown by Figure 6.

a. Resonator Controlled Oscillator. CKI and CKO are connected to an external ceramic resonator. The instruction cycle frequency equals the resonator frequency divided by 8. This is not available in the ET L 9411.

b. External Oscillator. CKI is an external clock input signal. The external frequency is divided by 8 to give the instruction frequency time. CKO is now available to be used as the RAM power supply (V_R), as a SYNC input, or no connection. (Note: No CKO on ET L 9411.

c. RC Controlled Oscillator. CKI is configured as a single pin RC controlled Schmitt trigger oscillator. The instruction cycle equals the oscillation frequency divided by 4. CKO is available as the RAM power supply (V_R) or no connection.

d. Externally Synchronized Oscillator. Intended for use in multi-COP systems, CKO is programmed to function as an input connected to the SK output of another chip operating at the same frequency (chip with L or C suffix) with CKI connected as shown. In this configuration, the SK output connected to CKO must provide a SYNC (instruction cycle) signal to CKO, thereby allowing synchronous data transfer between the chips using only the SI and SO serial I/O pins in conjunction with the XAS instruction. Note that on power-up SK is automatically enabled as a SYNC output. (See Functional Description, Initialization, above). This is not available in the ETL 9411.



Ceramic Resonator Oscillator

Resonator Value	Component Values			
	R1 (Ω)	R2 (Ω)	C1 (pF)	C2 (pF)
455 kHz	1k	1M	80	80

This circuit and these values are for indication only. As the oscillator characteristics are not guaranteed, please consider and examine the circuit constants carefully on your application.

RC Controlled Oscillator

R (kΩ)	C (pF)	Instruction Cycle Time in μs)
51	100	19 ± 15%
82	56	19 ± 13%

Note: 200 kΩ ⩾ R ⩾ 25 kΩ
360 pF ⩾ C ⩾ 50 pF

FIGURE 6 – ETL9410 / L 9411 - OSCILLATOR

CKO Pin Options

In a resonator controlled oscillator system, CKO is used as an output to the resonator network. As an option CKO can be a SYNC input as described above. As another option, CKO can be a RAM power supply pin (V_R), allowing its connection to a standby/backup power supply to maintain the integrity of RAM data with minimum power drain when the main supply is inoperative or shut down to conserve power. Using no connection option is appropriate in applications where the ET L 9410 system timing configuration does not require use of the CKO pin.

RAM Keep-Alive Option

Selecting CKO as the RAM power supply (V_R) allows the user to shut off the chip power supply (V_{CC}) and maintain data in the RAM. To insure that RAM data integrity is maintained, the following conditions must be met:

1. $\overline{\text{RESET}}$ must go low before V_{CC} goes below spec during power-off; V_{CC} must be within spec before $\overline{\text{RESET}}$ goes high on power-up.
2. During normal operation, V_R must be within the operating range of the chip with $(V_{CC} - 1) \leqslant V_R \leqslant V_{CC}$.
3. V_R must be $\geqslant$ 3.3V with V_{CC} off.

I/O Options

ET L 9410 / L 9411 inputs and outputs have the following optional configurations, illustrated in Figure 7.

a. Standard — an enhancement-mode device to ground in conjunction with a depletion-mode device to V_{CC}, compatible with LSTTL and CMOS input requirements. Available on SO, SK, and all D and G outputs.

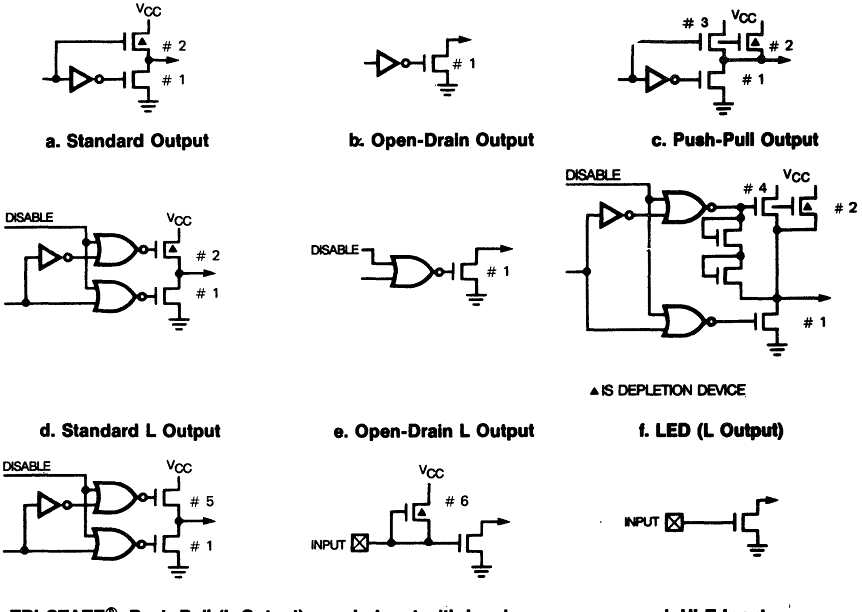
b. Open-Drain — an enhancement-mode device to ground only, allowing external pull-up as required by the user's application. Available on SO, SK, and all D and G outputs.

c. Push-Pull — an enhancement-mode device to ground in conjunction with a depletion-mode device paralleled by an enhancement-mode device to V_{CC}. This configuration has been provided to allow for fast rise and fall times when driving capacitive loads. Available on SO and SK outputs only.

d. Standard L — same as **a.**, but may be disabled. Available on L outputs only.

e. Open Drain L — same as **b.**, but may be disabled. Available on L outputs only.

f. LED Direct Drive — an enhancement mode device to ground and to V_{CC}, meeting the typical current sourcing requirements of the segments of an LED display. The sourcing device is clamped to limit current flow. These devices may be turned off under program control (see Functional Description, EN Register), placing the outputs in a high-impedance state to provide required LED segment blanking for a multiplexed display. Available on L outputs only.



a. Standard Output

b. Open-Drain Output

c. Push-Pull Output

▲ IS DEPLETION DEVICE

d. Standard L Output

e. Open-Drain L Output

f. LED (L Output)

g. TRI-STATE® Push-Pull (L Output)

h. Input with Load

i. Hi-Z Input

FIGURE 7 – INPUT AND OUTPUT CONFIGURATIONS

g. TRI-STATE® Push-Pull — an enhancement-mode device to ground and V_{CC}. These outputs are TRI-STATE® outputs, allowing for connection of these outputs to a data bus shared by other bus drivers. Available on L outputs only.

h. An on-chip depletion load device to V_{CC}.

i. A Hi-Z input which must be driven to a "1" or "0" by external components.

The above input and output configurations share common enhancement-mode and depletion-mode devices. Specifically, all configurations use one or more of six devices (numbered 1-6, respectively). Minimum and maximum current (I_{OUT} and V_{OUT}) curves are given in Figure 8 for each of these devices to allow the designer to effectivelv use these I/O configurations in designing a ETL 9410 / L 9411 svstem.

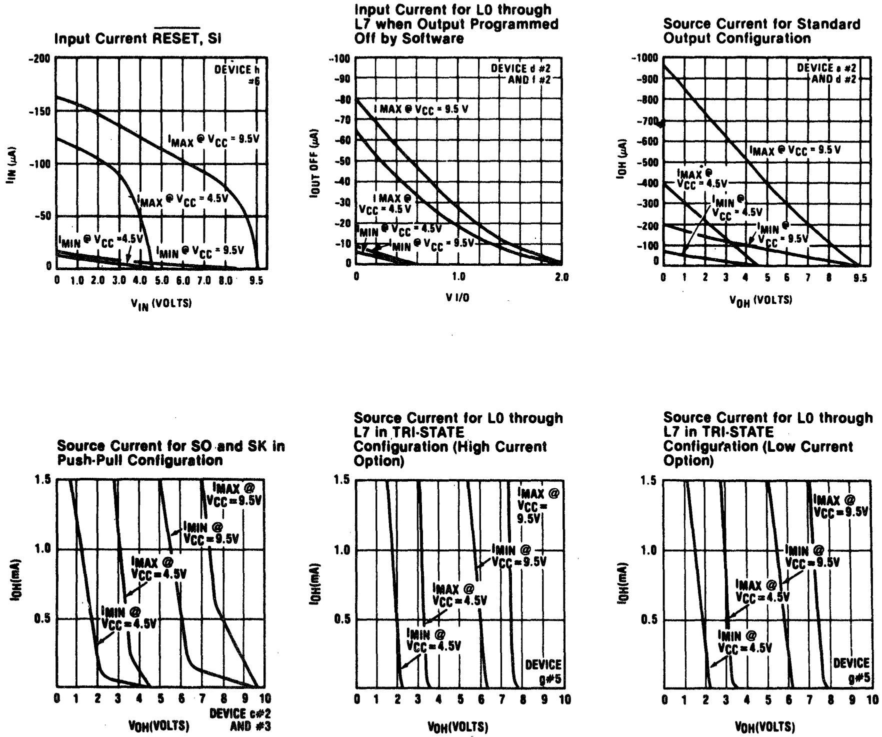
The SO, SK outputs can be configured as shown in **a.**, **b.**, or **c.** The D and G outputs can be configured as shown in **a.** or **b.** Note that when inputting data to the G ports, the G outputs should be set to "1." The L outputs can be configured as in **d.**, **e.**, **f.**, or **g.**

An important point to remember if using configuration **d.** or **f.** with the L drivers is that even when the L drivers are disabled, the depletion load device will source a small amount of current. (See Figure 8, device 2.) However, when the L port is used as input, the disabled depletion device CANNOT be relied on to source sufficient current to pull an input to a logic "1"

ET L 9411

If the ET L 9410 is bonded as a 20-pin device, it becomes the ET L 9411, illustrated in Figure 2, ET L 9410 / ET L 9411 Connection Diagrams. Note that the ET L 9411 does not contain D2, D3, G3, or CKO. Use of this option of course precludes use of D2, D3, G3, and CKO options. All other options are available for the ET L 9411.

TYPICAL PERFORMANCE CURVES


Input Current $\overline{RESET}$, SI

Input Current for L0 through L7 when Output Programmed Off by Software

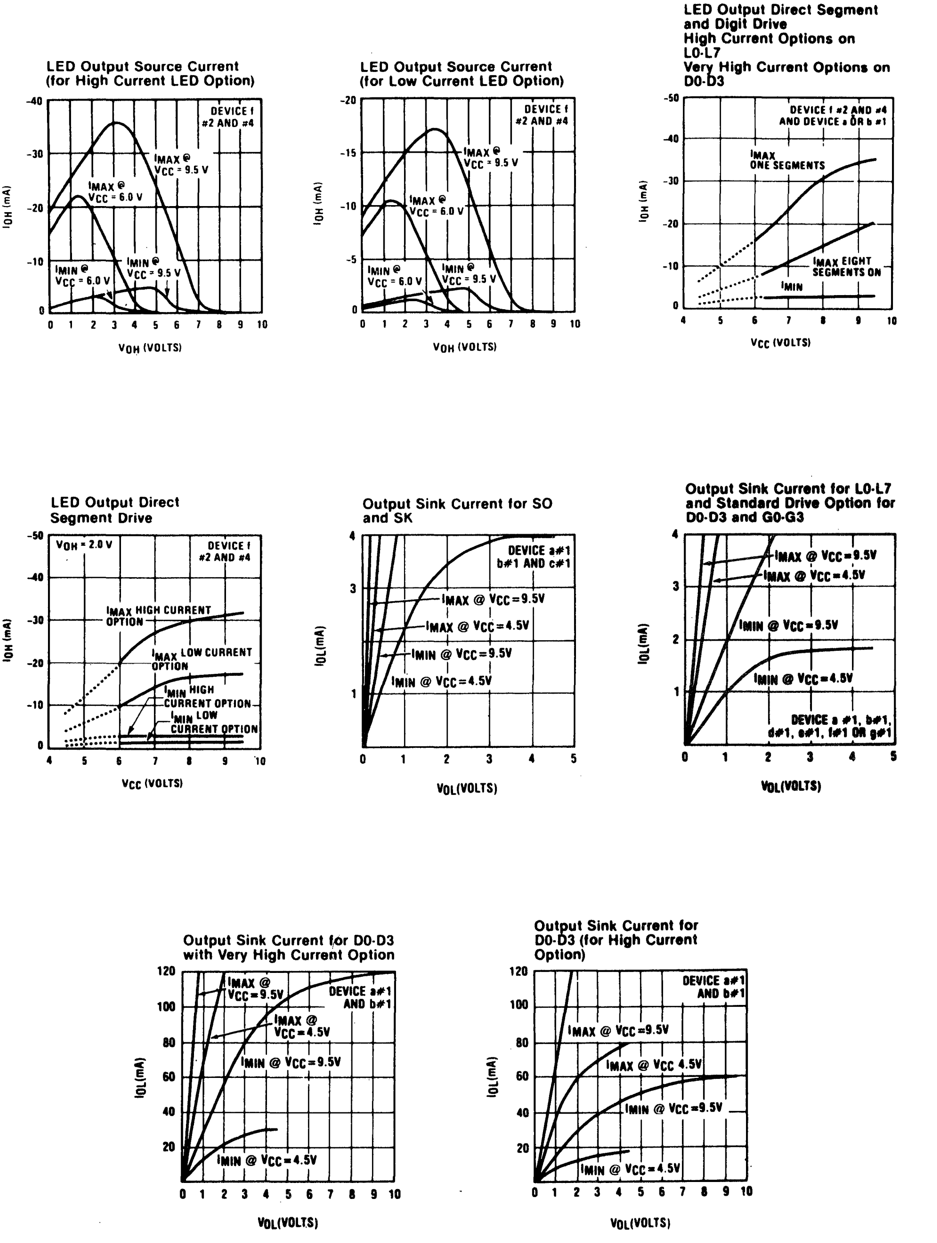
Source Current for Standard Output Configuration

Source Current for SO and SK in Push-Pull Configuration

Source Current for L0 through L7 in TRI-STATE Configuration (High Current Option)

Source Current for L0 through L7 in TRI-STATE Configuration (Low Current Option)

FIGURE 8a – ETL9410 / L 9411 - I/O DC CURRENT CHARACTERISTICS



LED Output Source Current (for High Current LED Option)

LED Output Source Current (for Low Current LED Option)

LED Output Direct Segment and Digit Drive High Current Options on L0-L7 Very High Current Options on D0-D3

LED Output Direct Segment Drive

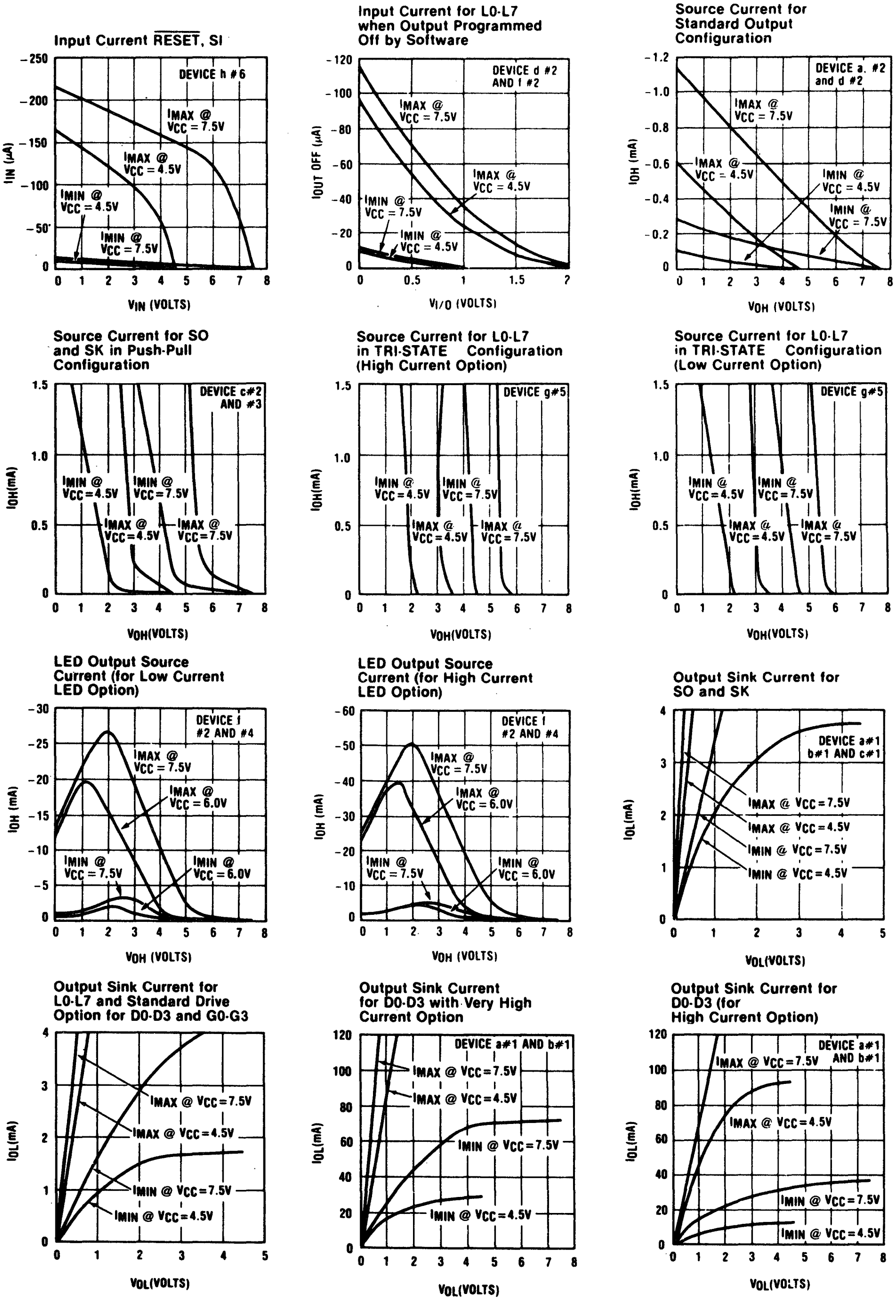
Output Sink Current for SO and SK

Output Sink Current for L0-L7 and Standard Drive Option for D0-D3 and G0-G3

Output Sink Current for D0-D3 with Very High Current Option

Output Sink Current for D0-D3 (for High Current Option)

FIGURE 8a – ETL9410 / L 9411 - I/O DC CURRENT CHARACTERISTICS



Input Current RESET, SI

Input Current for L0-L7 when Output Programmed Off by Software

Source Current for Standard Output Configuration

Source Current for SO and SK in Push-Pull Configuration

Source Current for L0-L7 in TRI-STATE Configuration (High Current Option)

Source Current for L0-L7 in TRI-STATE Configuration (Low Current Option)

LED Output Source Current (for Low Current LED Option)

LED Output Source Current (for High Current LED Option)

Output Sink Current for SO and SK

Output Sink Current for L0-L7 and Standard Drive Option for D0-D3 and G0-G3

Output Sink Current for D0-D3 with Very High Current Option

Output Sink Current for D0-D3 (for High Current Option)

FIGURE 8b – ETL9310 / L9311 - INPUT/OUTPUT CHARACTERISTICS

ETL 9410/L 9411 - INSTRUCTION SET

Table 2 is a symbol table providing internal architecture, instruction operand and operational symbols used in the instruction set table.

Table 3 provides the mnemonic, operand, machine, code, data flow, skip conditions and description associated with each instruction in the ET L 9410 / L 9411- instruction set.

TABLE 2 – ETL9410 / L9411 - INSTRUCTION SET TABLE SYMBOLS

Symbol	Definition
INTERNAL ARCHITECTURE SYMBOLS	
A	4-bit Accumulator
B	6-bit RAM Address Register
Br	Upper 2 bits of B (register address)
Bd	Lower 4 bits of B (digit address)
C	1-bit Carry Register
D	4-bit Data Output Port
EN	4-bit Enable Register
G	4-bit Register to latch data for G I/O Port
L	8-bit TRI-STATE I/O Port
M	4-bit contents of RAM Memory pointed to by B Register
PC	9-bit ROM Address Register (program counter)
Q	8-bit Register to latch data for L I/O Port
SA	9-bit Subroutine Save Register A
SB	9-bit Subroutine Save Register B
SIO	4-bit Shift Register and Counter
SK	Logic-Controlled Clock Output

Symbol	Definition
INSTRUCTION OPERAND SYMBOLS	
d	4-bit Operand Field, 0 - 15 binary (RAM Digit Select)
r	2-bit Operand Field, 0 - 3 binary (RAM Register Select)
a	9-bit Operand Field, 0 - 511 binary (ROM Address)
y	4-bit Operand Field 0 - 15 binary (Immediate Data)
RAM(s)	Contents of RAM location addressed by s
ROM(t)	Contents of ROM location addressed by t
OPERATIONAL SYMBOLS	
+	Plus
—	Minus
→	Replaces
↔	Is exchanged with
=	Is equal to
A	The one's complement of A
⊕	Exclusive-OR
:	Range of values

TABLE 3 – ETL9410 / L9411 - INSTRUCTION SET

Mnemonic	Operand	Hex Code	Machine Language Code (Binary)	Data Flow	Skip Conditions	Description
ARITHMETIC INSTRUCTIONS						
ASC		30	|0 0 1 1|0 0 0 0|	A + C + RAM(B) → A Carry → C	Carry	Add with Carry, Skip on Carry
ADD		31	|0 0 1 1|0 0 0 1|	A + RAM(B) → A	None	Add RAM to A
AISC	y	5-	|0 1 0 1| y |	A + y → A	Carry	Add Immediate, Skip on Carry (y ≠ 0)
CLRA		00	|0 0 0 0|0 0 0 0|	0 → A	None	Clear A
COMP		40	|0 1 0 0|0 0 0 0|	Ā → A	None	One's complement of A to A
NOP		44	|0 1 0 0|0 1 0 0|	None	None	No Operation
RC		32	|0 0 1 1|0 0 1 0|	"0" → C	None	Reset C
SC		22	|0 0 1 0|0 0 1 0|	"1" → C	None	Set C
XOR		02	|0 0 0 0|0 0 1 0|	A⊕RAM(B) → A	None	Exclusive-OR RAM with A

TABLE 3 – ETL9410 / L9411 - INSTRUCTION SET (continued)

Mnemonic	Operand	Hex Code	Machine Language Code (Binary)	Data Flow	Skip Conditions	Description
TRANSFER OF CONTROL INSTRUCTIONS						
JID		FF	1111 1111	ROM (PC_8,A,M) → $PC_{7:0}$	None	Jump Indirect (Note 2)
JMP	a	6- --	0110 000 a_8 $a_{7:0}$	a → PC	None	Jump
JP	a	-- --	1 $a_{6:0}$ (pages 2,3 only) or 11 $a_{5:0}$ (all other pages)	a → $PC_{6:0}$ a → $PC_{5:0}$	None	Jump within Page (Note 3)
JSRP	a	--	10 $a_{5:0}$	PC + 1 → SA → SB 010 → $PC_{8:6}$ a → $PC_{5:0}$	None	Jump to Subroutine Page (Note 4)
JSR	a	6- --	0110 100 a_8 $a_{7:0}$	PC + 1 → SA → SB a → PC	None	Jump to Subroutine
RET		48	0100 1000	SB → SA → PC	None	Return from Subroutine
RETSK		49	0100 1001	SB → SA → PC	Always Skip on Return	Return from Subroutine then Skip
MEMORY REFERENCE INSTRUCTIONS						
CAMQ		33 3C	0011 0011 0011 1100	A → $Q_{7:4}$ RAM(B) → $Q_{3:0}$	None	Copy A, RAM to Q
LD	r	-5	00 r 0101	RAM(B) → A Br ⊕ r → Br	None	Load RAM into A, Exclusive-OR Br with r
LQID		BF	1011 1111	ROM(PC_8,A,M) → Q SA → SB	None	Load Q Indirect (Note 2)
RMB	0 1 2 3	4C 45 42 43	0100 1100 0100 0101 0100 0010 0100 0011	0 → $RAM(B)_0$ 0 → $RAM(B)_1$ 0 → $RAM(B)_2$ 0 → $RAM(B)_3$	None	Reset RAM Bit
SMB	0 1 2 3	4D 47 46 4B	0100 1101 0100 0111 0100 0110 0100 1011	1 → $RAM(B)_0$ 1 → $RAM(B)_1$ 1 → $RAM(B)_2$ 1 → $RAM(B)_3$	None	Set RAM Bit
STII	y	7-	0111 y	y → RAM(B) Bd + 1 → Bd	None	Store Memory Immediate and Increment Bd
X	r	-6	00 r 0110	RAM(B) ⟷ A Br ⊕ r → Br	None	Exchange RAM with A, Exclusive-OR Br with r
XAD	3,15	23 BF	0010 0011 1011 1111	RAM(3,15) ⟷ A	None	Exchange A with RAM (3,15)
XDS	r	-7	00 r 0111	RAM(B) ⟷ A Bd – 1 → Bd Br ⊕ r → Br	Bd decrements past 0	Exchange RAM with A and Decrement Bd, Exclusive-OR Br with r
XIS	r	-4	00 r 0100	RAM(B) ⟷ A Bd + 1 → Bd Br ⊕ r → Br	Bd increments past 15	Exchange RAM with A and Increment Bd, Exclusive-OR Br with r

TABLE 3 – ETL9410 / L9411 - INSTRUCTION SET (continued)

Mnemonic	Operand	Hex Code	Machine Language Code (Binary)	Data Flow	Skip Conditions	Description
REGISTER REFERENCE INSTRUCTIONS						
CAB		50	0101 0000	A → Bd	None	Copy A to Bd
CBA		4E	0100 1110	Bd → A	None	Copy Bd to A
LBI	r.d	--	00 r (d – 1) (d = 0, 9:15)	r.d → B	Skip until not a LBI	Load B Immediate with r.d (Note 5)
LEI	y	33 6-	0011 0011 0110 y	y → EN	None	Load EN Immediate (Note 6)
TEST INSTRUCTIONS						
SKC		20	0010 0000		C = "1"	Skip if C is True
SKE		21	0010 0001		A = RAM(B)	Skip if A Equals RAM
SKGZ		33 21	0011 0011 0010 0001		$G_{3:0}$ = 0	Skip if G is Zero (all 4 bits)
SKGBZ		33	0011 0011	1st byte		Skip if G Bit is Zero
	0	01	0000 0001	2nd byte	G_0 = 0	
	1	11	0001 0001		G_1 = 0	
	2	03	0000 0011		G_2 = 0	
	3	13	0001 0011		G_3 = 0	
SKMBZ	0	01	0000 0001		$RAM(B)_0$ = 0	Skip if RAM Bit is Zero
	1	11	0001 0001		$RAM(B)_1$ = 0	
	2	03	0000 0011		$RAM(B)_2$ = 0	
	3	13	0001 0011		$RAM(B)_3$ = 0	
INPUT/OUTPUT INSTRUCTIONS						
ING		33 2A	0011 0011 0010 1010	G → A	None	Input G Ports to A
INL		33 2E	0011 0011 0010 1110	$L_{7:4}$ → RAM(B) $L_{3:0}$ → A	None	Input L Ports to RAM, A
OBD		33 3E	0011 0011 0011 1110	Bd → D	None	Output Bd to D Outputs
OMG		33 3A	0011 0011 0011 1010	RAM(B) → G	None	Output RAM to G Ports
XAS		4F	0100 1111	A ←→ SIO, C → SKL	None	Exchange A with SIO (Note 2)

Note 1 : All subscripts for alphabetical symbols indicate bit numbers unless explicitly defined (e.g., Br and Bd are explicitly defined). Bits are numbered 0 to N where 0 signifies the least significant bit (low-order, right-most bit). For example, A_3 indicates the most significant (left-most) bit of the 4-bit A register.

Note 2 : For additional information on the operation of the XAS, JID, and LQID instructions, see below.

Note 3 : The JP instruction allows a jump, while in subroutine pages 2 or 3, to any ROM location within the two-page boundary of pages 2 or 3. The JP instruction, otherwise, permits a jump to a ROM location within the current 64-word page. JP may not jump to the last word of a page.

Note 4 : A JSRP transfers program control to subroutine page 2 (0010 is loaded into the upper 4 bits of P). A JSRP may not be used when in pages 2 or 3. JSRP may not jump to the last word in page 2.

Note 5 : The machine code for the lower 4 bits of the LBI instruction equals the binary value of the "d" data *minus 1*, e.g., to load the lower four bits of B (Bd) with the value 9 (1001_2), the lower 4 bits of the LBI instruction equal 8 (1000_2). To load 0, the lower 4 bits of the LBI instruction should equal 15 (1111_2).

Note 6 : Machine code for operand field y for LEI instruction should equal the binary value to be latched into EN, where a "1" or "0" in each bit of EN corresponds with the selection or deselection of a particular function associated with each bit. (See Functional Description, EN Register).

The following information is provided to assist the user in uderstanding the operation of several unique instructions and to provide notes useful to programmers in writing ET L 9410 / L 9411.

XAS Instruction

XAS (Exchange A with SIO) exchanges the 4-bit contents of the accumulator with the 4-bit contents of the SIO register. The contents of SIO will contain serial-in/serial-out shift register or binary counter data, depending on the value of the EN register. An XAS instruction will also affect the SK output. (See Functional Description, EN Register, above.) If SIO is selected as a shift register, an XAS instruction must be performed once every 4 instruction cycles to effect a continuous data stream.

JID Instruction

JID (Jump Indirect) is an indirect addressing instruction, transferring program control to a new ROM location pointed to indirectly by A and M. It loads the lower 8 bits of the ROM address register PC with the *contents* of ROM addressed by the 9-bit word, PC_8, A, M. PC_8 is not affected by this instruction.

Note that JID requires 2 instruction cycles to execute.

LQID Instruction

LQID (Load Q Indirect) loads the 8-bit Q register with the contents of ROM pointed to by the 9-bit word PC_8, A, M. LQID can be used for table lookup or code conversion such as BCD to seven-segment. The LQID instruction "pushes" the stack (PC + 1 → SA → SB) and replaces the least significant 8 bits of PC as follows: A → $PC_{7:4}$, RAM(B) → $PC_{3:0}$, leaving PC_8 unchanged. The ROM data pointed to by the new address is fetched and loaded into the Q latches. Next, the stack is "popped" (SB → SA → PC), restoring the saved value of PC to continue sequential program execution. Since LQID pushes SA → SB, the previous contents of SB are lost. Also, when LQID pops the stack, the previously pushed contents of SA are left in SB. The net result is that the contents of SA are placed in SB (SA → SB). Note that LQID takes two instruction cycle times to execute.

Instruction Set Notes

a. The firs word of a ET L 9410 / ET L 9411 program (ROM address 0) must be a CLRA (Clear A) instruction.

b. Although skipped instructions are not executed, one instruction cycle time is devoted to skipping each byte of the skipped instruction. Thus all program paths except JID and LQID take the same number of cycle times whether instructions are skipped or executed. JID and LQID instructions take 2 cycles if executed and 1 cycle if skipped.

c. The ROM is organized into 8 pages of 64 words each. The Program Counter is a 9-bit binary counter, and will count through page boundaries. If a JP, JSRP, JID or LQID instruction is located in the last word of a page, the instruction operates as if it were in the next page. For example: a JP located in the last word of a page will jump to a location in the next page. Also, a LQID or JID located in the last word of page 3 or 7 will access data in the next group of 4 pages.

OPTION LIST

The ET L 9410 / ET L 9411 - mask programmable options are assigned numbers which correspond with the ET L 9410 pins.

The following is a list of ET L 9410 options. When specifying a ET L 9411 chip, Option 2 must be set to 3, Options 20, 21, and 22 to 0. The options are programmed at the same time as the ROM pattern to provide the user with the hardware flexibility to interface to various I/O components using little or no external circuitry.

Option 1 = 0: Ground Pin — no options available

Option 2: CKO Output (no option available for ET L 9411)
= 0: Clock output to ceramic resonator
= 1: Pin is RAM power supply (V_R) input
= 2: Multi-COP SYNC input
= 3: No connection

Option 3: CKI Input
= 0: Oscillator input divided by 8 (500 kHz max.)
= 1: Single-pin RC controlled oscillator divided by 4
= 2: External Schmitt trigger level clock divided by 4

Option 4: $\overline{\text{RESET}}$ Input
= 0: Load device to V_{CC}
= 1: Hi-Z input

Option 5: L_7 Driver
= 0: Standard output
= 1: Open-drain output
= 2: High current LED direct segment drive output
= 3: High current TRI-STATE® push-pull output
= 4: Low-current LED direct segment drive output
= 5: Low-current TRI-STATE® push-pull output

Option 6: L_6 Driver
same as Option 5

Option 7: L_5 Driver
same as Option 5

Option 8: L_4 Driver
same as Option 5

Option 9: V_{CC} Pin
= 0: 4.5V to 6.3V operation
= 1: 4.5V to 9.5V operation

Option 10: L_3 Driver
same as Option 5

Option 11: L_2 Driver
same as Option 5

Option 12: L_1 Driver
same as Option 5

Option 13: L_0 Driver
same as Option 5

Option 14: SI Input
= 0: load device to V_{CC}
= 1: HI-Z input

Option 15: SO Driver
= 0: Standard Output
= 1: Open-drain output
= 2: Push-pull output

Option 16: SK Driver
same as Option 15

Option 17: G_0 I/O Port
= 0: Standard output
= 1: Open-drain output

Option 18: G_1 I/O Port
same as Option 17

Option 19: G_2 I/O Port
same as Option 17

Option 20: G_3 I/O Port (no option available for ET L 9411)
same as Option 17

Option 21: D_3 Output (no option available for ET L 9411)
= 0: Very-high sink current standard output
= 1: Very-high sink current open-drain output
= 2: High sink current standard output
= 3: High sink current open-drain output
= 4: Standard LSTTL output (fanout = 1)
= 5: Open-drain LSTTL output (fanout = 1)

Option 22: D_2.Output (no option available for ET L 9411)
same as Option 21

Option 23: D_1 Output
same as Option 21

Option 24: D_0 Output
same as Option 21

Option 25: L Input Levels
= 0: Standard TTL input levels ("0" = 0.8V, "1" = 2.0V)
= 1: Higher voltage input levels ("0" = 1.2V, "1" = 3.6V)

Option 26: G Input Levels
same as Option 25

Option 27: SI Input Levels
same as Option 25

Option 28 : Bonding
= 0: ET L 9410 (24-pin device)
= 1: ET L 9411 (20-pin device)
= 2: Both 24- and 20-pin versions

Test Mode (Non-Standard Operation)

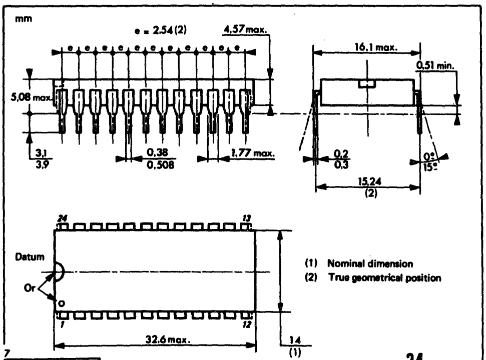
The SO output has been configured to provide for standard test procedures for the custom-programmed ET L 9410. With SO forced to logic "1", two test modes are provided, depending upon the value of SI:

a. RAM and Internal Logic Test Mode (SI = 1)

b. ROM Test Mode (SI = 0)

These special test modes should not be employed by the user; they are intended for manufacturing test only.

PHYSICAL DIMENSIONS

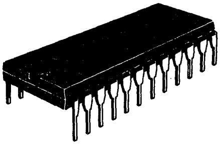


(1) Nominal dimension
(2) True geometrical position

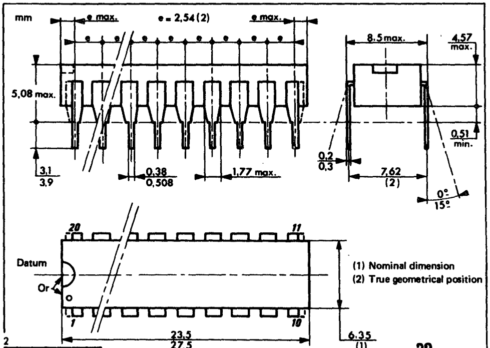
24 pins

DIN				
A51D CEI	D.A.T.A.	JEDEC	F 118 SITELESC	CB-68

CB-68



N SUFFIX
PLASTIC PACKAGE

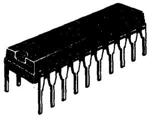


(1) Nominal dimension
(2) True geometrical position

20 pins

DIN				
CEI	D.A.T.A.	JEDEC	SITELESC	CB-194

CB-194



N SUFFIX
PLASTIC PACKAGE

These specifications are subject to change without notice.
Please inquire with our sales offices about the availability of the different products.

Printed in France

Ref.DSETL.93941011

HOMSON
COMPONENTS
EMICONDUCTEURS

ETL9413 • ETL9313

SINGLE CHIP MICROCONTROLLERS

The ETL9413 and ETL9313 Single-Chip N-Channel Microcontrollers are fully compatible with the COPS ® family, fabricated using N-channel, silicon gate MOS technology. The Controller Oriented Processors are complete microcomputers containing all system timing, internal logic, ROM, RAM and I/O necessary to implement dedicated control functions in a variety of applications. Features include single supply operation, a variety of output configuration options, with an instruction set, internal architecture and I/O scheme designed to facilitate keyboard input, display output and BCD data manipulation. They are an appropriate choice for use in numerous human interface control environments. Standard test procedures and reliable high-density fabrication techniques provide the medium to large volume customers with a customized Controller Oriented Processor at a low end-product cost.

The ETL9413 is identical to the ETL9410, but with 15 I/O lines instead of 19.

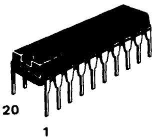
The ETL9313 is exact functional equivalent but extended temperature version of ETL9413.

- Low cost
- Powerful instruction set
- Ceramic or crystal oscillator input/output
- 512 × 8 ROM. 32 × 4 RAM
- Two-level subroutine stack
- 16μs instruction time
- Single supply operation (4.5 - 6.3V)
- Low current drain (6mA max.)
- Internal binary counter register with serial I/O capability
- General purpose and TRI-STATE® outputs
- LSTTL/CMOS compatible in and out
- Direct drive of LED digit and segment lines
- Software/hardware compatible with other members of ETL9313 family
- Extended temperature range device ETL9310 ETL9311 (-40° C to + 85° C)
- Wider supply range (4.5 - 9.5V) optionally available

NMOS

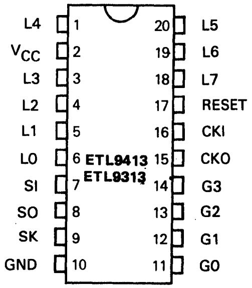
CASE

ETL9413 / ETL9313



N SUFFIX
PLASTIC PACKAGE

PIN ASSIGMENT



JULY 1986 1/3

OPTION LIST

The ETL9413/ETL9313 - mask programmable options are assigned numbers which correspond with the ETL9410 pins.

The options are programmed at the same time as the ROM pattern to provide the user with the hardware flexibility to interface to various I/O components using little or no external circuitry.

Option 1 = 0: Ground Pin — no options available

Option 2: CKO Output
= 0: Clock output to ceramic resonator
= 1: Pin is RAM power supply (V_R) input
= 2: Multi-COP SYNC input
= 3: No connection

Option 3: CKI Input
= 0: Oscillator input divided by 8 (500 kHz max.)
= 1: Single-pin RC controlled oscillator divided by 4
= 2: External Schmitt trigger level clock divided by 4

Option 4: $\overline{\text{RESET}}$ Input
= 0: Load device to V_{CC}
= 1: Hi-Z input

Option 5: L_7 Driver
= 0: Standard output
= 1: Open-drain output
= 2: High current LED direct segment drive output
= 3: High current TRI-STATE® push-pull output
= 4: Low-current LED direct segment drive output
= 5: Low-current TRI-STATE® push-pull output

Option 6: L_6 Driver
same as Option 5

Option 7: L_5 Driver
same as Option 5

Option 8: L_4 Driver
same as Option 5

Option 9: V_{CC} Pin
= 0: 4.5 V to 6.3 V operation
= 1: 4.5 V to 9.5 V operation

Option 10: L_3 Driver
same as Option 5

Option 11: L_2 Driver
same as Option 5

Option 12: L_1 Driver
same as Option 5

Option 13: L_0 Driver
same as Option 5

Option 14: SI Input
= 0: load device to V_{CC}
= 1: HI-Z input

Option 15: SO Driver
= 0: Standard Output
= 1: Open-drain output
= 2: Push-pull output

Option 16: SK Driver
same as Option 15

Option 17: G_0 I/O Port
= 0: Standard output
= 1: Open-drain output

Option 18: G_1 I/O Port
same as Option 17

Option 19: G_2 I/O Port
same as Option 17

Option 20: G_3 I/O Port
same as Option 17

Option 21: must be set to zero

Option 22: same as Option 21

Option 23: same as Option 21

Option 24: same as Option 21

Option 25: L Input Levels
= 0: Standard TTL input levels ("0" = 0.8 V, "1" = 2.0 V)
= 1: Higher voltage input levels ("0" = 1.2 V, "1" = 3.6 V)

Option 26: G Input Levels
same as Option 25

Option 27: SI Input Levels
same as Option 25

Option 28: Bonding, must be set to 3 for ETL9413/-9313

Test Mode (Non-standard Operation)

The SO output has been configured to provide for standard test procedures for the custom-programmed ETL9413. With SO forced to logic "1", two test modes are provided, depending upon the value of SI:

a. RAM and Internal Logic Test Mode (SI = 1)
b. ROM Test Mode (SI = 0)

These special test modes should not be employed by the user; they are intended for manufacturing test only.

The ETL9413 is the same chip as the ETL9410 but in a 20-pin package. The four D ports are not bonded.

Consequently, the electrical characteristics, the functional description and the instruction set are identical to the ETL9410 but without the D outputs. Please, refer to the ETL9410/9411 data sheet.

Note that the OBD instruction can be used but the D outputs are not available.

These specifications are subject to change without notice.
Please inquire with our sales offices about the availability of the different products.

THOMSON
COMPONENTS
SEMICONDUCTEURS

ET9420/21/22 • ET9320/21/22

SINGLE CHIP MICROCONTROLLERS

The ET9420/9421/9422, ET9320/9321 and 9322 Single-Chip N-Channel Microcontrollers are fully compatible with the COPS ® family, fabricated using N-channel, silicon gate XMOS technology. They are complete microcomputers containing all system timing, internal logic, ROM, RAM and I/O necessary to implement dedicated control functions in a variety of applications. Features include single supply operation, a variety of output configuration options, with an instruction set, internal architecture and I/O scheme designed to facilitate keyboard input, display output and BCD data manipulation. The ET9421 is identical to the ET9420, except with 19 I/O lines instead of 23 ; the ET9422 has 15 I/O lines. They are an appropriate choice for use in numerous human interface control environments. Standard test procedures and reliable high-density fabrication techniques provide the medium to large volume customers with a customized Controller Oriented Processor at a low end-product cost.

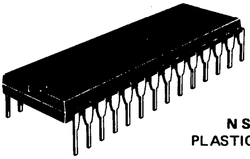
The ET9320 is the extended temperature range version of the ET9420 (likewise the ET9321 and ET9322 are the extended temperature range versions of the ET9421/ET9422). The ET9320/9321/9322 are exact functional equivalents of the ET9420/9421/9422.

- Low cost
- Powerful instruction set
- 1 K x 8 ROM, 64 x 4 RAM
- 23 I/O lines (ET9420, ET9320)
- True vectored interrupt, plus restart
- Three-level subroutine stack
- 4.0 μs instruction time
- Single supply operation
- Internal time-base counter for real-time processing
- Internal binary counter register with MICROWIRE® compatible serial I/O capability
- General purpose and TRI-STATE® outputs
- TTL/CMOS compatible in and out
- LED direct drive outputs
- MICROBUS ® compatible
- Software/hardware compatible with other members of ET9400
- Extended temperature range device ET9320/9321/9322 (-40° C to + 85° C)

NMOS

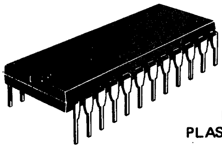
CASES

ET9420 • ET9320



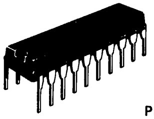
N SUFFIX
PLASTIC PACKAGE

ET9421 • ET9321



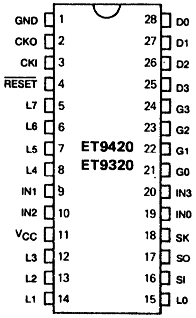
N SUFFIX
PLASTIC PACKAGE

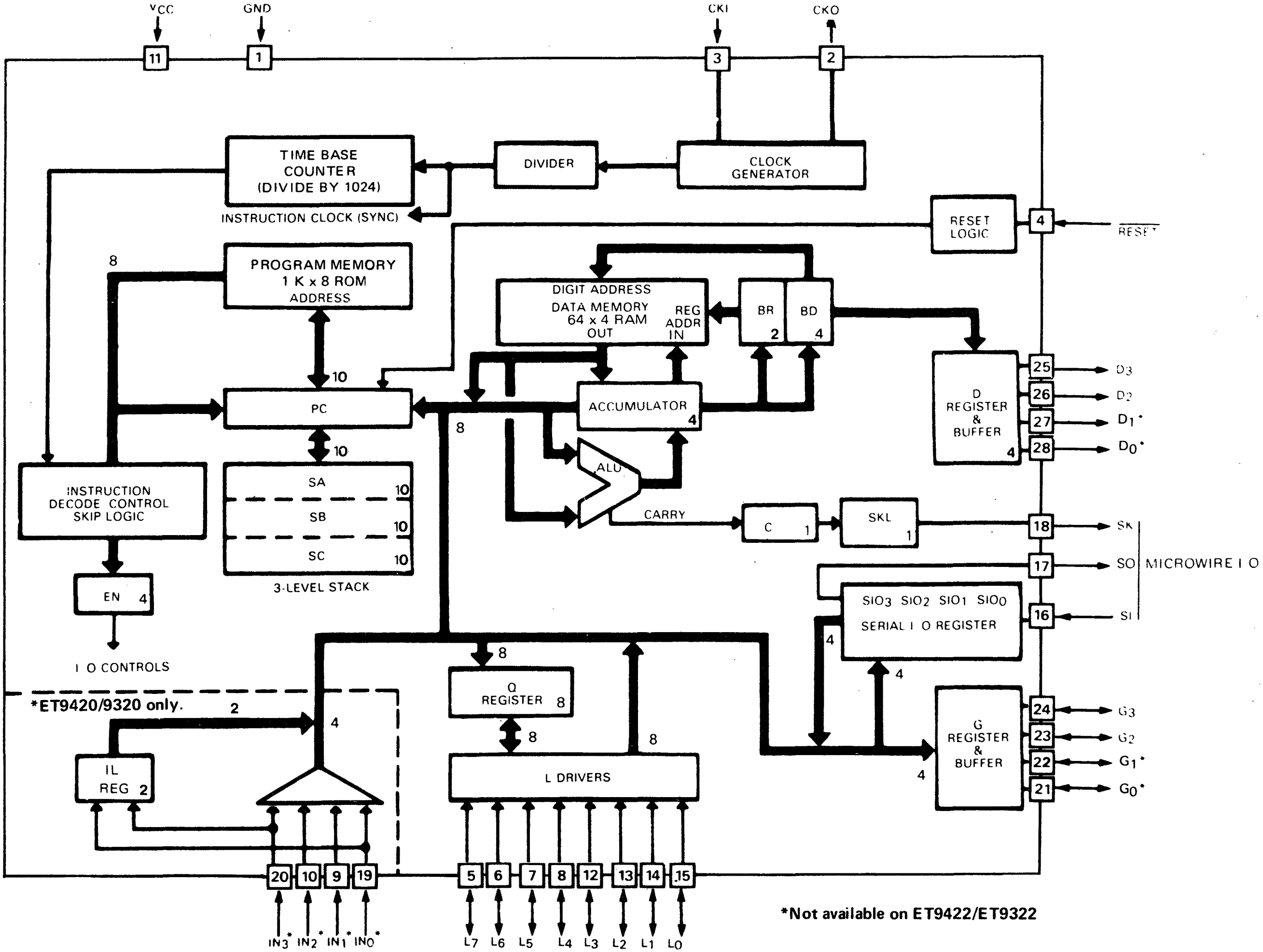
ET9422 • ET9322



N SUFFIX
PLASTIC PACKAGE

PIN ASSIGNMENT





*Not available on ET9422/ET9322

FIGURE 1 – BLOCK DIAGRAM (28-pin version)

ET9420/9421/9422 and ET9320/9321/9322

ABSOLUTE MAXIMUM RATINGS

Voltage at Any Pin	−0.3V to +7V	Package Power Dissipation	750 mW at 25°C
Operating Temperature Range		24 and 28 pin	400 mW at 70°C
ET9420/9421/9422	0°C to 70°C		250 mW at 85°C
ET9320/9321/9322	−40°C to +85°C	Package Power Dissipation	650 mW at 25°C
Storage Temperature Range	−65°C to +150°C	20 pin	300 mW at 70°C
Total Sink Current	75 mA		200 mW at 85°C
Total Source Current	95 mA	Lead Temperature (soldering, 10 sec.)	300°C

Absolute maximum ratings indicate limits beyond which damage to the device may occur. DC and AC electrical specifications are not ensured when operating the device at absolute maximum ratings.

ET9420/9421/9422

DC ELECTRICAL CHARACTERISTICS 0°C ≤ T_A ≤ +70°C,4.5C ≤ V_{CC} ≤ 6.3V (Unless otherwise specified)

Parameter	Conditions	Min.	Max.	Units
Operation Voltage		4.5	6.3	V
Power Supply Ripple	Peak to Peak (Note 3)	–	0.4	V
Supply Current	Outputs Open	–	38	mA
Supply Current	Outputs Open, V_{CC} = 5V, T_A = 25°C	–	30	mA
Input Voltage Levels				
CKI Input Levels				
Crystal Input				
Logic High	V_{CC} = Max.	3.0	–	
Logic High	V_{CC} = 5V ±5%	2.0	–	V
Logic Low		−0.3	0.4	V
TTL Input	V_{CC} = 5V ± 5%			
Logic High		2.0	–	V
Logic Low		−0.3	0.8	V
Schmitt Trigger Inputs				
RESET, CKI (÷4)				
Logic High		0.7 V_{CC}	–	V
Logic Low		−0.3	0.6	V
SO Input Level (Test Mode)		2.0	3.0	V
All Other Inputs				
Logic High	V_{CC} = Max.	3.0	–	V
Logic High	V_{CC} = 5V ± 5%	2.0	–	V
Logic Low		−0.3	0.8	V
Input Levels High Trip Option				
Logic High		3.6	–	V
Logic Low		−0.3	1.2	V
Input Load Source Current	V_{CC} = 5V, V_{IN} = 0V			
CKO		−4	−800	µA
All Others		−100	−800	µA
Input Capacitance		–	7	pF
Hi-Z Input Leakage	V_{CC} = 5V	−1	+1	µA
Output Voltage levels				
Standard Outputs				
TTL Operation	V_{CC} = 5V ± 5%			
Logic High	I_{OH} = −100µA	2.4	–	V
Logic Low	I_{OL} = 1.6 mA	−0.3	0.4	V
CMOS Operation				
Logic High	I_{OH} = −10µA	V_{CC} − 1	–	V

ET9420/9421/9422

DC ELECTRICAL CHARACTERISTICS (Cont'd) 0°C $\leqslant T_A \leqslant$ +70°C, 4.5V $\leqslant V_{CC} \leqslant$ 6.3V (Unless otherwise specified)

Parameter	Conditions	Min.	Max.	Units
Output Current Levels				
LED Direct Drive Output	$V_{CC} = 6V$			
Logic High	$V_{OH} = 2.0V$	2.5	14	mA
CKI Sink Current (R/C Option)	$V_{IN} = 3.5V$	2	–	mA
CKO (RAM Supply Current)	$V_R = 3.3V$	–	3	mA
TRI-STATE® or Open Drain Leakage Current	$V_{CC} = 5V$	−2.5	+2.5	µA
Output Current Levels				
Output Sink Current (I_{OL})	$V_{CC} = 6.3V$, $V_{OL} = 0.4V$	−2.0	–	mA
	$V_{CC} = 4.5V$, $V_{OL} = 0.4V$	−1.0	–	mA
Output Source Current (I_{OH})				
Standard Configuration				
All Outputs	$V_{CC} = 6.3V$, $V_{OH} = 3.0V$	−200	−900	µA
	$V_{CC} = 4.5V$, $V_{OH} = 2.0V$	−100	−500	µA
Push-Pull Configuration				
SO, SK Outputs	$V_{CC} = 6.3V$, $V_{OH} = 3.0V$	−1.0	–	mA
	$V_{CC} = 4.5V$, $V_{OH} = 2.0V$	−0.4	–	mA
TRI-STATE Configuration				
L_0-L_7 Outputs	$V_{CC} = 6.3V$, $V_{OH} = 3.0V$	−2.0	–	mA
	$V_{CC} = 4.5V$, $V_{OH} = 2.0V$	−0.8	–	mA
LED Configuration				
L_0-L_7 Outputs	$V_{CC} = 6.3V$, $V_{OH} = 3.0V$	−1.0	–	mA
	$V_{CC} = 4.5V$, $V_{OH} = 2.0V$	−0.5	–	mA
Allowable Sink Current				
Per Pin (L, D, G)		–	10	mA
Per Pin (All Others)		–	2	mA
Per Port (L)		–	16	mA
Per Port (D, G)		–	10	mA
Allowable Source Current				
Per Pin (L)		–	−15	mA
Per Pin (All Others)		–	−1.5	mA

ET9320/9321/9322

DC ELECTRICAL CHARACTERISTICS - 40° ≤ T_A ≤ +85°C, 4.5V ≤ V_{CC} ≤ 5.5V (Unless otherwise specified).

Parameter	Conditions	Min.	Max.	Units
Operation Voltage		4.5	5.5	V
Power Supply Ripple	Peak to Peak (Note 3)	–	0.4	V
Supply Current	$T_A = -40°C$, Outputs Open	–	40	mA
Input Voltage Levels				
CKI Input Levels				
Crystal Input				
Logic High		2.2	–	V
Logic Low		−0.3	0.3	V
TTL Input	$V_{CC} = 5V \pm 5\%$			
Logic High		2.2	–	V
Logic Low		−0.3	0.6	V
Schmitt Trigger Inputs				
RESET, CKI (÷4)				
Logic High		$0.7\,V_{CC}$	–	V
Logic Low		−0.3	0.4	V
SO Input Level (Test Mode)		2.0	3.0	V
All Other Inputs				
Logic High	V_{CC} = Max.	3.0	–	V
Logic High	$V_{CC} = 5V \pm 5\%$	2.2	–	V
Logic Low		−0.3	0.6	V
Input Levels High Trip Option				
Logic High		3.6	–	V
Logic Low		−0.3	1.2	V
Input Load Source Current	$V_{CC} = 5V$, $V_{IN} = 0V$			
CKO		−4	−800	μA
All Others		−100	−800	μA
Input Capacitance		–	7	pF
Hi-Z Input Leakage	$V_{CC} = 5V$	−2	+2	μA
Output Voltage levels				
Standard Outputs				
TTL Operation	$V_{CC} = 5V \pm 5\%$			
Logic High	$I_{OH} = -75\mu A$	2.4	–	V
Logic Low	$I_{OL} = 1.6 mA$	−0.3	0.4	V
CMOS Operation				
Logic High	$I_{OH} = -10\mu A$	$V_{CC} - 1$	–	V
Logic Low	$I_{OL} = 10\mu A$	−0.3	0.2	V
Output Current Levels				
LED Direct Drive Output	$V_{CC} = 5V$ (Note 4)			
Logic High	$V_{OH} = 2.0V$	1.0	12	mA
CKI Sink Current (R/C Option)	$V_{IN} = 3.5V$	2	–	mA
CKO (RAM Supply Current)	$V_R = 3.3V$	–	4	mA
TRI-STATE® or Open Drain Leakage Current	$V_{CC} = 5V$	−5	+5	μA
Allowable Sink Current				
Per Pin (L, D, G)		–	10	mA
Per Pin (All Others)		–	2	mA
Per Port (L)		–	16	mA
Per Port (D, G)		–	10	mA
Allowable Source Current				
Per Pin (L)		–	−15	mA
Per Pin (All Others)		–	−1.5	mA

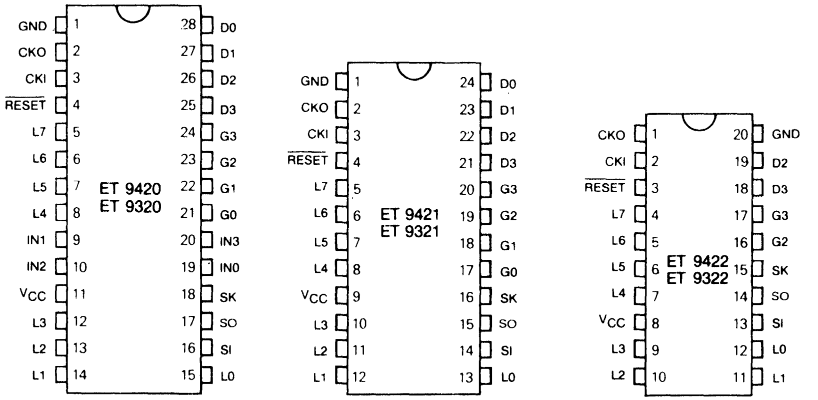
AC ELECTRICAL CHARACTERISTICS

ET9420/9421/9422 0°C ≤ T_A ≤ +70°C, 4.5V ≤ V_{CC} ≤ 6.3V (Unless otherwise specified)
ET9320/9321/9322 − 40°C ≤ T_A ≤ +85°C, 4.5V ≤ V_{CC} ≤ 5.5V (Unless otherwise specified)

Parameter	Conditions	Min.	Max.	Units
Instruction Cycle Time		4	10	µs
Operating CKI Frequency	÷16 mode	1.6	4.0	MHz
	÷8 mode	0.8	2.0	MHz
CKI Duty Cycle (Note 1)		40	60	%
Rise Time	Freq. = 4 MHz	–	60	ns
Fall Time	Freq. = 4 MHz	–	40	ns
CKI Using RC (Figure 8c)	÷4 mode			
Frequency	R = 15 kΩ ± 5%, C = 100 pF ± 10%	0.5	1.0	MHz
Instruction Cycle Time		4	8	µs
CKO as SYNC input (Figure 8d)				
t_{SYNC}	Figure 3a	50	–	ns
Inputs:				
SI				
t_{SETUP}		0.3	–	µs
t_{HOLD}		250	–	ns
All Other Inputs				
t_{SETUP}		1.7	–	µs
t_{HOLD}		300		ns
Output Propagation Delay	Test Conditions: $R_L = 5\,k\Omega$, $C_L = 50\,pF$, $V_{OUT} = 1.5V$	300	–	ns
SO and SK				
t_{pd1}		–	1.0	µs
t_{pd0}		–	1.0	µs
CKO				
t_{pd1}		–	0.25	µs
t_{pd0}		–	0.25	µs
All Other Outputs				
t_{pd1}		–	1.4	µs
t_{pd0}		–	1.4	µs
MICROBUS® Timing	$C_L = 100\,pF$, $V_{CC} = 5V \pm 5\%$			
Read Operation (Figure 4)				
Chip Select Stable before $\overline{RD}$—t_{CSR}		65	–	ns
Chip Select Hold Time for $\overline{RD}$—t_{RCS}		20	–	ns
$\overline{RD}$ Pulse Width—t_{RR}		400	–	ns
Data Delay from $\overline{RD}$—t_{RD}		–	375	ns
$\overline{RD}$ to Data Floating—t_{DF}		–	250	ns
Write Operation (Figure 5)				
Chip Select Stable before $\overline{WR}$—t_{CSW}		65	–	ns
Chip Select Hold Time for $\overline{WR}$—t_{WCS}		20	–	ns
$\overline{WR}$ Pulse Width—t_{WW}		400	–	ns
Data Set-Up Time for $\overline{WR}$—t_{DW}		320	–	ns
Data Hold Time for $\overline{WR}$—t_{WD}		100	–	ns
INTR Transition Time from $\overline{WR}$—t_{WI}		–	700	ns

Note 1 : Duty cycle = t_{W1} (t_{W1} + t_{W0}).
Note 2 : See Figure 9 for additional I/O characteristics.
Note 3 : Voltage change must be less than 0.5 volt in a 1 ms period.
Note 4 : Exercise great care not to exceed maximum device power dissipation limits when direct driving LEDs (or sourcing similar loads) at high temperature.

FIGURE 2 – CONNECTION DIAGRAMS


GND 1, CKO 2, CKI 3, RESET 4, L7 5, L6 6, L5 7, L4 8, IN1 9, IN2 10, VCC 11, L3 12, L2 13, L1 14
28 D0, 27 D1, 26 D2, 25 D3, 24 G3, 23 G2, 22 G1, 21 G0, 20 IN3, 19 IN0, 18 SK, 17 SO, 16 SI, 15 L0
ET 9420
ET 9320
GND 1, CKO 2, CKI 3, RESET 4, L7 5, L6 6, L5 7, L4 8, VCC 9, L3 10, L2 11, L1 12
24 D0, 23 D1, 22 D2, 21 D3, 20 G3, 19 G2, 18 G1, 17 G0, 16 SK, 15 SO, 14 SI, 13 L0
ET 9421
ET 9321
CKO 1, CKI 2, RESET 3, L7 4, L6 5, L5 6, L4 7, VCC 8, L3 9, L2 10
20 GND, 19 D2, 18 D3, 17 G3, 16 G2, 15 SK, 14 SO, 13 SI, 12 L0, 11 L1
ET 9422
ET 9322

Order number ET9420/N ET9320/N
Package N28A

Order number ET9421/N, ET9321/N
Package N24A

Order number ET9422/N ET9322/N
Package N20A

Pin	Description
L_7 - L_0	8 bidirectional I/O ports with TRI-STATE®
G_3 - G_0	4 bidirectional I/O ports
D_3 - D_0	4 general purpose outputs
IN_3 - IN_0	4 general purpose inputs (ET9420/9320 only)
SI	Serial input (or counter input)
SO	Serial output (or general purpose output)

Pin	Description
SK	Logic-controlled clock (or general purpose output)
CKI	System oscillator input
CKO	System oscillator output (or general purpose input or RAM power supply)
RESET	System reset input
V_{CC}	Power supply
GND	Ground

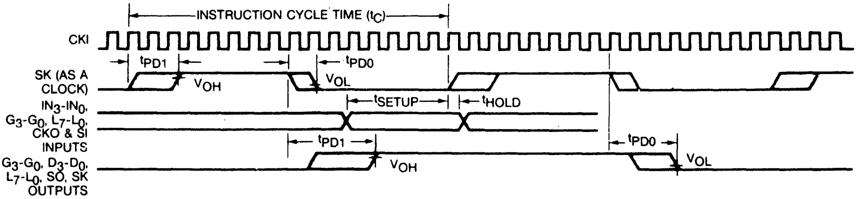

INSTRUCTION CYCLE TIME (t_C)
CKI
SK (AS A CLOCK)
t_{PD1}
t_{PD0}
V_{OH}
V_{OL}
IN_3-IN_0, G_3-G_0, L_7-L_0, CKO & SI INPUTS
t_{SETUP}
t_{HOLD}
G_3-G_0, D_3-D_0, L_7-L_0, SO, SK OUTPUTS
t_{PD1}
t_{PD0}
V_{OH}
V_{OL}

FIGURE 3 – INPUT/OUTPUT TIMING DIAGRAMS (CRYSTAL DIVIDE BY 16 MODE)

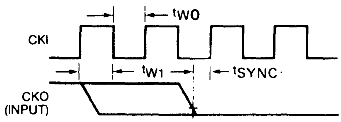


FIGURE 3a – SYNCHRONIZATION TIMING

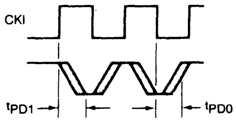


FIGURE 3b – CKO OUTPUT TIMING

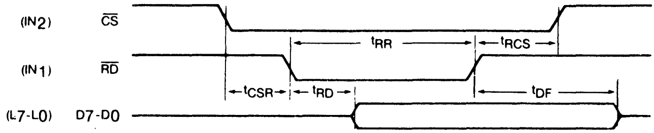


FIGURE 4 – MICROBUS® READ OPERATION TIMING

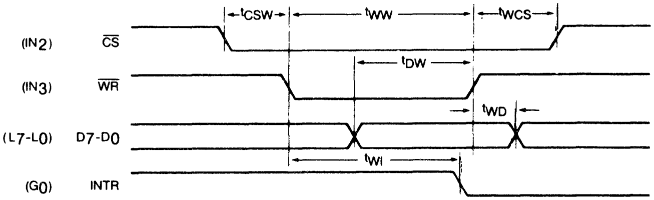


FIGURE 5 – MICROBUS® WRITE OPERATION TIMING

FUNCTIONAL DESCRIPTION

For ease of reading this description, only ET9420 and/or ET9421 are referenced ; however, all such references apply equally to the ET9422, 9322, 9320 and/or ET9321, respectively.

A block diagram of the ET9420 is given in figure 1. Data paths are illustrated in simplified form to depict how the various logic elements communicate with each other in implementing the instruction set of the device. Positive logic is used. When a bit is set, it is a logic "1" (greater than 2 volts). When a bit is reset, it is a logic "0" (less than 0.8 volts).

Program Memory

Programm Memory consists of a 1.024 byte ROM. As can be seen by an examination of the ET9420/9421/9422 instruction set, these words may be program instructions, program data or ROM addressing data. Because of the special characteristics associated with the JP, JSRP, JID and LQID instructions, ROM must often be thought of as being organized into 16 pages of 64 words each.

ROM addressing is accomplished by a 10-bit PC register. Its binary value selects one of the 1,024 8-bit words contained in ROM. A new address is loaded into the PC register during each instruction cycle. Unless the instruction is a transfer of control instruction, the PC register is loaded with the next sequential 10-bit binary count value. Three levels of subroutine nesting are implemented by the 10-bit subroutine save registers, SA, SB and SC, providing a last-in, first-out (LIFO) hardware subroutine stack.

ROM instruction words are fetched, decoded and executed by the Instruction Decode, Control and Skip Logic circuitry.

Data Memory

Data memory consists of a 256-bit RAM, organized as 4 data registers of 16 4-bit digits. RAM addressing is implemented by a 6-bit **B register** whose upper 2 bits (Br) select 1 of 4 data registers and lower 4 bits (Bd) select 1 of 16 4-bit digits in the selected data register. While the 4-bit contents of the selected RAM digit (M) is usually loaded into or from, or exchanged with, the A register (accumulator), it may also be loaded into or from the Q latches or loaded from the L ports. RAM addressing may also be performed directly by the LDD and XAD instructions based upon the 6-bit contents of the operand field of these instructions. The Bd register also serves as a source register for 4-bit data sent directly to the D outputs.

Internal Logic

The 4-bit **A register** (accumulator) is the source and destination register for most I/O, arithmetic, logic and data memory access operations. It can also be used to load the Br and Bd portions of the B register, to load and input 4 bits of the 8-bit Q latch data, to input 4 bits of the 8-bit L I/O port data and to perform data exchanges with the SIO register.

A 4-bit adder performs the arithmetic and logic functions of the ET9420/9421/9422, storing its results in A. It also outputs a carry bit to the 1-bit C register, most often employed to indicate arithmetic overflow. The C register, in conjunction with the XAS instruction and the EN register, also serves to control the SK output. C can be outputted directly to SK or can enable SK to be a sync clock each instruction cycle time. (See XAS instruction and EN register description, below).

Four **general-purpose inputs,** IN_3-IN_0, are provided; IN_1, IN_2 and IN_3 may be selected, by a mask-programmable option, as Read Strobe, Chip Select and Write Strobe inputs, respectively, for use in MICROBUS® applications.

The **D register** provides 4 general-purpose outputs and is used as the destination register for the 4-bit contents of Bd.

The **G register** contents are outputs to 4 general-purpose bidirectional I/O ports. G_0 may be mask-programmed as an output for MICROBUS® applications.

The **Q register** is an internal, latched, 8-bit register, used to hold data loaded to or from M and A, as well as 8-bit data from ROM. Its contents are output to the L I/O ports when the L drivers are enabled under program control. (See LEI instruction). With the MICROBUS™ option selected, Q can also be loaded with the 8-bit contents of the L I/O ports upon the occurence of a write strobe from the host CPU.

The **8 L drivers**,when enabled, output the contents of latched Q data to the L I/O ports. Also, the contents of L may be read directly into A and M. As explained above, the MICROBUS™ option allows L I/O port data to be latched into the Q register. L I/O ports can be directly connected to the segments of a multiplexed LED display (using the LED Direct Drive output configuration option) with Q data being outputted to the Sa-Sg and decimal point segments of the display.

The **SIO register** functions as a 4-bit serial-in/serial-out shift register or as a binary counter depending on the contents of the EN register. (See EN register description, below.) Its contents can be exchanged with A, allowing it to input or output a continuous serial data stream. SIO may also be used to provide additional parallel I/O by connecting SO to external serial-in/parallel-out shift registers.

The XAS instruction copies C into the **SKL latch**. In the counter mode, SK is the output of SKL; in the shift register modé, SK outputs SKL ANDed with the clock.

The **EN register** is an internal 4-bit register loaded under program control by the LEI instruction. The state of each bit of this register selects or deselects the particular feature associated with each bit of the EN register (EN_3-EN_0).

1. The least significant bit of the enable register, EN_0, selects the SIO register as either a 4-bit shift register or a 4-bit binary counter. With EN_0 set, SIO is an asynchronous binary counter, *decrementing* its value by one upon each low-going pulse ("1" to "0") ocurring on the SI input. Each pulse must be at least two instruction cycles wide. SK outputs the value of SKL. The SO output is equal to the value of EN_3. With EN_0 reset, SIO is a serial shift register shifting left each instruction cycle time. The data present at SI goes into the least significant bit of SIO. SO can be enabled to output the most significant bit of SIO each cycle time. (See 4 below.) The SK output becomes a logic-controlled clock.
2. With EN_1 set the IN_1 input is enabled as an interrupt input. Immediately following an interrupt, EN_1 is reset to disable further interrupts.
3. With EN_2 set, the L drivers are enabled to output the data in Q tó the L I/O ports. Resetting EN_2 disables the L drivers, placing the L I/O ports in a high-impedance input state.
4. EN_3, in conjunction with EN_0, affects the SO output. With EN_0 set (binary counter option selected) SO will output the value loaded into EN_3. With EN_0 reset (serial shift register option selected), setting EN_3 enables SO as the output of the SIO shift register, outputting serial shifted data each instruction time. Resetting EN_3 with the serial shift register option selected disables SO as the shift register output; data continues to be shifted through SIO and can be exchanged with A via an XAS instruction but SO remains reset to "0." The table below provides a summary of the modes associated with EN_3 and EN_0.

Enable Register Modes – Bits EN_3 and EN_0

EN_3	EN_0	SIO	SI	SO	SK
0	0	Shift Register	Input to Shift Register	0	If SKL = 1, SK = CLOCK If SKL = 0, SK = 0
1	0	Shift Register	Input to Shift Register	Serial Out	If SKL = 1, SK = CLOCK If SKL = 0, SK = 0
0	1	Binary Counter	Input to Binary Counter	0	If SKL = 1, SK = 1 If SKL = 0, SK = 0
1	1	Binary Counter	Input to Binary Counter	1	If SKL = 1, SK = 1 If SKL = 0, SK = 0

Interrupt

The following features are associated with the IN_1 interrupt procedure and protocol and must be considered by the programmer when utilizing interrupts.

a. The interrupt, once acknowledged as explained below, pushes the next sequential program counter address (PC + 1) onto the stack, pushing in turn the contents of the other subroutine-save registers to the next lower level (PC + 1 → SA → SB → SC). Any previous contents of SC are lost. The program counter is set to hex address 0FF (the last word of page 3) and EN_1 is reset.

b. An interrupt will be acknowledged only after the following conditions are met:

1. EN_1 has been set.
2. A low-going pulse ("1" to "0") at least two instruction cycles wide occurs on the IN_1 input.
3. A currently executing instruction has been completed.
4. All successive transfer of control instructions and successive LBIs have been completed (e.g., if the main program is executing a JP instruction which transfers program control to another JP instruction, the interrupt will not be acknowledged until the second JP instruction has been executed.

c. Upon acknowledgement of an interrupt, the skip logic status is saved and later restored upon popping of the stack. For example, if an interrupt occurs during the execution of ASC (Add with Carry, Skip on Carry) instruction which results in carry, the skip logic status is saved and program control is transferred to the interrupt servicing routine at hex address 0FF. At the *end* of the interrupt routine, a RET instruction is executed to "pop" the stack and return program control to the instruction following the original ASC. *At this time*, the skip logic is enabled and skips this instruction because of the previous ASC carry. Subroutines and LQID instructions should not be nested within the interrupt service routine, since their popping the stack will enable any previously saved main program skips, interfering with the orderly execution of the interrupt routine.

d. The first instruction of the interrupt routine at hex address 0FF must be a NOP.

e. A LEI instruction can be put immediately before the RET to re-enable interrupts.

Microbus® Interface

The ET9420 has an option which allows it to be used as a peripheral microprocessor device, inputting and outputting data from and to a host microprocessor (µP). IN_1, IN_2 and IN_3 general purpose inputs become **MICROBUS® compatible** read-strobe, chip-select, and write-strobe lines, respectively. IN_1 becomes $\overline{RD}$ — a logic "0" on this input will cause Q latch data to be enabled to the L ports for input to the µP. IN_2 becomes $\overline{CS}$ — a logic "0" on this line selects the ET9420 as the µP peripheral device by enabling the operation of the $\overline{RD}$ and $\overline{WR}$ lines and allows for the selection of one of several peripheral components. IN_3 becomes $\overline{WR}$ — a logic "0" on this line will write bus data from the L ports to the Q latches for input to the ET9420. G_0 becomes INTR a "ready" output, reset by a write pulse from the µP on the $\overline{WR}$ line, providing the "handshaking capability necessary for asynchronous data transfer between the host CPU and the ET9420.

This option has been designed for compatibility with National's MICROBUS™ — a standard interconnect system for 8-bit parallel data transfer between MOS/LSI CPUs and interfacing devices. (See MICROBUS™ National Publication.) The functioning and timing relationships between the ET9420 signal lines affected by this option are as specified for the MICROBUS® interface, and are given in the AC electrical characteristics and shown in the timing diagrams (figures 4 and 5). Connection of the ET9420 to the MICROBUS® is shown in Figure 6.

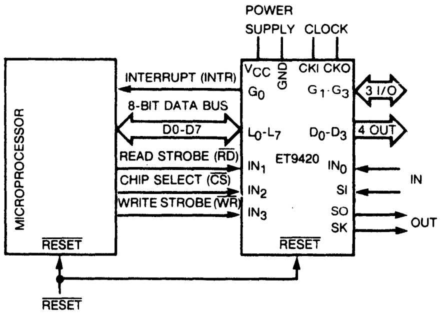
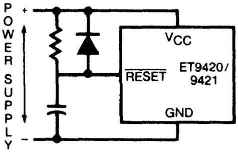


FIGURE 6 – MICROBUS® OPTION INTERCONNECT

Initialization

The Reset Logic, internal to the ET9420/9421/9422, will initialize (clear) the device upon power-up if the power supply rise time is less than 1ms and greater than 1µs. If the power supply rise time is greater than 1ms, the user must provide an external RC network and diode to the $\overline{RESET}$ pin as shown below. The $\overline{RESET}$ pin is configured as a Schmitt trigger input. If not used it should be connected to V_{CC}. Initialization will occur whenever a logic "0" is applied to the RESET input, provided it stays low for at least three instruction cycle times.

Upon initialization, the PC register is cleared to 0 (ROM address 0) and the A, B, C, D, EN, and G registers are cleared. The SK output is enabled as a SYNC output, providing a pulse each instruction cycle time. ***Data Memory (RAM) is not cleared upon initialization.*** The first instruction at address 0 must be a CLRA.



RC ≥ 5 x POWER SUPPLY RISE TIME

FIGURE 7 – POWER-UP CLEAR CIRCUIT

Oscillator

There are four basic clock oscillator configurations available as shown by figure 8.

a. **Crystal Controlled Oscillator.** CKI and CKO are connected to an external crystal. The instruction cycle time equals the crystal frequency divided by 16 (optional by 8).

b. **External Oscillator.** CKI is an external clock input signal. The external frequency is divided by 16 (optional by 8) to give the instruction cycle time. CKO is now available to be used as the RAM power supply (V_R) or as a general purpose input.

c. **RC Controlled Oscillator.** CKI is configured as a single pin RC controlled Schmitt trigger oscillator. The instruction cycle equals the oscillation frequency divided by 4. CKO is available for non-timing functions.

d. **Externally Synchronized Oscillator. Intended for use in multi-COP systems, CKO is programmed to function as an input connected to the SK output of another ET9420/-9421/9422 with CKI connected as shown. In this configuration, the SK output connected to CKO must provide a SYNC (instruction cycle) signal to CKO, thereby allowing synchronous data transfer between the COPs using only the SI and SO serial I/O pins in conjunction with the XAS instruction. Note that on power-up SK is automatically enabled as a SYNC output (See Functional Description, Initialization, above).**

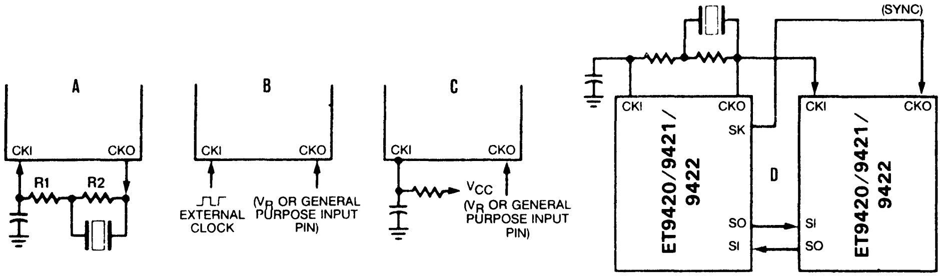
CKO Pin Options

In a crystal controlled oscillator system, CKO is used as an output to the crystal network. As an option CKO can be a SYNC input as described above. As another option CKO can be a general purpose input, read into bit 2 of A (accumulator) upon execution of an INIL instruction. As another option, CKO can be a RAM power supply pin (V_R), allowing its connection to a standby/backup power supply to maintain the integrity of RAM data with minimum power drain when the main supply is inoperative or shut down to conserve power. Using either option is appropriate in applications where the ET9420/-9421/9422 system timing configuration does not require use of the CKO pin.

RAM Keep-Alive Option (Not available on ET9422)

Selecting CKO as the RAM power supply (V_R) allows the user to shut off the chip power supply (V_{CC}) and maintain data in the RAM. To insure that RAM data integrity is maintained, the following conditions must be met:

1. $\overline{RESET}$ must go low before V_{CC} goes below spec during power off; V_{CC} must be within spec before $\overline{RESET}$ goes high on power up.
2. V_R must be within the operating range of the chip, and equal to V_{CC} ± 1V during normal operation.
3. V_R must be $\geq 3.3V$ with V_{CC} off.



Crystal Oscillator **External Oscillator** **RC Controlled Oscillator** **Externally Synchronized Oscillator**

Crystal Oscillator

Crystal Value	Component Values		
	R1 (Ω)	R2 (Ω)	C (pF)
4 MHz	1k	1M	27
3.58 MHz	1k	1M	27
2.09 MHz	1k	1M	56

RC Controlled Oscillator

R (kΩ)	C (pF)	Instruction Cycle Time (μs)
12	100	5 ± 20%
6.8	220	5.3 ± 23%
8 2	300	8 ± 29%
22	100	8.6 ± 16%

Note: 50 kΩ ≥ R ≥ 5 kΩ
360 pF ≥ C ≥ 50 pF

FIGURE 8 — ET9420/9421/9422/ET9320/9321/9322 OSCILLATOR

I/O Options

ET9420/9421/9422 outputs have the following optional configurations, illustrated in Figure 9a :

a. Standard — an enhancement mode device to ground in conjunction with a depletion-mode device to V_{CC}, compatible with TTL and CMOS input requirements. Available on SO, SK, and all D and G outputs.

b. Open-Drain — an enhancement-mode device to ground only, allowing external pull-up as required by the user's application. Available on SO, SK, and all D and G outputs.

c. Push-Pull — An enhancement-mode device to ground in conjunction with a depletion-mode device paralleled by an enhancement-mode device to V_{CC}. This configuration has been provided to allow for fast rise and fall times when driving capacitive loads. Available on SO and SK outputs only.

d. Standard L — same as **a.**, but may be disabled. Available on L outputs only.

e. Open Drain L — same as **b.**, but may be disabled. Available on L outputs only.

f. LED Direct Drive — an enhancement-mode device to ground and to V_{CC}, meeting the .typical current sourcing requirements of the segments of an LED display. The sourcing device is clamped to limit current flow. These devices may be turned off under program control (See Functional Description, EN Register), placing the outputs in a high-impedance state to provide required LED segment blanking for a multiplexed display.

g. TRI-STATE® Push-Pull — an enhancement-mode device to ground and V_{CC}. These outputs are TRI-STATE outputs, allowing for connection of these outputs to a data bus shared by other bus drivers.

ET9420/9421/9422 inputs have the following optional configurations :

h. An on-chip depletion load device to V_{CC}.

i. A Hi-Z input which must be driven to a "1" or "0" by external components.

The above input and output configurations share common enhancement-mode and depletion-mode devices. Specifically, all configurations use one or more of six devices (numbered 1-6, respectively). Minimum and maximum current (I_{OUT} and V_{OUT}) curves are given in Figure 9b for each of these devices to allow the designer to effectively use these I/O configurations in designing a ET9420/9421/9422 system.

The SO, SK outputs can be configured as shown in **a.**, **b.**, or **c.** The D and G outputs can be configured as shown in **a.** or **b.** Note that when inputting data to the G ports, the G outputs should be set to "1." The L outputs can be configured as in **d.**, **e.**, **f.** or **g.**

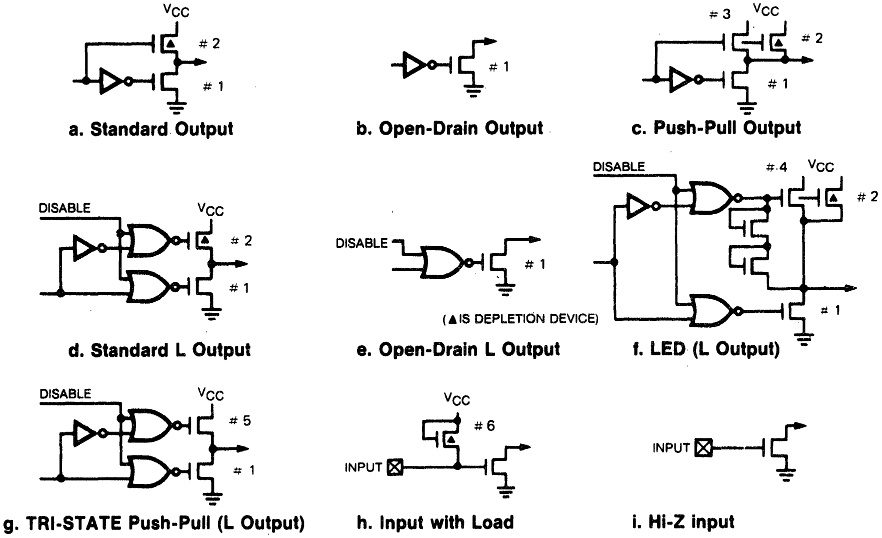
An important point to remember if using configuration **d.** or **f.** with the L drivers is that even when the L drivers are disabled, the depletion load device will source a small amount of current (see Figure 9b, device 2); however, when the L lines are used as inputs, the disabled depletion device can *not* be relied on to source sufficient current to pull an input to logic "1".

ET9421

If the ET9420 is bonded as a 24-pin device, it becomes the ET9421, illustrated in Figure 2, ET9420/9421 Connection Diagrams. Note that the ET9421 does not contain the four general purpose IN inputs (IN_3-IN_0). Use of this option precludes, of course, use of the IN options, interrupt feature, and the MICROBUS™ option which uses IN_1-IN_3. All other options are available for the ET9421.

ET9422

If the ET9420 is bonded as a 20-pin device, it becomes the ET9422, as illustrated in Figure 2. Note that the ETL9422 contains all the ET9421 pins except D0, D1, G0 and G1. ET9422 also does not allow RAM power supply input as a valid CKO pin option.



a. Standard Output

b. Open-Drain Output

c. Push-Pull Output

d. Standard L Output

e. Open-Drain L Output

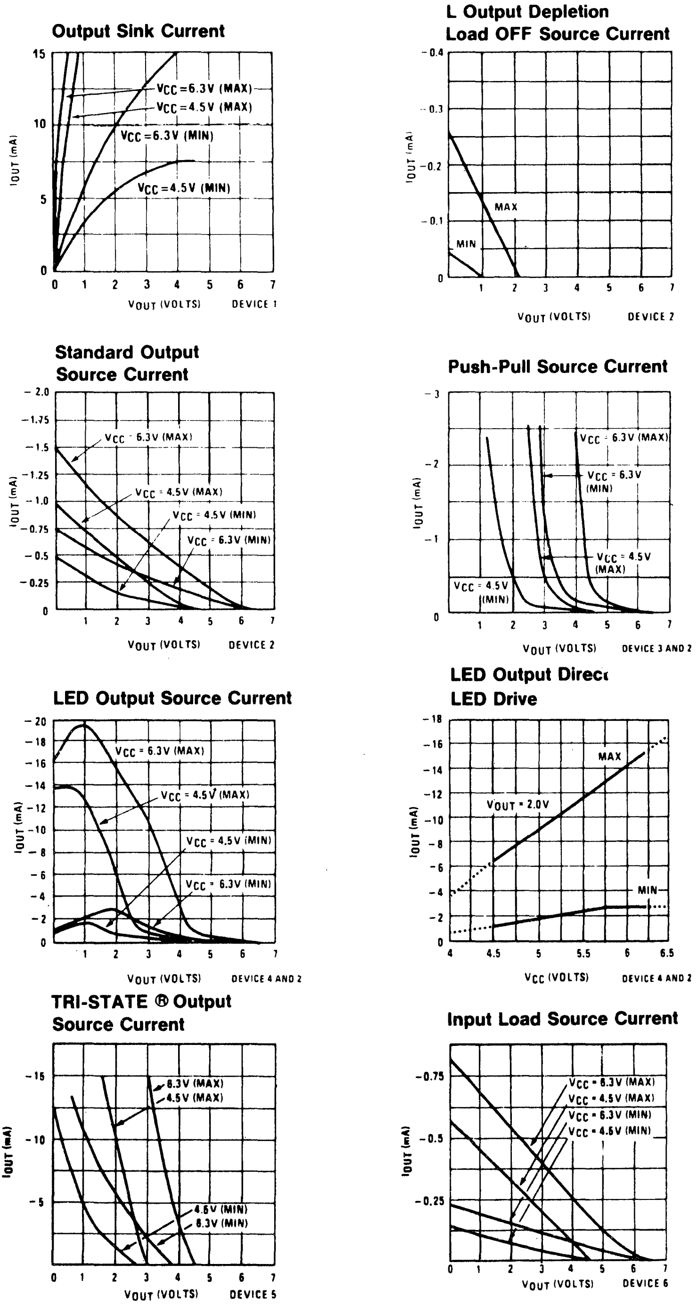
f. LED (L Output)

g. TRI-STATE Push-Pull (L Output)

h. Input with Load

i. Hi-Z input

FIGURE 9a – INPUT/OUTPUT CONFIGURATIONS



Output Sink Current

L Output Depletion Load OFF Source Current

Standard Output Source Current

Push-Pull Source Current

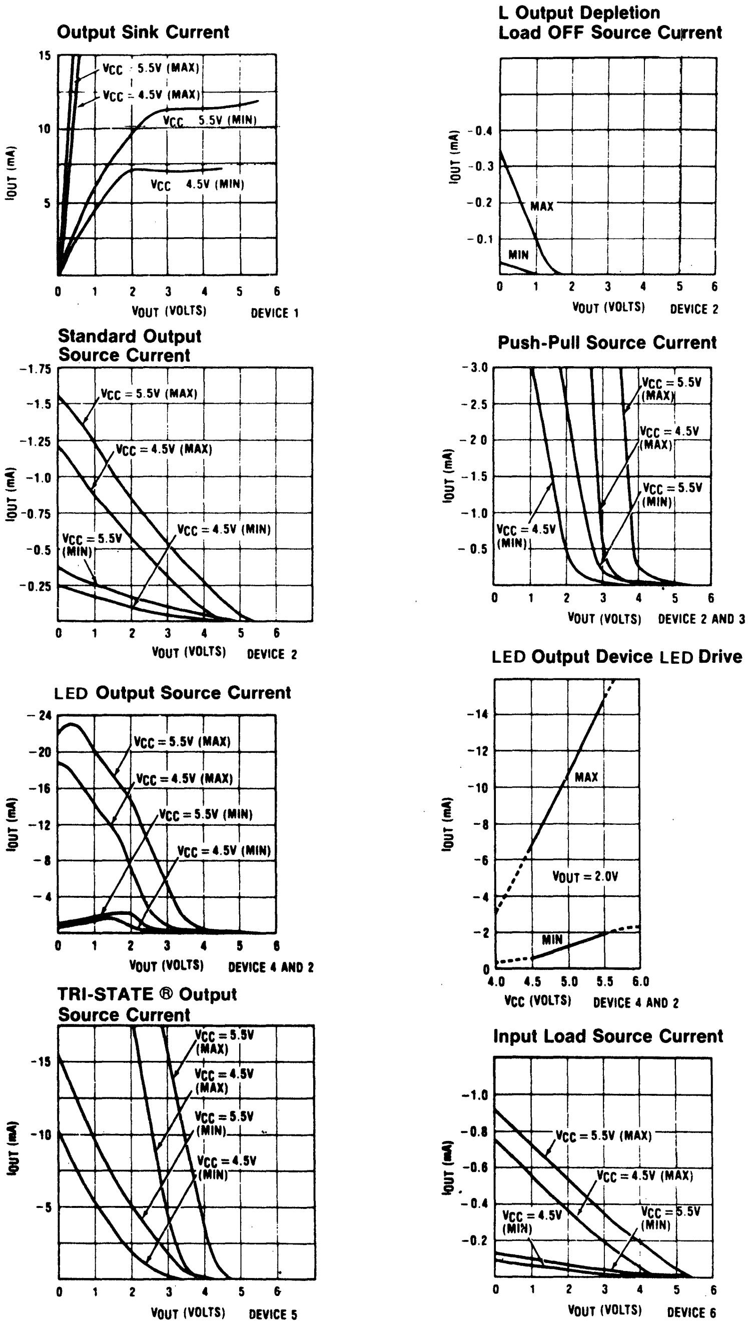
LED Output Source Current

LED Output Direct LED Drive

TRI-STATE ® Output Source Current

Input Load Source Current

FIGURE 9b — ET9420/9421/9422 INPUT/OUTPUT CHARACTERISTICS



Output Sink Current

L Output Depletion Load OFF Source Current

Standard Output Source Current

Push-Pull Source Current

LED Output Source Current

LED Output Device LED Drive

TRI-STATE ® Output Source Current

Input Load Source Current

FIGURE 9c — ET9320/9321/9322 INPUT/OUTPUT CHARACTERISTICS

INSTRUCTION SET

Table 1 is a symbol table providing internal architecture, instruction operand and operational symbols used in the instruction set table.

Table 2 provides the mnemonic, operand, machine code, data flow, skip conditions, and description associated with each instruction in the ET9420/9421/9422 instruction set.

TABLE 2 – ET9420/9421/9422. ET9320/9321/9322 INSTRUCTION SET TABLE SYMBOLS

Symbol	Definition
INTERNAL ARCHITECTURE SYMBOLS	
A	4-bit Accumulator
B	6-bit RAM Address Register
Br	Upper 2 bits of B (register address)
Bd	Lower 4 bits of B (digit address)
C	1-bit Carry Register
D	4-bit Data Output Port
EN	4-bit Enable Register
G	4-bit Register to latch data for G I/O Port
IL	Two 1-bit latches associated with the IN_3 or IN_0 inputs
IN	4-bit Input Port
L	8-bit TRI-STATE® I/O Port
M	4-bit contents of RAM Memory pointed to by B Register
PC	10-bit ROM Address Register (program counter)
Q	8-bit Register to latch data for L I/O Port
SA	10-bit Subroutine Save Register A
SB	10-bit Subroutine Save Register B
SC	10 Subroutine Save Register A
SIO	4-bit Shift Register and Counter
SK	Logic-Controlled Clock Output

Symbol	Definition
INSTRUCTION OPERAND SYMBOLS	
d	4-bit Operand Field, 0-15 binary (RAM Digit Select)
r	2-bit Operand Field, 0-3 binary (RAM Register Select)
a	10-bit Operand Field, 0-1023 binary (ROM Address)
y	4-bit Operand Field, 0-15 binary (Immediate Data)
RAM(s)	Contents of RAM location addressed by s
ROM(t)	Contents of ROM location addressed by t
OPERATIONAL SYMBOLS	
+	Plus
—	Minus
→	Replaces
↔	Is exchanged with
=	Is equal to
$\overline{A}$	The one's complement of A
⊕	Exclusive-OR
:	Range of values

TABLE 2 – ET9420/9421/9422. ET9320/9321/9322 INSTRUCTION SET

Mnemonic	Operand	Hex Code	Machine Language Code (Binary)	Data Flow	Skip Conditions	Description
ARITHMETIC INSTRUCTIONS						
ASC		30	0011 0000	A + C + RAM(B) → A Carry → C	Carry	Add with Carry, Skip on Carry
ADD		31	0011 0001	A + RAM(3) → A	None	Add RAM to A
ADT		4A	0100 1010	A + 10_{10} → A	None	Add Ten to A
AISC	y	5-	0101 y	A + y → A	Carry	Add Immediate, Skip on Carry (y ≠ 0)
CASC		10	0001 0000	$\overline{A}$ + RAM(B) + C → A Carry → C	Carry	Complement and Add with Carry, Skip on Carry
CLRA		00	0000 0000	0 → A	None	Clear A
COMP		40	0100 0000	$\overline{A}$ → A	None	One's complement of A to A
NOP		44	0100 0100	None	None	No Operation
RC		32	0011 0010	"0" → C	None	Reset C
SC		22	0010 0010	"1" → C	None	Set C
XOR		02	0000 0010	A ⊕ RAM(B) → A	None	Exclusive-OR RAM with A

TABLE 2 – ET9420/9421/9422. ET9320/9321/9322 INSTRUCTION SET (continued)

Mnemonic	Operand	Hex Code	Machine Language Code (Binary)	Data Flow	Skip Conditions	Description
TRANSFER OF CONTROL INSTRUCTIONS						
JID		FF	1111 | 1111	ROM $(PC_{9:8}, A,M) \rightarrow PC_{7:0}$	None	Jump Indirect (Note 3)
JMP	a	6- --	0110 | 00 | $a_{9:8}$ $a_{7:0}$	a → PC	None	Jump
JP	a	-- --	1 | $a_{6:0}$ (pages 2,3 only) or 11 | $a_{5:0}$ (all other pages)	$a \rightarrow PC_{6:0}$ $a \rightarrow PC_{5:0}$	None	Jump within Page (Note 4)
JSRP	a	--	10 | $a_{5:0}$	PC + 1 → SA → SB → SC $0010 \rightarrow PC_{9:6}$ $a \rightarrow PC_{5:0}$	None	Jump to Subroutine Page (Note 5)
JSR	a	6- --	0110 | 10 | $a_{9.8}$ $a_{7:0}$	PC + 1 → SA → SB → SC a → PC	None	Jump to Subroutine
RET		48	0100 | 1000	SC → SB → SA → PC	None	Return from Subroutine
RETSK		49	0100 | 1001	SC → SB → SA → PC	Always Skip on Return	Return from Subroutine then Skip
MEMORY REFERENCE INSTRUCTIONS						
CAMQ		33 3C	0011 | 0011 0011 | 1100	$A \rightarrow Q_{7:4}$ $RAM(B) \rightarrow Q_{3:0}$	None	Copy A, RAM to Q
CQMA		33 2C	0011 | 0011 0010 | 1100	$Q_{7:4} \rightarrow RAM(B)$ $Q_{3:0} \rightarrow A$	None	Copy Q to RAM, A
LD	r	-5	00 | r | 0101	RAM(B) → A $Br \oplus r \rightarrow Br$	None	Load RAM into A, Exclusive-OR Br with r
LDD	r,d	23 --	0010 | 0011 00 | r | d	RAM(r,d) → A	None	Load A with RAM pointed to directly by r,d
LQID		BF	1011 | 1111	$ROM(PC_{9:8},A,M) \rightarrow Q$ SB → SC	None	Load Q Indirect (Note 3)
RMB	0 1 2 3	4C 45 42 43	0100 | 1100 0100 | 0101 0100 | 0010 0100 | 0011	$0 \rightarrow RAM(B)_0$ $0 \rightarrow RAM(B)_1$ $0 \rightarrow RAM(B)_2$ $0 \rightarrow RAM(B)_3$	None	Reset RAM Bit
SMB	0 1 2 3	4D 47 46 4B	0100 | 1101 0100 | 1101 0100 | 0110 0100 | 1011	$1 \rightarrow RAM(B)_0$ $1 \rightarrow RAM(B)_1$ $1 \rightarrow RAM(B)_2$ $1 \rightarrow RAM(B)_3$	None	Set RAM Bit

TABLE 2 – ET9420/9421/9422. ET9320/9321/9322 INSTRUCTION SET (continued)

Mnemonic	Operand	Hex Code	Machine Language Code (Binary)	Data Flow	Skip Conditions	Description
MEMORY REFERENCE INSTRUCTIONS (continued)						
STII	y	7-	0111 y	y → RAM(B) Bd + 1 → Bd	None	Store Memory Immediate and Increment Bd
X	r	-6	00 r 0110	RAM(B) ↔ A Br ⊕ r → Br	None	Exchange RAM with A, Exclusive-OR Br with r
XAD	r,d	23 --	0010 0011 10 r d	RAM(r,d) ↔ A	None	Exchange A with RAM pointed to directly by r,d
XDS	r	-7	00 r 0111	RAM(B) ↔ A Bd – 1 → Bd Br ⊕ r → Br	Bd decrements past 0	Exchange RAM with A and Decrement Bd, Exclusive-OR Br with r
XIS	r	-4	00 r 0100	RAM(B) ↔ A Bd + 1 → Bd Br ⊕ r → Br	Bd increments past 15	Exchange RAM with A and Increment Bd, Exclusive-OR Br with r
REGISTER REFERENCE INSTRUCTIONS						
CAB		50	0101 0000	A → Bd	None	Copy A to Bd
CBA		4E	0100 1110	Bd → A	None	Copy Bd to A
LBI	r,d	-- 33 -- 	00 r (d – 1) (d = 0, 9:15) or 0011 0011 10 r d (any d)	r,d → B	Skip until not a LBI	Load B Immediate with r,d (Note 6)
LEI	y	33 6-	0011 0011 0110 y	y → EN	None	Load EN Immediate (Note 7)
XABR		12	0001 0010	A ↔ Br (0,0 → A_3,A_2)	None	Exchange A with Br
TEST INSTRUCTIONS						
SKC		20	0010 0000		C = "1"	Skip if C is True
SKE		21	0010 0001		A = RAM(B)	Skip if A Equals RAM
SKGZ		33 21	0011 0011 0010 0001		$G_{3:0} = 0$	Skip if G is Zero (all 4 bits)
SKGBZ	 0 1 2 3	33 01 11 03 13	0011 0011 0000 0001 0001 0001 0000 0011 0001 0011	1st byte 2nd byte	 $G_0 = 0$ $G_1 = 0$ $G_2 = 0$ $G_3 = 0$	Skip if G Bit is Zero
SKMBZ	0 1 2 3	01 11 03 13	0000 0001 0001 0001 0000 0011 0001 0011		$RAM(B)_0 = 0$ $RAM(B)_1 = 0$ $RAM(B)_2 = 0$ $RAM(B)_3 = 0$	Skip if RAM Bit is Zero
SKT		41	0100 0001		A time-base counter carry has occurred since last test	Skip on Timer (Note 3)

TABLE 2 – ET9420/9421/9422. ET9320/9321/9322 INSTRUCTION SET (continued)

Mnemonic	Operand	Hex Code	Machine Language Code (Binary)	Data Flow	Skip Conditions	Description
INPUT/OUTPUT INSTRUCTIONS						
ING		33 2A	0011|0011 0010|1010	G → A	None	Input G Ports to A
ININ		33 28	0011|0011 0010|1000	IN → A	None	Input IN Inputs to A (Note 2)
INIL		33 29	0011|0011 0010|1001	IL_3, CKO, "0", IL_0 → A	None	Input IL Latches to A (Note 3)
INL		33 2E	0011|0011 0010|1110	$L_{7:4}$ → RAM(B) $L_{3:0}$ → A	None	Input L Ports to RAM,A
OBD		33 3E	0011|0011 0011|1110	Bd → D	None	Output Bd to D Outputs
OGI	y	33 5-	0011|0011 0101| y	y → G	None	Output to G Ports Immediate
OMG		33 3A	0011|0011 0011|1010	RAM(B) → G	None	Output RAM to G Ports
XAS		4F	0100|1111	A ↔ SIO, C → SKL	None	Exchange A with SIO (Note 3)

Note 1 : All subscripts for alphabetical symbols indicate bit numbers unless explicitly defined (e.g., Br and Bd are explicitly defined). Bits are numbered 0 to N where 0 signifies the least significant bit (low-order, right-most bit). For example, A_3Indicates the most significant (left-most) bit of the 4-bit A register.

Note 2 : The ININ instruction is not available on the ET9421/ET9321 and ET9422/ET9322 since these devices do not contain the IN inputs.

Note 3 : For additional information on the operation of the XAS, JID, LQID, INIL, and SKT instructions, see below.

Note 4 : The JP instruction allows a jump, while in subroutine pages 2 or 3, to any ROM location within the two-page boundary of pages 2 or 3. The JP instruction, otherwise, permits a jump to a ROM location within the current 64-word page. JP may not jump to the last word of a page.

Note 5 : A JSRP transfers program control to subroutine page 2 (0010 is loaded into the upper 4 bits of P). A JSRP may not be used when in pages 2 or 3. JSRP may not jump to the last word in page 2.

Note 6 : LBI is a single-byte instruction if d = 0, 9, 10, 11, 12, 13, 14 or 15. The machine code for the lower 4 bits equals the binary value of the "d" data *minus 1*, e.g., to load the lower four bits of B (Bd) with the value 9 (1001_2), the lower 4 bits of the LBI instruction equals 8 (1000_2). To load 0, the lower 4 bits of the LBI instruction should equal 15 (1111_2).

Note 7 : Machine code for operand field y for LEI instruction should equal the binary value to be latched into EN, where a "1" or "0" in each bit of EN corresponds with the selection or deselection of a particular function associated with each bit. (See Functional Description, EN Register).

The following information is provided to assist the user in understanding the operation of several unique instructions and to provide notes useful to programmers in writing ET9420/9421/9422 programs.

XAS Instruction

XAS (Exchange A with SIO) exchanges the 4-bit contents of the accumulator with the 4-bit contents of the SIO register. The contents of SIO will contain serial-in/serial-out shift register or binary counter data, depending on the value of the EN register. An XAS instruction will also affect the SK output. (See Functional Description, EN Register, above.) If SIO is selected as a shift register, an XAS instruction must be performed once every 4 instruction cycles to effect a continuous data stream.

JID Instruction

JID (Jump Indirect) is an indirect addressing instruction, transferring program control to a new ROM location pointed to indirectly by A and M. It loads the lower 8 bits of the ROM address register PC with the *contents* of ROM addressed by the 10-bit word, $PC_{9:8}$, A, M. PC_9 and PC_8 are not affected by this instruction.

Note that JID requires 2 instruction cycles to execute.

INIL Instruction

INIL (Input IL Latches to A) inputs 2 latches, IL_3 and IL_0 (see figure 10) and CKO into A. The IL_3 and IL_0 latches are set if a low-going pulse ("1" to "0") has occurred on the IN_3 and IN_0 inputs since the last INIL instruction, provided the input pulse stays low for at least two instruction times. Execution of an INIL inputs IL_3 and IL_0 into A3 and A0 respectively, and resets these latches to allow them to respond to subsequent low-going pulses on the IN_3 and IN_0 lines. If CKO is mask programmed as a general purpose input, an INIL will input the state of CKO into A2. If CKO has not been so programmed, a "1" will be placed in A2. A "0" is always placed in A1 upon the execution of an INIL. The general purpose inputs IN_3-IN_0 are input to A upon execution of an ININ instruction. (See table 2, ININ instruction.) INIL is useful in recognizing pulses of short duration or pulses which occur too often to be read conveniently by an ININ instruction.

Note: IL latches are not cleared on reset.

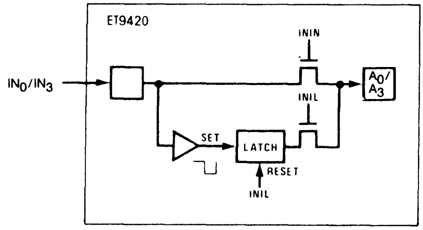


FIGURE 10.

LQID Instruction

LQID (Load Q Indirect) loads the 8-bit Q register with the contents of ROM pointed to by the 10-bit word PC_9, PC_8, A, M. LQID can be used for table lookup or code conversion such as BCD to seven-segment. The LQID instruction "pushes" the stack (PC + 1 → SA → SB →SC) and replaces the least significant 8 bits of PC as follows: A → $PC_{7:4}$, RAM(B) → $PC_{3:0}$, leaving PC_9 and PC_8 unchanged. The ROM data pointed to by the new address is fetched and loaded into the Q latches. Next, the stack is "popped" (SC → SB → SA →PC), restoring the saved value of PC to continue sequential program execution. Since LQID pushes SB → SC, the previous contents of SC are lost. Also, when LQID pops the stack, the previously pushed contents of SB are left in SC. The net result is that the contents of SB are placed in SC (SB → SC). Note that LQID takes two instruction cycle times to execute.

SKT Instruction

The SKT (Ship On Timer) instruction tests the state of an internal 10-bit time-base counter. This counter divides the instruction cycle clock frequency by 1024 and provides a latched indication of counter overflow. The SKT instruction tests this latch, executing the next program instruction if the latch is not set. If the latch has been set since the previous test, the next program instruction is skipped and the latch is reset. The features associated with this instruction, therefore, allow the ET9420/9421/9422 to generate its own time-base for real-time processing rather than relying on an external input signal.

For example, using a 2.097 MHz crystal as the time-base to the clock generator, the instruction cycle clock frequency will be 131 kHz (crystal frequency ÷ 16) and the binary counter output pulse frequency will be 128 Hz. For time-of-day or similar real-time processing, the SKT instruction can call a routine which increments a "seconds" counter every 128 ticks.

Instruction Set Notes

a. **The first word of a ET9420/9421/9422 program (ROM address 0) must be a CLRA (Clear A) instruction.**

b. Although skipped instructions are not executed, one instruction cycle time is devoted to skipping each byte of the skipped instruction. Thus all program paths take the same number of cycle times whether instructions are skipped or executed except JID and LQID. LQID and JID take two cycle times if executed and one if skipped.

c. The ROM is organized into 16 pages of 64 words each. The Program Counter is an 10-bit binary counter, and will count through page boundaries. If a JP, JSRP, JID or LQID instruction is located in the last word of a page, the instruction operates as if it were in the next page. For example: a JP located in the last word of a page will jump to a location in the next page. Also, a LQID or JID located in the last word of page 3, 7, 11 or 15 will access data in the next group of four pages.

OPTION LIST

The ET9420/9421/9422 mask programmable options are assigned numbers which correspond with the ET9420 pins.

The following is a list of ET9420 options. When specifying a ET9421 or ET9422 chip, Options 9, 10, 19, 20 and 29 must all be set to zero. When specifying a ET9422 chip. Options 21, 22, 27 and 28 must also be zero, and Option 2 must not be a 1. The options are programmed at the same time as the ROM pattern to provide the user with the hardware flexibility to interface to various I/O components using little or no external circuitry.

Option 1 = 0: Ground Pin — no options available

Option 2: CKO Pin
= 0: clock generator output to crystal
(0 not available if opticn 3 = 4 or 5)
= 1: pin is RAM power supply (V_R) input
(Not available on ET9422, ET9322)
= 2: general purpose input with load device
= 3: multi-COP SYNC input
= 4: general purpose Hi Z input

Option 3: CKI Input
= 0: crystal input divided by 16
= 1: crystal input divided by 8
= 2: TTL external clock input divided by 16
= 3: TTL external clock input divided by 8
= 4: single-pin RC controlled oscillator (÷4)
= 5: Schmitt trigger clock input (÷4)

Option 4: RESET Pin
= 0: Load devices to V_{CC}
= 1: Hi-Z input

Option 5: L_7 Driver
= 0: Standard output (figure 9D)
= 1: Open-Drain output (E)
= 2: LED direct drive output (F)
= 3: TRI-STATE®push-pull output (G)

Option 6: L_6 Driver
same as Option 5

Option 7: L_5 Driver
same as Option 5

Option 8: L_4 Driver
same as Option 5

Option 9: IN_1 Input
= 0: load device to V_{CC} (H)
= 1: Hi-Z input (I)

Option 10: IN_2 Input
same as Option 9

Option 11 = 0: V_{CC} Pin — no options available

Option 12: L_3 Driver
same as Option 5

Option 13: L_2 Driver
same as Option 5

Option 14: L_1 Driver
same as Option 5

Option 15: L_0 Driver
same as Option 5

Option 16 : SI Input
same as Option 9

Option 17: SO Driver
= 0: standard output (A)
= 1: open-drain output (B)
= 2: push-pull output (C)

Option 18: SK Driver
same as Option 17

Option 19: IN_0 Input
same as Option 9

Option 20: IN_3 Input
same as Option 9

Option 21: G_0 I/O Port
= 0: Standard output (A)
= 1: Open-Drain output (B)

Option 22: G_1 I/O Port
same as Option 21

Option 23: G_2 I/O Port
same as Option 21

Option 24: G_3 I/O Port
same as Option 21

Option 25: D_3 Output
= 0: Standard output (A)
= 1: Open-Drain output (B)

Option 26: D_2 Output
same as Option 25

Option 27: D_1 Output
same as Option 25

Option 28: D_0 Output
same as Option 25

Option 29: Chip Function
= 0: normal operation
= 1: MICROBUS® option

Option 30: Chip Bonding
= 0: ET9420 (28-pin device)
= 1: ET9421 (24-pin device)
= 2: 28 and 24-pin versions
= 3: ET9422 (20-pin device)
= 4: 28- and 20-pin versions
= 5: 24- and 20-pin versions
= 6: 28-, 24-, and 20-pin versio

Option 31: IN Input Levels
= 0: normal input levels
= 1: Higher voltage input leve
("0" = 1.2V, "1" = 3.6V)

Option 32: G Input Levels
same as Option 31

Option 33: L Input Levels
same as Option 31

Option 34: CKO Input Levels
same as Option 31

Option 35: SI Input Levels
same as Option 31

TEST MODE (Non-Standard Operation)

The SO output has been configured to provide for standard test procedures for the custom-programmed ET9420. With SO forced to logic "1", two test modes are provided, depending upon the value of SI :

a. RAM and Internal Logic Test Mode (SI = 1)

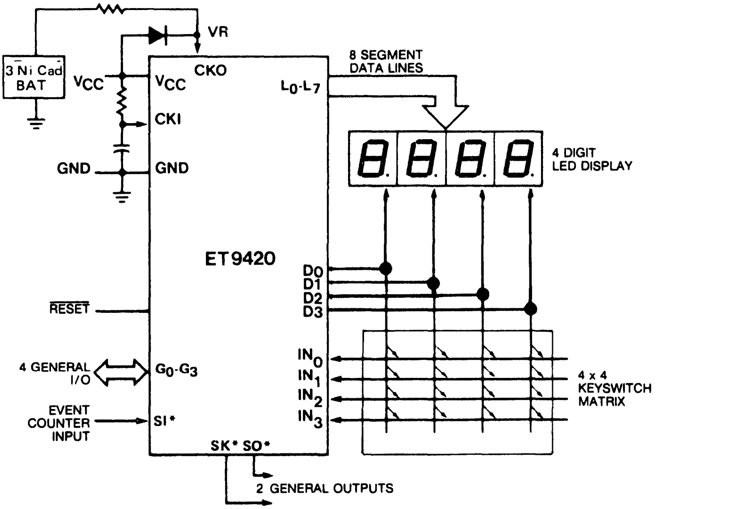
b. ROM Test Mode (SI = 0)

These special test modes should not be employed by the user; they are intended for manufacturing test only.

APPLICATION EXAMPLE : ET9420 General Controller

Figure 9 shows an interconnect diagram for a ET9420 used as a gene-al controller. Operation of the system is as follows :

1. The L_7-L_0 outputs are configured as LED Direct Drive outputs, allowing direct connection to the segments of the display.
2. The D_3-D_0 outputs drive the digits of the multiplexed display directly and scan the columns of the 4 x 4 keyboard matrix.
3. The IN_3-IN_0 inputs are used to input the 4 rows of the keyboard matrix. Reading the IN lines in conjunction with the current value of the D outputs allows detection, debouncing, and decoding of any one of the 16 keyswitches.
4. CKI is configured as a single-pin oscillator input allowing system timing to be controlled by a single-pin RC network. CKO is therefore available for use as a V_R RAM power supply pin. RAM data integrity is thereby assured when the main power supply is shut down (see RAM Keep-Alive Option description).
5. SI is selected as the input to a binary counter input. With SIO used as a binary counter, SO and SK ca- be used as general purpose outputs.
6. The 4 bidirectional G I/O ports (G_3-G_0) are available for use as required by the user s application.

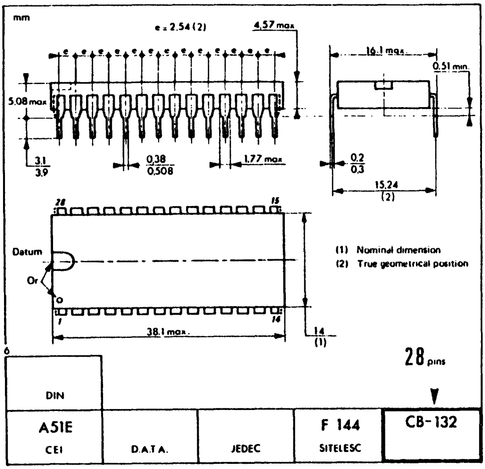


* SI, SO and SK may also be used for serial I/O

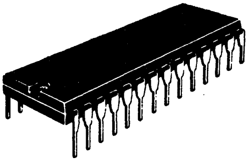
FIGURE 11 – ET9420 KEYBOARD/DISPLAY INTERFACE

COPS, MICROWIRE MICROBUS and TRI-STATE are registered trademarks of National Semiconductor Corp.

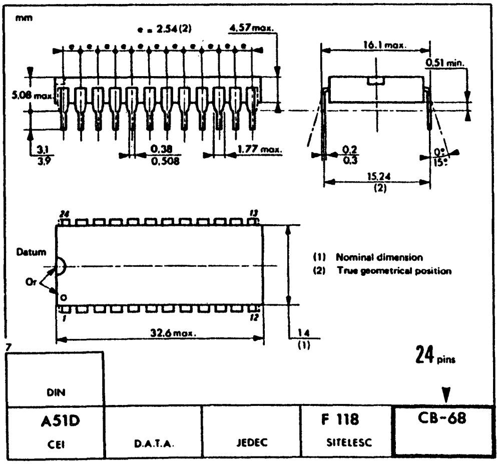
PHYSICAL DIMENSIONS



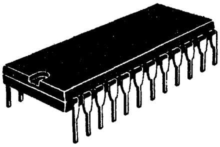
CB-132



N SUFFIX
PLASTIC PACKAGE

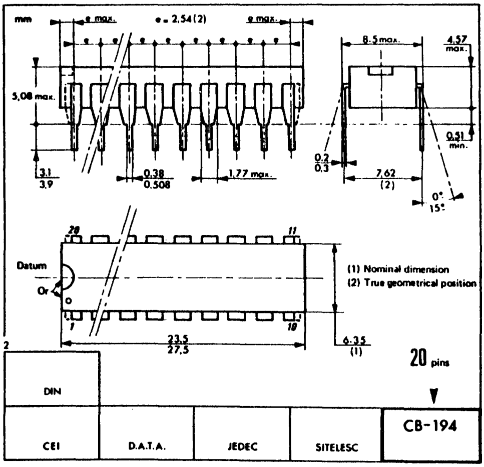


CB-68

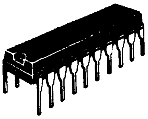


N SUFFIX
PLASTIC PACKAGE

PHYSICAL DIMENSIONS



CB-194



N SUFFIX
PLASTIC PACKAGE

These specifications are subject to change without notice.
Please inquire with our sales offices about the availability of the different products.

THOMSON
COMPONENTS
SEMICONDUCTEURS

ETC9420/21/22 • ETC9320/21/22
ETC9444/45 • ETC9344/45

SINGLE CHIP MICROCONTROLLERS

The ETC 9420, C 9421, C 9422, C 9320, C 9321, C 9322, and ETC 9445, C 9344, C 9345 fully static single-chip CMOS microcontrollers are fully compatible with the COPS® family, fabricated using double-poly, silicon gate complementary MOS technology. These Controller Oriented Processors are complete microcontrollers containing all system timing, internal logic, ROM, RAM and I/O necessary to implement dedicated control functions in a variety of applications.

Features include single supply operation, a variety of output configuration options, with an instruction set, internal architecture and I/O scheme designed to facilitate keyboard input, display output and BCD data manipulation. The ETC 9420, C 9444 are 28-pin chips. The ETC 9421, C 9445 are 24-pin versions (4 inputs removed) and C 9422 is a 20-pin version with 15 I/O lines. Standard test procedures and reliable high-density fabrication techniques provide the medium to large volume customers with a customized microcontroller at a low end -product cost.

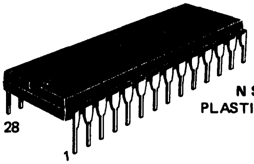
These microcontrollers are appropriate choices in many demanding control environments especially those with human interface.

- Lowest power dissipation (50 μW typical).
- Power saving IDLE state and HALT mode.
- Fully Static (can turn off the clock).
- 2 K x 8 ROM, 128 x 4 RAM (ETC 9444, C 9445).
- 1 K x 8 ROM, 64 x 4 RAM (ETC 9420, C 9421, C 9422).
- True vectored interrupt, plus restart.
- 3-level subroutine stack.
- 4 μsec intruction time, plus software selectable clocks.
- 23 I/Olines (ETC 9444, C 9420).
- Single supply operation (2.4 V to 5.5 V).
- Programmable read/write 8-bit timer/event counter.
- Internal binary counter register with MICROWIRE® serial I/O Capability.
- General purpose and tri-state® outputs.
- LSTTL/C MOS compatible.
- MICROBUS® compatible.
- Software/hardware compatible with other members of ET 9400 family.
- Extended temperature range devices ETC 9320/ETC 9321, ETC 9322 and ETC 9344/ETC 9345 (— 40° C to +85°C).
- Military devices (—55°C to +125°C) to be available.

CMOS

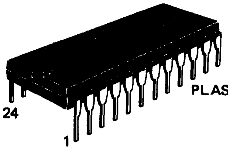
CASES

ETC9420 • ETC9444
ETC9320 • ETC9344



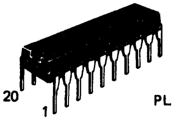
N SUFFIX
PLASTIC PACKAGE

ETC9421 • ETC9344
ETC9321 • ETC9345



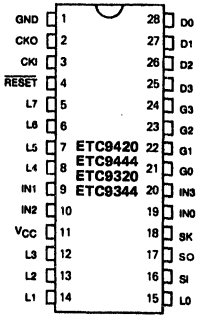
N SUFFIX
PLASTIC PACKAGE

ETC9422 • ETC9322



N SUFFIX
PLASTIC PACKAGE

PIN ASSIGNMENT



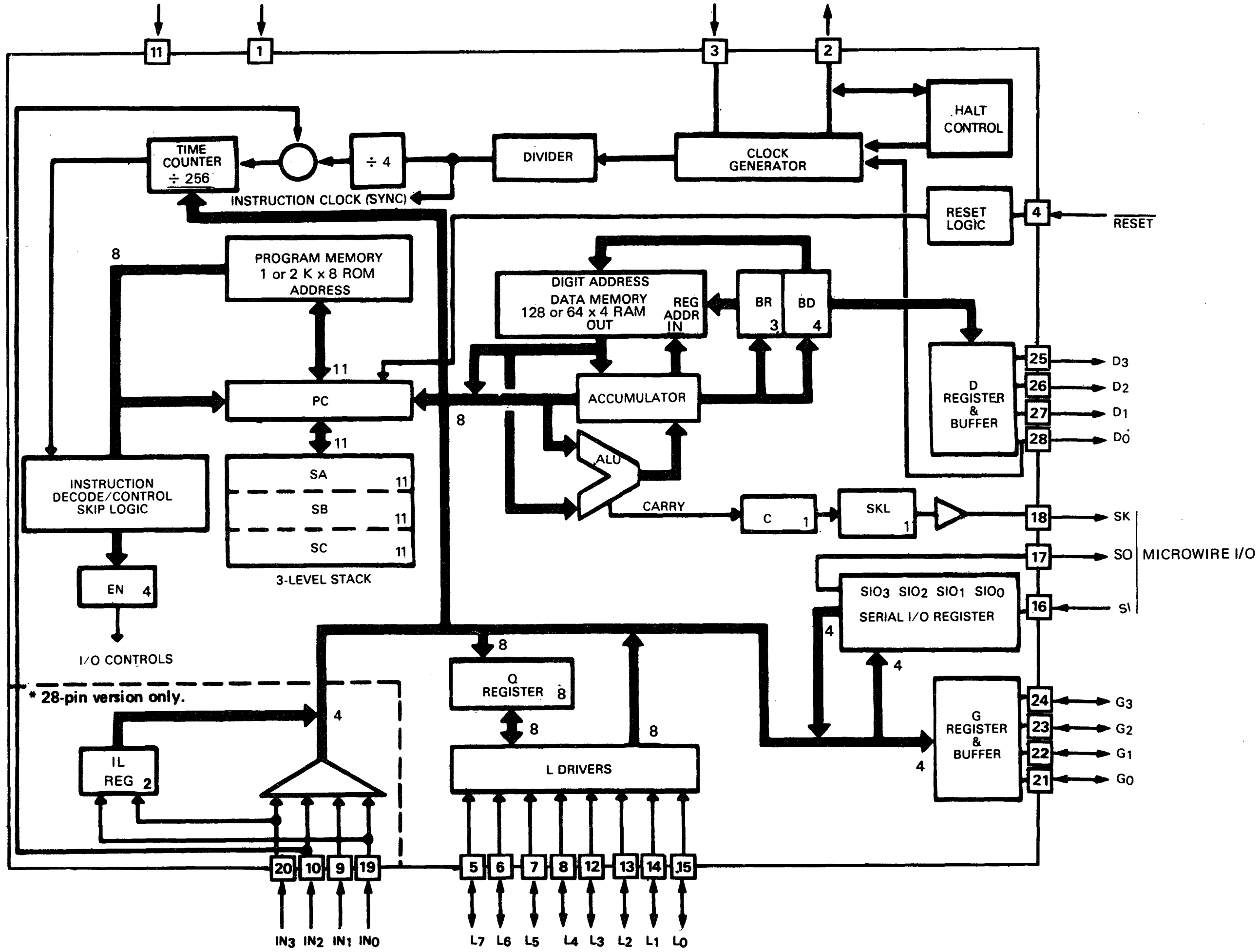


FIGURE 1 – BLOCK DIAGRAM (28-pin version)

ETC9420, ETC9421, ETC9422 • ETC9444, ETC9445

ABSOLUTE MAXIMUM RATINGS

Supply voltage (Vcc)	6 V
Voltage at any pin	−0.3 V to V_{CC} + 0.3 V
Total Allowable Source current	25 mA
Total Allowable Sink Current	25 mA
Operating Temperature Range	0° C to + 70° C
Storage Temperature Range	−65° C to + 150° C
Lead Temperature (soldering 10 seconds)	300° C

Absolute maximum ratings indicate limits beyond which damage to the device may occur. DC and AC electrical specifications are note insured when operating the device at absolute maximum ratings.

DC ELECTRICAL CHARACTERISTICS 0°C ⩽ T_A ⩽ +70°C (Unless otherwise specified)

Parameter	Conditions	Min	Max	Units
Operating Voltage		2.4	5.5	V
Power Supply Ripple (Note 5)	Peak to Peak	–	0.1 V_{CC}	V
Supply Current (Note 1)	V_{CC} = 2.4 V, tc = 64 µs	-	120	µA
	V_{CC} = 5.0 V, tc = 16 µs	-	700	µA
	V_{CC} = 5.0 V, tc = 4 µs	-	3000	µA
	(tc = instruction cycle time)			
Halt Mode Current (Note 2)	V_{CC} = 5.0 V, Fin = 0 kHz	-	40	µA
	V_{CC} = 2.4 V, Fin = 0 kHz	-	12	µA
Input Voltage levels				
$\overline{RESET}$ CKI (RC or crystal opt.)				
Do (Clock input)				
Logic High		0.9 V_{CC}	–	V
Logic Low			0.1 V_{CC}	V
All other inputs				
Logic High		0.7 V_{CC}	–	
Logic Low		–	0.2 V_{CC}	V
Input Pull-up current	V_{CC} = 4.5 V, V_{IN} = 0	30	330	µA
HI-Z input leakage		−1	+1	µA
Input capacitance (Note 4)		–	7	pF
Output Voltage Levels	Standard Outputs			
LSTTL Operation	V_{CC} = 5.0 V ± 5 %			
Logic High	I_{OH} = −100 µA	2.7	–	V
Logic Low	I_{OL} = 400 µA		0.4	V
CMOS Operation				
Logic High	I_{OH} = −10 µA	V_{CC}−0.2	–	V
Logic Low	I_{OL} = 10 µA	–	0.2	V
Output current Level (except CKO)				
Sink (Note 6)	V_{CC} = 4.5 V, Vout = V_{CC}	1.2	–	mA
	V_{CC} = 2.4 V, Vout = V_{CC}	0.2	–	mA
Source (standard option)	V_{CC} = 4.5 V, Vout = 0V	0.5	–	mA
	V_{CC} = 2.4 V, Vout = 0V	0.1	–	mA
Source (Low current option)	V_{CC} = 4.5 V, Vout = 0V	30	330	µA
	V_{CC} = 2.4 V, Vout = 0V	6	80	µA
CKO Current level (As clock out)				
Sink ÷ 4	V_{CC} = 4.5 V, CKI = V_{CC} V_{OUT} = V_{CC}	0.3	–	mA
÷ 8		0.6	–	mA
÷ 16		1.2	–	mA
Source ÷ 4	V_{CC} = 4.5 V, CKI = 0V, V_{OUT} = 0V	0.3	–	mA
÷ 8		0.6	–	mA
÷ 16		1.2	–	mA
Allowable Sink Source current per pin (Note 6)		–	5	mA
Allowable loading on CKO (as HALT)		–	100	pF
Current needed to over-ride HALT (Note 3)				
To continue	V_{CC} = 4.5 V, Vin = 0.2 V_{CC}	-	0.7	mA
To halt	V_{CC} = 4.5 V, Vin = 0.7 V_{CC}	–	1.6	mA
TRI-STATE or open drain Leakage current		−2.5	+2.5	µA

AC ELECTRICAL CHARACTERISTICS 0°C < T_A < +70°C (Unless otherwise specified)

Parameter	Conditions	Min	Max	Units
Instruction cycle time (tc)	$V_{CC} \geq 4.5$ V	4	DC	µs
	4.5 V > $V_{CC} \geq 2.4$ V	16	DC	µs
Operating CKI Frequency				
÷ 4 mode	$V_{CC} \geq 4.5$ V	DC	1.0	MHz
÷ 8 mode		DC	2.0	MHz
÷ 16 mode		DC	4.0	MHz
÷ 4 mode	4.5 V > $V_{CC} \geq 2.4$ V	DC	250	kHZ
÷ 8 mode		DC	500	kHZ
÷ 16 mode		DC	1.0	MHz
Duty cycle (Note 4)	F1 = 4 MHz	40	60	%
Rise time (Note 4)	F1 = 4 MHz ext. dock	-	60	ns
Fall time (Note 4)	F1 = 4 MHz ext. dock	-	40	ns

Instruction cycle time (RC oscillator) (Note 4)	R = 30 K V_{CC} = 5 V
	C = 82 pF, (÷ 4 Mode)
		8	16	µs

INPUTS (fig. 3)				
t_{SETUP}	G inputs $V_{CC} \geq 4.5$ V	tc/4 +.7	-	µs
	SI input	0.3	-	µs
	All others	1.7	-	µs
t_{HOLD}	$V_{CC} \geq 4.5$ V	0.25	-	µs
	4.5 V > $V_{CC} \geq 2.4$ V	1.0	-	µs
OUTPUT PROPAGATION DELAY	Vout = 1.5 V, CL = 100 pF, R_L = 5 K			
t_{PD1}, t_{PDO}	$V_{CC} \geq 4.5$ V	-	1.0	µs
t_{PD1}, t_{PDO}	4.5 V > $V_{CC} \geq 2.4$ V	-	4.0	µs
MICROBUS® TIMING	C_L = 50 pF, V_{CC} = 5 V ± 5 %			µs
Read operation (fig. 4)				
Chip select stable before $\overline{RD}$ - t_{CSR}		65	-	ns
Chip select hold time for $\overline{RD}$ - t_{RCS}		20	-	ns
$\overline{RD}$ pulse width - t_{RR}		400	-	ns
Data delay from $\overline{RD}$ - t_{RD}		-	375	ns
$\overline{RD}$ to data floating - t_{DF} (note 4)		-	250	ns
Write operation (fig. 5)				
Chip select stable before $\overline{WR}$ - t_{CSW}		65	-	ns
Chip select hold time for $\overline{WR}$ - t_{WCS}		20	-	ns
$\overline{WR}$ pulse width - t_{WW}		400	-	ns
Data set-up time for $\overline{WR}$ - t_{DW}		320	-	ns
Data hold time for $\overline{WR}$ - t_{WD}		100	-	ns
INTR transition time for $\overline{WR}$ - t_{WI}		-	700	ns

Note 1 : Supply current is measured after running for 2000 cycle times with a square-wave clock on CKI, CKO open, and all other pins pulled up to V_{CC} with 20k resistors. See current drain equation on page 17.

Note 2 : The HALT mode will stop CKI from oscillating in the RC and crystal configurations. Test conditions : all inputs tied to V_{CC}, L lines in TRI-STATE mode and tied to ground, all output low and tied to ground.

Note 3 : When forcing HALT, current is only needed for a short time (approx. 200 ns) to flip the HALT flip-flop.

Note 4 : This parameter is only sampled and not 100 % tested.

Note 5 : Voltage change must be less than 0.5 volt in a 1 ms period.

Note 6 : SO output sink current must be limited to keep V_{OL} below 0.2 V_{CC} when port is running in order to prevent entering test mode.

ETC9320, ETC9321, ETC9322 • ETC9344, ETC9345

ABSOLUTE MAXIMUM RATINGS

Supply voltage	6 V
Voltage at any pin	− 0.3 V to V_{CC} + 0.3 V
Total Allowable Source current	25 mA
Total Allowable Sink Current	25 mA
Operating Temperature Range	− 40° C to + 85° C
Storage Temperature Range	− 65° C to + 150° C
Lead Temperature (soldering 10 seconds)	300° C

Absolute maximum ratings indicate limits beyond which damage to the device may occur. DC and AC electrical specifications are note insured when operating the device at absolute maximum ratings.

DC ELECTRICAL CHARACTERISTICS — 40° C $\leq T_A \leq$ + 85° C (Unless otherwise specified)

Parameter	Conditions	Min	Max	Units
Operating Voltage		3.0	5.3	V
Power Supply Ripple (Note 5)	Peak to Peak	−	0.1 V_{CC}	V
Supply Current (Note 1)	V_{CC} = 3.0 V, t_C = 64 µs	−	180	µA
	V_{CC} = 5.0 V, tc = 16 µs	−	800	µA
	V_{CC} = 5.0 V, tc = 4 µs (tc = instruction cycle time)	−	3600	µA
Halt Mode Current	V_{CC} = 5.0 V, Fin = 0 kHz	−	60	µA
(Note 2)	V_{CC} = 3.0 V, Fin = 0 kHz	−	30	µA
Input Voltage levels $\overline{\text{RESET}}$ CKI (RC or crystal opt.) Do (Clock input)				
Logic High		0.9 V_{CC}	−	V
Logic Low		−	0.1 V_{CC}	V
All other inputs				
Logic High		0.7	−	
Logic Low		−	0.2 V_{CC}	V
Input Pull-up current	V_{CC} = 4.5 V, V_{IN} = 0	30	440	µA
HI-Z input leakage		−2	+2	µA
Input capacitance (Note 4)		−	7	pF
Output Voltage Levels	Standard Outputs			
LSTTL Operation	V_{CC} = 5.0 V ± 5 %			
Logic High	I_{OH} = −100 µA	2.7	−	V
Logic Low	I_{OL} = 400 µA	−	0.4	V
CMOS Operation				
Logic High	I_{OH} = −10 µA	V_{CC} −0.2	−	V
Logic Low	I_{OL} = 10 µA	−	0.2	V
Output current Level (except CKO)				
Sink (Note 6)	V_{CC} = 4.5 V, Vout = V_{CC}	1.2	−	mA
	V_{CC} = 3.0 V, Vout = V_{CC}	0.2	−	mA
Source (standard option)	V_{CC} = 4.5 V, Vout = 0V	0.5	−	mA
	V_{CC} = 3.0 V, Vout = 0V	0.1	−	mA
Source (Low current option)	V_{CC} = 4.5 V, Vout = 0V	30	440	µA
	V_{CC} = 3.0 V, Vout = 0V	8	200	µA
CKO Current level (As clock out)				
Sink ÷ 4		0.3	−	mA
÷ 8	V_{CC} = 4.5 V, CKI = V_{CC}, Vout = V_{CC}	0.6	−	mA
÷ 16		1.2	−	mA
Source ÷ 4		0.3	−	mA
÷ 8	V_{CC} = 4.5 V, CKI = 0V, Vout = 0V	0.6	−	mA
÷ 16		1.2	−	mA
Allowable Sink / Source current per pin (Note 6)		−	5	mA
Allowable loading on CKO (as HALT)		−	100	pF
Current needed to over-ride HALT (Note 3)				
To continue	V_{CC} = 4.5 V, Vin = 0.2 V_{CC}	−	0.9	mA
To halt	V_{CC} = 4.5 V, Vin = 0.7 V_{CC}	−	2.1	mA
TRI-STATE or open drain Leakage current		−5	+5	µA

AC ELECTRICAL CHARACTERISTICS - 40°C < T_A < + 85°C (Unless otherwise specified)

Parameter	Conditions	Min	Max	Units
Instruction cycle time (tc)	$V_{CC} \geq$ 4.5 V 4.5 V > $V_{CC} \geq$ 3.0 V	4 16	DC DC	µs µs
Operating CKI Frequency				
÷ 4 mode ÷ 8 mode ÷ 16 mode	$V_{CC} \geq$ 4.5 V	DC DC DC	1.0 2.0 4.0	MHz MHz MHz
÷ 4 mode ÷ 8 mode ÷ 16 mode	4.5 V > $V_{CC} \geq$ 3.0 V	DC DC DC	250 500 1.0	kHZ kHZ MHz
Duty cycle (Note 4)	F1 = 4 MHz	40	60	%
Rise time (Note 4)	F1 = 4 MHz ext. clock	–	60	ns
Fall time (Note 4)	F1 = 4 MHz ext. clock	–	40	ns

Instruction cycle time (RC oscillator) (Note 4)	R=30 K, V_{CC}=5 V
	C=82 pF, (÷ 4 Mode)
		8	16	µs

INPUTS (fig. 3)				
t_{SETUP}	G inputs SI input All others — $V_{CC} \geq$ 4.5 V	(tc/4)+.7 0.3 1.7	– – –	µs µs µs
t_{HOLD}	$V_{CC} \geq$ 4.5 V 4.5 V > $V_{CC} \geq$ 3.0 V	0.25 1.0	– –	µs µs
OUTPUT				
PROPAGATION DELAY	Vout=1.5 V, CL=100 pF, RL=5k			
t_{PD1}, t_{PD0}	$V_{CC} \geq$ 4.5 V	–	1.0	µs
t_{PD1}, t_{PD0}	4.5 V > $V_{CC} \geq$ 3.0 V	–	4.0	µs
MICROBUS® TIMING	C_L=50 pF, V_{CC}=5 V ± 5 %			
Read operation (fig. 4)				
Chip select stable before $\overline{RD}$ - t_{CSR}		65	–	ns
Chip select hold tome for $\overline{RD}$ - t_{RCS}		20	–	ns
$\overline{RD}$ pulse width - t_{RR}		400	–	ns
Data delay from $\overline{RD}$ - t_{RD}		–	375	ns
$\overline{RD}$ to data floating - t_{DFI}(note 4)		–	250	ns
Write operation (fig. 5)				
Chip select stable before $\overline{WR}$ - t_{CSW}		65	–	ns
Chip select hold time for $\overline{WR}$ - t_{WCS}		20	–	ns
$\overline{WR}$ pulse width - t_{WW}		400	–	ns
Data set-up time for $\overline{WR}$ - t_{DW}		320	–	ns
Data hold time for $\overline{WR}$ - t_{WD}		100	–	ns
$\overline{INTR}$ transition time for $\overline{WR}$ - t_{WI}		–	700	ns

Note 1 : Supply current is measured after running for 2000 cycle times with a square-wave clock on CKI, CKO open, and all other pins pulled up to V_{CC} with 20k resistors. See current drain equation on page 17

Note 2 : The HALT mode will stop CKI from oscillating in the RC and crystal configurations. Test conditions : all inputs tied to V_{CC}, L lines in TRI-STATE mode and tied to ground, all output low and tied to ground.

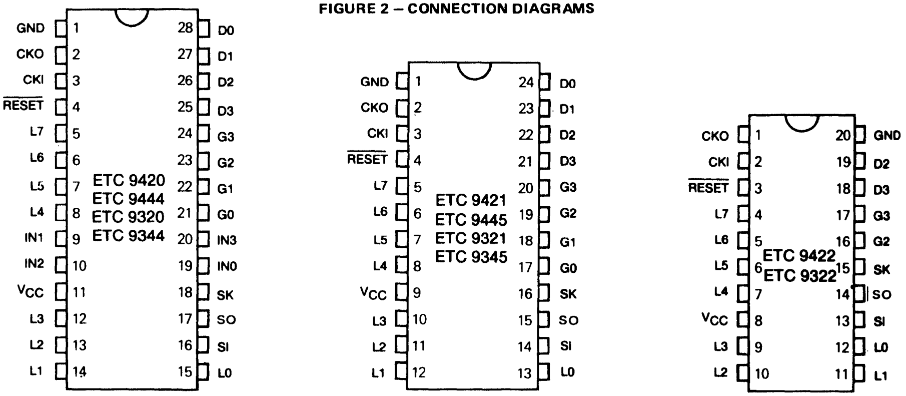
Note 3 : When forcing HALT, current is only needed for a short time (approx. 200 ns) to flip the HALT flip-flop.

Note 4 : This parameter is only sampled and not 100 % tested.

Note 5 : Voltage change must be less than 0.5 volt in a 1 ms period.

Note 6 : SO output sink current must be limited to keep V_{OL} below 0.2 V_{CC} when port is running in order to prevent entering test mode.

FIGURE 2 – CONNECTION DIAGRAMS



L_7-L_0	8 bit bidirectional I/O ports with TRI-STATE®
G_3-G_0	4 bit bidirectional I/O ports
D_3-D_1	3 bit general purpose outputs
D_0	General purpose output or oscillator input
IN_3-IN_0	4-bit input port (ETC 9420 or 9444 only)
SI	Serial input
SO	Serial output
SK	Logic-controlled clock or general purpose output
CKI	System oscillator input
CKO	System oscillator output (or general purpose Input or Halt I/O Port)
RESET	System reset input
V_{CC}	Power supply
GND	Ground

FUNCTIONAL DESCRIPTION

For ease of reading only the ETC 9420/9421/9422/9444/9445 are referenced. However, all such references apply equally to ETC 9320/9321/9322/9344/9345.

The internal architecture is shown in Figure 1. Data paths are illustrated in simplified form to depict how the various logic elements communicate with each other in implementing the instruction set of the device. Positive logic is used. When a bit is set, it is a logic "1" when it is reset, it is a logic "0".

Program memory

Program Memory consists of ROM, 1024 bytes for the ETC 9420/C 9421/ C 9422 and 2048 bytes for the ETC 9444/C 9445. These bytes of ROM may be program instructions, constants or ROM addressing data.

ROM addressing is accomplished by a 11-bit PC register which selects one of the 8-bit words contained in ROM. A new address is loaded into the PC register during each instruction cycle. Unless the instruction is a transfer of control instruction, the PC register is loaded with the next sequential 11-bit binary count value.

Three levels of subroutine nesting are implemented by a three level deep stack. Each subroutine call or interrupt pushes the next PC address into the stack. Each return pops the stack back into the PC register.

Data memory

Data memory consists of a 512-bit RAM for the ETC 9444/C 9445, organized as 8 data registers of 16 x 4-bit digits. RAM addressing is implemented by a 7-bit B register whose upper 3 bits (Br) select 1 of 8 data registers and lower 4 bits (Bd) select 1 of 16 4-bit digits in the selected data register.

Data memory consists of a 256-bit RAM for the ETC 9420/C 9421/C 9422, organized as 4 data registers of 16 × 4 bits digits. The B register is 6 bits long. Upper 2 bis (Br) select 1 of 4 data registers and lower 4 bits (Bd) select 1 of 16 4-bit digits in the selected data register. While the 4-bit contents of the selected RAM digit (M) are usually loaded into or from, or exchanged with the A register (accumulator), they may also be loaded into the Q latches or loaded from the L ports. RAM addressing may also be performed directly by the LDD and XAD instructions based upon the immediate operand field of these instructions.

The Bd register also serves as a source register for 4-bit data sent directly to the D outputs.

Internal Logic

The processor contains its own 4-bit A register (accumulator) which is the source and destination register for most I/O, arithmetic, logic, and data memory access operations. It can also be used to load the Br and Bd portions of the B register, to load and input 4 bits of the 8-bit Q latch, to input 4 bits of a ROM word, L I/O ports data, to input 4-bit G, or IN ports, and to perform data exchanges with the SIO register.

A 4-bit adder performs the arithmetic and logic functions, storing the results in A. It also outputs a carry bit to the 1-bit C register, most often employed to indicate arithmetic overflow. The C register in conjunction with the XAS instruction and the EN register, also serves to control the SK output.

The 8-bit T counter is a binary up counter which can be loaded to and from M and A using CAMT and CTMA instructions. This counter may be operated in two modes depending on a mask-programmable option : as a timer or as an external event counter. When the T counter overflows, an overflow flag will be set (see SKT and IT instructions below). The T counter is cleared on reset. A functional block diagram of the timer/counter is illustrated in *Figure 10 a.*

Four general-purpose inputs, IN3-IN0, are provided. IN1, IN2 and IN3 may be selected, by a mask-programmable option as Read Strobe, Chip Select, and Write Strobe inputs, respectively, for use in MICROBUS application.

The D register provides 4 general-purpose outputs and is used as the destination register for the 4-bit contents of Bd. In the dual clock mode. D0 latch controls the clock selection (see dual oscillator below).

The G register contents are outputs to a 4-bit general-purpose bidirectional I/O port. G0 may be mask-programmed as an output for MICROBUS applications.

The Q register is an internal, latched, 8-bit register, used to hold data loaded to or from M and A, as well as 8-bit data from ROM. Its contents are outputted to the L I/O ports when the L drivers are enabled under program control (see LEI instruction). With the MICROBUS®option selected, Q can also be loaded with the 8-bit contents of the L I/O ports upon the occurence of a write strobe from the host CPU.

The **8 L drivers,** when enabled, output the contents of latched Q data to the L I/O ports. Also, the contents of L may be read directly into A and M. As explained above, the MICROBUS® option allows L I/O port data to be latched into the Q register.

The **SIO register** functions as a 4-bit serial-in/serial-out shift register for MICROWIRE I/O and peripherals, or as a binary counter (depending on the contents of the EN register). Its contents can be exchanged with A.

The **XAS instruction** copies C into the SKL Latch. In the counter mode, SK is the output of SKL ; in the shift register mode, SK is a sync clock, inhibited when SKL is a logic "O".

The **"EN register** is an internal 4-bit register loaded under programm control by the LEI instruction. The state of each bit of this register selects or deselects the particular feature associated with each bit of the EN register (EN$_3$-EN$_0$).

0. The least significant bit of the enable register, EN$_0$, selects the SIO register as either a 4-bit shift register or as a 4-bit binary counter. With EN$_0$ set, SIO is as asynchronous binary counter, DECREMENTING its value by one upon each low going pulse ("1" to "0") occuring on the SI input. Each pulse must be at least 2 (two) instruction cycles wide. SK outputs the value of SKL. The SO output is equal to the value of EN$_3$. With EN$_0$ reset, SIO is a serial shift register, shifting left each instruction cycle time. The data present at SI is shifted into the least significant bit of SIO, SO can be enabled to output the most significant bit of SIO each instruction cycle time. The SK output SKL ANDed with the instruction cycle clock.

1. With EN$_1$ set interrupt is enabled. Immediatly following an interrupt, EN$_1$ is reset to disable further interrupts.

2. With EN$_2$ set, the L drivers are enabled to output the data in Q to the L I/O ports. Resetting EN$_2$ disables the L drivers, placing the L I/O ports in a high impedance input state.

3. EN$_3$, in conjunction with EN$_0$, affects the SO output. With EN$_0$ set(binary counter option selected), SO will output the value loaded into EN$_3$. With EN$_0$ reset (serial shift register option selected), setting EN$_3$ enables SO as the output of the SIO shift register, outputting serial shifted data each instruction time. Resetting EN$_3$ with the serial shift register option selected, disables SO as the shift register output ; data continues to be shifted through SIO and can be exchanged with A via an XAS instruction but SO remains reset to "0".

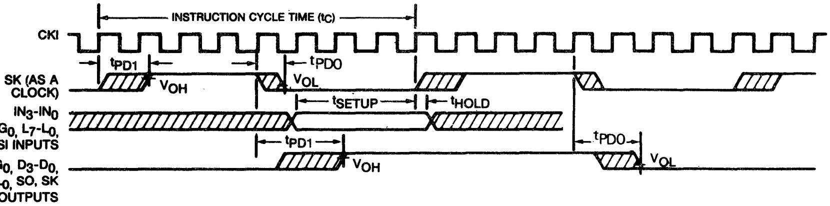

INSTRUCTION CYCLE TIME (tC)
CKI
SK (AS A CLOCK)
IN3-IN0 G3-G0, L7-L0, CKO & SI INPUTS
G3-G0, D3-D0, L7-L0, SO, SK OUTPUTS
tPD1
tPD0
VOH
VOL
tSETUP
tHOLD
tPD1
VOH
tPD0
VOL

FIGURE 3 – INPUT/OUTPUT TIMING DIAGRAMS (Divide-by-8 Mode)

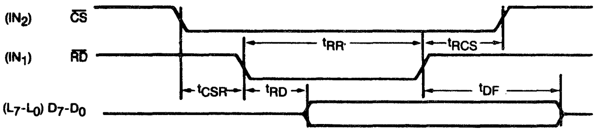


FIGURE 4 – MICROBUS READ OPERATION TIMING

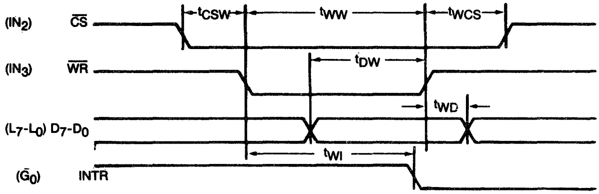


FIGURE 5 – MICROBUS WRITE OPERATION TIMING

FUNCTIONAL DESCRIPTION (continued)

TABLE 1 – ENABLE REGISTER MODES – BITS EN_0 AND EN_3

EN_0	EN_3	SIO	SI	SO	SK after XAS
0	0	Shift Register	Input to shift Register	0	If SKL = 1, SK = clock If SKL = 0, SK = 0
0	1	Shift Register	Input to Shift Register	Serial out	If SKL = 1, SK = clock If SKL = 0, SK = 0
1	0	Binary counter	Input to Counter	0	SK = SKL
1	1	Binary counter	Input to Counter	1	SK = SKL

Interrupt

The following features are associated with interrupt procedure and protocol and must be considered by the programmer when utilizing interrupts.

a. The interrupt, once acknowledged as explained below, pushes the next sequential program counter address (PC + 1) onto the stack. Any previous contents at the bottom of the stack are lost. The program counter is set to hex address OFF (the last word of page 3) and EN_1 is reset.

b. An interrupt will be acknowledged only after the following conditions are met :

1. EN_1 has been set.
2. A low-going pulse ("1" to "0") at least two instruction cycles wide occurs on the IN_1 input.
3. A currently executing instruction has been completed.
4. ALL successive transfer of control instructions and successive LBIs have been completed (e.g., if the main program is executing a JP instruction which transfers program control to another JP instruction, the interrupt will not be acknowledged until the second JP instruction has been executed.

c. Upon acknowledgement of an interrupt, the skip logic status is saved and later restored upon "popping" of the stack. For example, if an interrupt occurs during the execution of ASC (Add with Carry, Skip on Carry) instruction which results in carry, the skip logic status is saved and program control is transferred to the interrupt servicing routine at hex address OFF. At the end of the interrupt routine, a RET instruction is executed to "pop" the stack and return program control to the instruction following the original ASC. At this time, the skip logic is enabled and skips this instruction because of the previous ASC carry. **Subroutines should not be nested within the interrupt servicing routine** since their "popping" of the stack will enable any previously saved main program skips, interfering with the orderly execution of the interrupt routine.

d. The first instruction at hex address OFF must be a NOP.

e. A LEI instruction may be put immediately before the RET to re-enable interrupts.

MICROBUS interface

The ETC 9420/C 9444 has an option which allows it to be used as a peripheral microprocessor device, inputting and outputting data from to a host microprocessor (µP). IN_1,IN_2 and IN_3 general purposes input become **MICROBUS INTERFACE** read-strobe, chip-select, and write-strobe lines, respectively. IN_1 become $\overline{RD}$ – a logic "0" on this input will cause Q latch data to be enabled to the L ports for input to the µP. IN_2 becomes $\overline{CS}$ – a logic "0" on this line selects the ETC 9420/C 9444 as the uP peripheral device by enabling the operation of the $\overline{RD}$ and $\overline{WR}$ lines and allows for the selection of one of several peripheral components. IN_3 becomes $\overline{WR}$ – a logic "0" on this line will write bus data from the L ports to the Q latches for input to the ETC 9420/C 9444. G_0 becomes $\overline{INTR}$, a "ready" output, reset by a write pulse from the uP on the $\overline{WR}$ line, providing the "handshaking" capability necessary for asynchronous data transfer between the host CPU and the ETC 9420/C 9444

This option has been designed for compatibility with MICROBUS a standard interconnect system for 8-bit parallel data transfer between MOS/LSI CPUs and interfacing devices. (See MICROBUS National Publication). The functioning and timing relationships between the signal lines affected by this option are as specified for the MICROBUS interface, and are given in the AC electrical characteristics and shown in the timing diagrams (figure 4 and 5). Connection of the ETC 9420/C 9444 to the MICROBUS is shown in figure 6.

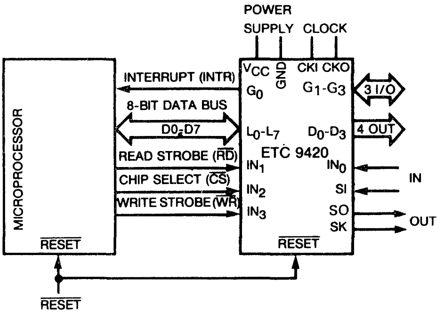
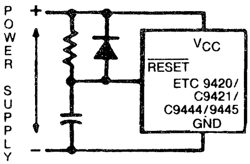


FIGURE 6 – MICROBUS®OPTION INTERCONNECT

Initialization

The internal reset logic will initialize the device upon power-up if the power supply rise time is less than 1 ms and if the operating frequency at CKI is greater than 32 kHz, otherwise the external RC network shown in Figure 7 must be connected to the $\overline{RESET}$ pin. The $\overline{RESET}$ pin is configured as a Schmitt trigger input. If not used, it should be connected to V_{CC}. Initialization will occur whenever a logic "0" is applied to the $\overline{RESET}$ input, providing it stays low for at least three instruction cycle times.

NOTE : if CKI clock is less than 32 kHz, the internal reset logic (option = 29 = 1) must be disabled and the external RC circuit must be used.



RC ≥ 5 x Power Supply Rise Time and RC ≥ 100 x CKI period

FIGURE 7 – POWER-UP CIRCUIT

FUNCTIONAL DESCRIPTION (continued)

Upon initialization, the PC register is cleared to 0 (ROM address 0) and the A, B, C, D, EN, IL, T and G registers are cleared. The SKL Latch is set, this enabling SK as a clock output. Data Memory (RAM) is not cleared upon initialization. The first instruction at address 0 must be a CLRA (clear A register).

Timer

There are two modes selected by mask option :

a. Time-base counter. In this mode, the instruction cycle frequency generated from CKI passes through a 2-bit divide-by-4 prescaler. The output of this prescaler increments the 8-bit T counter thus providing a 10-bit timer. The prescaler is cleared during execution of a CAMT instruction and on reset.

   For example, using a 4 MHz crystal with a divide-by-16 option, the instruction cycle frequency of 250 kHz increments the 10-bit timer every 4 µs. By presetting the counter and detecting overflow, accurate timeouts between 16 µs (4 counts) and 4.096 ms (1024 counts) are possible. Longer timeouts can be achieved by accumulating, under software control, multiple overflows.

b. External event counter. In this mode, a low-going pulse ("1" to "0") at least 2 instruction cycles wide on the IN2 input will increment the 8-bit T counter.

   Note : The IT instruction is not allowed in this mode.

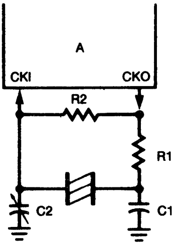
Halt mode

The ETC 9420/9421/ C 9422/ C 9444/ C 9445 is a FULLY STATIC circuit ; therefore, the user may stop the system oscillator at any time to halt the chip. The chip may be also halted by the HALT instruction or by forcing CKO high when it is used as an HALT/I/O port. Once in the HALT mode, the internal circuitry does not receive any clocksignal, and is therefore frozen in the exact state it was in when halted. All information is retained until continuing. The chip may be awakened by one of the two different methods.

1. Continue function — by forcing CKO low, if it is mask-programmed as an HALT I/O port, the system clock is reenabled and the circuit continues to operate from the point where it was stopped.
2. Restart — by forcing the $\overline{\text{RESET}}$ pin low (see initialization).

The HALT mode is the mininum power dissipation state.

NOTE : If the user has selected dual clock with D0 as external oscillator (option 30 = 2) AND the ETC 9444/C 9420 is running with the D0 clock, the HALT mode — either hardware or software — will NOT be entered. Thus, the user should switch to the CKI clock to HALT. Alternatively, the user may stop the D0 clock to minimize power.



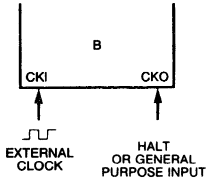
R/C controlled Oscillator

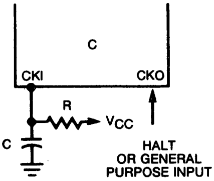
R	C	Cycle time	V_{CC}
15 k	82 pF	4 to 9 µs	≥ 4.5 V
30 k	82 pF	8 to 16 µs	≥ 4.5 V
60 k	100 pF	16 to 32 µs	2.4 to 4.5 V

Note : 15 k ≤ R ≤ 150 k ; 50 pF ≤ C ≤ 150 pF

Crystal or resonator

Crystal value	Component Values			
	R1	R2	C1 (pF)	C2 (pF)
32 kHz	220 k	20 M	30	6-36
455 kHz	5 k	10 M	80	40
2.086 MHz	2 k	1 M	30	6-36
4.0 MHz	1 k	1 M	30	6-36





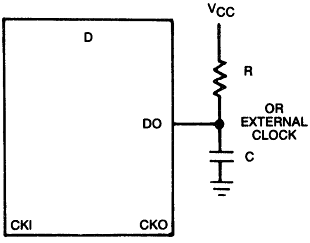


FIGURE 8 – OSCILLATOR COMPONENTS VALUES

This circuit and these values are for indication only. As the oscillator characteristics are not guaranteed, please consider and examine the circuit constants carefully on your application.

FUNCTIONAL DESCRIPTION (continued)

CKO Pin Options

a. Two-pin oscillator — (Crystal). See *Figure 9A.*

In a crystal controlled oscillator system, CKO is used as an output to the crystal network. The HALT mode may be entered by program control (HALT instruction) which forces CKO high, thus inhibiting the crystal network. The circuit can be awakened only by forcing the RESET pin to a logic "0" (restart).

b. One-pin oscillator — (RC or external). See *Figure 9B.*

If a one-pin oscillator system is chosen, two options are available for CKO :

- CKO can be selected as the HALT I/O port. In that case, it is an I/O flip-flop which is an indicator of the HALT status. An external signal can over-ride this pin to start and stop the chip. By forcing a high level to CKO, the chip will stop as soon as CKI is high and CKO output will stay high to keep the chip stopped if the external driver returns to high impedance state.

  By forcing a low level to CKO, the chip will continue and CKO will stay low.
- As another option, CKO can be a general purpose input, read into bit 2 of A (accumulator) upon execution of an INIL instruction.

Oscillator options

There are four basic clock oscillator configuration available as shown by *Figure 8.*

a. Cyrstal Controlled Oscillator. CKI and CKO are connected to an external crystal. The instruction cycle time equals the crystal frequency optionally divided by 4, 8 or 16.

b. External Oscillator. The external frequency is optionally divided by 4, 8 or 16 to give the instruction cycle time. CKO is the HALT I/O port or a general purpose input.

c. RC Controlled Oscillator. CKI is configured as a single pin RC controlled Schmitt trigger oscillator. The instruction cycle equals the oscillation frequency divided by 4. CKO is the HALT I/O port or a general purpose input.

d. Dual oscillator. By selecting the dual clock option, pin D0 is now a single pin oscillator input. Two configurations are available : RC controlled Schmitt trigger oscillator or external oscillator.

The user may software select between the D0 oscillator (in that case, the instruction cycle time equals the D0 oscillation frequency divided by 4) by setting the D0 latch high or the CKI (CKO) oscillator by resetting D0 Latch low. Note that even in dual clock mode, the counter, if mask-programmed as a time-base counter, is always connected to the CKI oscillator.

For example, the user may connect up to a 1 MHz RC circuit to D0 for faster processing and a 32 kHz watch crystal to CKI and CKO for minimum current drain and time keeping.

Note : CTMA instructions is not allowed when chip is running from D0 clock.

Figures 10A and *10B* show the clock and timer diagrams with and without Dual clock.

ETC 9445/C 9421 24-Pin package option

If the ETC 9444/C 9420 is bonded in a 24-pin pckage, it becomes the ETC 9445/C 9421 illustrated in *Figure 2,* Connection diagrams. Note that the ETC 9445/C 9421 does not contain the four general purpose IN inputs (IN_3-IN_0). Use of this option precludes, ocourse, use of th IN optons, interrupt feature, external event counter feature, and the MICROBUS option which uses IN_1-IN_3. All other options are available for the ETC 9445/C 9421.

Note : If user selects 24-pin package, options 9, 10, 19 and 20 must be selected as a "0" (load to V_{CC} on the IN intputs). See option list.

BLOCK DIAGRAMS

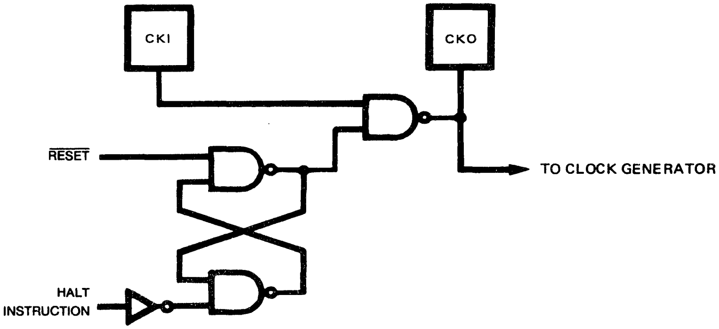


FIGURE 9 A – HALT MODE - TWO-PIN OSCILLATOR

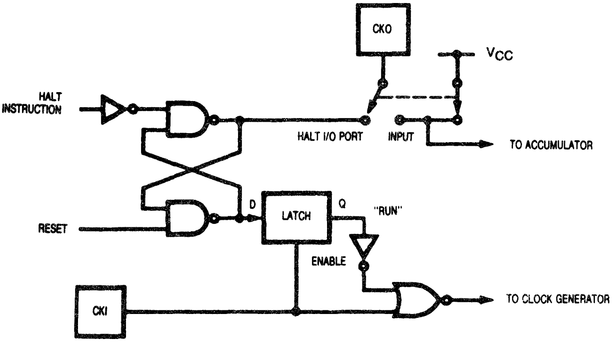


FIGURE 9 B – HALT MODE - ONE PIN OSCILLATOR

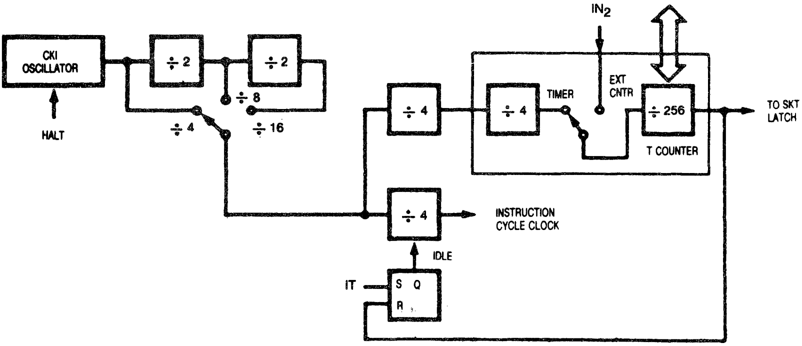


FIGURE 10 A – CLOCK AND TIMER BLOCK DIAGRAM WITHOUT DUAL-CLOCK

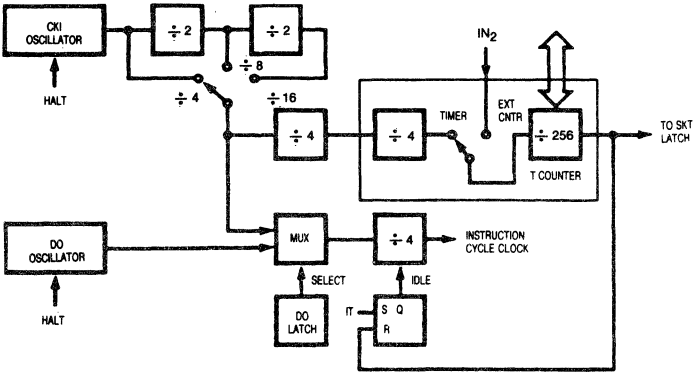


FIGURE 10 B – CLOCK AND TIMER BLOCK DIAGRAM WITH DUAL-CLOCK

INSTRUCTION SET

Table 2 is a symbol table providing internal architectûre, instruction operand and operation symbols used in the instruction set table.

Table 3 provides the mnemonic, operand, machine code data flow, skip conditions and description of each instruction.

TABLE 2 – INSTRUCTION SET TABLE SYMBOLS

Symbol	Definition
Internal Architecture Symbols	
A	4-bit Accumulator
B	7-bit RAM address register (6-bit for ETC 9420)
Br	Upper 3 bits of B (register address) (2-bit for ETC 9420)
Bd	Lower 4 bits of B (digit address)
C	1-bit Carry register
D	4-bit Data output port
EN	4-bit Enable register
G	4-bit General purpose I/O port
IL	Two 1-bit (IN_0 and IN_3) latches
IN	4-bit input port
L	8-bit TRI-STATE I/Oport
M	4-bit contents of RAM addressed by B
PC	11-bit ROM address program counter
Q.	8-bit latch for L port
SA, SB, SC	11-bit 3-level subroutine stack
SIO	4 bit Shift register and counter
SK	Logic-controlled clock output
SKL	1-bit lach for SK output
T	8-bit timer
Instruction operand symbols	
d	4-bit operand field, 0-15 binary (RAM digit select)
r	3(2)-bit operand field, 0-7(3)binary (RAM register select)
a	11-bit operand field, 0-2047 (1023)
y	4-bit Operand field, 0-15 (immediate data)
RAM(x)	RAM addressed by variable x
ROM(x)	ROM addressed by variable x
Operational Symbols	
+	Plus
—	Minus
→	Replaces
↔	is exchanged with
=	is equal to
$\bar{A}$	one's complement of A
⊕	exclusive-or
:	range of values

TABLE 3 – ETC9444/C 9420 INSTRUCTION SET

Mnemonic	Operand	Hx Code	Machine Language Code (Binary)	Date Flow	Skip Conditions	Description
ARITHMETIC INSTRUCTIONS						
ASC		30	0 0 1 1 0 0 0 0	A + C + RAM(B) → A Carry → C	Carry	Add with CARRY Skip on Carry
ADD		31	0 0 1 1 0 0 0 1	A + RAM(B) → A	None	Add RAM to A
ADT		4A	0 1 0 0 1 0 1 0	$A + 10_{10} \to A$	None	Add TEN TO A
AISC	y	5–	0 1 0 1 y	A + y → A	Carry	Add Immediate Skip on Carry (y ≠ 0)
CASC		10	0 0 0 1 0 0 0 0	$\bar{A}$ + RAM(B) + C → A Carry → C	Carry	Complement and Add with Carry. Skip on Carry
CLRA		00	0 0 0 0 0 0 0 0	0 → A	None	Clear A
COMP		40	0 1 0 0 0 0 0 0	$\bar{A}$ → A	None	Ones complement of A to A
NOP		44	0 1 0 0 0 1 0 0	None	None	No Operation
RC		32	0 0 1 1 0 0 1 0	0 → C	none	Reset C
SC		22	0 0 1 0 0 0 1 0	1 → C	None	Set C
XOR		02	0 0 0 0 0 0 1 0	A ⊕ RAM(B) → A	None	Exclusive-OR RAM with A

INSTRUCTION SET (continued)

Mnemonic	Operand	Hx Code	Machine Language Code (Binary)	Data Flow	Skip Conditions	Description
TRANSFER OF CONTROL INSTRUCTIONS						
JID		FF	1 1 1 1 1 1 1 1	ROM($PC_{10:8}$A,M) → $PC_{7:0}$	None	Jump Indirect (Note 3)
JMP	a	6 - - -	0 1 1 0 0 $a_{10:8}$ $a_{7:0}$	a → PC	None	Jump
JP	a	- - - -	1 $a_{6:0}$ (pages 2, 3, only) or 1 1 $a_{5:0}$ (all other pages)	a → $PC_{6:0}$ a → $PC_{5:0}$	None	Jump within Page (Note 4)
JSRP	a	- -	1 0 $a_{5:0}$	PC + 1 → SA → SB → SC 00010 → $PC_{10:6}$ a → $PC_{5:0}$	None	Jump to Subroutine Page (Note 5)
JSR	a	6 -	0 1 1 0 1 $a_{10:8}$ $a_{7:0}$	PC + 1 → SA → SB → SC a → PC	None	Jump to Subroutine
RET		48	0 1 0 0 1 0 0 0	SC → SB → SA → PC	None	Return from Subroutine
RETSK		49	0 1 0 0 1 0 0 1	SC → SB → SA → PC	Always Skip on Return	Return from Subroutine then Skip
HALT		33 38	0 0 1 1 0 0 1 1 0 0 1 1 1 0 0 0		None	HALT processor
IT		33 39	0 0 1 1 0 0 1 1 0 0 1 1 1 0 0 1		None	IDLE till timer Overflows then continues
MEMORY REFERENCE INSTRUCTIONS						
CAMT		33 3F	0 0 1 1 0 0 1 1 0 0 1 1 1 1 1 1	A → $T_{7:4}$ RAM(B) → $T_{3:0}$	None	Copy A, RAM to T
CTMA		33 2F	0 0 1 1 0 0 1 1 0 0 1 0 1 1 1 1	$T_{7:4}$ → RAM(B) $T_{3:0}$ → A	None	Copy T to RAM, A (Note 9)
CAMQ		33. 3C	0 0 1 1 0 0 1 1 0 0 1 1 1 1 0 0	A → $Q_{7:4}$ RAM(B) → $Q_{3:0}$	None	Copy A, RAM to Q
CQMA		33 2C	0 0 1 1 0 0 1 1 0 0 1 0 1 1 0 0	$Q_{7:4}$ → RAM(B) $Q_{3:0}$ → A	None	Copy Q to RAM, A
LD	r	· 5	0 0 r 0 1 0 1 (r – 0 3)	RAM(B) → A Br ⊙ r → Br	None	Load RAM into A Exclusive-OR Br with r
LDD	r d	23	0 0 1 0 0 0 1 1 0 r d	RAM(r d) → A	None	Load A with RAM pointed to directly by r.d
LQID		BF	1 0 1 1 1 1 1 1	ROM($PC_{10\,8}$ A M) → Q	None	Load Q Indirect (Note 3)
RMB	0 1 2 3	4C 45 42 43	0 1 0 0 1 1 0 0 0 1 0 0 0 1 0 1 0 1 0 0 0 0 1 0 0 1 0 0 0 0 1 1	0 · $RAM(B)_0$ 0 · $RAM(B)_1$ 0 · $RAM(B)_2$ 0 · $RAM(B)_3$	None	Reset RAM Bit

INSTRUCTION SET (continued)

Mnemonic	Operand	Hx Code	Machine Language Code (Binary)	Data Flow	Skip Conditions	Description
SMB	0 1 2 3	4D 47 46 4B	0100 1101 0100 0111 0100 0110 0100 1011	1 → RAM(B)$_0$ 1 → RAM(B)$_1$ 1 → RAM(B)$_2$ 1 → RAM(B)$_3$	None	Set RAM Bit
STII	y	7 -	0111 y	y → RAM(B) Bd ⊕ 1 → Bd	None	Store Memory Immediate and increment Bd
X	r	- 6	00 r 0110 (r = 0:3)	RAM(B) ↔ A Br ⊕ r → Br	None	Exchange RAM with A, Exclusive-OR Br with r
XAD	r,d	23 --	0010 0011 1 r d	RAM(r,d) ↔ A	None	Exchange A with RAM pointed to directly by r, d.
XDS	r	- 7	00 r 0111 (r = 0:3)	RAM(B) ↔ A Bd—1 → Bd Br ⊕ → Br	Bd decrements past 0	Exchange RAM with A and Decrement Bd. Exclusive-Or Br with r
XIS	r	- 4	00 r 0100 (r = 0:3)	RAM(B) ↔ A Bd + 1 → Bd Br ⊕ r → Br	Bd increments past 15	Exchange RAM with A and Increment Bd, Exclusive-Or Br with r

REGISTER REFERENCE INSTRUCTIONS

CAB		50	0101 0000	A → Bd	None	Copy A to Bd
CBA		4E	0100 1110	Bd → A	None	Copy Bd to A
LBI	r,d	- - 33 - -	00 r (d-1) (r = 0:3) (d = 0,9:15) or 0011 0011 1 r d (any r, any d)	r,d → B	Skip until not a LBI	Load B immediate with r,d (Note 6)
LEI	y	33 6 -	0011 0011 0110 y	y → EN	None	Load EN Immediate (Note 7)
XABR		12	0001 0010	A ↔ Br	None	Exchange A with Br (Note 8)

TEST INSTRUCTIONS

SKC		20	0010 0000		C = "1"	Skip if C is True
SKE		21	0010 0001		A = RAM(B)	Skip if A Equals RAM
SKGZ		33 21	0011 0011 0010 0001		$G_{3:0}$ = 0	Skip if G is Zero (all 4 bits)
SKGBZ	 0 1 2 3	33 01 11 03 13	0011 0011 0000 0001 0001 0001 0000 0011 0001 0011	1st byte 2nd byte	 G_0 = 0 G_1 = 0 G_2 = 0 G_3 = 0	Skip if G Bit is Zero

INSTRUCTION SET (continued)

Mnemonic	Operand	Hx Code	Machine Language Code (Binary)	Data Flow	Skip Conditions	Description
SKMBZ	0	01	0 0 0 0 0 0 0 1		RAM $(B)_0=0$	Skip if RAM Bit is Zero
	1	11	0 0 0 1 0 0 0 1		RAM $(B)_1=0$	
	2	03	0 0 0 0 0 0 1 1		RAM $(B)_2=0$	
	3	13	0 0 0 1 0 0 1 1		RAM $(B)_3=0$	
SKT		41	0 1 0 0 0 0 0 1		A time-base counter carry has occured since last test	Skip on Timer (Note 3)

INPUT/OUTPUT INSTRUCTIONS

Mnemonic	Operand	Hx Code	Machine Language Code (Binary)	Data Flow	Skip Conditions	Description
ING		33	0 0 1 1 0 0 1 1	G → A	None	Input G Ports to A
		2A	0 0 1 0 1 0 1 0			
ININ		33	0 0 1 1 0 0 1 1	IN → A	None	Input IN inputs to A
		28	0 0 1 0 1 0 0 0			(Note 2)
INIL		33	0 0 1 1 0 0 1 1	IL_3, CKO, "0", IL_0 → A	None	Input IL Latches to A
		29	0 0 1 0 1 0 0 1			(Note 3)
INL		33	0 0 1 1 0 0 1 1	$L_{7:4}$ → RAM(B)	None	Input L Ports to RAM, A
		2E	0 0 1 0 1 1 1 0	$L_{3:0}$ → A		
OBD		33	0 0 1 1 0 0 1 1	Bd → D	None	Output Bd to D Outputs
		3E	0 0 1 1 1 1 1 0			
OGI	y	33	0 0 1 1 0 0 1 1	y → G	None	Output to G Ports
		5 -	0 1 0 1 y			Immediate
OMG		33	0 0 1 1 0 0 1 1	RAM(B) → G	None	Output RAM to G Ports
		3A	0 0 1 1 1 0 1 0			
XAS		4F	0 1 0 0 1 1 1 1	A ↔ SIO, C → SKL	None	Exchange A with SIO (Note 3)

Note 1 : All subscripts for alphabetical symbols indicate bit numbers unless explicitly defined (e.g., Br and Bd are explicitly defined). Bits are numbered 0 to N where 0 signifies the least significant bit (low-order, right-most bit). For example, A_3 indicates the most significant (left-most) bit of the 4-bit A register.

Note 2 : The ININ instruction is not available on the 24-pin packages since these devices do not contain the IN inputs.

Note 3 : For additional on the operation of the XAS, JID, LQID, INIL, and SKT instructions, see below.

Note 4 : The JP instruction allows a jump, while in subroutine page 2 or 3, to any ROM location within the two-page boundary of pages 2 or 3. The JP instruction, otherwise, permits a jump to a ROM location within the current 64-word page. JP may not jump to the last word of a page

Note 5 : A JSRP transfers program control to subroutine page 2 (0010 is loaded into the upper 4 bits of P). A JSRP may not be used when in pages 2 or 3. JSRP may not jump to the last word in page 2.

Note 6 : LBI is a single-byte instruction if d=0, 9, 10, 11, 12, 13, 14, or 15. The machine code for the lower 4 bits equals the binary value of the "d" data minus 1, e.g., to load the lower four bits of B (Bd) with the value 9(1001_2), the lower 4 bits of the LBI instruction equal 8 (1000_2). To load 0, the lower 4 bits of the LBI instruction should equal 15 (1111_2).

Note 7 : Machine code for operand field y for LEI instruction should equal the binary value to be latched into EN, where a "1" or "0" in each bit of EN corresponds with the selection or deselection of a particular function associated with each bit. (See Functional Description, EN Register)

Note 8 : For 2K ROM devices. A ↔ Br (0 → A3). For 1K ROM devices. A ↔ Br (0,0 → A3, A2).

Note 9 : Do not use CTMA instruction when dual - option is selected and part is running from D0 Clocks.

DESCRIPTION OF SELECTED INSTRUCTIONS

Xas Instruction

XAS (Exchange A with SIO) copies C to the SKL latch and exchanges the accumulator with the 4-bit contents of the SIO register. The contents of SIO will contain serial-in/serial-out shift register or binary counter data, depending on the value of the EN register. If SIO is selected as a shift register, an XAS instruction can be performed once every 4 instruction cycles to effect a continuous data stream.

LQID Instruction

LQID (Load Q Indirect) loads the 8-bit Q register with the contents of ROM pointed to by the 11-bit word PC10:PC8,A,M. LQID can be used for table lookup or code conversion such as BCD to seven-segment. The LQID instruction "pushes" the stack (PC + 1 → SA → SB → SC) and replaces the least significant 8 bits of the PC as follows : A → PC(7:4), RAM(B) → PC(3:0), leaving PC(10), PC(9) and PC(8) unchanged. The ROM data pointed to by the new address is fetched and loaded into the Q latches. Next, the stack is "popped" (SC → SB → SA → PC), restoring the saved value of PC to continue sequential program excecution. Sincê LQID pushes SB → SC, the previous contents of SC are lost.

Note : LQID uses 2 instruction cycles if executed, one if skipped.

JID Instruction

JID (Jump Indirect) is an indirect addressing instruction, transferring program control to a new ROM location pointed to indirectly by A and M. It loads the lower 8 bits of the ROM address register PC with the contents of ROM addressed by the 11-bit word, PC10:8,A,M. PC10, PC9 and PC8 are not affected by JID.

Note : JID uses 2 instruction cycles if executed, one if skipped.

SKT Instruction

The SKT (Skip On Timer) instruction tests the state of the T counter overflow latch (see internal logic, above), executing the next program instruction if the latch is not set. If the latch has been set since the previous test, the next program instruction is skipped and the latch is reset. The features associated with this instruction allow the processor to generate its own time-base for real-time processing, rather than relying on an external input signal.

Note : If the most significant bit of the T counter is a 1 when a CAMT instruction loads the counter, the overflow flag will be set. The following sample of codes should be used when loading the counter :

```
CAMT ; load T counter
SKT   : skip if overflow flag is set and reset it
NOP
```

IT Instruction

The IT (idle till timer) instruction halts the processor and puts it in an idle state until the time-base counter overflows. Upon overflow, the processor will restart with a delay shorter than one cycle time. This idle state reduces current drain since all logic (except the oscillator and time base counter) is stopped. IT instruction is not allowed if the T counter is mask-programmed as an external event counter (option ≠ 31 = 1).

INIL Instruction

INIL (Input IL Latches to A) inputs 2 latches, IL3 and IL0, CKO and 0 into A. The IL3 and IL0 latches are set if a low-going pulse ("1" to "0") has occured on the IN3 and IN0 inputs since the last INIL instruction, provided the input pulse stays low for at least two instruction cycles. Execution of an INIL inputs IL3 and IL0 into A3 and A0 respectively, and resets these latches to allow them to respond to subsequent low-going pulses on the IN3 and IN0 lines. If CKO is mask programmed as a general purpose input, an INIL will input the state of CKO INTO A2. If CKO has not been so programmed, a "1" will be placed in A2. A 0 is input into A1. IL latches are cleared on reset. IL latches are not available on the ETC 9445/C 9421.

Instruction set notes

a. The first word of a program (ROM address 0) must be a CLRA (Clear A) instruction.

b. Although skipped instructions are not executed, they are still fetched from the program memory. Thus program paths take the same number of cycles whether instructions are skipped or executed except for JID, and LQID.

c. The ROM is organized into pages of 64 words each The Program Counter is a 11-bit binary counter, and will count through page boundaries. If a JP, JSRP, JID, or LQID is the last word of a page, it operates as if it were in the next page. For example : a JP Located in the last word of a page will jump to a location in the next page. Also, a JID or LQID located in the last word of every fourth page (i.e. hex address 0FF, 1FF, 2FF, 3FF, 4FF, etc.) will access data in the next group of four pages.

Note : The ETC 9420/C 9421/C 9422 needs only 10 bits to address its ROM. Therefore, the eleventh bit (10) is ignored.

POWER DISSIPATION

The lowest power drain is when the clock is stopped. As the frequency increases so does current. Current is also lower at lower operating voltages. Therefore, the user should run at the lowest speed and voltage that his application will allow. The user should take care that all pins swing to full supply levels to insure that outputs are not loaded down and that inputs are not at some intermediate level which may draw current. Any input with a slow rise or fall time will draw additional current. A crystal or resonator generated clock input will draw additional current. For example, a 500 kHz crystal input will typical draw 100 μA more than a square-wave input. An R/C oscillator will draw even more current since the input is a slow rising signal.

If using an external squarewave oscillator, the following equation can be used to calculate the ETC 9444/C 9445 operating current drain.

$I_{CO} = I_Q + V \times 40 \times Fi + V \times 1400 \times Fi/Dv$

where I_{CO} = chip operating current drain in microamps
I_Q = quiescent leakage current (from curve)
Fi = CKI frequency in MegaHertz
V = chip V_{CC} in volts
Dv = divide by option selected.

For example at 5 volts V_{CC} and 400 kHz (divide by 4)
$I_{CO} = 20 + 5 \times 40 \times 0.4 + 5 \times 1400 \times 0.4/4$
$I_{CO} = 20 + 80 + 700 = 800\,\mu A$

At 2.4 volts V_{CC} and 30 kHz (divide by 4)
$I_{CO} = 6 + 2.4 \times 40 \times 0.03 + 2.4 \times 1400 \times 0.03/4$
$I_{CO} = 6 + 2.88 + 25.2 = 34.08\,\mu A$

POWER DISSIPATION (continued)

If an IT instruction is executed, the chip goes into the IDLE mode until the timer overflows. In IDLE mode, the current drain can be calculated from the following equation :

$$Ici = I_Q + V \times 40 \times Fi$$

For example, at 5 volts V_{CC} and 400 kHz

$$Ici = 20 + 5 \times 40 \times 0.4 = 100\ \mu A$$

The total average current will then be the weighted average of the operation current and the idle current :

$$Ita = I_{CO} \times \frac{To}{To+Ti} + Ici \times \frac{Ti}{To+Ti}$$

where : Ita = total average current
I_{CO} = operating current
Ici = idle current
To = operating time
Ti = idle time

I/O options

ETC 9444/C 9445 outputs have the following optional configurations, illustrated in *Figure 11* :

a. Standard—A CMOS push-pull buffer with an N-channel device to ground in conjunction with a P-channel device to V_{CC}, compatible with CMOS and LSTTL.

b. Low Current—This is the same configuration as a. above except that the sourcing current is much less.

c. Open Drain—An N-channel device to ground only, allowing external pull-up as required by the user's application.

d. Standard TRI-STATE L Ouput—A CMOS output buffer similar to a. witch may be disabled by program control.

e. Low-Current TRI-STATE L Output—This is the same as d. above except that the sourcing current is much less.

f. Open-Drain TRI-STATE L Output—This has the N-channel device to ground only.

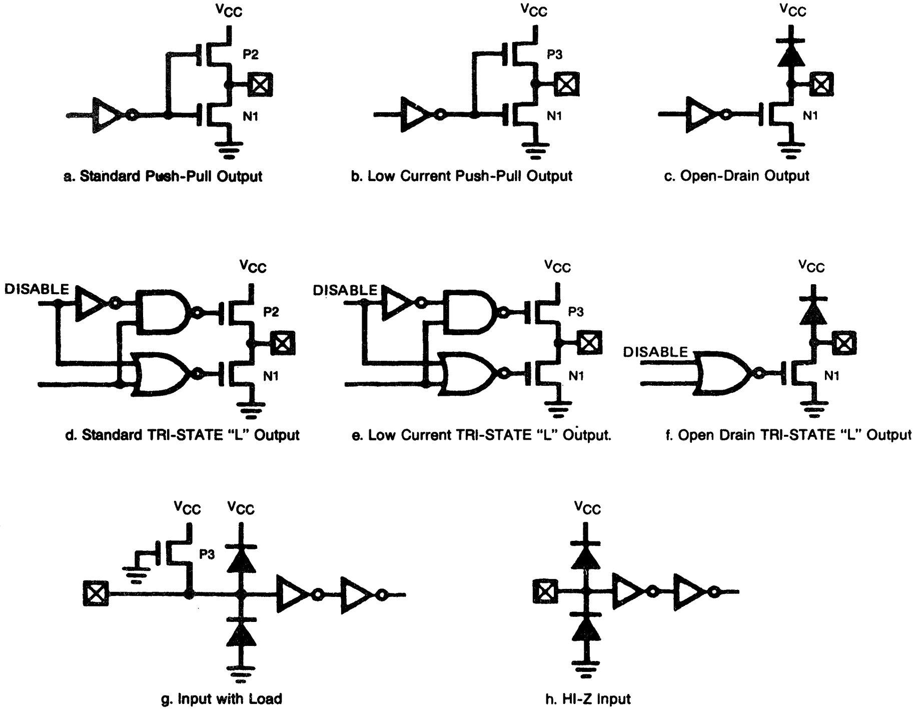
All inputs have the following options :

g. Input with on chip load device to V_{CC}.

h. Hi-Z input which must be driven by the users logic.

When using either the G or L I/O ports as inputs, a pull-up device is necessary. This can be an external device or the following alternative is available : Select the low-current output option. Now, by setting the output registers to a logic "1" level, the P-channel devices will act as the pull-up load. Note that when using the L ports in this fashion the Q registers must be set to a logic "1" level and the L drivers MUST BE ENABLED by an LEI instruction (see description above).

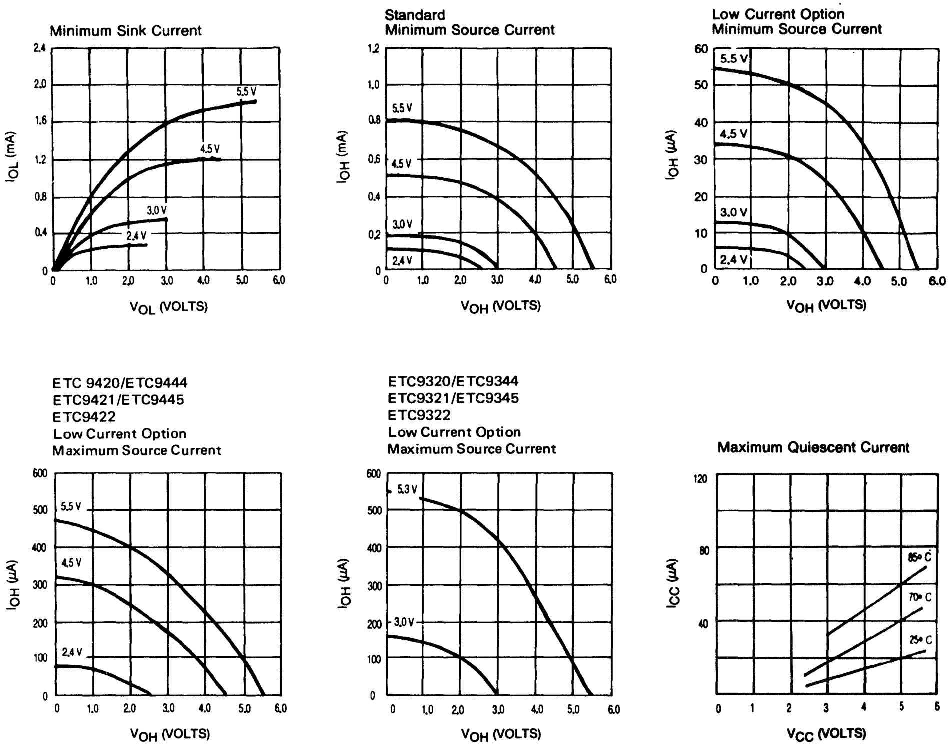
All output drivers use one or more of three common devices numbered 1 to 3. Minimum and maximum current (I_{OUT} and V_{OUT}) curves are given in *Figure 12* for each of these devices to allow the designer to effectively use these I/O configurations.



a. Standard Push-Pull Output
b. Low Current Push-Pull Output
c. Open-Drain Output
d. Standard TRI-STATE "L" Output
e. Low Current TRI-STATE "L" Output.
f. Open Drain TRI-STATE "L" Output
g. Input with Load
h. HI-Z Input

FIGURE 11 – INPUT/OUTPUT CONFIGURATIONS

POWER DISSIPATION (continued)



Minimum Sink Current

Standard Minimum Source Current

Low Current Option Minimum Source Current

ETC 9420/ETC9444
ETC9421/ETC9445
ETC9422
Low Current Option
Maximum Source Current

ETC9320/ETC9344
ETC9321/ETC9345
ETC9322
Low Current Option
Maximum Source Current

Maximum Quiescent Current

FIGURE 12 – INPUT/OUTPUT CHARACTERISTICS

OPTION LIST

The ETC 9444/C 9420/C 9445/C 9421/C 9422 mask-programmable options are assigned numbers which correspond with the ETC 9344/C 9320/C 9345/C 9321/C 9322 pins.

The following is a list of options. The options are programmed at the same time as the ROM pattern to provide the user with the hardware flexibility to interface to various I/O components using little or no external circuitry.

Option 1 = 0 : Ground Pin — no option available.

Option 2 : CKO Pin
= 0 : clock generator output to crystal/resonator
= 1 : HALT I/O port
= 2 : general purpose input with load device to V_{CC}
= 3 : general purpose input, high-Z

Option 3 : CKI input
= 0 : Crystal controlled oscillator input divide by 4
= 1 : Crystal controlled oscillator input divide by 8
= 2 : Crystal controlled oscillator input divide by 16
= 4 : Single-pin RC controlled oscillator (divide by 4)
= 5 : External oscillator input divide by 4
= 6 : External oscillator input divide by 8
= 7 : External oscillator input divide by 16

Option 4 : $\overline{\text{RESET}}$ input
= 0 : load device to V_{CC}
= 1 : Hi-Z input

Option 5 : L7 Driver
= 0 : Standard TRI-STATE push-pull output
= 1 : Low-current TRI-STATE push-pull output
= 2 : Open-drain TRI-STATE output

Option 6 : L6 Driver—(same as option 5)

Option 7 : L5 Driver—(same as option 5)

Option 8 : L4 Driver—(same as option 5)

Option 9 : IN1 input
= 0 : load device to V_{CC}
= 1 : Hi-Z input

Option 10 : IN2 input—(same as option 9)

Option 11 = 0 : V_{CC} Pin—no option available

Option 12 : L3 Driver—(same as option 5)

Option 13 : L2 Driver—(same as option 5)

Option 14 : L1 Driver—(same as option 5)

Option 15 : L0 Driver—(same as option 5)

Option 16 : Si Input — (same as option 9)

Option 17 : SO Driver
= 0 : Standard push-pull output
= 1 : Low-current push-pull output
= 2 : Open-drain output

Option 18 : SK Driver — (same as option 17)

Option 19 : IN0 Input — (same as option 9)

Option 20 : IN3 Input — (same as option 9)

Option 21 : G0 I/O Port — (same as option 17)

Option 22 : G1 I/O Port — (same as option 17)

Option 23 : G2 I/O Port — (same as option 17)

Option 24 : G3 I/O Port — (same as option 17)

Option 25 : D3 Output — (same as option 17)

Option 26 : D2 Output — (same as option 17)

Option 27 : D1 Output — (same as option 17)

Option 28 : D0 Output — (same as option 17)

Option 29 : Internal Initialization Logic
= 0 : Normal operation
= 1 : No internal initialization logic

Option 30 : Dual Clock
= 0 : Normal operation
= 1 : Dual Clock. D0 RC oscillator } (opt. #28 must = 2)
= 2 : Dual Clock. D0 ext. clock input }

Option 31 : Timer
= 0 : Time-base counter
= 1 : External event counter

Option 32 : MICROBUS
= 0 : Normal
= 1 : MICROBUS (opt. #31 must = 0)

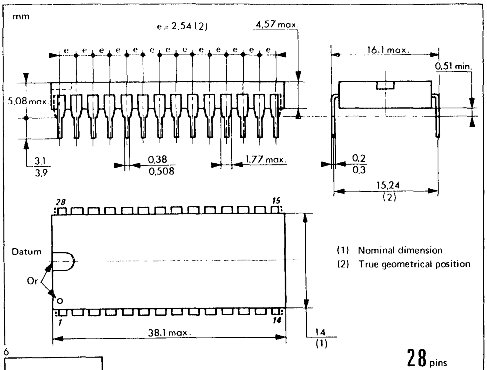
Option 33 : Chip bonding
(1 K and 2 K Microcontroller)
= 0 : 28 - pin package
= 1 : 24 - pin package
= 2 : Same die purchased in both 24 and 28 pin version
(1 K Microcontroller only)
= 3 : 20 - pin package
= 4 : 28- and 20- pin package
= 5 : 24- and 20- pin package
= 6 : 28-, 24- and 20- pin package

Note : - If opt#33 = 2 then opt#9, 10, 19, 20 and 32 must = 0
- If opt#33 = 3, 4, 5 or 6 then opt#9, 10, 19, 20, 21, 22, 30 and 32 must = 0

CROSS REFERENCE	
THOMSON SC	NS
ETC 9420 ETC 9421 ETC 9422 ETC 9444 ETC 9445	COP 424 C COP 425 C COP 426 C COP 444 C COP 445 C

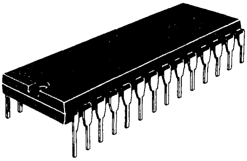
COPS, MICROWIRE and TRI-STATE are registered trademarks of National Semiconductor Corp.

PHYSICAL DIMENSIONS

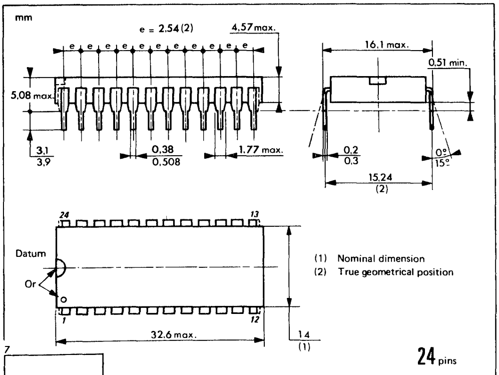


DIN				
A51E CEI	D.A.T.A.	JEDEC	F 144 SITELESC	CB-132

CB-132

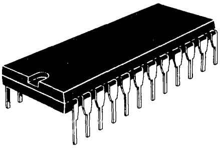


N SUFFIX
PLASTIC PACKAGE



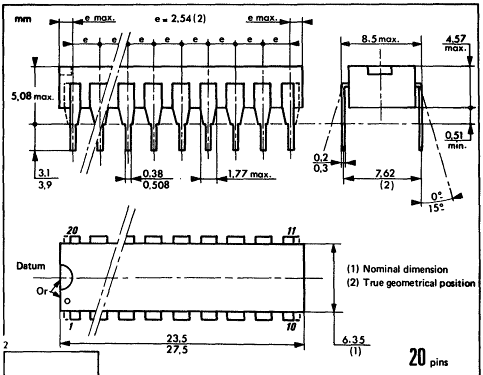
DIN				
A51D CEI	D.A.T.A.	JEDEC	F 118 SITELESC	CB-68

CB-68



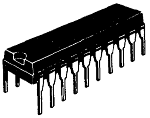
N SUFFIX
PLASTIC PACKAGE

PHYSICAL DIMENSIONS



DIN				
CEI	D.A.T.A.	JEDEC	SITELESC	CB-194

CB-194



N SUFFIX
PLASTIC PACKAGE

These specifications are subject to change without notice.
Please inquire with our sales offices about the availability of the different products.

THOMSON
COMPONENTS
SEMICONDUCTEURS

ETL9420/21/22 • ETL9320/21/22

SINGLE CHIP MICROCONTROLLERS

The ETL9420, ETL9421, ETL9422, ELT9320, ETL9321, and ETL9322 Single-Chip N-Channel Microcontrollers are fully compatible with the COPS® family, fabricated using N-channel, silicon gate XMOS technology. These controller oriented processors are complete microcomputers containing all system timing, internal logic, ROM, RAM, and I/O necessary to implement dedicated control functions in a variety of applications. Features include single supply operation, a variety of output configuration options, with an instruction set, internal architecture, and I/O scheme designed to facilitate keyboard input, display output, and BCD data manipulation. The ETL9421 and ETL9422 are identical to the ETL9420, but with 19 and 15 I/O lines, respectively, instead of 23. They are an appropriate choice for use in numerous human interface control environments. Standard test procedures and reliable high-density fabrication techniques provide the medium to large volume customers with a customized controller oriented processor at a low end-product cost.

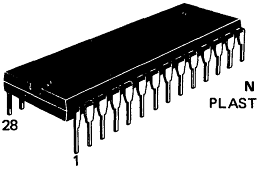
The ETL9320/L9321/L9322 are exact functional equivalents, but extended temperature range versions, of the ETL9420, ETL9421 and ETL9422 respectively.

- Low cost
- Powerful instruction set
- 1 K x 8 ROM, 64 x 4 RAM
- 23 I/O lines (ETL9420)
- True vectored interrupt, plus restart
- Three-level subroutine stack
- 16μs instruction time
- Single supply operation (4.5-6.3V)
- Low current drain (9mA max.)
- Internal time-base counter for real-time processing
- Internal binary counter register with MICROWIRE® compatible serial I/O
- General purpose and TRI-STATE® outputs
- LSTTL/CMOS compatible in and out
- Direct drive of LED digit and segment lines
- Software/hardware compatible with other members of ET9400 family
- Extended temperature range device ETL9320/L9321/L9322 (−40°C to + 85°)
- Wider supply range (4.5V – 9.5V) optionally available

NMOS

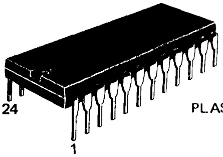
CASES

ETL9420 • ETL9320



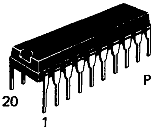
N SUFFIX
PLASTIC PACKAGE

ETL9421 • ETL9321



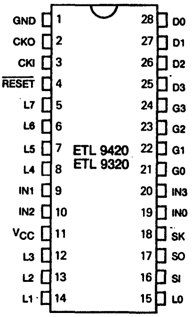
N SUFFIX
PLASTIC PACKAGE

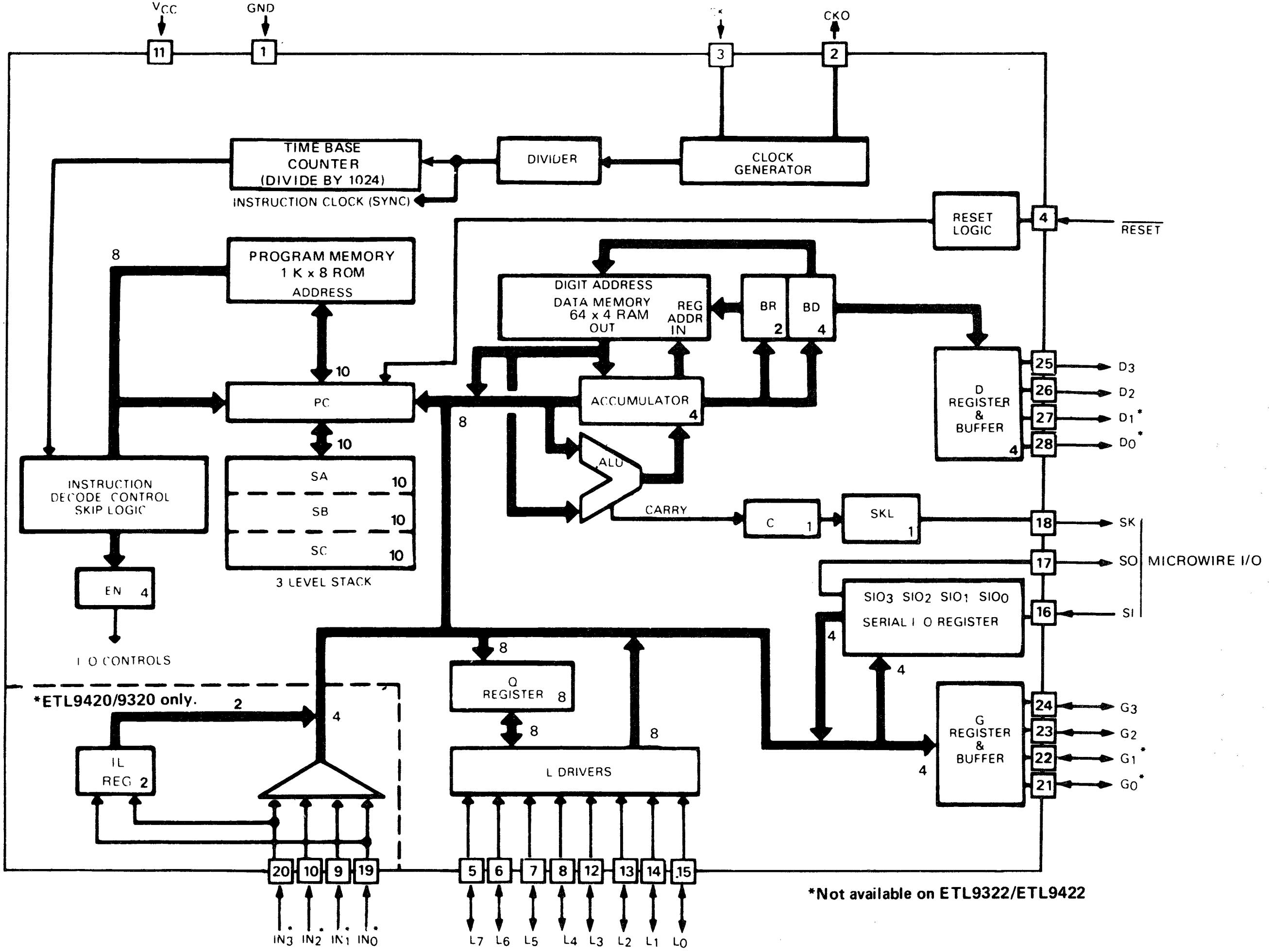
ETL9422 • ETL9322



N SUFFIX
PLASTIC PACKAGE

PIN ASSIGNMENT





*ETL9420/9320 only.

*Not available on ETL9322/ETL9422

FIGURE 1 – BLOCK DIAGRAM (28-pin version)

ETL9420/L9421/L9422

ABSOLUTE MAXIMUM RATINGS

Voltage at Any Pin Relative to GND	−0.5V to +10V
Ambient Operating Temperature	0°C to +70°C
Ambient Storage Temperature	−65°C to +150°C
Lead Temperature (Solderir.g, 10 seconds)	300°C
Power Dissipation	
ETL9420/L9421	0.75 Watt at 25°C
	0.4 Watt at 70°C
ETL9422	0.65 Watt at 25°C
	0.3 Watt at 70°C
Total Source Current	120 mA
Total Sink Current	120 mA

Absolute maximum ratings indicate limits beyond which damage to the device may occur. DC and AC electrical specifications are not ensured when operating the device at absolute maximum ratings.

DC ELECTRICAL CHARACTERISTICS $0° \leqslant T_A \leqslant +70°C$, $4.5V \leqslant V_{CC} \leqslant 9.5V$ (Unless otherwise specified).

Parameter	Conditions	Min.	Max.	Units
Standard Operating Voltage (V_{CC})	Note 1	4.5	6.3	V
Optional Operating Voltage (V_{CC})		4.5	9.5	V
Power Supply Ripple	peak to peak	—	0.5	V
Operating Supply Current	all inputs and outputs open	—	9	mA
Input Voltage Levels				
CKI Input Levels				
Crystal Input (÷32, ÷16, ÷8)				
Logic High (V_{IH})		2.0	—	V
Logic Low (V_{IL})		−0.3	0.4	V
Schmitt Trigger Input (÷4)				
Logic High (V_{IH})		0.7 V_{CC}	—	V
Logic Low (V_{IL})		−0.3	0.6	V
$\overline{\text{RESET}}$ Input Levels	Schmitt Trigger Input			
Logic High		0.7 V_{CC}	—	V
Logic Low		−0.3	0.6	V
SO Input Level (Test mode)		2.0	2.5	V
Al; Other Inputs				
Logic High	V_{CC} = Max.	3.0	—	V
Logic High	with TTL trip level options	2.0	—	V
Logic Low	selected, $V_{CC} = 5V \pm 5\%$	−0.3	0.8	V
Logic High	with high trip level options	3.6	—	V
Logic Low	selected	−0.3	1.2	V
Input Capacitance		—	7	pF
Hi-Z Input Leakage		−1	+1	µA
Output Voltage Levels				
LSTTL Operation	$V_{CC} = 5V \pm 5\%$			
Logic High (V_{OH})	$I_{OH} = -25 \mu A$	2.7	—	V
Logic Low (V_{OL})	$I_{OL} = 0.36 mA$	—	0.4	V
CMOS Operation				
Logic High	$I_{OH} = -10 \mu A$	$V_{CC} - 1$	—	V
Logic Low	$I_{OL} = +10 \mu A$	—	0.2	V

Note 1 : V_{CC} voltage change must be less than 0.5V in a 1 ms period to maintain proper operation.

ETL9420/L9421/L9422

DC ELECTRICAL CHARACTERISTICS (continued) 0°C ⩽ T_A ⩽ +70°C, 4.5V ⩽ V_{CC} ⩽ 9.5V (Unless otherwise specified)

Parameter	Conditions	Min.	Max.	Units
Output Current Levels				
Output Sink Current				
SO and SK Outputs (I_{OL})	V_{CC} = 9.5V, V_{OL} = 0.4V	1.8	—	mA
	V_{CC} = 6.3V, V_{OL} = 0.4V	1.2	—	mA
	V_{CC} = 4.5V, V_{OL} = 0.4V	0.9	—	mA
L_0-L_7 Outputs and Standard	V_{CC} = 9.5V, V_{OL} = 0.4V	0.8	—	mA
G_0-G_3, D_0-D_3 Outputs (I_{OL})	V_{CC} = 6.3V, V_{OL} = 0.4V	0.5	—	mA
	V_{CC} = 4.5V, V_{OL} = 0.4V	0.4	—	mA
G_0-G_3 and D_0-D_3 Outputs with	V_{CC} = 9.5V, V_{OL} = 1.0V	15	—	mA
High Current Options (I_{OL})	V_{CC} = 6.3V, V_{OL} = 1.0V	11	—	mA
	V_{CC} = 4.5V, V_{OL} = 1.0V	7.5	—	mA
G_0-G_3 and D_0-D_3 Outputs with	V_{CC} = 9.5V, V_{OL} = 1.0V	30	—	mA
Very High Current Options (I_{OL})	V_{CC} = 6.3V, V_{OL} = 1.0V	22	—	mA
	V_{CC} = 4.5V, V_{OL} = 1.0V	15	—	mA
CKI (Single-pin RC oscillator)	V_{CC} = 4.5V, V_{IH} = 3.5V	2	—	mA
CKO	V_{CC} = 4.5V, V_{OL} = 0.4V	0.2	—	mA
Output Source Current				
Standard Configuration,	V_{CC} = 9.5V, V_{OH} = 2.0V	−140	−800	μA
All Outputs (I_{OH})	V_{CC} = 6.3V, V_{OH} = 2.0V	−75	−480	μA
	V_{CC} = 4.5V, V_{OH} = 2.0V	−30	−250	μA
Push-Pull Configuration	V_{CC} = 9.5V, V_{OH} = 4.75V	−1.4	—	mA
SO and SK Outputs (I_{OH})	V_{CC} = 6.3V, V_{OH} = 2.4V	−1.4	—	mA
	V_{CC} = 4.5V, V_{OH} = 1.0V	−1.2	—	mA
LED Configuration, L_0-L_7				
Outputs, Low Current	V_{CC} = 9.5V, V_{OH} = 2.0V	−1.5	−18	mA
Driver Option (I_{OH})	V_{CC} = 6.0V, V_{OH} = 2.0V	−1.5	−13	mA
LED Configuration, L_0-L_7				
Outputs, High Current	V_{CC} = 9.5V, V_{OH} = 2.0V	−3.0	−35	mA
Driver Option (I_{OH})	V_{CC} = 6.0V, V_{OH} = 2.0V	−3.0	−25	mA
TRI-STATE® Configuration,	V_{CC} = 9.5V, V_{OH} = 5.5V	-0.75	—	mA
L_0-L_7 Outputs, Low	V_{CC} = 6.3V, V_{OH} = 3.2V	-0.8	—	mA
Current Driver Option (I_{OH})	V_{CC} = 4.5V, V_{OH} = 1.5V	-0.9	—	mA
TRI-STATE® Configuration,	V_{CC} = 9.5V, V_{OH} = 5.5V	-1.5	—	mA
L_0-L_7 Outputs, High	V_{CC} = 6.3V, V_{OH} = 3.2V	-1.6	—	mA
Current Driver Option (I_{OH})	V_{CC} = 4.5V, V_{OH} = 1.5V	-1.8	—	mA
Input Load Source Current	V_{CC} = 5.0V, V_{IL} = 0V	-10	-140	μA
CKO Output				
RAM Power Supply Option				
Power Requirement	V_R = 3.3V	—	3.0	mA
TRI-STATE® Output Leakage Current		-2.5	+2.5	μA
Total Sink Current Allowed				
All Outputs Combined		—	120	mA
D, G Ports		—	120	mA
L_7-L_4		—	4	mA
L_3-L_0		—	4	mA
All Other Pins		—	1.5	mA
Total Source Current Allowed				
All I/O Combined		—	120	mA
L_7-L_4		—	60	mA
L_3-L_0		—	60	mA
Each L Pin		—	30	mA
All Other Pins		—	1.5	mA

ETL9320/L9321/L9322

ABSOLUTE MAXIMUM RATINGS

Voltage at Any Pin Relative to GND	−0.5V to +10V
Ambient Operating Temperature	−40°C to +85°C
Ambient Storage Temperature	−65°C to +150°C
Lead Temperature (Soldering, 10 seconds)	300°C
Power Dissipation	
ETL9320 L9321	0.75 Watt at 25°C
	0.4 Watt at 85°C
ETL9322	0.65 Watt at 25°C
	0.20 Watt at 85°C
Total Source Current	120 mA
Total Sink Current	120 mA

Absolute maximum ratings indicate limits beyond which damage to the device may occur. DC and AC electrical specifications are not ensured when operating the device at absolute maximum ratings.

DC ELECTRICAL CHARACTERISTICS – 40°C ⩽ T_A ⩽ +85°C, 4.5V ⩽ V_{CC} ⩽ 7.5V (Unless otherwise specified)

Parameter	Conditions	Min.	Max.	Units
Standard Operating Voltage (V_{CC})	Note 1	4.5	5.5	V
Optional Operating Voltage (V_{CC})		4.5	7.5	V
Power Supply Ripple	peak to peak	–	0.5	V
Operating Supply Current	all inputs and outputs open	–	11	mA
Input Voltage Levels				
CKI Input Levels				
Crystal Input				
Logic High (V_{IH})		2.2	–	V
Logic Low (V_{IL})		−0.3	0.3	V
Schmitt Trigger Input				
Logic High (V_{IH})		0.7 V_{CC}	–	V
Logic Low (V_{IL})		−0.3	0.4	V
$\overline{\text{RESET}}$ Input Levels	Schmitt Trigger Input			
Logic High		0.7 V_{CC}	–	V
Logic Low		−0.3	0.4	V
SO Input Level (Test mode)		2.2	2.5	V
All Other Inputs				
Logic High	V_{CC} = Max.	3.0	–	V
Logic High	with TTL trip level options	2.2	–	V
Logic Low	selected, V_{CC} = 5V ± 5%	−0.3	0.6	V
Logic High	with high trip level options	3.6	–	V
Logic Low	selected	−0.3	1.2	V
Input Capacitance		–	7	pF
Hi-Z Input Leakage		−2	+2	μA
Output Voltage Levels				
LSTTL Operation	V_{CC} = 5V ± 5%			
Logic High (V_{OH})	I_{OH} = −20μA	2.7	–	V
Logic Low (V_{OL})	I_{OL} = 0.36mA	–	0.4	V
CMOS Operation				
Logic High	I_{OH} = −10μA	V_{CC} − 1	–	V
Logic Low	I_{OL} = +10μA	–	0.2	V

Note 1 : V_{CC} voltage change must be less than 0.5V in a 1 ms period to maintain proper operation.

ETL9320/L9321/L9322

DC ELECTRICAL CHARACTERISTICS (continued − 40°C $\leqslant T_A \leqslant$ +85°C, 4.5V $\leqslant V_{CC} \leqslant$ 7.5V (Unless otherwise specified).

Parameter	Conditions	Min.	Max.	Units
Output Current Levels				
Output Sink Current				
SO and SK Outputs (I_{OL})	$V_{CC} = 7.5V, V_{OL} = 0.4V$	1.4	—	mA
	$V_{CC} = 5.5V, V_{OL} = 0.4V$	1.0	—	mA
	$V_{CC} = 4.5V, V_{OL} = 0.4V$	0.8	—	mA
L_0-L_7 Outputs and Standard	$V_{CC} = 7.5V, V_{OL} = 0.4V$	0.6	—	mA
G_0-G_3, D_0-D_3 Outputs (I_{OL})	$V_{CC} = 5.5V, V_{OL} = 0.4V$	0.5	—	mA
	$V_{CC} = 4.5V, V_{OL} = 0.4V$	0.4	—	mA
G_0-G_3 and D_0-D_3 Outputs with	$V_{CC} = 7.5V, V_{OL} = 1.0V$	12	—	mA
High Current Options (I_{OL})	$V_{CC} = 5.5V, V_{OL} = 1.0V$	9	—	mA
	$V_{CC} = 4.5V, V_{OL} = 1.0V$	7	—	mA
G_0-G_3 and D_0-D_3 Outputs with	$V_{CC} = 7.5V, V_{OL} = 1.0V$	24	—	mA
Very High Current Options (I_{OL})	$V_{CC} = 5.5V, V_{OL} = 1.0V$	18	—	mA
	$V_{CC} = 4.5V, V_{OL} = 1.0V$	14	—	mA
CKI (Single-pin RC oscillator)	$V_{CC} = 4.5V, V_{IH} = 3.5V$	2	—	mA
CKO	$V_{CC} = 4.5V, V_{OL} = 0.4V$	0.2	—	mA
Output Source Current				
Standard Configuration,	$V_{CC} = 7.5V, V_{OH} = 2.0V$	−100	−900	µA
All Outputs (I_{OH})	$V_{CC} = 5.5V, V_{OH} = 2.0V$	−55	−600	µA
	$V_{CC} = 4.5V, V_{OH} = 2.0V$	−28	−350	µA
Push-Pull Configuration	$V_{CC} = 7.5V, V_{OH} = 3.75V$	−0.85	—	mA
SO and SK Outputs (I_{OH})	$V_{CC} = 5.5V, V_{OH} = 2.0V$	−1.1	—	mA
	$V_{CC} = 4.5V, V_{OH} = 1.0V$	−1.2	—	mA
LED Configuration, L_0-L_7	$V_{CC} = 7.5V, V_{OH} = 2.0V$	−1.4	−27	mA
Outputs, Low Current	$V_{CC} = 6.0V, V_{OH} = 2.0V$	−1.4	−17	mA
Driver Option (I_{OH})	$V_{CC} = 5.5V, V_{OH} = 2.0V$	−0.7	−15	mA
LED Configuration, L_0-L_7	$V_{CC} = 7.5V, V_{OH} = 2.0V$	−2.7	−54	mA
Outputs, High Current	$V_{CC} = 6.0V, V_{OH} = 2.0V$	−2.7	−34	mA
Driver Option (I_{OH})	$V_{CC} = 5.5V, V_{OH} = 2.0V$	−1.4	−30	mA
TRI-STATE® Configuration,	$V_{CC} = 7.5V, V_{OH} = 4.0V$	−0.7	—	mA
L_0-L_7 Outputs, Low	$V_{CC} = 5.5V, V_{OH} = 2.7V$	−0.6	—	mA
Current Driver Option (I_{OH})	$V_{CC} = 4.5V, V_{OH} = 1.5V$	−0.9	—	mA
TRI-STATE® Configuration,	$V_{CC} = 7.5V, V_{OH} = 4.0V$	−1.4	—	mA
L_0-L_7 Outputs, High	$V_{CC} = 5.5V, V_{OH} = 2.7V$	−1.2	—	mA
Current Driver Option (I_{OH})	$V_{CC} = 4.5V, V_{OH} = 1.5V$	−1.8	—	mA
Input Load Source Current	$V_{CC} = 5.0V, V_{IL} = 0V$	−10	−200	µA
CKO Output				
RAM Power Supply Option				
Power Requirement	$V_R = 3.3V$	—	4.0	mA
TRI-STATE® Output Leakage Current		−5	+5	µA
Total Sink Current Allowed				
All Outputs Combined		—	120	mA
D, G Ports		—	120	mA
L_7-L_4		—	4	mA
L_3-L_0		—	4	mA
All Other Pins		—	1.5	mA
Total Source Current Allowed				
All I/O Combined		—	120	mA
L_7-L_4		—	60	mA
L_3-L_0		—	60	mA
Each L Pin		—	30	mA
All Other Pins		—	1.5	mA

AC ELECTRICAL CHARACTERISTICS

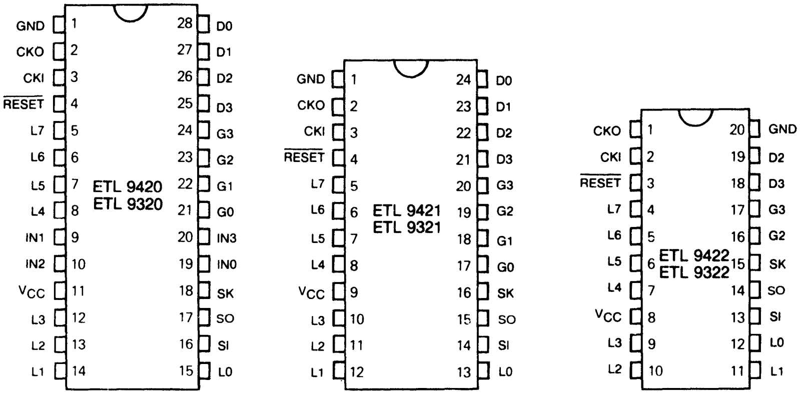
ETL9420/L9421/L9422 : 0°C $\leqslant T_A \leqslant$ +70°C, 4.5V $\leqslant V_{CC} \leqslant$ 9.5V (Unless otherwise specified)
ETL9320/L9321/L9322 : −40°C $\leqslant T_A \leqslant$ +85°C, 4.5V $\leqslant V_{CC} \leqslant$ 7.5V (Unless otherwise specified).

Parameter	Conditions	Min.	Max.	Units
Instruction Cycle Time — t_C		16	40	µs
CKI				
Input Frequency — f_I	÷32 mode	0.8	2.0	MHz
	÷16 mode	0.4	1.0	MHz
	÷8 mode	0.2	0.5	MHz
	÷4 mode	0.1	0.25	MHz
Duty Cycle		30	60	%
Rise Time	f_I = 2 MHz	—	120	ns
Fall Time		—	80	ns

CKI Using RC (÷4)	R = 56 kΩ ± 5%
	C = 100 pF ± 10%			

Instruction Cycle Time		16	28	µs
CKO as SYNC Input				
t_{SYNC}		400	—	ns
INPUTS:				
IN_3-IN_0, G_3-G_0, L_7-L_0				
t_{SETUP}		—	8.0	µs
t_{HOLD}		—	1.3	µs
SI				
t_{SETUP}		—	2.0	µs
t_{HOLD}		—	1.0	µs
OUTPUT PROPAGATION DELAY	Test condition: C_L = 50 pF, R_L = 20 kΩ, V_{OUT} = 1.5V			
SO, SK Outputs				
t_{PD1}, t_{PD0}		—	4.0	µs
All Other Outputs				
t_{PD1}, t_{PD0}		—	5.6	µs

FIGURE 2 – CONNECTION DIAGRAMS



Order number ETL9420/N ETL9320/N
Package N28A

Order Number ETL9421/N, ETL9321/N
Package N24A

Order number ETL9422/N ETL9322/N
Package N20A

Pin	Description
L_7-L_0	8 bidirectional I/O ports with TRI-STATE®
G_3-G_0	4 bidirectional I/O ports
D_3-D_0	4 general purpose outputs
IN_3-IN_0	4 general purpose inputs (ETL9420/L9320 only)
SI	Serial input (or counter input)
SO	Serial output (or general purpose output)

Pin	Description
SK	Logic-controlled clock (or general purpose output)
CKI	System oscillator input
CKO	System oscillator output (or general purpose input, RAM power supply or SYNC input)
RESET	System reset input
V_{CC}	Power supply
GND	Ground

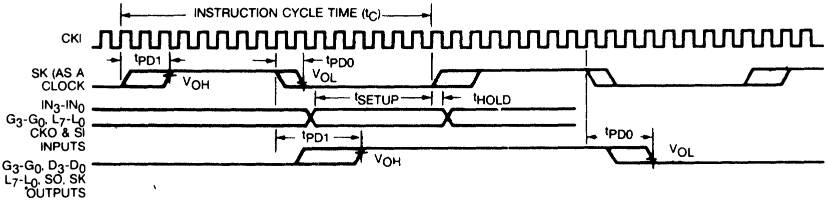


FIGURE 3 – INPUT/OUTPUT TIMING DIAGRAMS (CRYSTAL DIVIDE-BY-16 MODE)

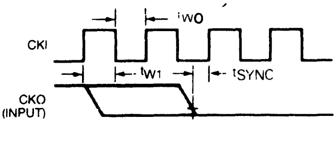


FIGURE 3a – SYNCHRONIZATION TIMING

FUNCTIONAL DESCRIPTION

For ease of reading this description, only ETL9420 and/or ETL9421 are referenced ; however, all such references apply also to ETL9320, L9321, L9322 or ETL9422.

A block diagram of the ETL9420 is given in Figure 1. Data paths arle illustrated in simplified form to depict how the various logic elements communicate with each other in implementing the instruction set of the device. Positive logic is used. When a bit is set, it is a logic "1" (greater than 2 volts). When a bit is reset, it is a logic "0" (less than 0.8 volts).

Program Memory

Programm Memory consists of a 1,024 byte ROM. As can be seen by an examination of the ETL9420, L9421, L9422 instruction set, these words may be program instructions, program data or ROM addressing data. Because of the special characteristics associated with the JP, JSRP, JID and LQUID instructions, ROM must often be thought of as being organized into 16 pages of 64 words each.

ROM addressing is accomplished by a 10-bit PC register. Its binary value selects one of the 1,024 8-bit words contained in ROM. A new address is loaded into the PC register during each instruction cycle. Unless the instruction is a transfer of control instruction, the PC register is loaded with the next sequential 10-bit binary count value. Three levels of subroutine nesting are implemented by the 10-bit subroutine save registers, SA, SB and SC, providing a last-in, first-out (LIFO) hardware subroutine stack.

ROM instruction words are fetched, decoded and executed by the Instruction Decode, Control and Skip Logic circuitry.

Data Memory

Data memory consists of a 256-bit RAM, organized as 4 data registers of 16 4-bit digits. RAM addressing is implemented by a 6-bit B register whose upper 2 bits (Br) select 1 of 4 data registers and lower 4 bits (Bd) select 1 of 16 4-bit digits in the selected data register. While the 4-bit contents of the selected RAM digit (M) is usually loaded into or from, or exchanged with, the A register (accumulator), it may also be loaded into or from the Q latches or loaded from the L ports. RAM addressing may also be performed directly by the LDD and XAD instructions based upon the 6-bit contents of the operand field of these instructions. The Bd register also serves as a source register for 4-bit data sent directly to the D outputs.

Internal Logic

The 4-bit A register (accumulator) is the source and destination register for most I/O, arithmetic, logic and data memory access operations. It can also be used to load the Br and Bd portions of the B register, to load and input 4 bits of the 8-bit Q latch data, to input 4 bits of the 8-bit L I/O port data and to perform data exchanges with the SIO register.

At 4-bit adder performs the arithmetic and logic functions of the ELT9420, L9421, L9422, storing its results in A. It also outputs a carry bit tho the 1 bit C register, most often employed to indicate arithmetic overflow. The C register, in conjuction with the XAS instruction and the EN register, also serves to control the SK output. C can be outputted directly to SK or can enable SK to be a sync clock each instruction cycle time.(See XAS instruction and EN register description, below.)

Four general-purpose inputs, IN_3-IN_0, are provided.

The D register provides 4 general-purpose outputs and is used as the destination register for the 4-bit contents of Bd. The D outputs can be directly connected to the digits of a multiplexed LED display.

The G register contents are outputs to 4 general-purpose bidirectional I/O ports. G I/O ports can be directly connected to the digits of a multiplexed LED display.

The Q register is an internal, latched, 8-bit register, used to hold data loaded to or from M and A, as well as 8-bit data from ROM. Its contents are outputted to the L I/O ports when the L drivers are enabled under program control. (See LEI instruction.)

The 8 L drivers, when enabled, output the contents of latched Q data to the L I/O ports. Also, the contents of L may be read directly into A and M. L I/O ports can be directly connected to the segments of a multiplexed LED display (using the LED Direct Drive output configuration option) with Q data being outputted to the Sa-Sg and decimal point segments of the display.

The SIO register functions as a 4-bit serial-in/serial-out shift register or as a binary counter depending on the contents of the EN register. (See EN register description, below.) Its contents can be exchanged with A, allowing it to input or output a continuous serial data stream. SIO may also be used to provide additional parallel I/O by connecting SO to external serial-in/parallel-out shift registers.

The XAS instruction copies C into the SKL latch. In the counter mode, SK is the output of SKL; in the shift register mode, SK outputs SKL ANDed with the clock.

The EN register is an internal 4-bit register loaded under program contol by the LEI instruction. The state of each bit of this register selects or deselects the particular feature associated with each bit of the EN register (EN_3-EN_0).

1. The least significant bit of the enable register, EN_0, selects the SIO register as either a 4-bit shift register or a 4-bit binary counter. With EN_0 set, SIO is an asynchronous binary counter, *decrementing* its value by one upon each low-going pulse ("1" to "0") occurring on the SI input. Each pulse must be at least two instruction cycles wide. SK outputs the value of SKL. The SO output is equal to the value of EN_3. With EN_0 reset, SIO is a serial shift register shifting left each instruction cycle time. The data present at SI goes into the least significant bit of SIO. SO can be enabled to output the most significant bit of SIO each cycle time. (See 4 below.) The SK output becomes a logic-controlled clock.
2. With EN_1 set the IN_1 input is enabled as an interrupt input. Immediately following an interrupt, EN_1 is reset to disable further interrupts.
3. With EN_2 set, the L drivers are enabled to output the data in Q to the L I/O ports. Resetting EN_2 disables

the L drivers, placing the L I/O ports in a high-impedance input state.

4. EN_3 in conjunction with EN_0, affects the SO output. With EN_0 set (binary counter option selected) SO will output the value loaded into EN_3. With EN_0 reset (serial shift register option selected), setting EN_3 enables SO as the output of the SIO shift register outputting serial shifted data each instruction time. Resetting EN_3 with the serial shift register option selected disables SO as the shift register output ; data continues to be shifted through SIO and can be exchanged with A via an XAS instruction but SO remains reset to "0". The table below provides a summary of the modes associated with EN_3 and EN_0.

Enable Register Modes — Bits EN_3 and EN_0

EN_3	EN_0	SIO	SI	SO	SK
0	0	Shift Register	Input to Shift Register	0	If SKL = 1, SK = Clock If SKL = 0, SK = 0
1	0	Shift Register	Input to Shift Register	Serial Out	If SKL = 1, SK = Clock If SKL = 0, SK = 0
0	1	Binary Counter	Input to Binary Counter	0	If SKL = 1, SK = 1 If SKL = 0, SK = 0
1	1	Binary Counter	Input to Binary Counter	1	If SKL = 1, SK = 1 If SKL = 0, SK = 0

Interrupt

The following features are associated with the IN_1 interrupt procedure and protocol and must be considered by the programmer when utilizing interrupts.

a. The interrupt, once acknowledged as explained below, pushes the next sequential program counter address (PC + 1) onto the stack, pushing in turn the contents of the other subroutine-save registers to the next lower level (PC + 1 → SA → SB → SC). Any previous contents of SC are lost. The program counter is set to hex address 0FF (the last word of page 3) and EN_1 is reset.

b. An interrupt will be acknowledged only after the following conditions are met:
   1. EN_1 has been set.
   2. A low-going pulse ("1" to "0") at least two instruction cycles wide occurs on the IN_1 input.
   3. A currently executing instruction has been completed.
   4. All successive transfer of control instructions and successive LBIs have been completed (e.g., if the main program is executing a JP instruction which transfers program control to another JP instruction, the interrupt will not be acknowledged until the second JP instruction has been executed).

c. Upon acknowledgement of an interrupt, the skip logic status is saved and later restored upon popping of the stack. For example, if an interrupt occurs during the execution of ASC (Add with Carry, Skip on Carry) instruction which results in carry, the skio logic status is saved and program control is transferred to the interrupt servicing routine at hex address 0FF. At the *end* of the interrupt routine, a RET instruction is executed to "pop" the stack and return program control to the instruction following the original ASC. *At this time*, the skip logic is enabled and skips this instruction because of the previous ASC carry. Subroutines and LQID instructions should not be nested within the interrupt servicing routine since their popping the stack will enable any previously saved main program skips, interfering with the orderly execution of the interrupt routine.

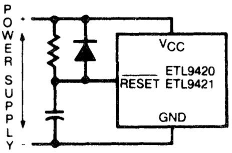
d. The first instruction of the interrupt routine at hex address 0FF must be a NOP.

e. A LEI instruction can be put immediately before the RET to re-enable interrupts.

Initialization

The Reset Logic will initialize (clear) the device upon power-up if the power supply rise time is less than 1ms and greater than 1µs. If the power supply rise time is greater than 1ms, the user must provide an external RC network and diode to the $\overline{\text{RESET}}$ pin as shown below. The $\overline{\text{RESET}}$ pin is configured as a Schmitt trigger input. If not used it should be connected to V_{CC}. Initialization will occur whenever a logic "0" is applied to the $\overline{\text{RESET}}$ input, provided it stays low for at least three instructio cycle times.

Upon initialization, the PC register is cleared to 0 (ROM address 0) and the A, B, C, D, EN, and G registers are cleared. The SK output is enabled as a SYNC output, providing a pulse each instruction cycle time. *Data Memory (RAM) is not cleared upon initialization.* The first instruction at address 0 must be a CLRA.



RC ⩾ 5 X POWER SUPPLY RISE TIME

Power-Up Clear Circuit

Oscillator

There are four basic clock oscillator configurations available as shown by Figure 4.,

a. Crystal Controlled Oscillator. CKI and CKO are connected to an external crystal. The instruction cycle time equals the crystal frequency divided by 32 (optional by 16 or 8).

b. External Oscillator. CKI is an external clock input signal. The external frequency is divided by 32 (optional by 16 or 8) to give the instruction cycle time. CKO is now available to be used as the RAM power supply (V_R), as a general purpose input, or as a SYNC input.

c. RC Controlled Oscillator. CKI is configured as a single pin RC controlled Schmitt trigger oscillator. The instruction cycle equals the oscillation frequency divided by 4. CKO is available as the RAM power supply (V_R) or as a general purpose input

d. Externally Synchronized Oscillator. Intended for use in multi-COP systems, CKO is programmed to function as an input connected to the SK output of another COP chip operating at the same frequency (COP chip with L or C suffix) with CKI connected as shown. In this configuration, the SK output connected to CKO must provide a SYNC (instruction cycle) signal to CKO, thereby allowing synchronous data transfer between the COPs using only the SI and SO serial I/O pins in conjunction with the XAS instruction. Note that on power-up SK is automatically enabled as a SYNC output (See Functional Description, Initialization, above).

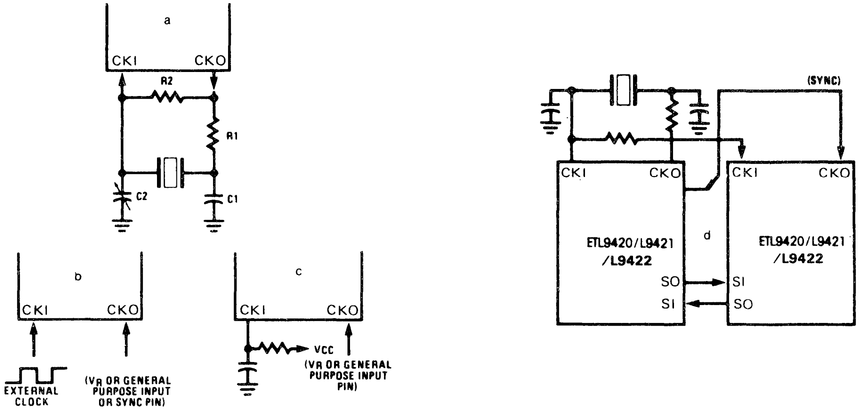
CKO Pin Options

In a crystal controlled oscillator system, CKO is used as an output to the crystal network. As an option CKO can be a SYNC input as described above. As another option CKO can be a general purpose input, read into bit 2 of A (accumulator) upon execution of an INIL instruction. As another option, CKO can be a RAM power supply pin (V_R), allowing its connection to a standby/backup power supply to maintain the integrity of RAM data with minimum power drain when the main supply is inoperative or shut down to conserve power. Using either option is appropriate in application where the ETL9420/L9421/L9422 system timing configuration does not require use of the CKO pin.

RAM Keep-Alive Option (Not available on ETL9422)

Selecting CKO as the RAM power supply (V_R) allows the user to shut off the chip power supply (V_{CC}) and maintain data in the RAM. To insure that RAM data integrity is maintained, the following conditions must be met:

1. $\overline{RESET}$ must go low before V_{CC} goes below spec during power-off; V_{CC} must be within spec before $\overline{RESET}$ goes high on power-up.
2. During normal operation V_R must be within the operating range of the chip, with $(V_{CC} - 1) \leq V_R \leq V_{CC}$.
3. V_R must be $\geq$ 3.3V with V_{CC} off.



Crystal Oscillator

Crystal Value	Component Values			
	R1 (Ω)	R2 (Ω)	C1 (pF)	C2 (pF)
455 kHz	4.7k	1M	220	220
2.097 MHz	1k	1M	30	6-36

RC Controlled Oscillator

R (kΩ)	C (pF)	Instruction Cycle Time (μs)
51	100	19 ± 15%
82	56	19 ± 13%

Note: 200k ≥ R ≥ 25k
360 pF ≥ C ≥ 50 pF

FIGURE 4—ETL9420/L9421/L9422 OSCILLATOR

I/O Options

ETL9420/L9421/L9422 outputs have the following optional configurations, illustrated in Figure 5 :

a. Standard — an enhancement mode device to ground in conjunction with a depletion-mode device to V_{CC}, compatible with LSTTL and CMOS input requirements. Available on SO, SK, and all D and G outputs.

b. Open-Drain — an enhancement-mode device to ground only, allowing external pull-up as required by the user's application. Available on SO, SK, and all D and G outputs.

c. Push-Pull — An enhancement-mode device to ground in conjunction with a depletion-mode device paralleled by an enhancement-mode device to V_{CC}. This configuration has been provided to allow for fast rise and fall times when driving capacitive loads. Available on SO and SK outputs only.

d. Standard L — same as **a.**, but may be disabled. Available on L outputs only.

e. Open Drain L — same as **b.**, but may be disabled. Available on L outputs only.

f. LED Direct Drive — an enhancement-mode device to ground and to V_{CC}, meeting the typical current sourcing requirements of the segments of an LED display. The sourcing device is clamped to limit current flow. These devices may be turned off under program control (See Functional Description, EN Register), placing the outputs in a high-impedance state to provide required LED segment blanking for a multiplexed display. Available on L outputs only.

g. TRI-STATE® Push-Pull — an enhancement-mode device to ground and V_{CC}. These outputs are TRI-STATE outputs, allowing for connection of these outputs to a data bus shared by other bus drivers. Available on L outputs only.

ETL9420/L9421/L9422 inputs have the following optional configurations.

h. An on-chip depletion load device to V_{CC}.

I. A Hi-Z input which must be driven to a "1" or "0" by external components.

The above input and output configurations share common enhancement-mode and depletion-mode devices. Specifically, all configurations use one or more of six devices (numbered 1-6, respectively). Minimum and maximum current (I_{OUT} and V_{OUT}) curves are given in Figure 6 for each of these devices to allow the designer to effectively use these I/O configurations in designing a ETL9420/L9421 system.

The SO, SK outputs can be configured as shown in **a.**, **b.**, or **c.** The D and G outputs can be configured as shown in **a.** or **b.** Note that when inputting data to the G ports, the G outputs should be set to "1". The L outputs can be configured as in **d.**, **e.**, **f.** or **g.**

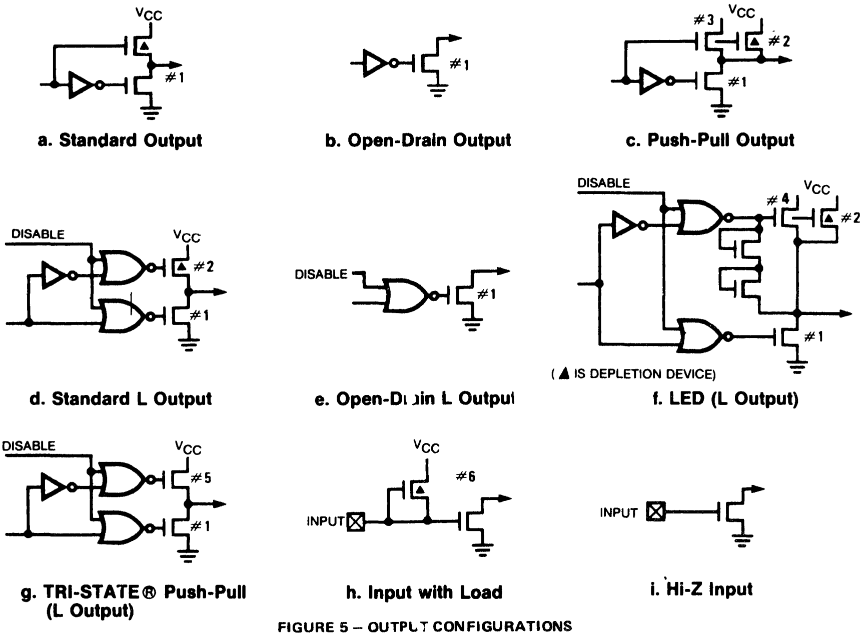
An important point to remember if using configuration **d.** or **f.** with the L drivers is that even when the L drivers are disabled, the depletion load device will source a small amount of current (see Figure 6, device 2); however, when the L lines are used as inputs, the disabled depletion device *cannot* be relied on to source sufficient current to pull an input to a logic 1.

ETL9421

If the ETL9420 is bonded as a 24-pin device, it becomes the ETL9421, illustrated in Figure 2, ETL9420/L9421 Connection Diagrams. Note that the ETL9421 does not contain the four general purpose IN inputs (IN_3-IN_0). Use of this option precludes, of course, use of the IN options and the interrupt feature. All other options are available for the ETL9421.

ETL9422

If the ETL9420is bonded as a 20-pin device, it becomes the ETL9422, as illustrated in Figure 2. Note that the ETL9422 contains all the ETL9421 pins except D_0, D_1, G_0, and G_1, ETL9422 also does not allow RAM power supply input as a valid CKO pin option.



a. Standard Output

b. Open-Drain Output

c. Push-Pull Output

d. Standard L Output

e. Open-Dı ɔin L Outpul

f. LED (L Output)

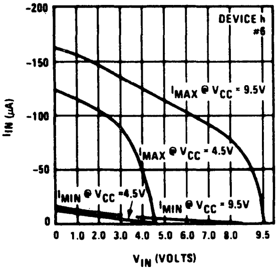
g. TRI-STATE® Push-Pull (L Output)

h. Input with Load

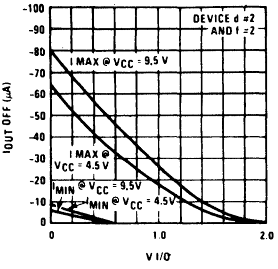
i. Hi-Z Input

FIGURE 5 – OUTPLT CONFIGURATIONS

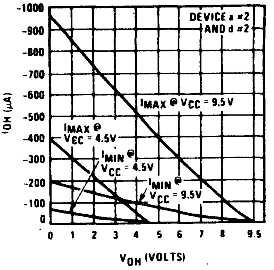
Input current IN_0-IN_3



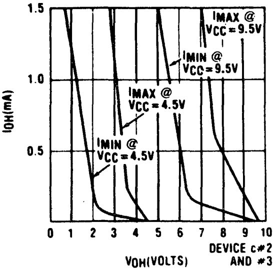
Input current for L_0-L_7 when output programmed OFF by software



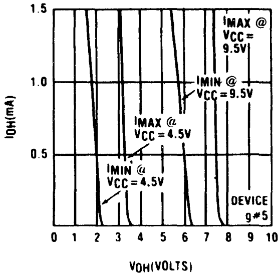
Source current for standard output configuration



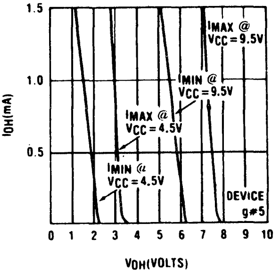
Source current for SO and SK in push-pull configuration

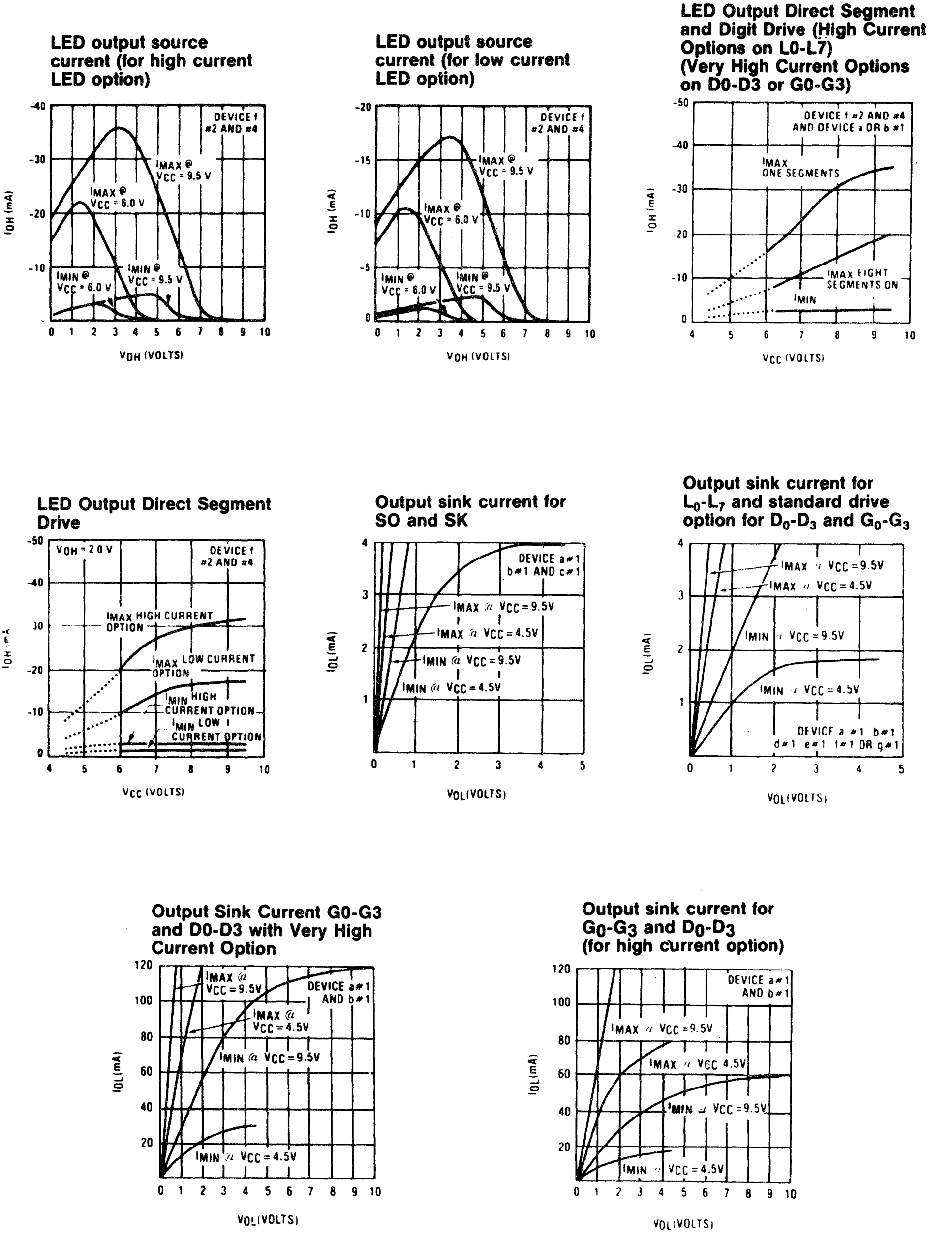


Source Current for L0-L7 in TRI-STATE® Configuration (High Current Option)



Source Current for L0-L7 in TRI-STATE® Configuration (Low Current Option)





LED output source current (for high current LED option)

LED output source current (for low current LED option)

LED Output Direct Segment and Digit Drive (High Current Options on L0-L7) (Very High Current Options on D0-D3 or G0-G3)

LED Output Direct Segment Drive

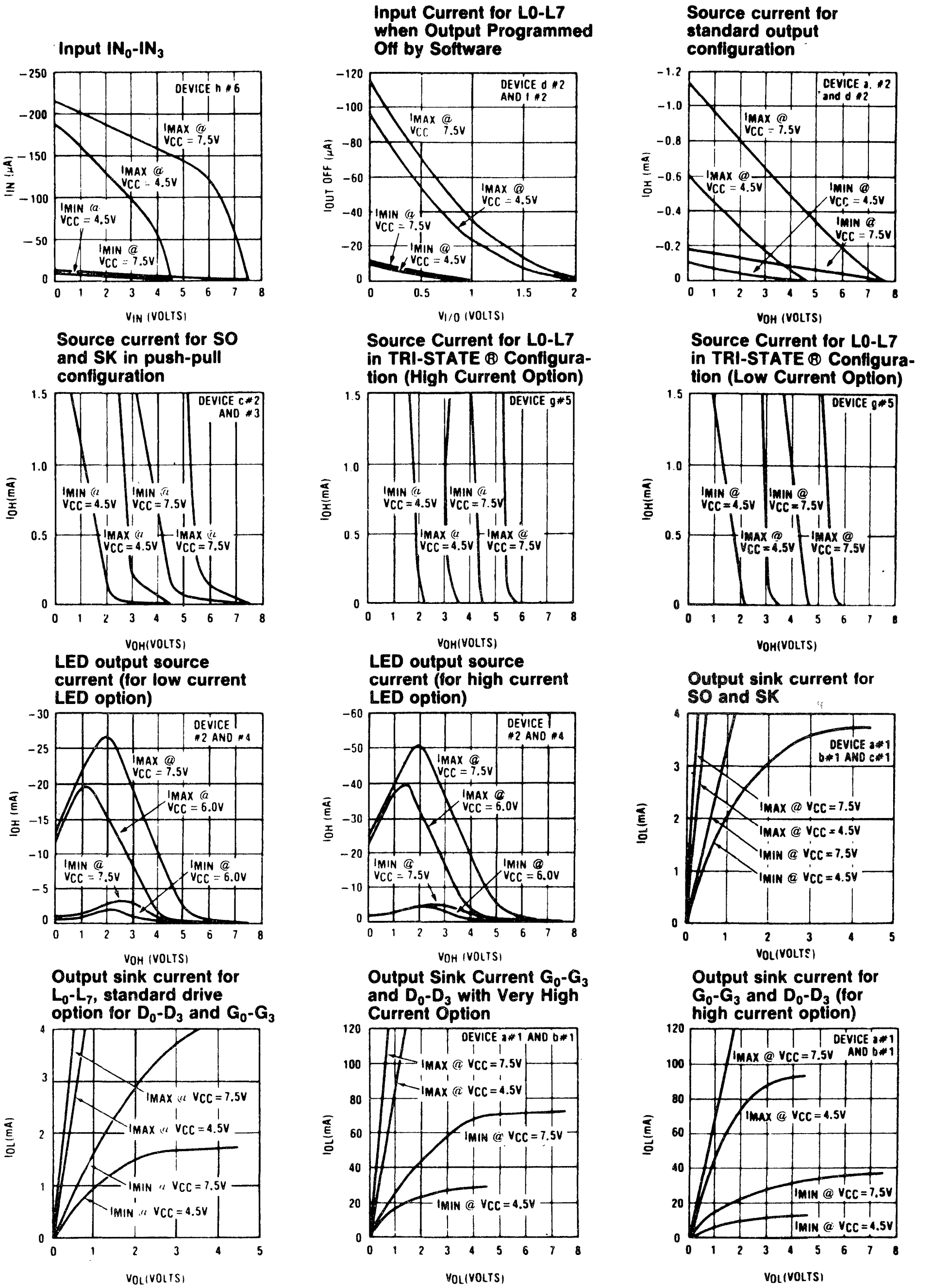
Output sink current for SO and SK

Output sink current for L_0-L_7 and standard drive option for D_0-D_3 and G_0-G_3

Output Sink Current G0-G3 and D0-D3 with Very High Current Option

Output sink current for G_0-G_3 and D_0-D_3 (for high current option)

FIGURE 6 – ETL9420/L9421/L9422 INPUT/OUTPUT CHARACTERISTICS



Input IN_0-IN_3

Input Current for L0-L7 when Output Programmed Off by Software

Source current for standard output configuration

Source current for SO and SK in push-pull configuration

Source Current for L0-L7 in TRI-STATE ® Configuration (High Current Option)

Source Current for L0-L7 in TRI-STATE ® Configuration (Low Current Option)

LED output source current (for low current LED option)

LED output source current (for high current LED option)

Output sink current for SO and SK

Output sink current for L_0-L_7, standard drive option for D_0-D_3 and G_0-G_3

Output Sink Current G_0-G_3 and D_0-D_3 with Very High Current Option

Output sink current for G_0-G_3 and D_0-D_3 (for high current option)

FIGURE 7 – ETL9320/L9321/L9322 INPUT/OUTPUT CHARACTERISTICS

ETL9420/L9421/L9422 - INSTRUCTION SET

Table 1 is a symbol table providing internal architecture, instruction operand and operational symbols used in the instruction set table.

Table 2 provides the mnemonic, operand, machine code, data flow, skip conditions, and description associated with each instruction in the ETL9420/L9421/L9422 instruction set.

TABLE 1 — ETL9420/L9421/L9422 INSTRUCTION SET TABLE SYMBOLS

Symbol	Definition
INTERNAL ARCHITECTURE SYMBOLS	
A	4-bit Accumulator
B	6-bit RAM Address Register
Br	Upper 2 bits of B (register address)
Bd	Lower 4 bits of B (digit address)
C	1-bit Carry Register
D	4-bit Data Output Port
EN	4-bit Enable Register
G	⁴-bit Register to latch data for G I/O Port
IL	Two 1-bit Latches associated with the IN_3 or IN_0 Inputs
IN	4-bit Input Port
L	8-bit TRI-STATE I/O Port
M	4-bit contents of RAM Memory pointed to by B Register
PC	10-bit ROM Address Register (program counter)
Q	8-bit Register to latch data for L I/O Port
SA	10-bit Subroutine Save Register A
SB	10-bit Subroutine Save Register B
SC	10-bit Subroutine Save Register C
SIO	4-bit Shift Register and Counter
SK	Logic-Controlled Clock Output

Symbol	Definition
INSTRUCTION OPERAND SYMBOLS	
d	4-bit Operand Field, 0-15 binary (RAM Digit Select)
r	2-bit Operand Field, 0-3 binary (RAM Register Select)
a	10-bit Operand Field, 0-1023 binary (ROM Address)
y	4-bit Operand Field, 0-15 binary (Immediate Data)
RAM(s)	Contents of RAM location addressed by s
ROM(t)	Contents of ROM location addressed by t

OPERATIONAL SYMBOLS	
+	Plus
−	Minus
→	Replaces
←→	Is exchanged with
=	Is equal to
$\overline{A}$	The ones complement of A
⊕	Exclusive-OR
:	Range of values

TABLE 2 — ETL9420/L9421/L9422 INSTRUCTION SET

Mnemonic	Operand	Hex Code	Machine Language Code (Binary)	Data Flow	Skip Conditions	Description
ARITHMETIC INSTRUCTIONS						
ASC		30	0011 | 0000	A + C + RAM(B) → A Carry → C	Carry	Add with Carry, Skip on Carry
ADD		31	0011 | 0001	A + RAM(B) → A	None	Add RAM to A
ADT		4A	0100 | 1010	A + 10_{10} → A	None	Add Ten to A
AISC	y	5-	0101 | y	A + y → A	Carry	Add Immediate, Skip on Carry (y ≠ 0)
CASC		10	0001 | 0000	$\bar{A}$ + RAM(B) + C → A Carry → C	Carry	Complement and Add with Carry, Skip on Carry
CLRA		00	0000 | 0000	0 → A	None	Clear A
COMP		40	0100 | 0000	$\bar{A}$ → A	None	Ones complement of A to A
NOP		44	0100 | 0100	None	None	No Operation
RC		32	0011 | 0010	"0" → C	None	Reset C
SC		22	0010 | 0010	"1" → C	None	Set C
XOR		02	0000 | 0010	A $\oplus$ RAM(B) → A	None	Exclusive-OR RAM with A
TRANSFER OF CONTROL INSTRUCTIONS						
JID		FF	1111 | 1111	ROM ($PC_{9:8}$, A,M) → $PC_{7:0}$	None	Jump Indirect (Note 3)
JMP	a	6- --	0110 | 00 | $a_{9:8}$ $a_{7:0}$	a → PC	None	Jump
JP	a	-- --	1 | $a_{6:0}$ (pages 2,3 only) or 11 | $a_{5:0}$ (all other pages)	a → $PC_{6:0}$ a → $PC_{5:0}$	None	Jump within Page (Note 4)
JSRP	a	--	10 | $a_{5:0}$	PC + 1 → SA → SB → SC 0010 → $PC_{9:6}$ a → $PC_{5:0}$	None	Jump to Subroutine Page (Note 5)
JSR	a	6- --	0110 | 10 | $a_{9:8}$ $a_{7:0}$	PC + 1 → SA → SB → SC a → PC	None	Jump to Subroutine
RET		48	0100 | 1000	SC → SB → SA → PC	None	Return from Subroutine
RETSK		49	0100 | 1001	SC → SB → SA → PC	Always Skip on Return	Return from Subroutine then Skip

TABLE 2 — ETL9420/L9421/L9422 INSTRUCTION SET (continued)

Mnemonic	Operand	Hex Code	Machine Language Code (Binary)	Data Flow	Skip Conditions	Description
MEMORY REFERENCE INSTRUCTIONS						
CAMQ		33 3C	0011 0011 0011 1100	A → $Q_{7:4}$ RAM(B) → $Q_{3:0}$	None	Copy A, RAM to Q
CQMA		33 2C	0011 0011 0010 1100	$Q_{7:4}$ → RAM(B) $Q_{3:0}$ → A	None	Copy Q to RAM, A
LD	r	-5	00 r 0101	RAM(B) → A Br ⊕ r → Br	None	Load RAM into A, Exclusive-OR Br with r
LDD	r,d	23 --	0010 0011 00 r d	RAM(r,d) → A	None	Load A with RAM pointed to directly by r,d
LQID		BF	1011 1111	ROM($PC_{9:8}$,A,M) → Q SB → SC	None	Load Q Indirect (Note 3)
RMB	0 1 2 3	4C 45 42 43	0100 1100 0100 0101 0100 0010 0100 0011	0 → $RAM(B)_0$ 0 → $RAM(B)_1$ 0 → $RAM(B)_2$ 0 → $RAM(B)_3$	None	Reset RAM Bit
SMB	0 1 2 3	4D 47 46 4B	0100 1101 0100 1101 0100 0110 0100 1011	1 → $RAM(B)_0$ 1 → $RAM(B)_1$ 1 → $RAM(B)_2$ 1 → $RAM(B)_3$	None	Set RAM Bit
STII	y	7-	0111 y	y → RAM(B) Bd + 1 →Bd	None	Store Memory Immediate and Increment Bd
X	r	-6	00 r 0110	RAM(B) ↔ A Br ⊕ r → Br	None	Exchange RAM with A, Exclusive-OR Br with r
XAD	r,d	23 --	0010 0011 10 r d	RAM(r,d) ↔ A	None	Exchange A with RAM pointed to directly by r,d
XDS	r	-7	00 r 0111	RAM(B) ↔ A Bd − 1 → Bd Br ⊕ r → Br	Bd decrements past 0	Exchange RAM with A and Decrement Bd, Exclusive-OR Br with r
XIS	r	-4	00 r 0100	RAM(B) ↔ A Bd + 1 → Bd Br ⊕ r → Br	Bd increments past 15	Exchange RAM with A and Increment Bd, Exclusive-OR Br with r
REGISTER REFERENCE INSTRUCTIONS						
CAB		50	0101 0000	A → Bd	None	Copy A to Bd
CBA		4E	0100 1110	Bd → A	None	Copy Bd to A
LBI	r,d	-- 33 --	00 r (d − 1) (d = 0, 9:15) or 0011 0011 10 r d (any d)	r,d → B	Skip until not a LBI	Load B Immediate with r,d (Note 6)
LEI	y	33 6-	0011 0011 0110 y	y → EN	None	Load EN Immediate (Note 7)
XABR		12	0001 0010	A ↔ Br (0,0 → A_3,A_2)	None	Exchange A with Br

TABLE 2 — ETL9420/L9421/L9422 INSTRUCTION SET (continued)

Mnemonic	Operand	Hex Code	Machine Language Code (Binary)	Data Flow	Skip Conditions	Description
TEST INSTRUCTIONS						
SKC		20	0010 0000		C = "1"	Skip if C is True
SKE		21	0010 0001		A = RAM(B)	Skip if A Equals RAM
SKGZ		33 21	0011 0011 0010 0001		$G_{3:0} = 0$	Skip if G is Zero (all 4 bits)
SKGBZ		33	0011 0011	1st byte		Skip if G Bit is Zero
	0	01	0000 0001	2nd byte	$G_0 = 0$	
	1	11	0001 0001		$G_1 = 0$	
	2	03	0000 0011		$G_2 = 0$	
	3	13	0001 0011		$G_3 = 0$	
SKMBZ	0	01	0000 0001		$RAM(B)_0 = 0$	Skip if RAM Bit is Zero
	1	11	0001 0001		$RAM(B)_1 = 0$	
	2	03	0000 0011		$RAM(B)_2 = 0$	
	3	13	0001 0011		$RAM(B)_3 = 0$	
SKT		41	0100 0001		A time-base counter carry has occurred since last test	Skip on Timer (Note 3)
INPUT/OUTPUT INSTRUCTIONS						
ING		33 2A	0011 0011 0010 1010	G → A	None	Input G Ports to A
ININ		33 28	0011 0011 0010 1000	IN → A	None	Input IN Inputs to A (Note 2)
INIL		33 29	0011 0011 0010 1001	IL_3, CKO, "0", IL_0 → A	None	Input IL Latches to A (Note 3)
INL		33 2E	0011 0011 0010 1110	$L_{7:4}$ → RAM(B) $L_{3:0}$ → A	None	Input L Ports to RAM,A
OBD		33 3E	0011 0011 0011 1110	Bd → D	None	Output Bd to D Outputs
OGI	y	33 5-	0011 0011 0101 y	y → G	None	Output to G Ports Immediate
OMG		33 3A	0011 0011 0011 1010	RAM(B) → G	None	Output RAM to G Ports
XAS		4F	0100 1111	A ↔ SIO, C → SKL	None	Exchange A with SIO (Note 3)

Note 1 : All subscripts for alphabetical symbols indicate bit numbers unless explicitly defined (e.g., Br and Bd are explicitly defined). Bits are numbered 0 to N where 0 signifies the least significant bit (low-order, right-most bit). For example, A_3 indicates the most significant (left-most) bit of the 4-bit A register.

Note 2 : The ININ instruction is only available on the 28-pin ETL9420 as the other devices do not contain the IN inputs.

Note 3 : For additional information on the operation of the XAS, JID, LQID, INIL, and SKT instructions, see below.

Note 4 : The JP instruction allows a jump, while in subroutine pages 2 or 3, to any ROM location within the two-page boundary of pages 2 or 3. The JP instruction, otherwise, permits a jump to a ROM location within the current 64-word page. JP may not jump to the last word of a page.

Note 5 : A JSRP transfers program control to subroutine page 2 (0010 is loaded into the upper 4 bits of P). A JSRP may not be used when in pages 2 or 3. JSRP may not jump to the last word in page 2.

Note 6 : LBI is a single-byte instruction if d = 0, 9, 10, 11, 12, 13, 14 or 15. The machine code for the lower 4 bits equals the binary value of the "d" data *minus 1*, e.g., to load the lower four bits of B (Bd) with the value 9 (1001_2), the lower 4 bits of the LBI instruction equal 8 (1000_2). To load 0, the lower 4 bits of the LBI instruction should equal 15 (1111_2).

Note 7 : Machine code for operand field y for LEI instruction should equal the binary value to be latched into EN, where a "1" or "0" in each bit of EN corresponds with the selection or deselection of a particular function associated with each bit. (See Functional Description EN Register).

The following information is provided to assist the user in understanding the operation of several unique instructions and to provide note useful to programmers in writing ETL9420/L9421/L9422 programs.

XAS Instruction

XAS (Exchange A with SIO) exchanges the 4-bit contents of the accumulator with the 4-bit contents of the SIO register. The contents of SIO will contain serial-in/serial-out shift register or binary counter data, depending on the value of the EN register. An XAS instruction will also affect the SK output. (See Functional Description, EN Register, above.) If SIO is selected as a shift register, an XAS instruction must be performed once every 4 instruction cycles to effect a continuous data stream.

JID Instruction

JID (Jump Indirect) is an indirect addressing instruction, transferring program control to a new ROM location pointed to indirectly by A and M. It loads the lower 8 bits of the ROM address register PC with the *contents* of ROM addressed by the 10-bit word, $PC_{9:8}$, A, M, PC_9 and PC_8 are not affected by this instruction.

Note that JID requires 2 instruction cycles to execute.

INIL Instruction

INIL (Input IL Latches to A) inputs 2 latches, IL_3 and IL_0 (see Figure 8) and CKO into A. The IL_3 and IL_0 latches are set if a low-going pulse ("1" to "0") has occurred on the IN_3 and IN_0 inputs since the last INIL instruction, provided the input pulse stays low for at least two instruction times. Execution of an INIL inputs IL_3 and IL_0 into A3 and A0 respectively, and resets these latchês to allow them to respond to subsequent low-going pulses on the IN_3 and IN_0 lines. If CKO is mask programmed as a general purpose input, an INIL will input the state of CKO into A2. If CKO has not been so programmed, a "1" will be placed in A2. A "0" is always placed in A1 upon the execution of an INIL. The general purpose inputs IN_3-IN_0 are input to A upon execution of an ININ instruction. (See Table 2, ININ instruction.) INIL is useful in recognizing pulses of short duration or pulses which occur too often to be read conveniently by an ININ instruction. IL latches are *not cleared* on reset.

LQID Instruction

LQID (Load Q Indirect) loads the 8-bit Q register with the contents of ROM pointed to by the 10-bit word PC_9, PC_8, A, M. LQID can be used for table lookup or code conversion such as BCD to seven-segment. The LQID instruction "pushes" the stack (PC + 1 → SA → SB → SC) and replaces the least significant 8 bits of PC as follows: A → $PC_{7:4}$, RAM(B) → $PC_{3:0}$, leaving PC_9 and PC_8 unchanged. The ROM data pointed to by the new address is fetched and loaded into the Q latches. Next, the stack is "popped" (SC → SB → SA → PC), restoring the saved value of PC to continue sequential program execution. Since LQID pushes SB → SC, the previous contents of SC are lost. Also, when LQID pops the stack, the previously pushed contents of SB are left in SC. The net result is that the contents of SB are placed in SC (SB → SC). Note that LQID takes two instruction cycle times to execute.

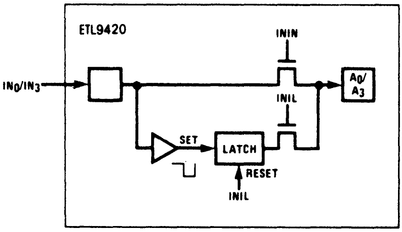


FIGURE 8 – INIL HARDWARE IMPLEMENTATION

SKT Instruction

The SKT (Skip On Timer) instruction tests the state of an internal 10-bit time-base counter. This counter divides the instruction cycle clock frequency by 1024 and provides a latched indication of counter overflow. The SKT instruction tests this latch, executing the next program instruction if the latch is not set. If the latch has been set since the previous test, the next program instruction is skipped and the latch is reset. The features associated with this instruction, therefore, allow the ELT9420/L9421/L9422 to generate its own tim-base for real-time processing rather than relying on an external input signal.

For example, using a 2.097 MHz crystal as the time-base to the clock generator, the instruction cycle clock frequency will be 65kHz (crystal frequency ÷ 32) and the binary counter output pulse frequency will be 64 Hz. For time-of-day or similar real-time processing, the SKT instruction can call a routine which increments a "seconds" counter every 64 ticks.

Instruction Set Notes

a. The first word of a ETL9420/L9421/L9422 programm (ROM address 0) must be a CLRA (Clear A) instruction.

b. Although skipped instructions are not executed, one instruction cycle time is devoted to skipping each byte of the skipped instruction. Thus all program paths except JID and LQID take the same number of cycle times whether instructions are skipped or executed. JID and LQID instructions take 2 cycles if executed and 1 cycle if skipped.

c. The ROM is organized into 16 pages of 64 words each. The Program Counter is a 10-bit binary counter, and will count through page boundaries. If a JP, JSRP, JID or LQID instruction is located in the last word of a page, the instruction operates as if it were in the next page. For example: a JP located in the last word of a page will jump to a location in the next page. Also, a LQID or JID located in the last word of page 3, 7, 11, or 15 will access data in the next group of four pages.

OPTION LIST

The ETL9420/L9421/L9422 mask-programmable options ar assigned numbers which correspond with the ETL9420 pins.

The following is a list of ETL9420 options. When specifying a ETL9421 chip, Options 9, 10, 19, and 20 must all be set to zero. When specifying a ETL9422 chip, options 9, 10, 19, and 20 must all be set to zero ; options 21 and 22 may not be set to one, three or five ; and option 2 may not be set to one. The options are programmed at the same time as the ROM pattern to provide the user with the hardware flexibility to interface to various I/O components using little or no external circuitry.

Option 1 = 0: Ground Pin — no options available

Option 2: CKO Output
- = 0: clock generator output to crystal/resonator (0 not allowable value if Option 3 = 3)
- = 1: pin is RAM power supply (V_R) input (not available on the ETL9422)
- = 2: general purpose input with load device to V_{CC}
- = 3: general purpose input, Hi-Z
- = 4: multi-COP SYNC input (CKI ÷ 32, CKI ÷ 16)
- = 5: multi-COP SYNC input (CKI ÷ 8)

Option 3: CKI Input
- = 0: oscillator input divided by 32 (2 MHz max.)
- = 1: oscillator input divided by 16 (1 MHz max.)
- = 2: oscillator input divided by 8 (500 kHz max.)
- = 3: single-pin RC controlled oscillator (÷ 4)
- = 4: Schmitt trigger clock input (÷ 4)

Option 4: RESET Input
- = 0: load device to V_{CC}
- = 1: Hi-Z input

Option 5: L_7 Driver
- = 0: Standard output
- = 1: Open-drain output
- = 2: High current LED direct segment drive output
- = 3: High current TRI-STATE® push-pull output
- = 4: Low-current LED direct segment drive output
- = 5: Low-current TRI-STATE® push-pull output

Option 6: L_6 Driver
same as Option 5

Option 7: L_5 Driver
same as Option 5

Option 8: L_4 Driver
same as Option 5

Option 9: IN_1 Input
- = 0: load device to V_{CC}
- = 1: Hi-Z input

Option 10: IN_2 Input:
same as Option 9

Option 11: V_{CC} pin
- = 0: Standard V_{CC}
- = 1: Optional higher voltage V_{CC}

Option 12: L_3 Driver
same as Option 5

Option 13: L_2 Driver
same as Option 5

Option 14: L_1 Driver
same as Option 5

Option 15: L_0 Driver
same as Option 5

Option 16: SI Input
same as Option 9

Option 17: SO Driver
- = 0: standard output
- = 1: open-drain output
- = 2: push-pull output

Option 18: SK Driver
same as Option 17

Option 19 : IN_0 Input
same as Option 9

Option 20: IN_3 Input
same as Option 9

Option 21: G_0 I/O Port
- = 0: very-high current standard output
- = 1: very-high current open-drain output
- = 2: high current standard output
- = 3: high current open-drain output
- = 4: standard LSTTL output (fanout = 1)
- = 5: open-drain LSTTL output (fanout = 1)

Option 22: G_1 I/O Port
same as Option 21

Option 23: G_2 I/O Port
same as Option 21

Option 24: G_3 I/O Port
same as Option 21

Option 25: D_3 Output
same as Option 21

Option 26: D_2 Output
same as Option 21

Option 27: D_1 Output
same as Option 21

Option 28: D_0 Output
same as Option 21

Option 29: L Input Levels
- = 0: standard TTL input levels ("0" = 0.8V, "1" = 2.0V)
- = 1: higher voltage input levels ("0" = 1.2V, "1" = 3.6V)

Option 30: IN Input Levels
same as Option 29

Option 31: G Input Levels
same as Option 29

Option 32: SI Input Levels
same as Option 29

Option 33: $\overline{\text{RESET}}$ Input
- = 0: Schmitt trigger input
- = 1: standard TTL input levels
- = 2: higher voltage input levels

Option 34: CKO Input Levels (CKO = input; Option 2 = 2,3)
same as Option 29

Option 35 COP Bonding
- = 0: ETL9420 (28-pin device)
- = 1: ETL9421 (24-pin device)
- = 2: 28 and 24-pin versions
- = 3: ETL9422 (20-pin device)
- = 4: 28- and 20-pin versions
- = 5: 24- and 20-pin versions
- = 6: 28-, 24-, and 20-pin versions

TEST MODE (Non-Standard Operation)

The SO output has been configured to provide for standard test procedures for the custom-programmed ETL9420. With SO forced to logic "1", two test modes are provided, depending upon the value of SI :

a. RAM and Internal Logic Test Mode (SI = 1)

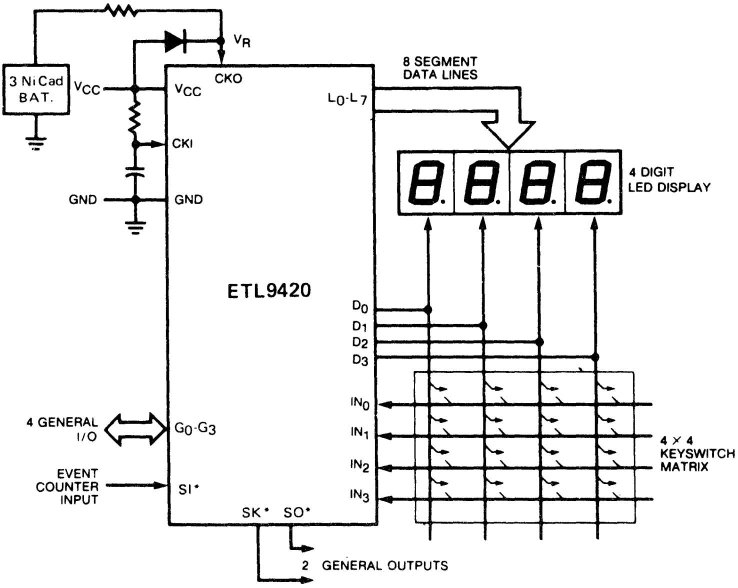
b. ROM Test Mode (SI = 0)

These special test modes should not be employed by the user; they are intended for manufacturing test only.

APPLICATION 1 : ETL9420 General Controller

Figure 9 shows an interconnect diagram for a ETL9420 used as a general controller. Operation of the system is as follows :

1. The L_7-L_0 outputs are configured as LED Direct Drive outputs, allowing direct connection to the segments of the display.
2. The D_3-D_0 outputs drive the digits of the multiplexed display directly and scan the columns of the 4 × 4 keyboard matrix.
3. The IN_3-IN_0 inputs are used to input the 4 rows of the keyboard matrix. Reading the IN lines in conjunction with the current value of the D outputs allows detection, debouncing, and decoding of any one of the 16 keyswitches.
4. CKI is configured as a single-pin oscillator input allowing system timing to be controlled by a single-pin RC network. CKO is therefore available for use as a V_R RAM power supply pin. RAM data integrity is thereby assured when the main power supply is shut down (see RAM Keep-Alive option description).
5. SI is selected as the input to a binary counter input. With SIO used as a binary counter, SO and SK can be used as general purpose outputs.
6. The 4 bidirectional G I/O ports (G_3-G_0) are available for use as required by the user's application.

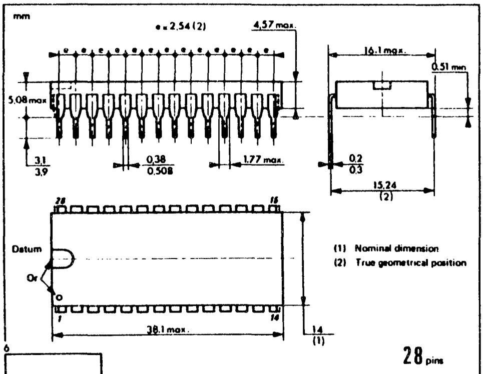


*SO, SI, SK MAY ALSO BE USED FOR SERIAL I/O

FIGURE 9 – ETL9420 KEYBOARD/DISPLAY INTERFACE

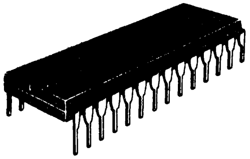
COPS, MICROWIRE and TRI-STATE are registered trademarks of National Semiconductor Corp.

PHYSICAL DIMENSIONS

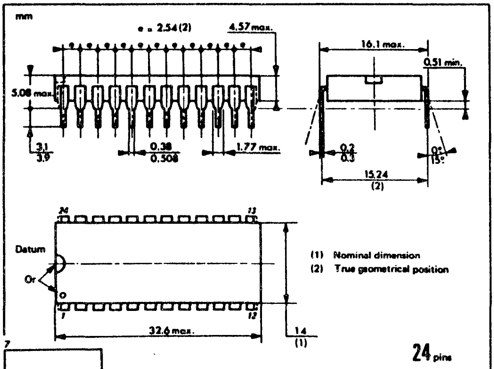


DIN				
A51E CEI	D.A.T.A.	JEDEC	F 144 SITELESC	CB-132

CB-132

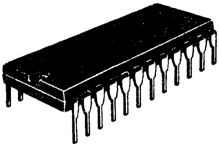


N SUFFIX
PLASTIC PACKAGE



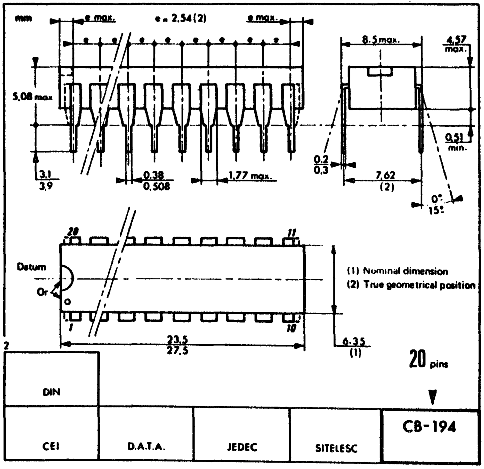
DIN				
A51D CEI	D.A.T.A.	JEDEC	F 118 SITELESC	CB-68

CB-68

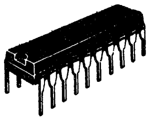


N SUFFIX
PLASTIC PACKAGE

PHYSICAL DIMENSIONS



CB-194



N SUFFIX
PLASTIC PACKAGE

These specifications are subject to change without notice.
Please inquire with our sales offices about the availability of the different products.

Printed in France

Ref.DSETL9394202122

THOMSON COMPONENTS
SEMICONDUCTEURS

ETL9444 • ETL9445 • ETL9344 • ETL9345

SINGLE CHIP MICROCONTROLLERS

The ETL9444/L9445 and ETL9344/L9345 Single-Chip N-Channel Microcontrollers are fully compatible with the COPS® family, fabricated using N-channel, silicon gate XMOS technology. They are complete microcomputers containing all system timing, internal logic, ROM, RAM and I/O necessary to implement dedicated control functions in a variety of applications. Features include single supply operation, a variety of output configuration options, with an instruction set, internal architecture and I/O scheme designed to facilitate keyboard input, display output and BCD data manipulation. The ETL9445 is identical to the ETL9444, except with 19 I/O lines instead of 23 : They are an appropriate choice for use in numerous human interface control environments. Standard test procedures and reliable high-density fabrication techniques provide the medium to large volume customers with a customized controller oriented processor at a low end-product cost.

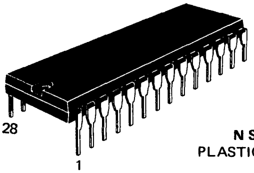
The ETL9344/L9345 are exact functional equivalents, but extended temperature range versions of the ETL9444/L9445 respectively.

- Low cost
- Powerful instruction set
- 2k × 8 ROM, 128 × 4 RAM
- 23 I/O lines (ETL9444)
- True vectored interrupt, plus restart
- Three-level subroutine stack
- 16 μs instruction time
- Single supply operation (4.5-6.3V)
- Low current drain (13mA max.)
- Internal time-base counter for real-time processing
- Internal binary counter register with MICROWIRE® serial I/O capability
- General purpose and TRI-STATE® outputs
- LSTTL/CMOS compatible in and out
- Direct drive of LED digit and segment lines
- Software/hardware compatible with other members of ET 9400 family
- Extended temperature range devices ETL9344/L9345 (-40° C to + 85° C)
- Wider supply range (4.5-9.5V) optionally available

NMOS

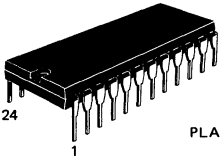
CASES

ETL9444 • ETL9344



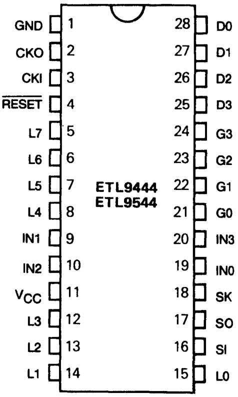
N SUFFIX
PLASTIC PACKAGE

ETL9345 • ETL9345



N SUFFIX
PLASTIC PACKAGE

PIN ASSIGNMENT



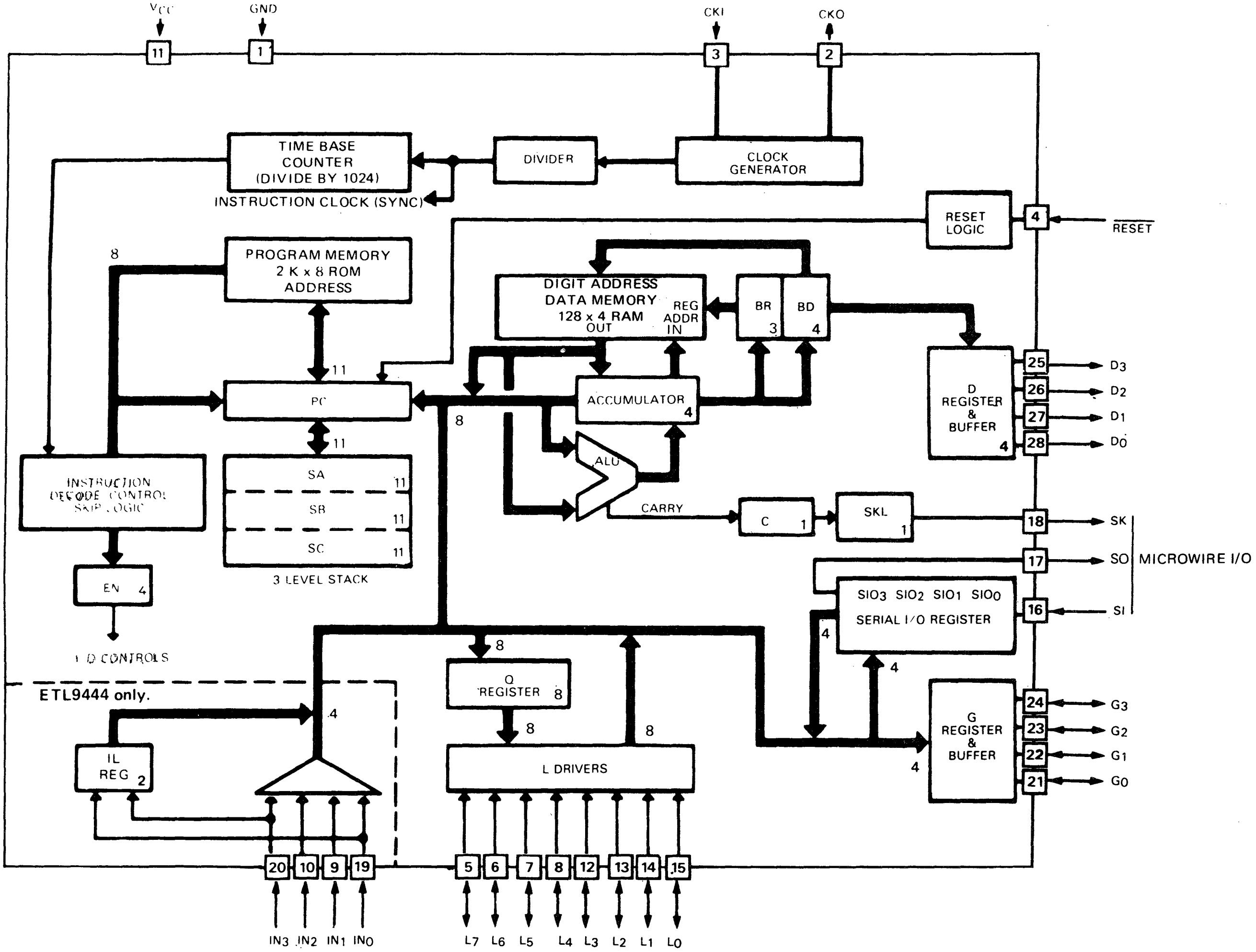


FIGURE 1 – BLOCK DIAGRAM (28-pin version)

ETL9444/L9445

ABSOLUTE MAXIMUM RATINGS

Voltage at Any Pin Relative to GND	−0.5V to +10V
Ambient Operating Temperature	0°C to +70°C
Ambient Storage Temperature	−65°C to +150°C
Lead Temperature (Soldering, 10 seconds)	300°C
Power Dissipation	0.75 Watt at 25°C 0.4 Watt at 70°C
Total Source Current	120 mA
Total Sink Current	120 mA

Absolute maximum ratings indicate limits beyond which damage to the device may occur. DC and AC electrical specifications are not ensured when operating the device at absolute maximum ratings.

DC ELECTRICAL CHARACTERISTICS 0°C ⩽ T_A ⩽ +70°C, 4.5V ⩽ V_{CC} ⩽ 9.5V (Unless otherwise specified)

Parameter	Conditions	Min.	Max.	Units
Standard Operating Voltage (V_{CC})	Note 1	4.5	6.3	V
Optional Operating Voltage (V_{CC})		4.5	9.5	V
Power Supply Ripple	peak to peak	–	0.5	V
Operating Supply Current	all inputs and outputs open	–	13	mA
Input Voltage Levels				
CKI Input Levels				
Crystal Input (÷32, ÷16, ÷8)				
Logic High (V_{IH})		2.0	–	V
Logic Low (V_{IL})		−0.3	0.4	V
Schmitt Trigger Input (÷4)				
Logic High (V_{IH})		0.7 V_{CC}	–	V
Logic Low (V_{IL})		−0.3	0.6	V
$\overline{\text{RESET}}$ Input Levels	Schmitt trigger input			
Logic High		0.7 V_{CC}	–	V
Logic Low		−0.3	0.6	V
SO Input Level (Test mode)		2.0	2.5	V
All Other Inputs				
Logic High	V_{CC} = Max.	3.0	–	V
Logic High	with TTL trip level options	2.0	–	V
Logic Low	selected, V_{CC} = 5V ± 5%	−0.3	0.8	V
Logic High	with high trip level options	3.6	–	V
Logic Low	selected	−0.3	1.2	V
Input Capacitance		–	7	pF
Hi-Z Input Leakage		−1	+1	μA
Output Voltage Levels				
LSTTL Operation	V_{CC} = 5V ± 5%			
Logic High (V_{OH})	I_{OH} = −25 μA	2.7	–	V
Logic Low (V_{OL})	I_{OL} = 0.36 mA	–	0.4	V
CMOS Operation				
Logic High	I_{OH} = −10 μA	V_{CC} − 1	–	V
Logic Low	I_{OL} = +10 μA	–	0.2	V

Note 1 : V_{CC} voltage change must be less than 0.5V in a 1 ms period to maintain proper operation.

ETL9444/L9445

DC ELECTRICAL CHARACTERISTICS (continued) 0°C ⩽ T_A ⩽ +70°C, 4.5V ⩽ V_{CC} ⩽ 9.5V (Unless otherwise specified)

Parameter	Conditions	Min.	Max.	Units
Output Current Levels				
Output Sink Current				
SO and SK Outputs (I_{OL})	$V_{CC} = 9.5V$, $V_{OL} = 0.4V$	1.8	–	mA
	$V_{CC} = 6.3V$, $V_{OL} = 0.4V$	1.2	–	mA
	$V_{CC} = 4.5V$, $V_{OL} = 0.4V$	0.9	–	mA
L_0-L_7 Outputs and Standard	$V_{CC} = 9.5V$, $V_{OL} = 0.4V$	0.8	–	mA
G_0-G_3, D_0-D_3 Outputs (I_{OL})	$V_{CC} = 6.3V$, $V_{OL} = 0.4V$	0.5	–	mA
	$V_{CC} = 4.5V$, $V_{OL} = 0.4V$	0.4	–	mA
G_0-G_3 and D_0-D_3 Outputs with	$V_{CC} = 9.5V$, $V_{OL} = 1.0V$	15	–	mA
High Current Options (I_{OL})	$V_{CC} = 6.3V$, $V_{OL} = 1.0V$	11	–	mA
	$V_{CC} = 4.5V$, $V_{OL} = 1.0V$	7.5	–	mA
G_0-G_3 and D_0-D_3 Outputs with	$V_{CC} = 9.5V$, $V_{OL} = 1.0V$	30	–	mA
Very High Current Options (I_{OL})	$V_{CC} = 6.3V$, $V_{OL} = 1.0V$	22	–	mA
	$V_{CC} = 4.5V$, $V_{OL} = 1.0V$	15	–	mA
CKI (Single-pin RC oscillator)	$V_{CC} = 4.5V$, $V_{IH} = 3.5V$	2	–	mA
CKO	$V_{CC} = 4.5V$, $V_{OL} = 0.4V$	0.2	–	mA
Output Source Current				
Standard Configuration,	$V_{CC} = 9.5V$, $V_{OH} = 2.0V$	−140	−800	μA
All Outputs (I_{OH})	$V_{CC} = 6.3V$, $V_{OH} = 2.0V$	−75	−480	μA
	$V_{CC} = 4.5V$, $V_{OH} = 2.0V$	−30	−250	μA
Push-Pull Configuration	$V_{CC} = 9.5V$, $V_{OH} = 4.75V$	−1.4	–	mA
SO and SK Outputs (I_{OH})	$V_{CC} = 6.3V$, $V_{OH} = 2.4V$	−1.4	–	mA
	$V_{CC} = 4.5V$, $V_{OH} = 1.0V$	−1.2	–	mA
LED Configuration, L_0-L_7				
Outputs, Low Current	$V_{CC} = 9.5V$, $V_{OH} = 2.0V$	−1.5	−18	mA
Driver Option (I_{OH})	$V_{CC} = 6.0V$, $V_{OH} = 2.0V$	−1.5	−13	mA
LED Configuration, L_0-L_7				
Outputs, High Current	$V_{CC} = 9.5V$, $V_{OH} = 2.0V$	−3.0	−35	mA
Driver Option (I_{OH})	$V_{CC} = 6.0V$, $V_{OH} = 2.0V$	−3.0	−25	mA
TRI-STATE® Configuration,	$V_{CC} = 9.5V$, $V_{OH} = 5.5V$	−0.75	–	mA
L_0-L_7 Outputs, Low	$V_{CC} = 6.3V$, $V_{OH} = 3.2V$	−0.8	–	mA
Current Driver Option (I_{OH})	$V_{CC} = 4.5V$, $V_{OH} = 1.5V$	−0.9	–	mA
TRI-STATE® Configuration,	$V_{CC} = 9.5V$, $V_{OH} = 5.5V$	−1.5	–	mA
L_0-L_7 Outputs, High	$V_{CC} = 6.3V$, $V_{OH} = 3.2V$	−1.6	–	mA
Current Driver Option (I_{OH})	$V_{CC} = 4.5V$, $V_{OH} = 1.5V$	−1.8	–	mA
Input Load Source Current	$V_{CC} = 5.0V$, $V_{IL} = 0V$	−10	−140	μA
CKO Output				
RAM Power Supply Option				
Power Requirement	$v_R = 3.3V$	–	6.0	mA
TRI-STATE® Output Leakage				
Current		−2.5	+2.5	μA
Total Sink Current Allowed				
All Outputs Combined		–	120	mA
D, G Ports		–	120	mA
L_7-L_4		–	4	mA
L_3-L_0		–	4	mA
All Other Pins		–	1.5	mA
Total Source Current Allowed				
All I/O Combined		–	120	mA
L_7-L_4		–	60	mA
L_3-L_0		–	60	mA
Each L Pin		–	30	mA
All Other Pins		–	1.5	mA

ETL9344/L9345

ABSOLUTE MAXIMUM RATINGS

Voltage at Any Pin Relative to GND	−0.5V to +10V
Ambient Operating Temperature	−40°C to +85°C
Ambient Storage Temperature	−65°C to +150°C
Lead Temperature (Soldering, 10 seconds)	300°C
Power Dissipation	0.75 Watt at 25°C 0.25 Watt at 85°C
Total Source Current	120 mA
Total Sink Current	120 mA

Absolute maximum ratings indicate limits beyond which damage to the device may occur. DC and AC electrical specifications are not ensured when operating the device at absolute maximum ratings.

DC ELECTRICAL CHARACTERISTICS – 40°C ⩽ T_A ⩽ +85°C, 4.5V ⩽ V_{CC} ⩽ 7.5V (Unless otherwise specified)

Parameter	Conditions	Min.	Max.	Units
Standard Operating Voltage (V_{CC})	Note 1	4.5	5.5	V
Optional Operating Voltage (V_{CC})		4.5	7.5	V
Power Supply Ripple	peak to peak	–	0.5	V
Operating Supply Current	all inputs and outputs open	–	15	mA
Input Voltage Levels				
CKI Input Levels				
Crystal Input				
Logic High (V_{IH})		2.2	–	V
Logic Low (V_{IL})		−0.3	0.3	V
Schmitt Trigger Input				
Logic High (V_{IH})		0.7 V_{CC}	–	V
Logic Low (V_{IL})		−0.3	0.4	V
$\overline{\text{RESET}}$ Input Levels	Schmitt Trigger Input			
Logic High		0.7 V_{CC}	–	V
Logic Low		−0.3	0.4	V
SO Input Level (Test mode)		2.2	2.5	V
All Other Inputs				
Logic High	V_{CC} = Max.	3.0	–	V
Logic High	with TTL trip level options	2.2	–	V
Logic Low	selected, V_{CC} = 5V ± 5%	−0.3	0.6	V
Logic High	with high trip level options	3.6	–	V
Logic Low	selected	−0.3	1.2	V
Input Capacitance		–	7	pF
Hi-Z Input Leakage		−2	+2	μA
Output Voltage Levels				
LSTTL Operation	V_{CC} = 5V ± 5%			
Logic High (V_{OH})	$I_{OH} = -20 \mu A$	2.7	–	V
Logic Low (V_{OL})	$I_{OL} = 0.36 mA$	–	0.4	V
CMOS Operation				
Logic High	$I_{OH} = -10 \mu A$	V_{CC} − 1	–	V
Logic Low	$I_{OL} = +10 \mu A$	–	0.2	V

Note 1 : V_{CC} voltage change must be less than 0.5V in a 1 ms period to maintain proper operation.

ETL9344/L9345

DC ELECTRICAL CHARACTERISTICS (continued) $-40°C \leq T_A \leq +85°C$, $4.5V \leq V_{CC} \leq 7.5V$ (Unless otherwise specified).

Parameter	Conditions	Min.	Max.	Units
Output Current Levels				
Output Sink Current				
SO and SK Outputs (I_{OL})	$V_{CC} = 7.5V$, $V_{OL} = 0.4V$	1.4	–	mA
	$V_{CC} = 5.5V$, $V_{OL} = 0.4V$	1.0	–	mA
	$V_{CC} = 4.5V$, $V_{OL} = 0.4V$	0.8	–	mA
L_0-L_7 Outputs, and Standard	$V_{CC} = 7.5V$, $V_{OL} = 0.4V$	0.6	–	mA
G_0-G_3, D_0-D_3 Outputs (I_{OL})	$V_{CC} = 5.5V$, $V_{OL} = 0.4V$	0.5	–	mA
	$V_{CC} = 4.5V$, $V_{OL} = 0.4V$	0.4	–	mA
G_0-G_3 and D_0-D_3 Outputs with	$V_{CC} = 7.5V$, $V_{OL} = 1.0V$	12	–	mA
High Current Options (I_{OL})	$V_{CC} = 5.5V$, $V_{OL} = 1.0V$	9	–	mA
	$V_{CC} = 4.5V$, $V_{OL} = 1.0V$	7	–	mA
G_0-G_3 and D_0-D_3 Outputs with	$V_{CC} = 7.5V$, $V_{OL} = 1.0V$	24	–	mA
Very High Current Options (I_{OL})	$V_{CC} = 5.5V$, $V_{OL} = 1.0V$	18	–	mA
	$V_{CC} = 4.5V$, $V_{OL} = 1.0V$	14	–	mA
CKI (Single-pin RC oscillator)	$V_{CC} = 4.5V$, $V_{IH} = 3.5V$	2	–	mA
CKO	$V_{CC} = 4.5V$, $V_{OL} = 0.4V$	0.2	–	mA
Output Source Current				
Standard Configuration,	$V_{CC} = 7.5V$, $V_{OH} = 2.0V$	−100	−900	μA
All Outputs (I_{OH})	$V_{CC} = 5.5V$, $V_{OH} = 2.0V$	−55	−600	μA
	$V_{CC} = 4.5V$, $V_{OH} = 2.0V$	−28	−350	μA
Push-Pull Configuration	$V_{CC} = 7.5V$, $V_{OH} = 3.75V$	−0.85	–	mA
SO and SK Outputs (I_{OH})	$V_{CC} = 5.5V$, $V_{OH} = 2.0V$	−1.1	–	mA
	$V_{CC} = 4.5V$, $V_{OH} = 1.0V$	−1.2	–	mA
LED Configuration, L_0-L_7	$V_{CC} = 7.5V$, $V_{OH} = 2.0V$	−1.4	−27	mA
Outputs, Low Current	$V_{CC} = 6.0V$, $V_{OH} = 2.0V$	−1.4	−17	mA
Driver Option (I_{OH})	$V_{CC} = 5.5V$, $V_{OH} = 2.0V$	−0.7	−15	mA
LED Configuration, L_0-L_7	$V_{CC} = 7.5V$, $V_{OH} = 2.0V$	−2.7	−54	mA
Outputs, High Current	$V_{CC} = 6.0V$, $V_{OH} = 2.0V$	−2.7	−34	mA
Driver Option (I_{OH})	$V_{CC} = 5.5V$, $V_{OH} = 2.0V$	−1.4	−30	mA
TRI-STATE® Configuration,	$V_{CC} = 7.5V$, $V_{OH} = 4.0V$	−0.7	–	mA
L_0-L_7 Outputs, Low	$V_{CC} = 5.5V$, $V_{OH} = 2.7V$	−0.6	–	mA
Current Driver Option (I_{OH})	$V_{CC} = 4.5V$, $V_{OH} = 1.5V$	−0.9	–	mA
TRI-STATE® Configuration,	$V_{CC} = 7.5V$, $V_{OH} = 4.0V$	−1.4	–	mA
L_0-L_7 Outputs, High	$V_{CC} = 5.5V$, $V_{OH} = 2.7V$	−1.2	–	mA
Current Driver Option (I_{OH})	$V_{CC} = 4.5V$, $V_{OH} = 1.5V$	−1.8	–	mA
Input Load Source Current	$V_{CC} = 5.0V$, $V_{IL} = 0V$	−10	−200	μA
CKO Output				
RAM Power Supply Option				
Power Requirement	$V_R = 3.3V$	–	8.0	mA
TRI-STATE® Output Leakage				
Current		−5	+5	μA
Total Sink Current Allowed				
All Outputs Combined		–	120	mA
D, G Ports		–	120	mA
L_7-L_4		–	4	mA
L_3-L_0		–	4	mA
All Other Pins		–	1.5	mA
Total Source Current Allowed				
All I/O Combined		–	120	mA
L_7-L_4		–	60	mA
L_3-L_0		–	60	mA
Each L Pin		–	30	mA
All Other Pins		–	1.5	mA

AC ELECTRICAL CHARACTERISTICS

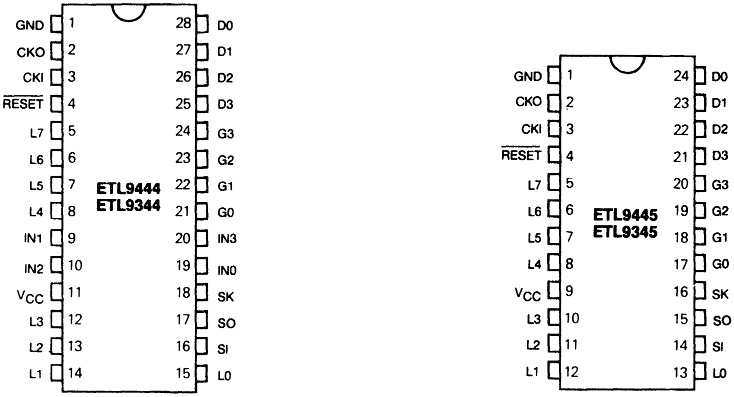
ETL9444/L9445 : 0°C ⩽ T_A ⩽ 70°C, 4.5V ⩽ V_{CC} ⩽ 9.5V (Unless otherwise specified)
ETL9344/L9345 : −40°C ⩽ T_A ⩽ +85°C, 4.5V ⩽ V_{CC} ⩽ 7.5V (Unless otherwise specified).

Parameter	Conditions	Min.	Max.	Units
Instruction Cycle Time — t_C		16	40	μs
CKI				
Input Frequency — f_I	÷32 mode	0 8	2.0	MHz
	÷16 mode	0.4	1.0	MHz
	÷8 mode	0.2	0.5	MHz
	÷4 mode	0.1	0.25	MHz
Duty Cycle		30	60	%
Rise Time	$f_I = 2\,MHz$	–	120	ns
Fall Time		–	80	ns

CKI Using RC (÷4)	R = 56 kΩ ± 5%
	C = 100 pF ± 10%			

Instruction Cycle Time		16	28	μs
CKO as SYNC Input				
t_{SYNC}		400	–	ns
INPUTS:				
IN_3-IN_0, G_3-G_0, L_7-L_0				
t_{SETUP}		–	8.0	μs
t_{HOLD}		–	1.3	μs
SI				
. t_{SETUP}		–	2.0	μs
t_{HOLD}		–	1.0	μs
OUTPUT PROPAGATION DELAY	Test condition: $C_L = 50\,pF$, $R_L = 20\,k\Omega$, $V_{OUT} = 1.5V$			
SO, SK Outputs				
t_{pd1}, t_{pd0}		–	4.0	μs
All Other Outputs				
t_{pd1}, t_{pd0}		–	5.6	μs

FIGURE 2 – CONNECTION DIAGRAMS



Pin	Description
L_7-L_0	8 bidirectional I/O ports with TRI-STATE®
G_3-G_0	4 bidirectional I/O ports
D_3-D_0	4 general purpose outputs
IN_3-IN_0	4 general purpose inputs (COP444L only)
SI	Serial input (or counter input)
SO	Serial output (or general purpose output)

Pin	Description
SK	Logic-controlled clock (or general purpose output)
CKI	System oscillator input
CKO	System oscillator output (or general purpose input, RAM power supply, or SYNC input)
$\overline{RESET}$	System reset input
V_{CC}	Power supply
GND	Ground

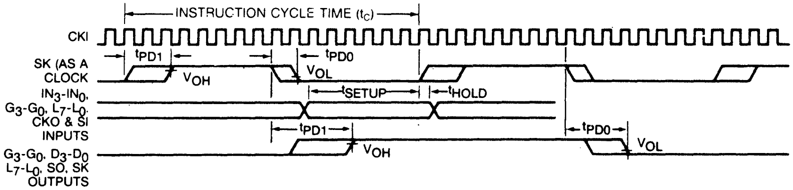

INSTRUCTION CYCLE TIME (tC)
CKI
SK (AS A CLOCK
tPD1
tPD0
VOH
VOL
IN3-IN0, G3-G0, L7-L0, CKO & SI INPUTS
tSETUP
tHOLD
tPD1
tPD0
VOH
VOL
G3-G0, D3-D0 L7-L0, SO, SK OUTPUTS

FIGURE 3 – INPUT/OUTPUT TIMING DIAGRAMS (CRYSTAL DIVIDE-BY-16 MODE)

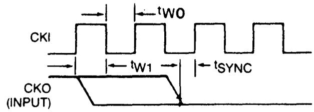


FIGURE 3a – SYNCHRONIZATION TIMING

FUNCTIONAL DESCRIPTION

A block diagram of the ETL9444 is given in Figure 1. Data paths are illustrated in simplified form to depict how the various logic elements communicate with each other in implementing the instruction set of the device. Positive logic is used. When a bit is set, it is a logic "1" (greater than 2 volts). When a bit is reset, it is a logic "0" (less than 0.8 volts).

All functional references to the ETL9444/L9445 also apply to the ETL9344/L9345.

Program Memory

Program Memory consists of a 2048 byte ROM. As can be seen by an examination of theETL9444/L9445 instruction set, these words may be program instructions, program data or ROM addressing data. Because of the special characteristics associated with the JP, JSRP, JID, and LQID instructions, ROM must often be thought of as being organized into 32 pages of 64 words each.

ROM addressing is accomplished by a 11-bit PC register. Its binary value selects one of the 2048 8-bit words contained in ROM. A new address is loaded into the PC register during each instruction cycle. Unless the instruction is a transfer of control instruction, the PC register is loaded with the next sequential 11-bit binary count value. Three levels of subroutine nesting are implemented by the 11-bit subroutine save registers, SA, SB, and SC; providing a last-in, first-out (LIFO) hardware subroutine stack.

ROM instruction words are fetched, decoded and executed by the Instruction Decode, Control and Skip Logic circuitry.

Data Memory

Data memory consists of a 512-bit RAM, organized as 8 data registers of 16 4-bit digits. RAM addressing is implemented by a 7-bit B register whose upper 3 bits (Br) select 1 of 8 data registers and lower 4 bits (Bd) select 1 of 16 4-bit digits in the selected data register. While the 4-bit contents of the selected RAM digit (M) is usually loaded into or from, or exchanged with, the A register (accumulator), it may also be loaded into or from the Q latches or loaded from the L ports. RAM addressing may also be performed directly by the LDD and XAD instructions based upon the 7-bit contents of the operand field of these instructions. The Bd register also serves as a source register for 4-bit data sent directly to the D outputs.

Internal Logic

The 4-bit A register (accumulator) is the source and destination register for most I/O, arithmetic, logic and data memory access operations. It can also be used to load the Br and Bd portions of the B register, to load and input 4 bits of the 8-bit Q latch data, to input 4 bits of the 8-bit L I/O port data and to perform data exchanges with the SIO register.

A,4-bit adder performs the arithmetic and logic functions, storing its results in A. It also outputs a carry bit to the 1-bit C register, most often employed to indicate arithmetic overflow. The C register, in conjunction with the XAS instruction and the EN register, also serves to control the SK output. C can be outputted directly to SK or can enable SK to be a sync clock each instruction cycle time. (See XAS instruction and EN register description, below.)

Four general-purpose inputs, IN_3-IN_0,are provided.

The D register provides 4 general-purpose outputs and is used as the destination register for the 4-bit contents of Bd. The D outputs can be directly connected to the digits of a multiplexed LED display.

The G register contents are outputs to 4 general-purpose bidirectional I/O ports. G I/O ports can be directly connected to the digits of a multiplexed LED display.

The Q register is an internal, latched, 8-bit register, used to hold data loaded to or from M and A, as well as 8-bit data from ROM. Its contents are output to the L I/O ports when the L drivers are enabled under program control. (See LEI instruction.)

The 8 L drivers,when enabled, output the contents of latched Q data to the L I/O ports. Also, the contents of L may be read directly into A and M. L I/O ports can be directly connected to the segments of a multiplexed LED display (using the LED Direct Drive output configuration option) with Q data being outputted to the Sa-Sg and decimal point segments of the display.

The SIO register functions as a 4-bit serial-in/serial-out shift register or as a binary counter depending on the contents of the EN register. (See EN register description, below.) Its contents can be exchanged with A, allowing it to input or output a continuous serial data stream. SIO may also be used to provide additional parallel I/O by connecting SO to external serial-in/parallel-out shift registers.

The XAS instruction copies C into the SKL latch. In the counter mode, SK is the output of SKL; in the shift register mode, SK outputs SKL ANDed with the clock.

The EN register is an internal 4-bit register loaded under program control by the LEI instruction. The state of each bit of this register selects or deselects the particular feature associated with each bit of the EN register (EN_3-EN_0)

1. The least significant bit of the enable register, EN_0,selects the SIO register as either a 4-bit shift register or a 4-bit binary counter. With EN_0set, SIO is an asynchronous binary counter, *decrementing* its value by one upon each low-going pulse ("1" to "0") ocurring on the SI input. Each pulse must be at least two instruction cycles wide. SK outputs the value of SKL. The SO output is equal to the value of EN_3. With EN_0 reset, SIO is a serial shift register shifting left each instruction cycle time. The data present at SI goes into the least significant bit of SIO. SO can be enabled to output the most significant bit of SIO each cycle time. (See 4 below.) The SK output becomes a logic-controlled clock.
2. With EN_1 set the IN_1 input is enabled as an interrupt input. Immediately following an interrupt, EN_1 is reset to disable further interrupts.

3. With EN_2 set, the L drivers are enabled to output the data in Q to the L I/O ports. Resetting EN_2 disables the L drivers, placing the L I/O ports in a high-impedance input state.
4. EN_3, in conjunction with EN_0, affects the SO output. With EN_0 set (binary counter option selected) SO will output the value loaded into EN_3. With EN_0 reset (serial shift register option selected), setting EN_3 enables SO as the output of the SIO shift register, outputting serial shifted data each instruction time. Resetting EN_3 with the serial shift register option selected disables SO as the shift register output; data continues to be shifted through SIO and can be exchanged with A via an XAS instruction but SO remains reset to "0". The table below provides a summary of the modes associated with EN_3 and EN_0.

Enable Register Modes — Bits EN_3 and EN_0

EN_3	EN_0	SIO	SI	SO	SK
0	0	Shift Register	Input to Shift Register	0	If SKL = 1, SK = CLOCK If SKL = 0, SK = 0
1	0	Shift Register	Input to Shift Register	Serial Out	If SKL = 1, SK = CLOCK If SKL = 0, SK = 0
0	1	Binary Counter	Input to Binary Counter	0	If SKL = 1, SK = 1 If SKL = 0, SK = 0
1	1	Binary Counter	Input to Binary Counter	1	If SKL = 1, SK = 1 If SKL = 0, SK = 0

Interrupt

The following features are associated with the IN_1 interrupt procedure and protocol and must be considered by the programmer when utilizing interrupts.

a. The interrupt, once acknowledged as explained below, pushes the next sequential program counter address (PC + 1) onto the stack, pushing in turn the contents of the other subroutine-save registers to the next lower level (PC + 1 → SA → SB → SC). Any previous contents of SC are lost. The program counter is set to hex address 0FF (the last word of page 3) and EN_1 is reset.

b. An interrupt will be acknowledged only after the following conditions are met:
   1. EN_1 has been set.
   2. A low-going pulse ("1" to "0") at least two instruction cycles wide occurs on the IN_1 input.
   3. A currently executing instruction has been completed.
   4. All successive transfer of control instructions and successive LBIs have been completed (e.g., if the main program is executing a JP instruction which transfers program control to another JP instruction, the interrupt will not be acknowledged until the second JP instruction has been executed.

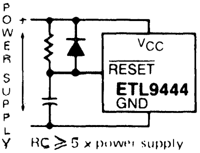
c. Upon acknowledgement of an interrupt, the skip logic status is saved and later restored upon popping of the stack. For example, if an interrupt occurs during the execution of ASC (Add with Carry, Skip on Carry) instruction which results in carry, the skip logic status is saved and program control is transferred to the interrupt servicing routine at hex address 0FF. At the *end* of the interrupt routine, a RET instruction is executed to "pop" the stack and return program control to the instruction following the original ASC. *At this time*, the skip logic is enabled and skips this instruction because of the previous ASC carry. Subroutines and LQID instructions should not be nested within the interrupt service routine, since their popping the stack will enable any previously saved main program skips, interfering with the orderly execution of the interrupt routine.

d. The first instruction of the interrupt routine at hex address 0FF must be a NOP.

e. A LEI instruction can be put immediately before the RET to re-enable interrupts.

Initialization

The Reset Logic will initialize (clear) the device upon power-up if the power supply rise time is less than 1ms and greater than 1µs. If the power supply rise time is greater than 1ms, the user use provide an external RC network and diode to the $\overline{RESET}$ pin as shown below. If the RC network is not used, the $\overline{RESET}$ pin must be pulled up to V_{CC} either by the internal load or by an external resistor (≥40kΩ) to V_{CC}. The $\overline{RESET}$ pin is configured as a Schmitt trigger input. Initialization will occur whenever a logic "0" is applied to the $\overline{RESET}$ input, provided it stays low for at least three instruction cycle times.



RC ≥ 5 x power supply Rise time (R ≥ 40 K)

Power up clear circuit

Upon initialization, the PC register is cleared to 0 (ROM address 0) and the A, B, C, D, EN, and G registers are cleared. The SK output is enabled as a SYNC output, providing a pulse each instruction cycle time. *Data Memory (RAM) is not cleared upon initialization*. The first instruction at address 0 must be a CLRA.

Oscillator

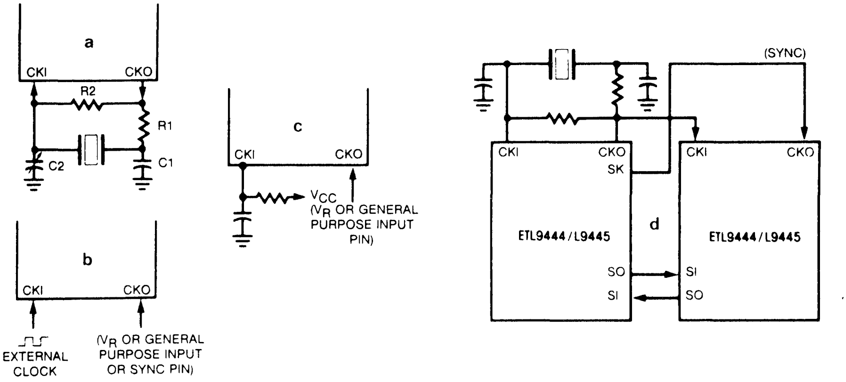
There are four basic clock oscillator configurations available as shown by Figure 4.

a. Crystal Controlled Oscillator. CKI and CKO are connected to an external crystal. The instruction cycle time equals the crystal frequency divided by 32 (optional by 16 or 8).

b. External Oscillator. CKI is an external clock input signal. The external frequency is divided by 32 (optional by 16 or 8) to give the instruction cycle time. CKO is now available to be used as the RAM power supply (V_R), as a general purpose input, or as a SYNC input.

c. RC Controlled Oscillator. CKI is configured as a single pin RC controlled Schmitt trigger oscillator. The instruction cycle equals the oscillation frequency divided by 4. CKO is available as the RAM power supply (V_R) or as a general purpose input.

d. Externally Synchronized Oscillator. Intended for use in multi-COP systems, CKO is programmed to function as an input connected to the SK output of another COP chip operating at the same frequency (COP chip with L or C suffix) with CKI connected as shown. In this configuration, the SK output connected to CKO must provide a SYNC (instruction cycle) signal to CKO, thereby allowing synchronous data transfer between the COPs using only the SI and SO serial I/O pins in conjunction with the XAS instruction. Note that on power-up SK is automatically enabled as a SYNC output. (See Functional Description, Initialization, above.)



Crystal Oscillator

Crystal Value	Component Values			
	R1 (Ω)	R2 (Ω)	C1 (pF)	C2 (pF)
455 kHz	4 7k	1M	220	220
2.097 MHz	1k	1M	30	6-36

RC Controlled Oscillator

R (kΩ)	C (pF)	Instruction Cycle Time (μs)
51	100	19 ± 15%
82	56	19 ± 13%

Note: 200 kΩ ≥ R ≥ 25 kΩ
360 pF ≥ C ≥ 50 pF

FIGURE 4 – ETL9444/L9445 OSCILLATOR

CKO Pin Options

In a crystal controlled oscillator system, CKO is used as an output to the crystal network. As an option CKO can be a SYNC input as described above. As another option CKO can be a general purpose input, read into bit 2 of A (accumulator) upon execution of an INIL instruction. As another option, CKO can be a RAM power supply pin (V_R), allowing its connection to a standby/backup power supply to maintain the integrity of RAM data with minimum power drain when the main supply is inoperative or shut down to conserve power. Using either option is appropriate in applications where the ETL9444/L9445 system timing configuration does not require use of the CKO pin.

I/O Options

ETL9444/L9445 outputs have the following optional configurations, illustrated in Figure 5 :

a. **Standard** — an enhancement mode device to ground in conjunction with a depletion-mode device to V_{CC}, compatible with LSTTL and CMOS input requirements. Available on SO, SK, and all D and G outputs.

b. **Open-Drain** — an enhancement-mode device to ground only, allowing external pull-up as required by the user's application. Available on SO, SK, and all D and G outputs.

c. **Push-Pull** — An enhancement-mode device to ground in conjunction with a depletion-mode device paralleled by an enhancement-mode device to V_{CC}. This configuration has been provided to allow for fast rise and fall times when driving capacitive loads. Available on SO and SK outputs only.

d. **Standard L** — same as **a.**, but may be disabled. Available on L outputs only.

e. **Open Drain L** — same as **b.**, but may be disabled. Available on L outputs only.

f. **LED Direct Drive** — an enhancement-mode device to ground and to V_{CC}, meeting the typical current sourcing requirements of the segments of an LED display. The sourcing device is clamped to limit current flow. These devices may be turned off under program control (See Functional Description, EN Register), placing the outputs in a high-impedance state to provide required LED segment blanking for a multiplexed display. Available on L outputs only.

g. **TRI-STATE® Push-Pull** — an enhancement-mode device to ground and V_{CC}. These outputs are TRI-STATE outputs, allowing for connection of these outputs to a data bus shared by other bus drivers. Available on L outputs only.

ETL9444/L9445 inputs have the following optional configurations :

h. An on-chip depletion load device to V_{CC}.

i. A Hi-Z input which must be driven to a "1" or "0" by external components.

The above input and output configurations share common enhancement-mode and depletion-mode devices. Specifically, all configurations use one or more of six devices (numbered 1-6, respectively). Minimum and maximum current (I_{OUT} and V_{OUT} curves are given in Figure 6 for each of these devices to allow the designer to effectively use these I/O configurations in designing a system.

The SO, SK outputs can be configured as shown in **a.**, **b.**, or **c.** The D and G outputs can be configured as shown in **a.** or **b.** Note that when inputting data to the G ports, the G outputs should be set to "1." The L outputs can be configured as in **d.**, **e.**, **f.** or **g.**

An important point to remember if using configuration **d.** or **f.** with the L drivers is that even when the L drivers are disabled, the depletion load device will source a small amount of current (see Figure 6, device 2); however, when the L-lines are used as inputs, the disabled depletion device can *not* be relied on to source sufficient current to pull an input to logic "1".

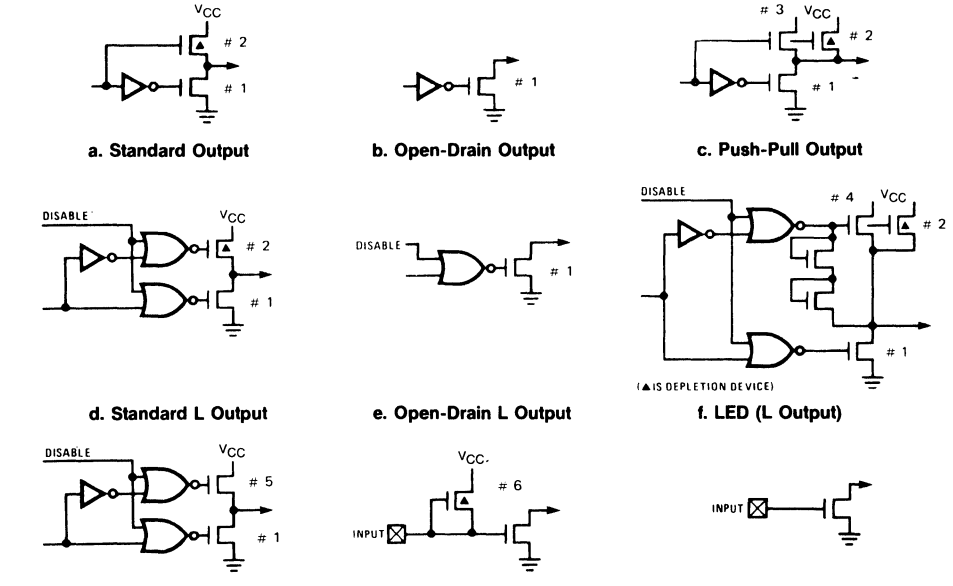
RAM Keep-Alive Option

Selecting CKO as the RAM power supply (V_R) allows the user to shut off the chip power supply (V_{CC}) and maintain data in the RAM.
To insure that RAM data integrity is maintained, the following conditions *must* be met:

1. $\overline{\text{RESET}}$ must go low before V_{CC} goes low during power off; V_{CC} must go high before $\overline{\text{RESET}}$ goes high on power-up.
2. V_R must be within the operating range of the chip, and equal to V_{CC} ± 1 V during normal operation.
3. V_R must be $\geqslant$ 3.3V with V_{CC} off.

ETL9445

If the ETL9444 Is bonded as a 24-pin device, it becomes the ETL9445, illustrated in Figure 2, ETL9444 Connection Diagrams. Note that the ETL9445 does not contain the four general purpose IN inputs (IN_3-IN_0). Use of this option precludes, of course, use of the IN options and the interrupt feature, which uses IN_1. All other options are available for the ETL9445.



a. Standard Output

b. Open-Drain Output

c. Push-Pull Output

d. Standard L Output

e. Open-Drain L Output

f. LED (L Output)

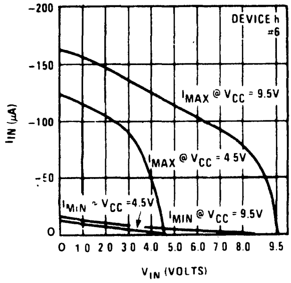
g. TRI-STATE®Push-Pull (L Output)

h. Input with Load

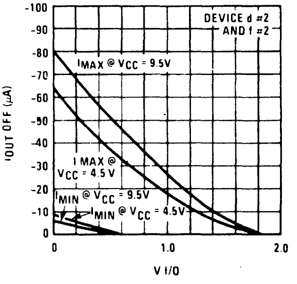
i. Hi-Z Input

FIGURE 5 – OUTPUT CONFIGURATIONS

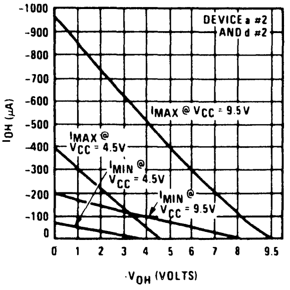
Current for Inputs with Load Device



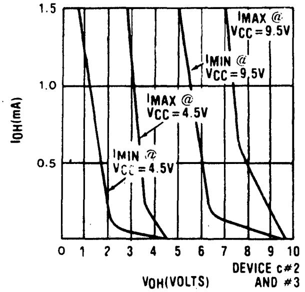
Input Current for L_0 through L_7 when Output Programmed Off by Software



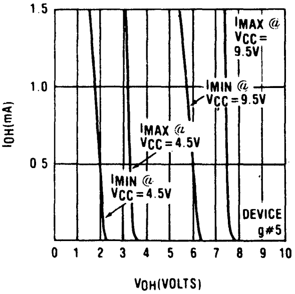
Source Current for Standard Output Configuration



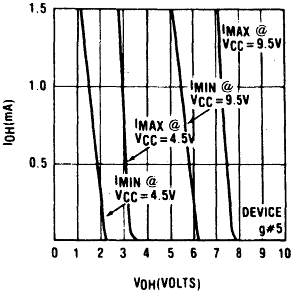
Source Current for SO and SK in Push-Pull Configuration



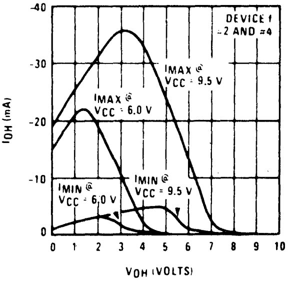
Source Current for L_0 through L_7 in TRI-STATE®Configuration (High Current Option)



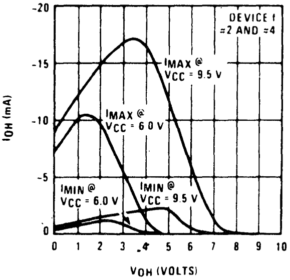
Source Current for L_0 through L_7 in TRI-STATE®Configuration (Low Current Option)



LED Output Source Current (for High Current LED Option)

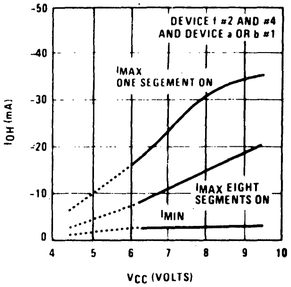


LED Output Source Current (for Low Current LED Option)

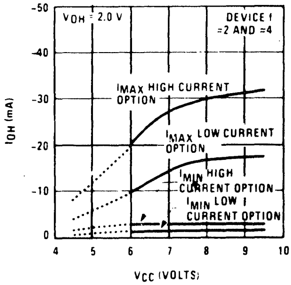


LED Output Direct Segment Drive

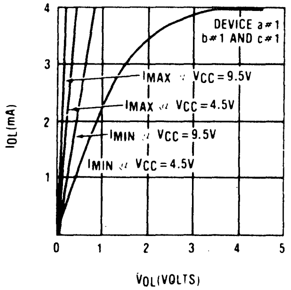
High Current Options on L_0-L_7 Very High Current Options on D_0-D_3 or G_0-G_3.



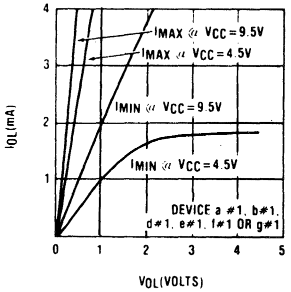
LED Output Direct Segment Drive



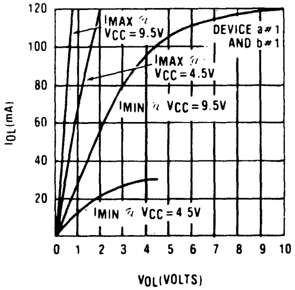
Output Sink Current for SO and SK



Output Sink Current for L_0-L_7 and Standard Drive Option for D_0-D_3 and G_0-G_3



Output Sink Current G_0-G_3 and D_0-D_3 with Very High Current Option



Output Sink Current for G_0-G_3 and D_0-D_3 (for High Current Option)

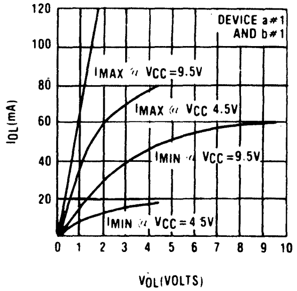
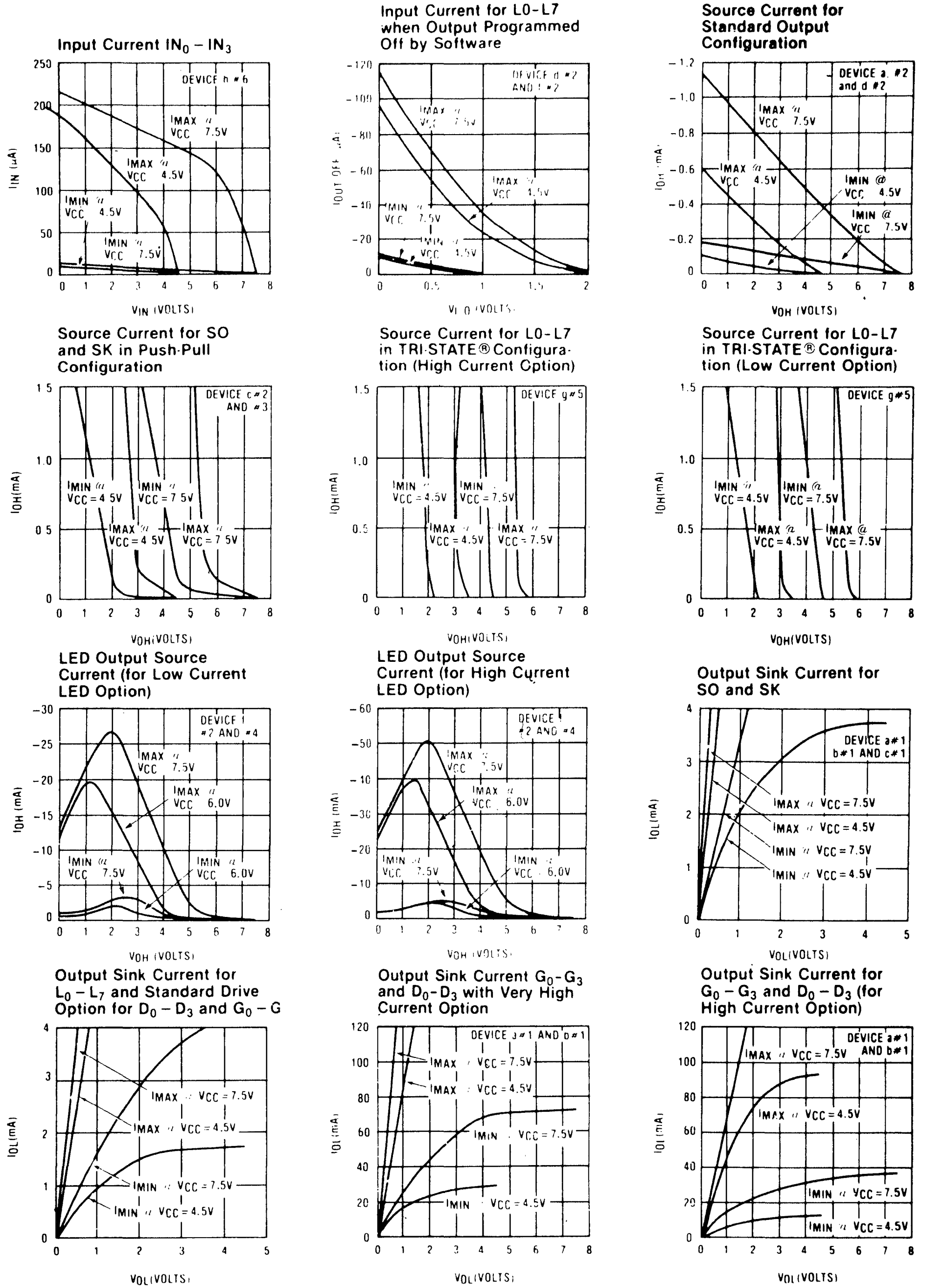


FIGURE 6a – ETL9444/L9445 INPUT/OUTPUT CHARACTERISTICS



Input Current $IN_0 - IN_3$

Input Current for L0-L7 when Output Programmed Off by Software

Source Current for Standard Output Configuration

Source Current for SO and SK in Push-Pull Configuration

Source Current for L0-L7 in TRI-STATE® Configuration (High Current Option)

Source Current for L0-L7 in TRI-STATE® Configuration (Low Current Option)

LED Output Source Current (for Low Current LED Option)

LED Output Source Current (for High Current LED Option)

Output Sink Current for SO and SK

Output Sink Current for $L_0 - L_7$ and Standard Drive Option for $D_0 - D_3$ and $G_0 - G$

Output Sink Current G_0-G_3 and D_0-D_3 with Very High Current Option

Output Sink Current for $G_0 - G_3$ and $D_0 - D_3$ (for High Current Option)

FIGURE 6b – ETL9344/L9345 INPUT/OUTPUT CHARACTERISTICS

ETL9444/L9445,ETL9344/L9345 Instruction Set

Table 1 is a symbol table providing internal architecture, instruction operand and operational symbols used in the instruction set table.

Table 2 provides the mnemonic, operand, machine code, data flow, skip conditions, and description associated with each instruction in the ETL9444/L9445 instruction set.

TABLE 1 – ETL9444/L9445 ETL9344/L9345 INSTRUCTION SET TABLE SYMBOLS

Symbol	Definition
INTERNAL ARCHITECTURE SYMBOLS	
A	4-bit Accumulator
B	7-bit RAM Address Register
Br	Upper 3 bits of B (register address)
Bd	Lower 4 bits of B (digit address)
C	1-bit Carry Register
D	4-bit Data Output Port
EN	4-bit Enable Register
G	4-bit Register to latch data for G I/O Port
IL	Two 1-bit latches associated with the IN_3 or IN_0 inputs
IN	4-bit Input Port
L	8-bit TRI-STATE® I/O Port
M	4-bit contents of RAM Memory pointed to by B Register
PC	11-bit ROM Address Register (program counter)
Q	8-bit Register to latch data for L I/O Port
SA	11-bit Subroutine Save Register A
SB	11-bit Subroutine Save Register B
SC	11-bit Subroutine Save Register C
SIO	4-bit Shift Register and Counter
SK	Logic-Controlled Clock Output

Symbol	Definition
INSTRUCTION OPERAND SYMBOLS	
d	4-bit Operand Field, 0-15 binary (RAM Digit Select)
r	3-bit Operand Field, 0-7 binary (RAM Register Select)
a	11-bit Operand Field, 0-2047 binary (ROM Address)
y	4-bit Operand Field, 0-15 binary (Immediate Data)
RAM(s)	Contents of RAM location addressed by s
ROM(t)	Contents of ROM location addressed by t
OPERATIONAL SYMBOLS	
+	Plus
—	Minus
→	Replaces
←→	Is exchanged with
=	Is equal to
$\overline{A}$	The one's complement of A
⊕	Exclusive-OR
:	Range of values

TABLE 2 – ETL9444/L9445 INSTRUCTION SET

Mnemonic	Operand	Hex Code	Machine Language Code (Binary)	Data Flow	Skip Conditions	Description
ARITHMETIC INSTRUCTIONS						
ASC		30	0011 0000	A + C + RAM(B) → A Carry → C	Carry	Add with Carry. Skip on Carry
ADD		31	0011 0001	A + RAM(B) → A	None	Add RAM to A
ADT		4A	0100 1010	A + 10_{10} → A	None	Add Ten to A
AISC	y	5-	0101 y	A + y → A	Carry	Add Immediate. Skip on Carry ($y \neq 0$)
CASC		10	0001 0000	$\overline{A}$ + RAM(B) + C → A Carry → C	Carry	Complement and Add with Carry. Skip on Carry
CLRA		00	0000 0000	0 → A	None	Clear A
COMP		40	0100 0000	$\overline{A}$ → A	None	Ones complement of A to A
NOP		44	0100 0100	None	None	No Operation
RC		32	0011 0010	"0" → C	None	Reset C
SC		22	0010 0010	"1" → C	None	Set C
XOR		02	0000 0010	A ⊕ RAM(B) → A	None	Exclusive-OR RAM with A
TRANSFER OF CONTROL INSTRUCTIONS						
JID		FF	1111 1111	ROM ($PC_{10:8}$, A,M) → $PC_{7:0}$	None	Jump Indirect (Note 3)
JMP	a	6- --	0110 0 $a_{10:8}$ $a_{7:0}$	a → PC	None	Jump
JP	a	-- --	1 $a_{6:0}$ (pages 2,3 only) or 11 $a_{5:0}$ all other pages)	a → $PC_{6:0}$ a → $PC_{5:0}$	None	Jump within Page (Note 4)
JSRP	a	--	10 $a_{5:0}$	PC + 1 → SA → SB → SC 00010 → $PC_{10:6}$ a → $PC_{5:0}$	None	Jump to Subroutine Page (Note 5)
JSR	a	6- --	0110 1 $a_{10:8}$ $a_{7:0}$	PC + 1 → SA → SB → SC a → PC	None	Jump to Subroutine
RET		48	0100 1000	SC → SB → SA → PC	None	Return from Subroutine
RETSK		49	0100 1001	SC → SB → SA → PC	Always Skip on Return	Return from Subroutine then Skip

TABLE 2 – ETL9444/L9445 INSTRUCTION SET (continued)

Mnemonic	Operand	Hex Code	Machine Language Code (Binary)	Data Flow	Skip Conditions	Description

MEMORY REFERENCE INSTRUCTIONS

Mnemonic	Operand	Hex Code	Machine Language Code (Binary)	Data Flow	Skip Conditions	Description
CAMQ		33 3C	0011 001.1 0011 1100	A → $Q_{7\,4}$ RAM(B) → $Q_{3\,0}$	None	Copy A, RAM to Q
CQMA		33 2C	0011 0011 0010 1100	$Q_{7\,4}$ → RAM(B) $Q_{3\,0}$ → A	None	Copy Q to RAM, A
LD	r	-5	00 r 0101 (r = 0:3)	RAM(B) → A Br ⊕ r → Br	None	Load RAM into A. Exclusive-OR Br with r
LDD	r,d	23 --	0010 0011 0 r d	RAM(r,d) → A	None	Load A with RAM pointed to directly by r,d
LQID		BF	1011 1111	ROM($PC_{10\cdot 8}$,A,M) → Q SB → SC	•None	Load Q Indirect (Note 3)
RMB	0 1 2 3	4C 45 42 43	0100 1100 0100 0101 0100 0010 0100 0011	0 → $RAM(B)_0$ 0 → $RAM(B)_1$ 0 → $RAM(B)_2$ 0 → $RAM(B)_3$	None	Reset RAM Bit
SMB	0 1 2 3	4D 47 46 4B	0100 1101 0100 1101 0100 0110 0100 1011	1 → $RAM(B)_0$ 1 → $RAM(B)_1$ 1 → $RAM(B)_2$ 1 → $RAM(B)_3$	None	Set RAM Bit
STII	y	7-	0111 y	y → RAM(B) Bd + 1 → Bd	None	Store Memory Immediate and Increment Bd
X	r	-6	00 r 0110 (r = 0:3)	RAM(B) ↔ A Br ⊕ r → Br	None	Exchange RAM with A, Exclusive-OR Br with r
XAD	r,d	23 --	0010 0011 1 r d	RAM(r,d) ↔ A	None	Exchange A with RAM pointed to directly by r,d
XDS	r	-7	00 r 0111 (r = 0:3)	RAM(B) ↔ A Bd – 1 → Bd Br ⊕ r → Br	Bd decrements past 0	Exchange RAM with A and Decrement Bd, Exclusive-OR Br with r
XIS	r	-4	00 r 0100 (r = 0:3)	RAM(B) ↔ A Bd + 1 → Bd Br ⊕ r → Br	Bd increments past 15	Exchange RAM with A and Increment Bd, Exclusive-OR Br with r

REGISTER REFERENCE INSTRUCTIONS

Mnemonic	Operand	Hex Code	Machine Language Code (Binary)	Data Flow	Skip Conditions	Description
CAB		50	0101 0000	A → Bd	None	Copy A to Bd
CBA		4E	0100 1110	Bd → A	None	Copy Bd to A
LBI	r,d	-- 33 --	00 r (d – 1) (r = 0:3: d = 0, 9:15) or 0011 0011 1 r d (any r, any d)	r,d → B	Skip until not a LBI	Load B Immediate with r,d (Note 6)
LEI	y	33 6-	0011 0001 0110 y	y → EN	None	Load EN Immediate (Note 7)
XABR		12	0001 0010	A ↔ Br (0 → A_3)	None	Exchange A with Br

TABLE 2 – ETL9444/L9445 INSTRUCTION SET (continued)

Mnemonic	Operand	Hex Code	Machine Language Code (Binary)	Data Flow	Skip Conditions	Description
TEST INSTRUCTIONS						
SKC		20	0010 0000		C = "1"	Skip if C is True
SKE		21	0010 0001		A = RAM(B)	Skip if A Equals RAM
SKGZ		33 21	0011 0011 0010 0001		$G_{3:0} = 0$	Skip if G is Zero (all 4 bits)
SKGBZ		33	0011 0011	1st byte		Skip if G Bit is Zero
	0	01	0000 0001	2nd byte	$G_0 = 0$	
	1	11	0001 0001		$G_1 = 0$	
	2	03	0000 0011		$G_2 = 0$	
	3	13	0001 0011		$G_3 = 0$	
SKMBZ	0	01	0000 0001		$RAM(B)_0 = 0$	Skip if RAM Bit is Zero
	1	11	0001 0001		$RAM(B)_1 = 0$	
	2	03	0000 0011		$RAM(B)_2 = 0$	
	3	13	0001 0011		$RAM(B)_3 = 0$	
SKT		41	0100 0001		A time-base counter carry has occurred since last test	Skip on Timer (Note 3)
INPUT/OUTPUT INSTRUCTIONS						
ING		33 2A	0011 0011 0010 1010	G → A	None	Input G Ports to A
ININ		33 28	0011 0011 0010 1000	IN → A	None	Input IN Inputs to A (Note 2)
INIL		33 29	0011 0011 0010 1001	IL_3, CKO, "0", IL_0 → A	None	Input IL Latches to A (Note 3)
INL		33 2E	0011 0011 0010 1110	$L_{7:4}$ → RAM(B) $L_{3:0}$ → A	None	Input L Ports to RAM, A
OBD		33 3E	0011 0011 0011 1110	Bd → D	None	Output Bd to D Outputs
OGI	y	33 5-	0011 0011 0101 y	y → G	None	Output to G Ports Immediate
OMG		33 3A	0011 0011 0011 1010	RAM(B) → G	None	Output RAM to G Ports
XAS		4F	0100 1111	A ⟷ SIO, C → SKL	None	Exchange A with SIO (Note 3)

Note 1 : All subscripts for alphabetical symbols indicate bit numbers unless explicitly defined (e.g., Br and Bd are explicitly defined). Bits are numbered 0 to N where 0 signifies the least significant bit (low-order, right-most bit). For example, A_3 indicates the most significant (left-most) bit of the 4-bit A register.

Note 2 : The ININ instruction is not available on the 24-pin ETL9445 or ETL9345 since these devices do not contain the IN inputs.

Note 3 : For additional information on the operation of the XAS, JID, LQUID, INIL, and SKT instructions, see below.

Note 4 : The JP instruction allows a jump, while in subroutine pages 2 or 3, to any ROM location within the two-page boundary of pages 2 or 3. The JP instruction, otherwise, permits a jump to a ROM location within the current 64-word page. JP may not jump to the last word of a page.

Note 5 : A JSRP transfers program control to subroutine page 2 (0010 is loaded into the upper 4 bits of P). A JSRP may not be used when in pages 2 or 3. JSRP may not jump to the last word in page 2.

Note 6 : LBI is a single-byte instruction if d = 0, 9, 10, 11, 12, 13, 14 or 15. The machine code for the lower 4 bits equals the binary value of the "d" data *minus 1*, e.g., to load the lower four bits of B (Bd) with the value 9 (1001_2), the lower 4 bits of the LBI instruction equal 8(1000_2). To load 0, the lower 4 bits of the LBI instruction should equal 15 (1111_2).

Note 7 : Machine code for operand field y for LEI instruction should equal the binary value to be latched into EN, where a "1" or "0" in each bit of EN corresponds with the selection or deselection of a particular function associated with each bit. (See Functional Description, EN Register).

The following information is provided to assist the user in understanding the operation of several unique instructions and to provide notes useful to programmers in writingETL9444/L9445programs.

XAS Instruction

XAS (Exchange A with SIO) exchanges the 4-bit contents of the accumulator with the 4-bit contents of the SIO register. The contents of SIO will contain serial-in/serial-out shift register or binary counter data, depending on the value of the EN register. An XAS instruction will also affect the SK output. (See Functional Description, EN Register, above.) If SIO is selected as a shift register, an XAS instruction must be performed once every 4 instruction cycles to effect a continuous data stream.

JID Instruction

JID (Jump Indirect) is an indirect addressing instruction, transferring program control to a new ROM location pointed to indirectly by A and M. It loads the lower 8 bits of the ROM address register PC with the *contents* of ROM addressed by the 11-bit word, $PC_{10:8}$ A, M. PC_{10} ,PC_9 and PC_8 are not affected by this instruction.

Note that JID requires 2 instruction cycles to execute.

INIL Instruction

INIL (Input IL Latches to A) inputs 2 latches, IL_3 and IL_0 (see Figure 7) and CKO into A. The IL_3 and IL_0 latches are set if a low-going pulse ("1" to "0") has occurred on the IN_3 and IN_0 inputs since the last INIL instruction, provided the input pulse stays low for at least two instruction times. Execution of an INIL inputs IL_3 and IL_0 into A3 and A0 respectively, and resets these latches to allow them to respond to subsequent low-going pulses on the IN_3 and IN_0 lines. If CKO is mask programmed as a general purpose input, an INIL will input the state of CKO into A2. If CKO has not been so programmed, a "1" will be placed in A2. A "0" is always placed in A1 upon the execution of an INIL. The general purpose inputs IN_3-IN_0 are input to A upon execution of an ININ instruction. (See Table 2, ININ instruction.) INIL is useful in recognizing pulses of short duration or pulses which occur too often to be read conveniently by an ININ instruction.

Note : IL latches are not cleared on reset : IL_3 and IL_0 *not* input on ETL9444/L9445.

LQID Instruction

LQID (Load Q Indirect) loads the 8-bit Q register with the contents of ROM pointed to by the 11-bit word PC_{10} ,PC_9, PC_8, A, M. LQID can be used for table lookup or code conversion such as BCD to seven-segment. The LQID instruction "pushes" the stack (PC + 1 → SA → SB →SC) and replaces the least significant 8 bits of PC as follows: A — $PC_{7:4}$, RAM(B) → $PC_{3:0}$, leaving PC_{10}, PC_9 and PC_8 unchanged. The ROM data pointed to by the new address is fetched and loaded into the Q latches. Next, the stack is "popped" (SC → SB → SA →PC), restoring the saved value of PC to continue sequential program execution. Since LQID pushes SB → SC, the previous contents of SC are lost. Also, when LQID pops the stack, the previously pushed contents of SB are left in SC. The net result is that the contents of SB are placed in SC (SB → SC). Note that LQID takes two instruction cycle times to execute.

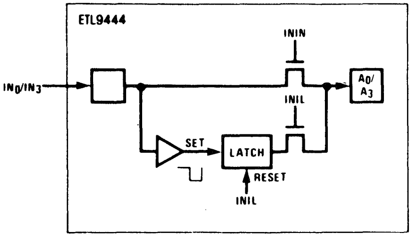


FIGURE 7 – INIL HARDWARE IMPLEMENTATION

SKT Instruction

The SKT (Skip On Timer) instruction tests the state of an internal 10-bit time-base counter. This counter divides the instruction cycle clock frequency by 1024 and provides a latched indication of counter overflow. The SKT instruction tests this latch, executing the next program instruction if the latch is not set. If the latch has been set since the previous test, the next program instruction is skipped and the latch is reset. The features associated with this instruction, therefore, allow the ETL9344/L9345 to generate its own time-base for real-time processing rather than relying on an external input signal.

For example, using a 2.097 MHz crystal as the time-base to the clock generator, the instruction cycle clock frequency will be 65kHz (crystal frequency ÷ 32) and the binary counter output pulse frequency will be 64Hz. For time-of-day or similar real-time processing, the SKT instruction can call a routine which increments a "seconds" counter every 64 ticks.

Instruction Set Notes

a. The first word of a ETL9444/L9445program (ROM address 0) must be a CLRA (Clear A) instruction.

b. Although skipped instructions are not executed, one instruction cycle time is devoted to skipping each byte of the skipped instruction. Thus all program paths except JID and LQID take the same number of cycle times whether instructions are skipped or executed. JID and LQID instructions take 2 cycles if executed and 1 cycle if skipped.

c. The ROM is organized into 32 pages of 64 words each. The Program Counter is an 11-bit binary counter, and will count through page boundaries. If a JP, JSRP, JID or LQID instruction is located in the last word of a page, the instruction operates as if it were in the next page. For example: a JP located in the last work of a page will jump to a location in the next page. Also, a LQID or JID located in the last word of page 3, 7, 11, 15, 19, 23, or 27 will access data in the next group of four pages.

OPTION LIST

The ETL9444/L9445 mask-programmable options are assigned numbers which correspond with the ETL9444 pins.

The following is a list of ETL9444 options. When specifying ETL9445 chip, Options 9, 10, 19, and 20 must all be set to zero. The options are programmed at the same time as the ROM pattern to provide the user with the hardware flexibility to interface to various I/O components using little or no external circuitry.

Option 1 = 0: Ground Pin — no options available

Option 2: CKO Output
= 0: clock generator output to crystal/resonator
(0 not allowable value if option 3 = 3)
= 1: pin is RAM power supply (V_R) input
= 2: general purpose input. load device to V_{CC}
= 3: general purpose input, Hi-Z
= 4: multi-COP SYNC input (CKI ÷ 32, CKI ÷ 16)
= 5: multi-COP SYNC input (CKI ÷ 8)

Option 3: CKI Input
= 0: oscillator input divided by 32 (2 MHz max.)
= 1: oscillator input divided by 16 (1 MHz max.)
= 2: oscillator input divided by 8 (500 kHz max.)
= 3: single-pin RC controlled oscillator divided by 4
= 4: oscillator input divided by 4 (Schmitt)

Option 4: $\overline{\text{RESET}}$ Input
= 0: load device to V_{CC}
= 1: Hi-Z input

Option 5: L_7 Driver
= 0: Standard output
= 1: Open-drain output
= 2: High current LED direct segment drive output
= 3: High current TRI-STATE® push-pull output
= 4: Low-current LED direct segment drive output
= 5: Low-current TRI-STATE® push-pull output

Option 6: L_6 Driver
same as Option 5

Option 7: L_5 Driver
same as Option 5

Option 8: L_4 Driver
same as Option 5

Option 9: IN_1 Input
= 0: load device to V_{CC}
= 1: Hi-Z input

Option 10: IN_2 Input
same as Option 9

Option 11: V_{CC} pin
= 0: 4.5V to 6.3V operation
= 1: 4.5V to 9.5V operation

Option 12: L_3 Driver
same as Option 5

Option 14: L_2 Driver
same as Option 5

Option 14: L_1 Driver
same as Option 5

Option 15: L_0 Driver
same as Option 5

Option 16: SI Input
same as Option 9

Option 17: SO Driver
= 0: standard output
= 1: open-drain output
= 2: push-pull output

Option 18: SK Driver
same as Option 17

Option 19: IN_0 Input
same as Option 9

Option 20: IN_3 Input
same as Option 9

Option 21: G_0 I/O Port
= 0: very-high current standard output
= 1: very-high current open-drain output
= 2: high current standard output
= 3: high current open-drain output
= 4: standard LSTTL output (fanout = 1)
= 5: open-drain LSTTL output (fanout = 1)

Option 22: G_1 I/O Port
same as Option 21

Option 23: G_2 I/O Port
same as Option 21

Option 24: G_3 I/O Port
same as Option 21

Option 25: D_3 Output
same as Option 21

Option 26: D_2 Output
same as Option 21

Option 27: D_1 Output
same as Option 21

Option 28: D_0 Output
same as Option 21

Option 29: L Input Levels
= 0: standard TTL input levels
("0" = 0.8V, "1" = 2.0V)
= 1: higher voltage input levels
("0" = 1.2V, "1" = 3.6V)

Option 30: IN Input Levels
same as Option 29

Option 31: G Input Levels
same as Option 29

Option 32: SI Input Levels
same as Option 29

Option 33: RESET Input
= 0: Schmitt trigger input
= 1: standard TTL input levels
= 2: higher voltage input levels

Option 34: CKO Input Levels (CKO = input. Option 2 = 2,3)
same as Option 29

Option 35 COP Bonding
= 0: ETL9444 (28-pin device)
= 1: ETL9445 (24-pin device)
= 2: both 28 and 24-pin versions

TEST MODE (Non-Standard Operation)

The SO output has been configured to provide for standard test procedures for the custom-programmed ETL9444. With SO forced to logic "1", two test modes are provided, depending upon the value of SI :

a. RAM and Internal Logic Test Mode (SI = 1)

b. ROM Test Mode (SI = 0)

These special test modes should not be employed by the user; they are intended for manufacturing test only.

APPLICATION EXAMPLE :
ETL9444 General Controller

Figure 8 shows and interconnect diagram for a ETL9444 used as a general controller. Operation of the system is as follows :

1. The L_7-L_0 outputs are configured as LED Direct Drive outputs, allowing direct connection to the segments of the display.
2. The D_3-D_0 outputs drive the digits of the multiplexed display directly and scan the columns of the 4 x 4 keyboard matrix.
3. The IN_3-IN_0 inputs are used to input the 4 rows of the keyboard matrix. Reading the IN lines in conjunction with the current value of the D outputs allows detection, debouncing, and decoding of any one of the 16 keyswitches.
4. CKI is configured as a single-pin oscillator input allowing system timing to be controlled by a single-pin RC network. CKO is therefore available for use as a general-purpose input.
5. SI is selected as the input to a binary counter input. With SIO used as a binary counter, SO and SK can be used as general purpose outputs.
6. The 4 bidirectional G I/O ports (G_3-G_0) are available for use as required by the user's application.
7. Normal reset operation is selected.

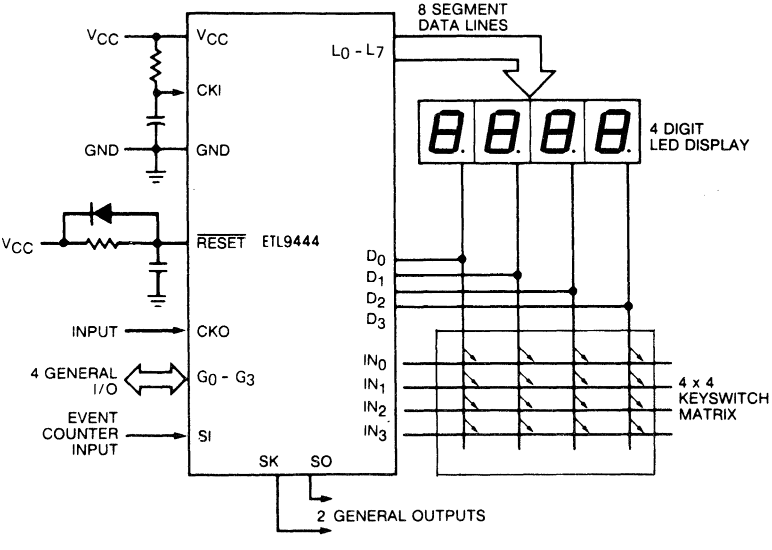
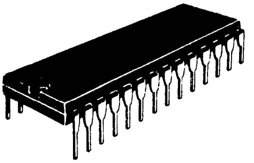


FIGURE 8 – ETL9444 KEYBOARD/DISPLAY INTERFACE

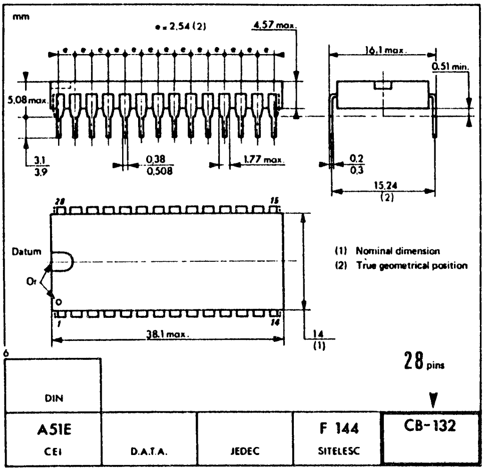
COPS, MICROWIRE and TRI-STATE are registered trademarks of National Semiconductor Corp.

PHYSICAL DIMENSIONS

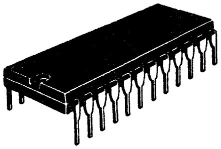
CB-132



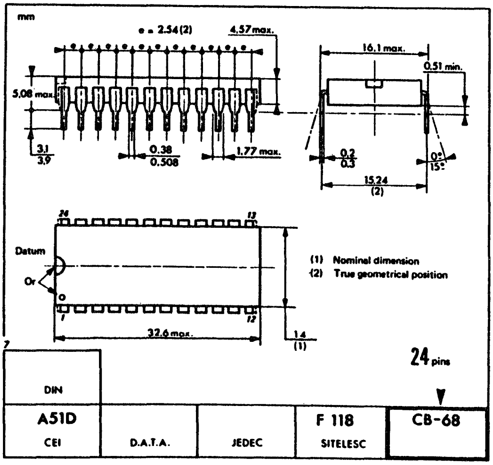
N SUFFIX
PLASTIC PACKAGE



CB-68



N SUFFIX
PLASTIC PACKAGE



These specifications are subject to change without notice.
Please inquire with our sales offices about the availability of the different products.

Ref.DSETL93944445

Printed in France

THOMSON
COMPONENTS
SEMICONDUCTEURS

TS94104

SINGLE CHIP MICROCONTROLLER

ADVANCE INFORMATION

NMOS

GENERAL DESCRIPTION

The TS94104 ROMless N-channel microcontroller is a member of the TS94100 family, fabricated using N-channel, silicon gate XMOS technology.
The TS94104 contains the internal logic of the TS94100 microcontroller family except ROM.

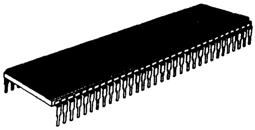
This internal logic is identical to the TS94144 except that the ROM is removed and extra pins are added to output the ROM address and to input the ROM data. The TS94104 can be configured, by means of external pins, to function as a TS94120 or TS94144. Pins have been added to allow the user to select the various functional options that are available on TS94100 family. The TS94104 is primarily entended for a program development and debug for the TS94120 and TS94144 devices prior to masking the final part.
The TS94104 is also appropriate in low volume applications or when the program might be evolutive.

FEATURES

- Accurate emulation of the TS94100 family.
- Powerful instruction set (52 instructions).
- Addresses 2K x 8 external ROM.
- 128 x 4 RAM.
- Interrupt sources: external, internal counter or dual-time peripheral.
- 3-level subroutine stack.
- 4 μs instruction cycle.
- Single supply operation (4.5 V - 5.5 V).
- Low-current drain (15 mA max.).
- Programmable read/write 8 bit internal counter.
- Internal shift register with serial I/O.
- Watch-dog feature.
- Dual timer peripheral for complex waveforms generation and measurements.
- General purpose and tri-state outputs.
- TTL compatible.
- Direct drive of LED digit and segment line.
- Direct drive of TRIAC.
- Extended temperature (— 40°C to + 85°C) available.
- Software compatible with the other member of ET9400, TS94100 families.

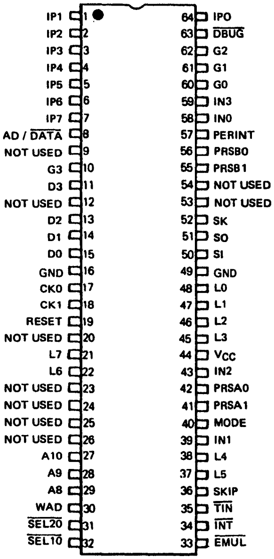
CASE

CB-506



PLASTIC PACKAGE

PIN ASSIGNMENT



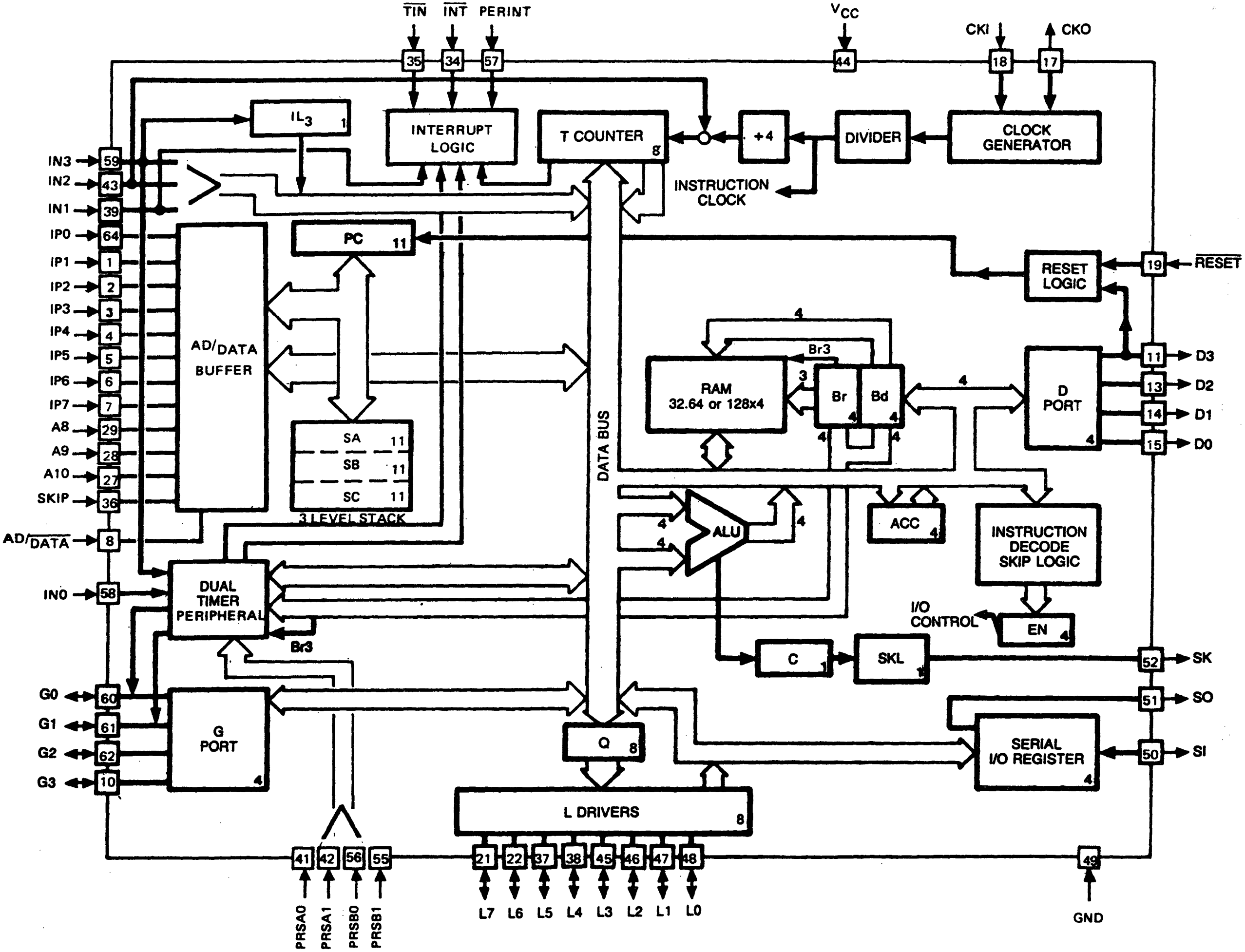


FIGURE 1 : BLOCK DIAGRAM

FUNCTIONAL DESCRIPTION

The internal architecture is shown in Figure 1. Data paths are illustred in simplified form to depict how the various logic elements communicate with each other in implementing the instruction set of the device. Positive logic is used. When a bit is set, it is a logic "1" when it is reset, it is a logic "0".

PROGRAM MEMORY

Program memory consists of a 2K byte external ROM memory (typically an EPROM).

These bytes of ROM may be program instructions, constants or ROM addressing data.

ROM addressing is accomplished by a 11-bit PC register which selects one of the 8-bit words contained in ROM. A new address is loaded into the PC register during each instruction cycle. Unless the instruction is a transfer of control instruction, the PC register is loaded with the next sequential 11-bit binary count value.

Three levels of subroutine nesting are implemented by a three levels deep stack. Each subroutine call (or interrupt) pushes the next PC address into the stack. Each return pops the stack back into the PC register.

DATA MEMORY

Data memory consists of a 512-bit RAM organized as 8 data registers of 16 x 4-bit digits. RAM addressing is implemented by a 7-bit B register whose upper 3 bits (Br) select 1 or 8 data registers and lower 4 bits (Bd) select 1 of 16 4-bit digits in the selected data register.

To emulate with accuracy the TS94120, the size of the RAM available to the user may be reduced by using $\overline{\text{SEL10}}$ and $\overline{\text{SEL20}}$ pins to obtain the following configurations:

— a 128 digits RAM is available, thus emulating an TS94144,

— the user RAM is physically halved to 64 digits (256 bits) thus emulating an TS94120,

The upper most bit of Br (Br3) register is only used to select between the RAM and the peripheral.

While the 4-bit contents of the selected RAM digit (M) are usually loaded into or from, or exchanged with the A register (accumulator), they may also be loaded into the Q latches or loaded from the L ports. RAM addressing may also be performed directly by the LDD and XAD instructions based upon the immediate operand field of these instructions.

The Bd register also serves as a source register for 4-bit data sent directly to the D outputs.

INTERNAL LOGIC

A 4-bit A register (accumulator) is the source and destination register for most I/O, arithmetic, logic and data memory access operations. It can also be used to load the Br and Bd field of the B register, to load and input 4 bits out of the 8-bits Q latch, to input 4 bits out of a ROM word, to input G or IN ports, and to perform data exchanges with the SIO register.

A 4-bit adder performs the arithmetic and logic functions, storing the results in A with eventually a carry bit in the 1-bit C register. The C register in conjunction with the XAS instruction and the EN register is also related to the SK output control.

The 8-bit T counter is a binary up counter which can be loaded to and from M and A using CAMT and CTMA instructions. This counter may be running in two modes depending on $\overline{\text{TIN}}$: as a timer or as an external event counter. When the T counter overflows, an overflow flag will be set (see SKT instruction below). Note that T counter is cleared on reset. A functional block diagram of the timer/counter is illustrated in Figure 4.

3 general-purpose inputs, IN3-IN1, are provided. Note that IN3 can be considered as a dual-timer input. IN0 is an analog input connected to an internal comparator.

The D register provides, 4 general-purpose outputs and is used as the destination register for the 4-bit contents of Bd. D3 is also used as the watchdog input when the watchdog option is selected.

The G register is a 2-bit general-purpose bidirectional I/O port (G3-G2). G0-G1 are the outputs of the peripheral and can also be used as general-purpose inputs.

The Q register is an internal, latched, 8-bit register, used to hold data loaded to or from M and A, as well as 8-bit data from ROM. Q register contents are transferred to the L I/O ports when the L drivers are enabled under program control (see LEI instruction).

L driver, when enabled, outputs the content of latched Q data to the L I/O ports. Contents of L may be read directly into A and M.

The SIO register is a 4-bit serial-in/serial-out shift register for the serial out. SIO content can be exchanged with A.

The XAS instruction copies C into the SKL Latch.

When SKL = 1, SK is a clock output
When SKL = 0, SK outputs a value depending upon the contents of EN0 and EN3.

The « EN register » is an internal 4-bit register loaded under program control by the LEI instruction. Each bit selects or inhibits the dedicated feature associated with each bit of the EN register (EN3-EN0).

- With EN1 set, the CPU interrupt is enabled.
- With EN2 set, the L drivers are enabled to output the data in Q to the L I/O ports. Resetting EN2 disables the L drivers, setting the L port as high impedance inputs.
- EN3, in conjunction with EN0, affects the SO and the SK outputs (see table 1).
  - — When EN3 = 0, SO outputs EN0 and SK outputs the clock anded with the content of the SKL latch,
  - — when EN3 = 1, SO is the output of the serial shifter, while SK outputs the clock (if SKL = 1) or the EN0 bit (if SKL = 0).

  Even when SO is disabled (as SIO output), data goes on shifting through SIO and can be exchanged with A via an XAS.

TABLE 1. ENABLE REGISTER MODES - BITS EN0 AND EN3

EN3	EN0	SO	SK after XAS
0	0	0	If SKL = 1, SK = Clock If SKL = 0, SK = 0
0	1	1	
1	0	Serial out	If SKL = 1, SK = Clock If SKL = 0, SK = EN0
1	1	Serial out	

INTERRUPT SOURCES

There are 3 possible interrupt sources (see Figure 4).

- Counter overflow or external interrupt ($\overline{\text{INT}}$ pin). External interrupt is triggered on the falling edge of a zero pulse. The zero pulse must be at least 2 instruction cycles wide on the external pin IN1 (or internal counter overflow).
- 2 peripheral interrupt sources (see peripheral description).

Each interrupt source has its own enable bit (EN1 for the CPU interrupt, ENA and ENB for the peripheral interrupts) and its own flag (F1 for the CPU interrupt, FA and FB for the peripheral interrupts, see Figure 15). An interrupt is enabled when its associated enable bit is set. Its associated flag is set when the interrupt is requested (for example, timer overflow for the CPU interrupt).

Note that the FA, FB, ENA, ENB bits, but also the F1 bit are implemented in the peripheral registers.

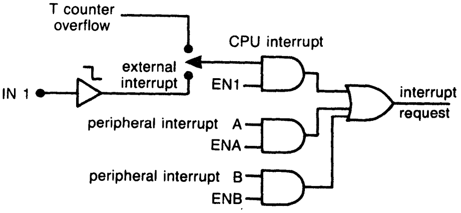


FIGURE 2: INTERRUPT BLOCK DIAGRAM

The following features are associated with interrupt procedure and protocol and must be considered by the programmer when utilizing interrupts.

- The interrupt, once acknowledged as explained below, pushes the next sequential program counter address (PC + 1) onto the stack:
  PC + 1 → SA → SB → SC. Any previous contents of SC is lost. The program counter is set to hex address 0FF (the last word of page 3) and all incoming interrupts are disabled without clearing the enable bits.
- An interrupt will be acknowledged only after the following conditions:
  - — An interrupt request has been generated.
  - — The associated enable bit is set.
  - — A currently executing instruction has been completed.
  - — All successive transfer of control instructions and successive LBIs have been completed (e.g., if the main program is executing a JP instruction which transfers program control to another JP instruction, the interrupt will not be acknowledged until the second JP instruction has been executed).
- Upon acknowledgement of an interrupt, bit Br3 of the B register is saved (and later restored upon « popping » off the stack by a RET instruction), then Br3 is cleared, allowing the user to begin his interrupt subroutine in the RAM space addressing.
- Upon acknowledgement of an interrupt, the skip logic status is saved and later restored upon « popping » off the stack. For example, if an interrupt occurs during the execution of ASC (Add with Carry, Skip on Carry) instruction which results in carry, the skip logic status is saved and program control is transferred to the interrupt servicing routine at hex address 0FF. At the end of the interrupt routine, a RET instruction is executed to « pop » the stack and return program control to the instruction following the original ASC. At this time, the skip logic is enabled and skips this instruction because of the previous ASC carry. Subroutines should not be nested within the interrupt servicing routine since « popping » the stack will enable any previously saved main program skips, interfering with the orderly execution sequence of the interrupt routine.
- The first instruction at hex address 0FF must be a NOP.
- When an interrupt is acknowledged, other interrupt sources are internally disabled until the end of the interrupt servicing routine.

RESET

The RESET pin is configured as a Schmitt trigger input. If not used it should be connected to Vcc. Initialization will occur whenever a logic « 0 » is applied to the RESET input for at least three instruction cycles.

Upon initialization, the PC register is cleared to 0 (ROM address 0) the A, B, C, D, EN, G2-G3, T counter, IL latches registers and the peripheral registers CRA, CRB, IMR are cleared. The SK output is enabled as a SYNC output, providing a pulse each instruction cycle time. Data Memory (RAM) is not cleared upon initialization. The first instruction at address 0 must be a CLRA. RC must be greather than 5 times power supply rise time.

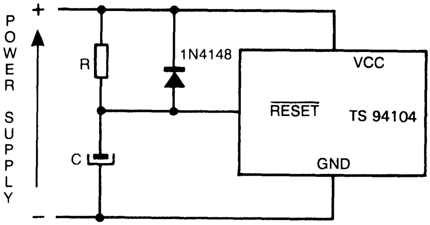


FIGURE 3: POWER-UP CLEAR CIRCUIT

T COUNTER

There are two modes selected by TIN pin.

— $\overline{\text{TIN}}$ = 0: TIME BASE COUNTER
Instruction cycle clock (CKI divided by 10) is divided by 4 x 256 = 1024 (see figure 6). For example, using a 1.0 MHz crystal, therefore getting a 10 µs instruction cycle period, TS94104 allows accurate timings between 10 µs* (1 count) and 10.24 ms (1024 counts). Longer timings can be achieved with software control (count of multiple overflows).

— $\overline{\text{TIN}}$ = 1: EXTERNAL EVENT COUNTER
Any pulse lasting more than 2 instructions cycle on IN2 will increment the T counter (on falling edge).

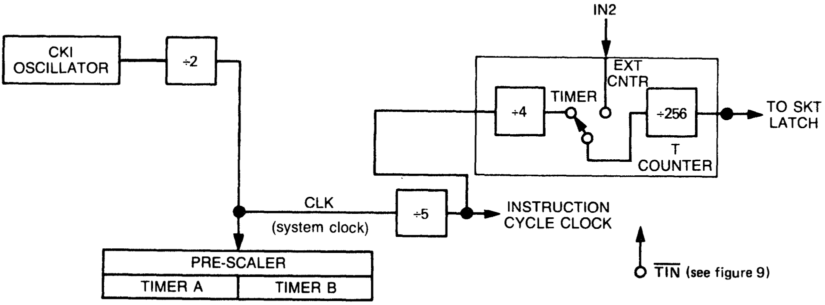


FIGURE 4: CLOCK AND CPU COUNTER BLOCK DIAGRAM

OSCILLATOR

CKI is configured as a TTL compatible external clock input (equivalent to option 3.2 for a TS94100). Instruction cycle time is the external frequency divided by 10. (equivalent to option 4.1 for a TS94100).

WATCHDOG

Watchdog mode is selected with WAD input

WAD = 0 NO WATCHDOG
WAD = 1 D3 IS THE WATCHDOG INPUT

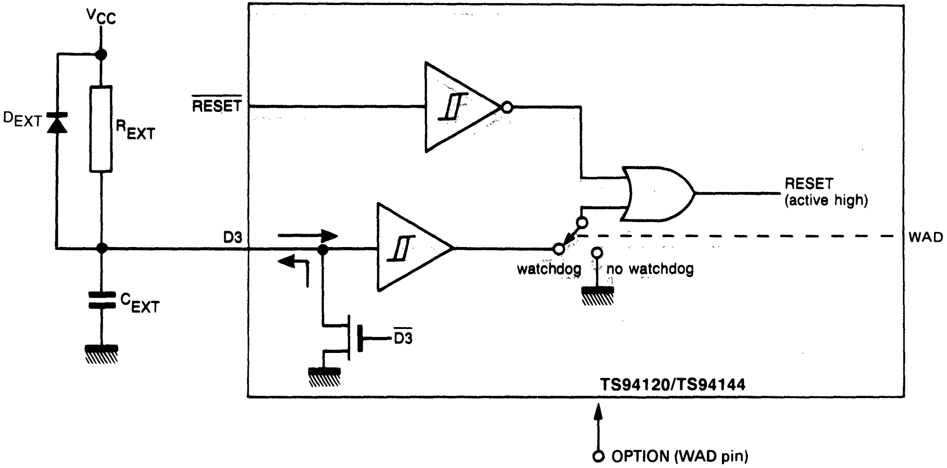


FIGURE 5: D3 AS WATCHDOG INPUT

In the watchdog mode, a « 1 » on pin D3 activates the internal reset (see Figure 5). The user needs only to connect a pull-up resistor and a capacitor on D3, and insert a watchdog loop in his program. This software loop will periodically outputs a « 0 » on D3 when the program is normally executed. When the microcontroller is trapped in another loop, the watchdog loop is not any more efficient and C_{ext} is charged up to Vcc, thus triggering the chip reset.

Open-drain option is selected on D3. The R_{EXT} x C_{EXT} time constant on D3 is related to the period of the software watchdog loop. D_{EXT} speeds up the C_{EXT} discharge time during power off.

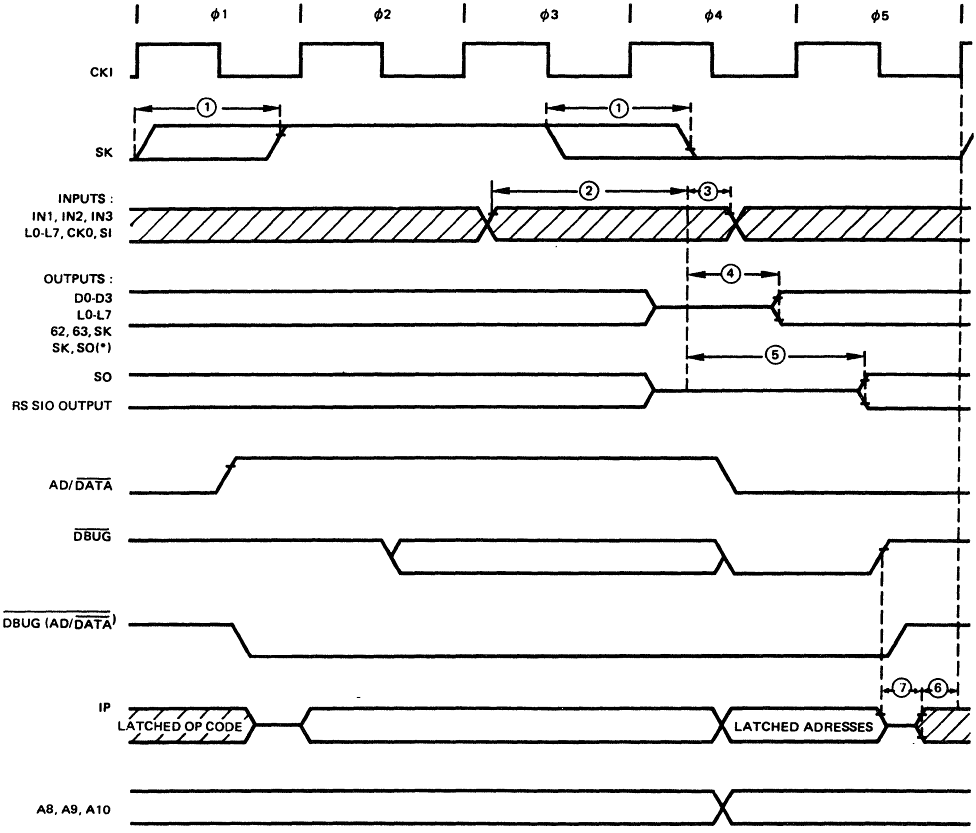
EXTERNAL MEMORY INTERFACE

The TS94104 is designed to be running with an external program memory. This memory requires the following characteristics:

- random addressing,
- TTL compatible 3-STATE OUTPUTS,
- TTL compatible inputs
- access time (from CS to output): 800 ns max.

These requirements are typically met with most EPROMS.
During operation, the address of the next instruction is sent out on A8, A9 and A10 and IP0-IP7 while $\overline{\text{AD/DATA}}$ is high (logic "1" = address mode).
Address data on the IP lines is stored into an external latch on the high to low transition of the AD/DATA line. When $\overline{\text{AD/DATA}}$ is low (logic "0" = data mode) the output of the memory is sent on the IP0-IP7 bus, and is latched inside the microcontroller on the rising edge of $\overline{\text{AD/DATA}}$. Note that $\overline{\text{AD/DATA}}$ output has a period of one instruction time, and specifies whether the IP bus is outputing addresses or inputing data. A simplified block diagram of the external memory interface is shown on Figure 6.

Ident. number	Characteristics	Symbol	Min.	Typ.	Max.	Unit
1	SK propagation delay	t_{s1}	—	—	2	μs
2	Input set up delay	t_{setup}	$t_c/10$ + 1.3	—	—	μs
3	Input Hold time	t_{hold}	100	—	—	ns
4	Output propagation delay	t_{pd1}	$t_c/10$	—	1.0	μs
5	SO propagation delay (SO as SIO output)	t_{pd0}	$t_c/10$	—	1.0	μs
6	IP set up delay (Opcode)	t_{opcod}	—	—	—	—
7	IP buffer propagation delay	t_{buff}	—	—	—	—

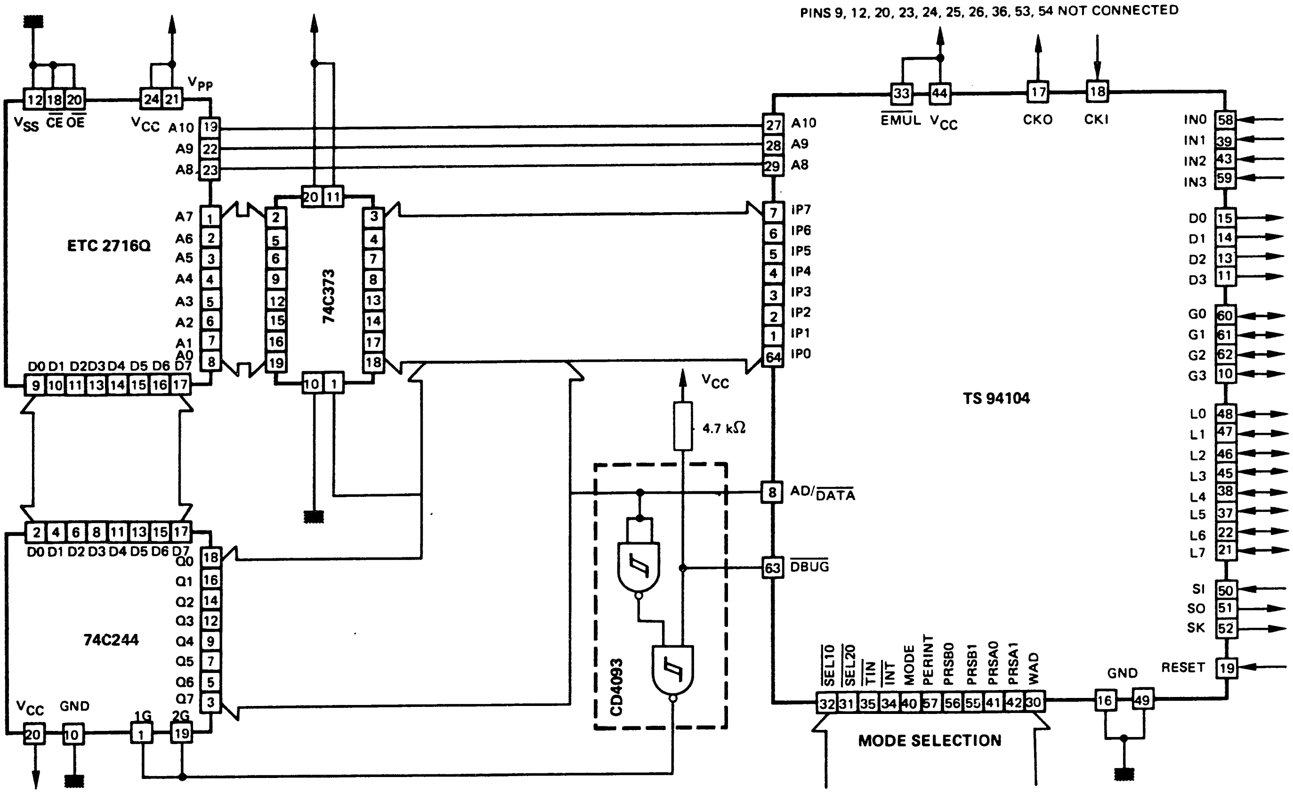


(*) SO AS AN OUTPUT WHEN (EN0, EN3) = (0,0) OR (1,0) OR (1,1)

- — SK has 50% duty cycle when used as a clock.
- — Outputs are sampled (by MCU) on the rising edge of SK (except IP0-IP7, A8-A10).
- — Inputs are sampled on the SK falling edge.
- — The external address bus (IP0-IP7, A8-A10) is sampled on the falling edge of AD/DATA.
- — The IP0-IP7 inputs have to be referenced to the rising edge of AD/$\overline{\text{DATA}}$.

Note:

- — Consider we are executing an instruction (from /1 to /5). The address sampled on /5 on the address bus is the address of next instruction, but the SKIP status sampled on the SKIP output is the status of the instruction currently under execution.
- — The SKIP output = 1 when the current instruction is skipped, except for LBI's skipped after an LBI.



PINS 9, 12, 20, 23, 24, 25, 26, 36, 53, 54 NOT CONNECTED

FIGURE 6

DUAL TIMER

Dual Timer is identical to TS94100 family. Please refer to the TS94100 specification.

INSTRUCTION SET

TS94104 instruction set is identical to the TS94100 instruction set. Please refer to the TS94100 specification.

ELECTRICAL CHARACTERISTICS

TS94104 electrical characteristics are identical to the TS94100. Please refer to the TS94100 specifications.

MODE SELECTION

TS94104 can emulate the whole TS94100 family (i.e. 1K and 2K ROM) by simply connecting $\overline{\text{SEL10}}$ and $\overline{\text{SEL20}}$ pins to Vcc or GND. In addition, most TS94100 family mask options can be choosen by setting extra pins to Vcc or GND as shown on figure 7, figure 8.

	TS94144	TS94120
$\overline{\text{SEL10}}$	1	1
$\overline{\text{SEL20}}$	1	0

MCU SIZE

	Prescaler	Event counter
$\overline{\text{TIN}}$	0	1

COUNTER INPUT

	No watchdog	D3 pin as watchdog input
WAD	0	1

WATCHDOG

	External interrupt	Timer overflow
$\overline{\text{INT}}$	0	1

INTERRUPT SOURCE

FIGURE 7

ROMLESS - I/O AND OSCILLATOR OPTIONS SELECTED

	÷ 1	÷ 5	÷ 20	÷ 40
PRSA0	0	1	0	1
PRSA1	0	0	1	1

TIMER A PRESCALER

	Period measurement	**Duty cycle measurement**
MODE	0	1

TIMER B: PERIOD/DUTY CYCLE (MRB0 = 0)
(MRB1 = 0)

	÷ 1	÷ 5	÷ 20	÷ 40
PRSB0	0	1	0	1
PRSB1	0	0	1	1

TIMER B PRESCALER

	High to low transition on IN0	**Every transition of IN0**
PERINT	0	1

IN0 TRANSITION

FIGURE 8

PIN NAME

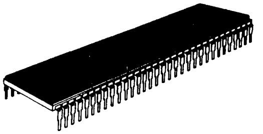
IP0-IP7	I/O for ROM address and data
A8-A9-A10	Address Most Significant Bits
SKIP	SKIP status output
AD/$\overline{\text{DATA}}$	Address out/data in (demux special)
$\overline{\text{INT}}$	Select pin for external interrupt or timer overflow (CPU)
$\overline{\text{TIN}}$	Select pin for prescaler or event counter (CPU)
$\overline{\text{SEL10-SEL20}}$	Select pin for the RAM size (TS94210 or TS94144)
$\overline{\text{DBUG}}$	Output enable signal for external EPROM, and select pin for debug modes (not to be used by the customer)
$\overline{\text{EMUL}}$	Emulation pin: internal timer and peripheral halt, interrupts masking
WAD	Select pin for watchdog option
MODE	Select pin for period measurement or duty cycle measurement
PERINT	Select pin for high to low or every transition of IN0 (dual timer)
PRSA0 PRSA1	Select pin for prescaler division of TIMER A
PRSB0 PRSB1	Select pin for prescaler division of TIMER B

ROMLESS EXTRA PINS - FUNCTIONAL DESCRIPTION

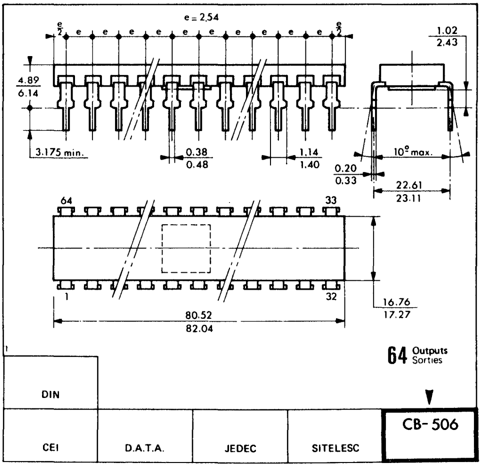
		OPTION
CKO PIN CONFIGURATION	GENERAL PURPOSE INPUT, LOAD DEVICE TO Vcc	1-1
CKO INPUT LEVEL	STANDARD TTL INPUT LEVELS	2-0
CKI INPUT CONFIGURATION	EXTERNAL CLOCK	3-2
CKI DIVIDER	CKI FREQUENCY DIVIDED BY 10	4-1
RESET PIN CONFIGURATION	LOAD DEVICE TO Vcc	5-0
IN1-IN3 INPUT CONFIGURATION	LOAD DEVICE TO Vcc	6-0
IN1-IN3 INPUT LEVELS	STANDARD TTL INPUT LEVELS	7-0
S1 INPUT LEVELS	STANDARD TTL INPUT LEVELS	8-0
SO OUTPUT CONFIGURATION	GENERAL PURPOSE TTL OUTPUT	9-0
SK OUTPUT CONFIGURATION	GENERAL PURPOSE HIGH CURRENT OUTPUT	10-1
LO, L1, L2 OUTPUT DRIVER	3 STATE PUSH-PULL OUTPUT	11-5, 12-5, 13-5
L3 OUTPUT CONFIGURATION	3 STATE GENERAL PURPOSE TTL OUTPUT	14-0
L4 OUTPUT CONFIGURATION	3 STATE GENERAL PURPOSE VERY HIGH CURRENT OUTPUT	15-1
L5 OUTPUT CONFIGURATION	OPEN DRAIN TTL OUTPUT	16-2
L6 OUTPUT CONFIGURATION	OPEN DRAIN VERY HIGH CURRENT OUTPUT	17-3
L7 OUTPUT CONFIGURATION	3 STATE PUSH-PULL TTL OUTPUT	18-4
L INPUT LEVEL	HIGHER VOLTAGE INPUT LEVEL	19-1
G2-G3 OUTPUT CONFIGURATION	GENERAL PURPOSE OUTPUT	20-0, 21-0
G INPUT LEVEL	STANDARD TTL INPUT LEVELS	22-0
D0-D3 OUTPUT CONFIGURATION	OPEN-DRAIN OUTPUT	23-1, 24-1, 25-1, 26-1
G0 OUTPUT LEVEL FOR STA=0	LOGICAL 1	34-1
G1 OUTPUT LEVEL FOR STB=0	LOGICAL 1	35-1

PHYSICAL DIMENSIONS

CB-506



PLASTIC PACKAGE



These specifications are subject to change without notice.
Please inquire with our sales offices about the availability of the different products.

HOMSON
COMPONENTS
EMICONDUCTEURS

TS94120 • TS94144

SINGLE CHIP MICROCONTROLLERS

ADVANCE INFORMATION

The TS94120/144 single chip NMOS microcontrollers are members of the TS94000 family, fabricated using N-channel silicon gate XMOS technology. The TS94120/144 contain the internal logic of the microcontrollers ET9410/20/44 (system timing, internal logic, ROM, RAM and I/O necessary to implement dedicated control functions). Features include single supply operation, a variety of output configuration options, with an instruction set, internal architecture, and I/O scheme designed to facilitate keyboard input, display output, BCD data manipulation and triac drive. They are an appropriate choice for use in numerous human interface control environments. Standard test procedures and reliable high-density fabrication techniques provide the medium to large volume customers with a customized low cost controller. TS94100 family offers a dual 16-bit timer perfectly suited for universal motor speed control (phase control), accurate time measurements and waveform generation.

- Low cost
- 23 I/O lines
- Single supply operation (4.5 V to 5.5 V)
- Low-current drain (13 mA max)
- General purpose and 3-state outputs
- TTL compatible inputs
- TTL compatible outputs
- Temperature range (0° to + 70°C)
- Extended temperature range (— 40°C to + 85°C) available.

CPU

- Powerful instruction set (52 instructions)
- ROM: 2K x 8 (TS94144), 1K x 8 (TS94120)
- RAM: 128 x 4 (TS94144), 64 x 4 (TS94120)
- 2 interrupt sources: external pin or internal timer overflow
- 3-level subroutine stack
- 4 μs instruction cycle time
- Programmable read/write 8-bit internal timer/event counter
- Internal shift register with serial I/O
- Watchdog feature
- Direct drive of LED digit and segment line
- Software compatible with the other members of ET9400, ETL9400 and ETC9400 families.

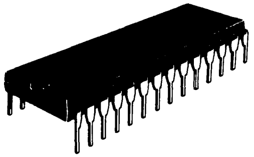
DUAL TIMER PERIPHERAL

- Two 8-bit and one 16-bit independent timers/counters
- 2 independent programmable prescalers
- Maximum working frequency: 1.25 MHz
- 16 software programmable working modes including delayed and calibrated pulse triggering, complex waveform generation, period and duty-cycle measurements
- 3 interrupt sources
- Zero-crossing detector
- Direct drive of triac

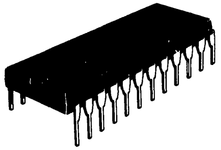
NMOS

CASES

CB-132

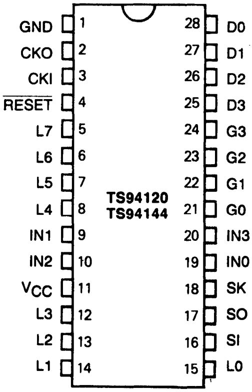


CB-68



P SUFFIX
PLASTIC PACKAGE

PIN ASSIGNMENT



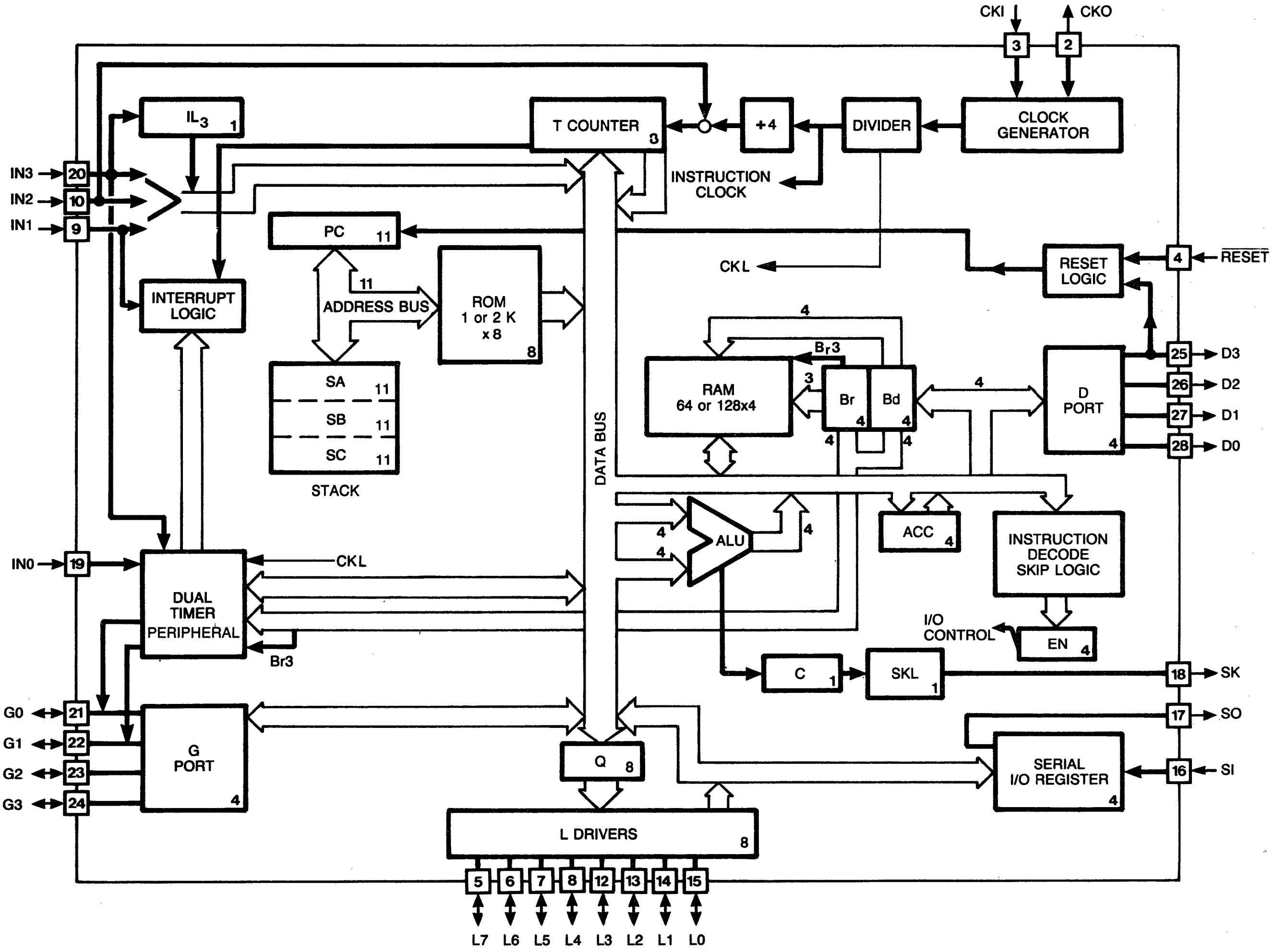


FIGURE 1 - BLOCK DIAGRAM

CPU FUNCTIONAL DESCRIPTION

The internal architecture is shown in Figure 1. Data paths are illustrated in simplified form to depict how the various logic elements communicate with each other in implementing the instruction set of the device. Positive logic is used. A bit set is a logic « 1 », when a bit reset is a logic « 0 ».

PROGRAM MEMORY

Program memory consists of ROM:

- 1024 bytes for the TS94120
- 2048 bytes for the TS94144

These bytes of ROM can be either program instructions, constants or ROM addressing data.

ROM addressing is accomplished by a 11-bit PC register which selects one of the 8-bit words contained in ROM. A new address is loaded into the PC register during each instruction cycle. Unless the instruction is a transfer of control instruction, the PC register is loaded with the next sequential 11-bit binary count value.

- For TS94120 the 10 lower bits of PC are used to address the internal ROM

Three levels of subroutine nesting are implemented by a three level deep stack. Each subroutine call (or interrupt) pushes the next PC address onto the stack. Each return (RET) pops off the stack back into the PC register.

DATA MEMORY

Data memory consists of a 512-bit RAM for the TS94144 organized as 8 data registers of 16 x 4-bit digits. RAM addressing is implemented by a 8-bit B register whose upper 3 bits Br0, Br1, Br2 select 1 of 8 data registers and lower 4 bits (Bd) select 1 of 16 4-bit digits in the selected data register.

Data memory consists of a 256-bit RAM for the TS94120 organized as 4 data registers of 16 x 4 bits digits. The B register is 8 bits long. Upper 2 bits Br0, Br1 select 1 of 4 data registers and lower 4 bits (Bd) select 1 of 16 x 4-bit digits in the selected data register. Br2 is not used to address the RAM and so 2 addresses with a different value of Br2 will access the same word in the RAM.

For TS94120/144, the uppermost bit Br3 of B register is only used to select either the peripheral or RAM.

While the 4-bit contents of the selected RAM digit (M) are usually loaded into or from, or exchanged with the A register (accumulator), they may also be loaded into the Q latches or loaded from the L ports. RAM addressing may also be performed directly by the LDD and XAD instructions based upon the immediate operand field of these instructions.

The Bd register also serves as a source register for 4-bit data sent directly to the D outputs.

INTERNAL LOGIC

A 4-bit A register (accumulator) is the source and destination register for most I/O, arithmetic, logic and data memory access operations. It can also be used to load the Br and Bd field of the B register, to load and input 4 bits out of the 8-bits Q latch, to input 4 bits out of a ROM word, to input G or IN ports, and to perform data exchanges with the SIO register.

A 4-bit adder performs the arithmetic and logic functions, storing the results in A with eventually a carry bit in the 1-bit C register. The C register in conjunction with the XAS instruction and the EN register is also related to the SK output control.

The 8-bit T counter is a binary up counter which can be loaded to and from M and A using CAMT and CTMA instructions. This counter may be running in two modes depending on a mask-programmable option: as a timer or as an external event counter. When the T counter overflows, an overflow flag will be set (see SKT instruction below). Note that T counter is cleared on reset. A functional block diagram of the timer/counter is illustrated in Figure 4.

3 general-purpose inputs, IN3-IN1, are provided. Note that IN3 can be considered as a dual-timer input.

IN0 is an analog input connected to an internal comparator.

The D register provides, 4 general-purpose outputs and is used as the destination register for the 4-bit contents of Bd. D3 is also used as the watchdog input when the watchdog option is selected.

The G register is a 2-bit general-purpose bidirectional I/O port (G3-G2). G0-G1 are the outputs of the peripheral and can also be used as general-purpose inputs.

The Q register is an internal, latched, 8-bit register, used to hold data loaded to or from M and A, as well as 8-bit data from ROM. Q register contents are transferred to the L I/O ports when the L drivers are enabled under program control (see LEI instruction).

L driver, when enabled, outputs the content of latched Q data to the L I/O ports. Contents of L may be read directly into A and M.

The SIO register is a 4-bit serial-in/serial-out shift register for the serial out. SIO content can be exchanged with A.

The XAS instruction copies C into the SKL Latch.

When SKL = 1, SK is a clock output
When SKL = 0, SK outputs a value depending upon the contents of EN0 and EN3.

THE « EN register » is an internal 4-bit register loaded under program control by the LEI instruction. Each bit selects or inhibits the dedicated feature associated with each bit of the EN register (EN3-EN0).

- With EN1 set, the CPU interrupt is enabled.
- With EN2 set, the L drivers are enabled to output the data in Q to the L I/O ports. Resetting EN2 disables the L drivers, setting the L port as high impedance inputs.
- EN3, in conjunction with EN0, affects the SO and the SK outputs (see table 1).
  - When EN3 = 0, SO outputs EN0 and SK outputs the clock anded with the content of the SKL latch,
  - when EN3 = 1, SO is the output of the serial shifter, while SK outputs the clock (if SKL = 1) or the EN0 bit (if SKL = 0).

  Even when SO is disabled (as SIO output), data goes on shifting through SIO and can be exchanged with A via an XAS.

TABLE 1. ENABLE REGISTER MODES - BITS EN0 AND EN3

EN3	EN0	SO	SK after XAS
0	0	0	If SKL = 1, SK = Clock If SKL = 0, SK = 0
0	1	1	
1	0	Serial out	If SKL = 1, SK = Clock If SKL = 0, SK = EN0
1	1	Serial out	

INTERRUPT SOURCES

There are 3 possible interrupt sources (see Figure 2).

- Counter overflow or external interrupt (mask option). External interrupt is triggered on the falling edge of a zero pulse. The zero pulse must be at least 2 instruction cycles wide on the external pin IN1 (or internal counter overflow).
- 2 peripheral interrupt sources (see peripheral description).

Each interrupt source has its own enable bit (EN1 for the CPU interrupt, ENA and ENB for the peripheral interrupts) and its own flag (F1 for the CPU interrupt, FA and FB for the peripheral interrupts, see Figure 13). An interrupt is enabled when its associated enable bit is set. Its associated flag is set when the interrupt is requested (for example, timer overflow for the CPU interrupt).

Note that the FA, FB, ENA, ENB bits, but also the F1 bit are implemented in the peripheral registers.

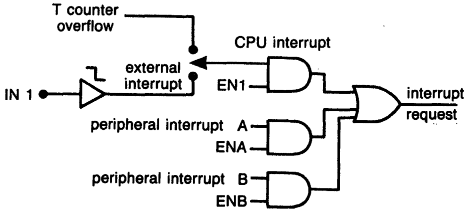


FIGURE 2: INTERRUPT BLOCK DIAGRAM

The following features are associated with interrupt procedure and protocol and must be considered by the programmer when utilizing interrupts.

- The interrupt, once acknowledged as explained below, pushes the next sequential program counter address (PC + 1) onto the stack:
  PC + 1 → SA → SB → SC. Any previous contents of SC is lost. The program counter is set to hex address 0FF (the last word of page 3) and all incoming interrupts are disabled without clearing the enable bits.
- An interrupt will be acknowledged only after the following conditions:
  - An interrupt request has been generated.
  - The associated enable bit is set.
  - A currently executing instruction has been completed.
  - All successive transfer of control instructions and successive LBIs have been completed (e.g., if the main program is executing a JP instruction which transfers program control to another JP, instruction, the interrupt will not be acknowledged until the second JP instruction has been executed).
- Upon acknowledgement of an interrupt, bit Br3 of the B register is saved (and later restored upon « popping » off the stack by a RET instruction), then Br3 is cleared, allowing the user to begin his interrupt subroutine in the RAM space addressing.
- Upon acknowledgement of an interrupt, the skip logic status is saved and later restored upon « popping » off the stack. For example, if an interrupt occurs during the execution of ASC (Add with Carry, Skip on Carry) instruction which results in carry, the skip logic status is saved and program control is transferred to the interrupt servicing routine at hex address 0FF. At the end of the interrupt routine, a RET instruction is executed to « pop » the stack and return program control to the instruction following the original ASC. At this time, the skip logic is enabled and skips this instruction because of the previous ASC carry. Subroutines should not be nested within the interrupt servicing routine since « popping » the stack will enable any previously saved main program skips, interfering with the orderly execution sequence of the interrupt routine.
- The first instruction at hex address 0FF must be a NOP.
- When an interrupt is acknowledged, other interrupt sources are internally disabled until the end of interrupt servicing subroutine.

RESET

The RESET pin is configured as a Schmitt trigger input. If not used it should be connected to Vcc. Initialization will occur whenever a logic « 0 » is applied to the RESET input for at least three instruction cycles.

Upon initialization, the PC register is cleared to 0 (ROM address 0) the A, B, C, D, EN, G2-G3, T counter, IL latches registers and the peripheral registers CRA, CRB, IMR are cleared. The SK output is enabled as a SYNC output, providing a pulse each instruction cycle time. Data

Memory (RAM) is not cleared upon initialization. The first instruction at address 0 must be a CLRA. RC must be greather than 5 times power supply rise time.

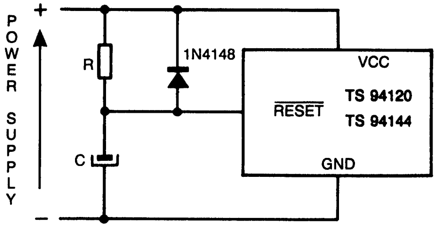


FIGURE 3: POWER-UP CLEAR CIRCUIT

T COUNTER

There are two modes selected by mask option:

- Time-base counter. In this mode, the instruction cycle frequency generated from CKI passes through a 2-bit (divide-by-4) prescaler. The output of this prescaler increments the 8-bit counter thus providing a 10-bit timer. T counter prescaler is cleared by a CAMT instruction or on reset.

  For example, using a 1.25 MHz crystal with a divide-by-5 option, the instruction cycle frequency of 250 KHz increments the 10-bit timer every 4 μs. By presetting the counter and detecting overflow, accurate timeouts between 16 μs (4 counts) and 4.096 ms (1024 counts) are possible. Longer timeouts can be achieved by accumulating, under software control, multiple overflows.

- External event counter. With this option, a low-going pulse (« 1 » to « 0 ») lasting more than 2 instruction cycles on the IN2 input will increment the 8-bit T counter.

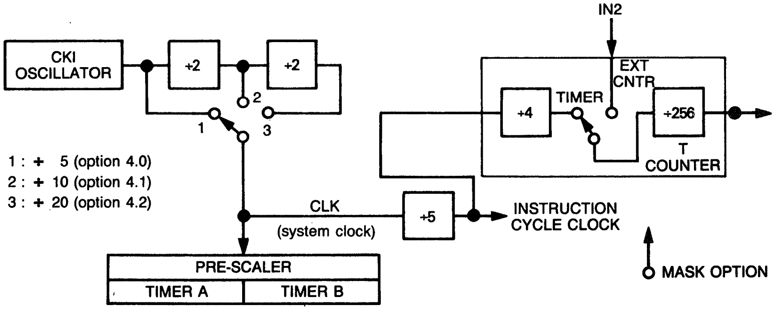


FIGURE 4: CLOCK AND CPU COUNTER BLOCK DIAGRAM

OSCILLATOR

There are three basic clock oscillator configurations available as shown in Figures 5 and 6.

- Crystal Controller Oscillator. CKI and CKO are connected to an external crystal. The instruction cycle time equals the crystal frequency divided by 20 (optional by 10 or 5).

- External Oscillator. CKI is an external clock input signal. The external frequency is divided by 10 to give the instruction cycle time. CKO is available to be used as a general purpose input.

- RC Controlled Oscillator. CKI is configured as a single pin RC controlled Schmitt trigger oscillator. The instruction cycle equals the oscillation frequency divided by 10 to give the instruction cycle time. CKO is in this case available to be used as a general purpose input (see INIL instruction).

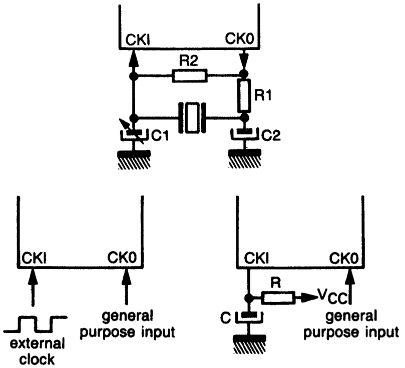


FIGURE 5: OSCILLATOR EXTERNAL CONFIGURATION

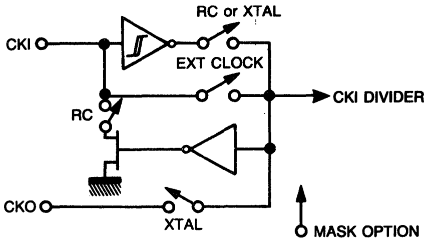


FIGURE 6: OSCILLATOR MASK OPTIONS

WATCHDOG (Figure 7)
This mode may be chosen by mask option. In this case, D3 becomes the watchdog input. This mode is a basic need for microcontrollers working in a noisy environment, where parasitics may crash the program.

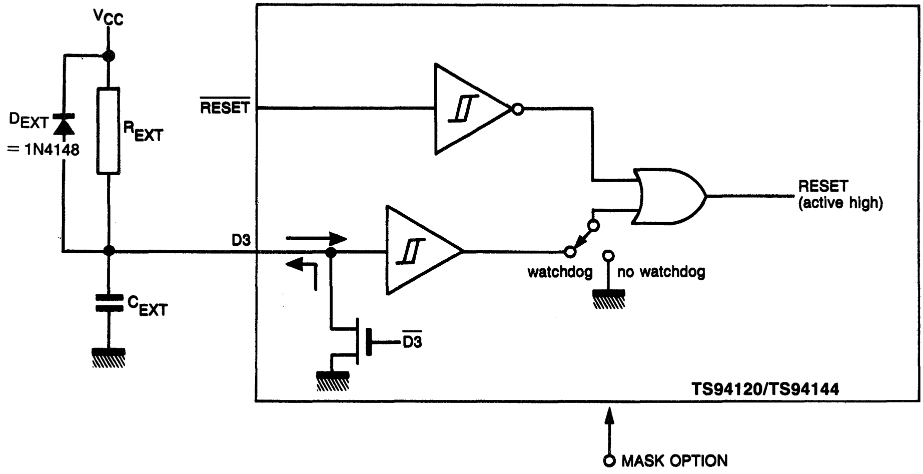


FIGURE 7: D3 AS WATCHDOG INPUT

In the watchdog mode, a « 1 » on pin D3 activates the internal reset (see Figure 7). The user needs only to connect a pull-up resistor and a capacitor on D3, and insert a watchdog loop in his program. This software loop will periodically outputs a « 0 » on D3 when the program is normally executed. When the microcontroller is trapped in another loop, the watchdog loop is not any more efficient and C_{ext} is charged up to Vcc, thus triggering the chip reset.

Open-drain option has to be selected on D3 when the watchdog mode is chosen. The R_{EXT} x C_{EXT} time constant on D3 is related to the period of the software watchdog loop. D_{EXT} speeds up the C_{EXT} discharge time during power off.

DUAL TIMER

FUNCTIONAL OVERVIEW

A block diagram of the peripheral is given in Figure 8. It consists of two truly independent blocks. Block A consists of two 8-bit registers (RW0-RW1, RD0-RD1) and two 8 bit binary down-counters (CW0-CW1, CD0-CD1) organized as two register-counter pairs. The block B consists of a sixteen bit register (RB0-RB3) and a 16-bit binary down counter (CB0-CB3). The operating modes are software programmable through CRA and CRB control registers. Most of the modes are driven by block A and B clocks (CLKA, CLKB) which are respectively generated by the A or B prescalers. These prescalers are mask-option programmable. The following options are available: the system clock (CLK) may be divided by 1, 5, 20, 40.

Block A is a complex waveform generator. In most of the modes, each 8 bit pair is running alternatively, using the autoreload procedure. The CD counter starts to run. It counts down to zero, at which time the output flip-flop of block A is toggled. The data in the RD register is then reloaded automatically into the CD counter, which stays idle. In the same time, the RW/CW 8 bit pair starts running. When CW counts down to zero, the output flip-flop is toggled again, CW is reloaded with RW contents and stays idle. The RD/CD pair can start running again and so on. In the pulse burst generation mode, the CD/RD pair is running alone: the output flip-flop is toggled each time the CD counter counts down to zero. The CW/RW pair does not drive the output flip-flop, but counts the pulses generated by the CD/RD pair. Block A output is G0, input is IN0.

Block B performs measurement and generation functions. In the frequency generation and frequency divider mode, the 16-bit pair is running, using the autoreload procedure. In the measurement mode, while the counter is running, an automatic transfer to the register is performed each time a measure is ended. This is determined by a mask-option: if the period measurement mode is selected, a measure is ended each time a high to low transition occurs on the IN3 pin. If the duty-cycle measurement mode is selected, a measure is ended each time a high to low or low to high transition occurs on the IN3 pin. In the external event counter mode, the counter counts down the external pulses, register RB latching up counter CB state every instruction cycle t_C. Block B output is G1, input is IN3.

In all modes of blocks A and B, there is no initial autoreload, and the counters start, regardless of their content.

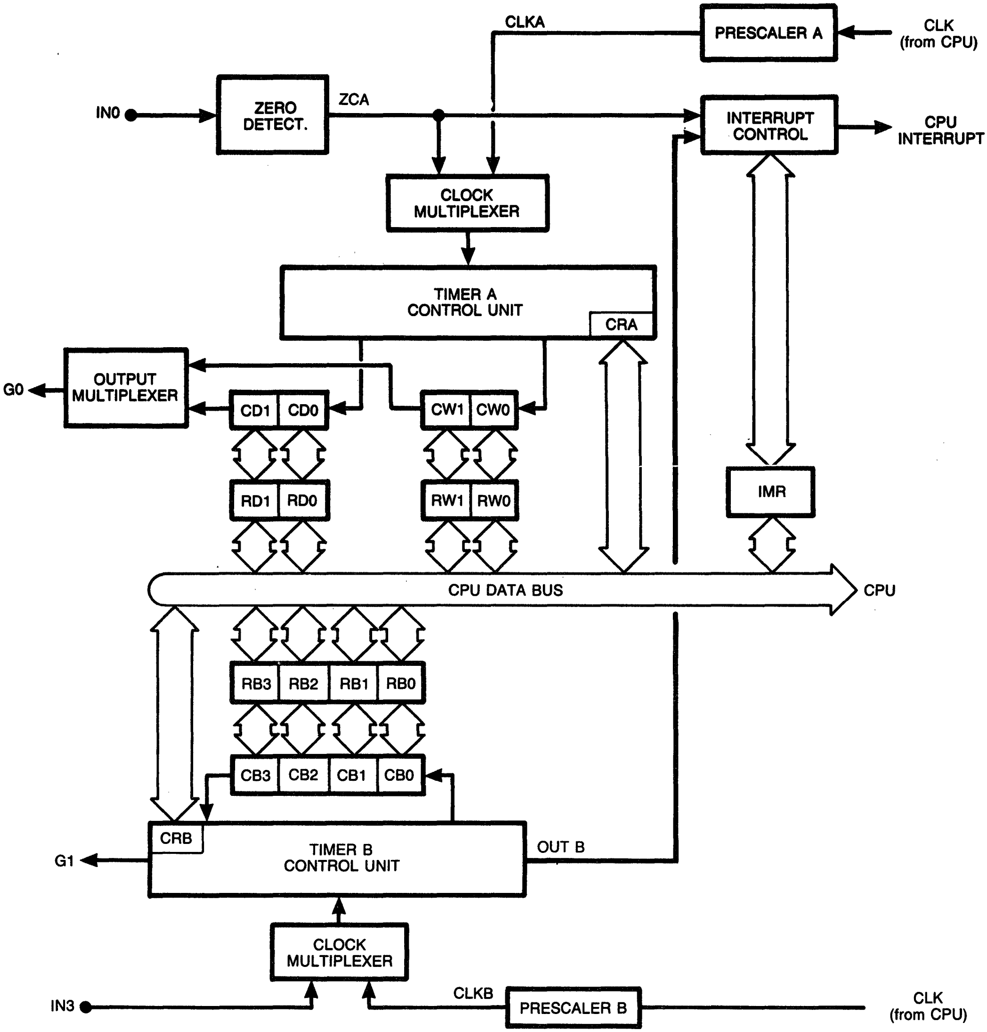

CLKA
PRESCALER A
CLK (from CPU)
IN0
ZERO DETECT.
ZCA
INTERRUPT CONTROL
CPU INTERRUPT
CLOCK MULTIPLEXER
TIMER A CONTROL UNIT
CRA
G0
OUTPUT MULTIPLEXER
CD1
CD0
CW1
CW0
RD1
RD0
RW1
RW0
IMR
CPU DATA BUS
CPU
RB3
RB2
RB1
RB0
CB3
CB2
CB1
CB0
CRB
G1
TIMER B CONTROL UNIT
OUT B
CLOCK MULTIPLEXER
IN3
CLKB
PRESCALER B
CLK (from CPU)

FIGURE 8

PERIPHERAL REGISTERS

The Dual-Timer contains several registers that interface with the CPU :

- two 8-bit data registers (RD0-RD1) and (RW0-RW1) for block A, and one-16 bit data register (RB0-RB1-RB2-RB3) for block B.
- two 4-bit control registers (1 per block) CRA and CRB managing the selection mode, the output on the pins G0 and G1 and start-up for both blocks.
- One 4-bit interrupt management register (IMR). This register contains the enable bits and the interrupt flags.

CRA: Control Register

F1	STA	MRA1	MRA0
3	2	1	0

STA: Start/Stop bit
and output enable
on pin G0
MRA1-MRA0: Block A
mode selection
F1: CPU interrupt flag

CRB: Control Register

CSB	STB	MRB1	MRB0
3	2	1	0

STB: Start/Stop bit

CSB: Output enable
on pin G1
MRB1-MRB0: Block B
mode selection

IMR: Interrupt Management Register

ENB	ENA	FB	FA
3	2	1	0

FA, FB: Interrupt flag of blocks A and B
ENA, ENB : Interrupt enable bits of blocks A and B

FIGURE 9: CONTROL REGISTERS (CRA AND CRB) AND INTERRUPT MANAGEMENT REGISTER (IMR)

All these registers are memory mapped as 4-bit memory words (See Figure 10) and are addressed by Bd and the lower bit Br0 of Br (See Figure 11).

Bd Br0	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	RD1	RD0	RW1	RW0												CRA
1	RB3	RB2	RB1	RB0			IMR									CRB

FIGURE 10: REGISTERS MEMORY MAPPING

All the instructions referenced to the RAM may be used for the peripheral, including bit manipulation instructions. Br3, the uppermost bit of Br controls the switching between the RAM and the peripheral when using memory reference instructions.

Two instructions are able to alter the content of Br3:

— XABr, that exchanges A with Br, and in particular A3 with Br3,

— TMB, that complements Br3.

When Br3 = 0, RAM is accessed
When Br3 = 1, peripheral registers are accessed

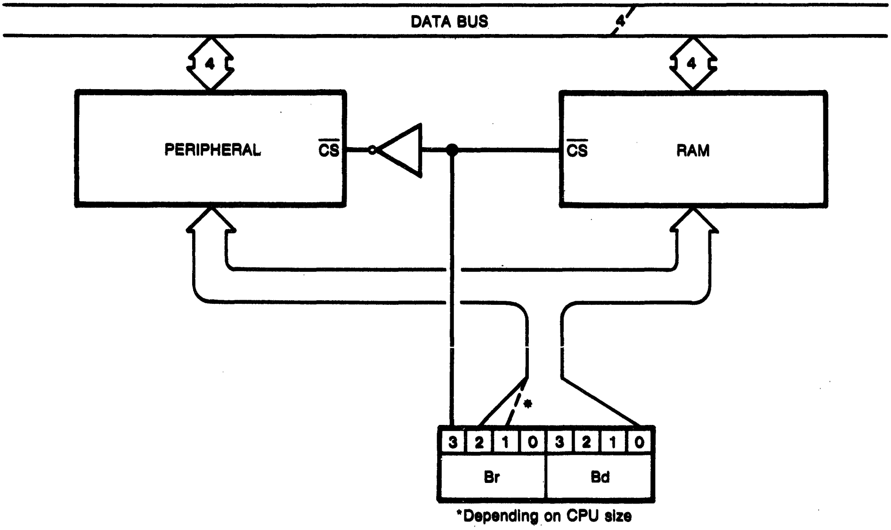


FIGURE 11: PERIPHERAL AND RAM ADDRESSING BLOCK DIAGRAM

Note that, as Br1 and Br2 are not used to address the registers, there is redundancy in the addressing: that is, 2 addresses where only Br1 (or Br2) is different will access the same register.

- Each peripheral memory word may be read at any time, even during reloading: the value read is always valid.
- On the other hand, the user cannot write in a peripheral memory word while the peripheral internal logic is trying to reload it.

To ensure a good start of the timer, the Start/Stop bit (STA, STB) should not be set while setting a new mode (MRA0-MRA1, MRB0-MRB1): one instruction cycle is required, for stabilization.

Counters access:
Data transfer between counters and registers (RD ↔ CD, RW ↔ CW, RB ↔ CB) exchanges words of 16 bits (or 8 bits). This transfer is done with a single instruction (2 bytes opcode) as shown on table 2.

MNEMONIC	OPCODE	DATA FLOW
TRCA	23 8A	RD → CD RW → CW
TCRA	23 8B	CD → RD CW → RW
TRCB	23 9A	RB → CB
TCRB	23 9B	CB → RB

TABLE 2: 16-BIT TRANSFER INSTRUCTIONS

Opcodes of these 16 bits transfer instructions are XAD opcodes addressing unused peripheral addresses, and are valid only when Br3 = 1. TRCA, TCRA, TRCB and TCRB instructions load accumulator A with F(Rex).

Important remark

Note that transfer is equivalent to:
XAD 0A for transfer (RW, RD) towards (CW, CD)
XAD 0B for transfer (CW, CD) towards (RW, RD)
XAD 1A for transfer RB towards CB
XAD 1B for transfer CB towards RB
} when $B_r3 = 1$

These four transfer instructions are therefore compatible with the ET9400 set of instructions. XAD instruction has a different meaning when Br3 = 1.

CONTROL REGISTERS CRA AND CRB

— The MRA0-MRA1 and MRB0-MRB1 bits select the operating mode of block A and block B. The mode selected by MRB0 = MRB1 = 0 is mask option programmable: period measurement, or duty-cycle measurement.

Block A

MRA1	MRA0	OPERATING MODE
0	0	Triggered pulse generation
0	1	Pulse burst generation
1	0	Duty cycle generation
1	1	Analog period counter

Block B

MRB1	MRB0	OPERATING MODE
0	0	Period measurement Duty-cycle measurement } mask programmable option
0	1	External event counter
1	0	Frequency generation
1	1	Frequency divider

TABLE 3: OPERATING MODE CODING THROUGH MRA0- MRA1 AND MRB0-MRB1 BITS

— Setting STA and STB to zero stops the counters which stays idle. At the same time, the A and B flip flop outputs are forced to predefined values (« 0 » or « 1 » selectable by mask option).

— Setting STA and STB to 1, makes the respective counters run in the selected mode. In the pulse burst mode, STA is automatically reset when the burst is generated.

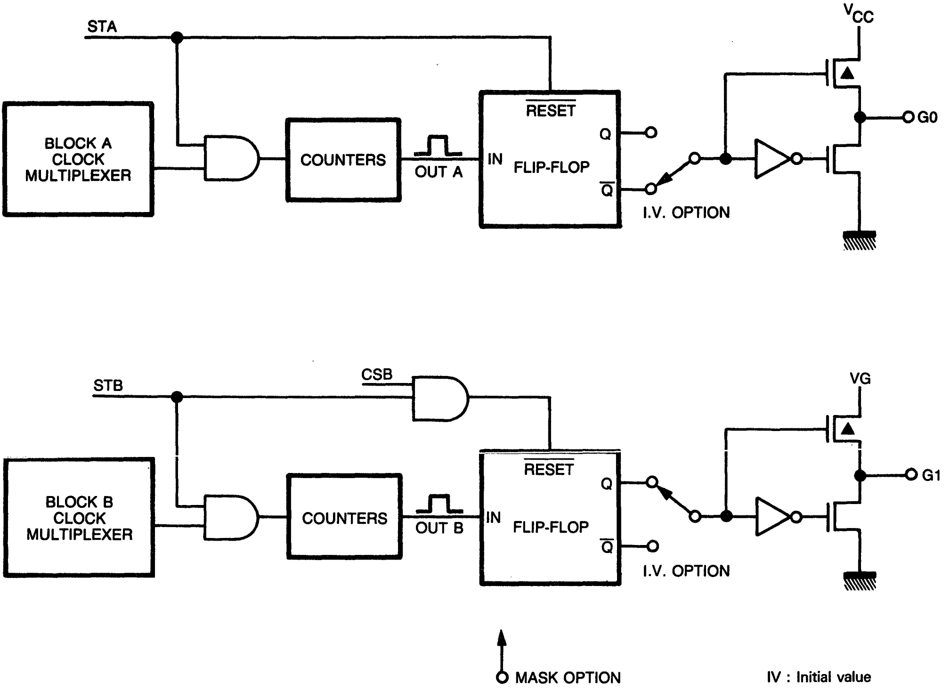


FIGURE 12: OUTPUT CONTROL

Setting the CSB bit to zero disconnects the pin G1 from the CB counter : counter is running and G1 is forced to 1 or 0 (depending on mask option)

G0 (and G1) pins can be used as a general purpose CPU input, as long as it has been previously set to output a 1.

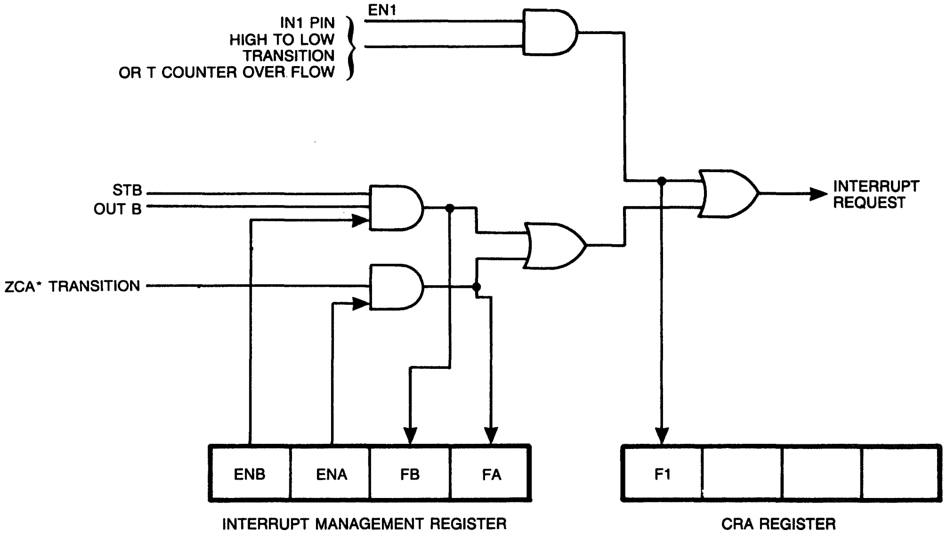
If proper start is required, the counters should be initialized before the start up. In other words, the first G0 (or G1) transition (counter underflow) occurs at a defined time after start up only if the counter C was previously loaded (see nota-1).

SYSTEM RESET

Upon reset CRA, CRB and IMR peripheral registers are cleared. CD/RD, CW/RW and CB/RB are unchanged.

DUAL TIMER INTERRUPT MANAGEMENT

The Dual Timer interrupts are controlled by the IMR register.


EN1
IN1 PIN HIGH TO LOW TRANSITION OR T COUNTER OVER FLOW
STB
OUT B
ZCA* TRANSITION
INTERRUPT REQUEST
ENB
ENA
FB
FA
INTERRUPT MANAGEMENT REGISTER
F1
CRA REGISTER

* : ZCA is the Zero crossing detector output (see Option 32)

FIGURE 13: INTERRUPT MANAGEMENT

The Dual Timer interrupt sources, one for each block, are software selectable through the enable bits ENA and ENB (IMR register). Each interrupt in turn sets the FA or FB flag in the IMR register when activated. F1 (bit 3 of CRA register) is set when a CPU interrupt is generated. These flags will be used in the interrupt subroutine to locate the interrupt source. These flags will have to be reset by the user interrupt subroutine. It should be noted that Timer B interrupt is disabled through the start bit STB, when this block is not running. The peripheral interrupt requests are enabled by the ENA bit, or by the ENB and STB bits. The block A interrupt source is mask option programmable: an interrupt request is generated either on each transition of the zero crossing detector output or on high to low transition of the zero crossing detector output. Block B interrupt requests are generated on each pulse on OUTB.

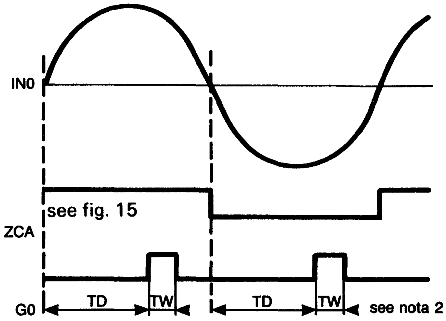
BLOCK A MODE DESCRIPTION

Remark: all the following timings are drawn in the case where I.V. (Initial value) of G0 is 0.

Triggered pulse generation

This mode outputs a pulse triggered by the zero-crossing of a signal at IN0. The delay from the zero-crossing is specified by the content of RD. The pulse width is specified by the content of RW.

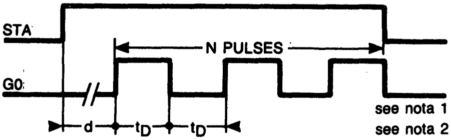
$t_D = D.\ t_{CLKA} + 2.5\ t_{CLK}$
$t_W = (W + 1).\ t_{CLKA}$
Where t_{CLK} = System clock period ;
t_{CLKA} = Block A clock period
D = Content of register RD.
W = Content of register RW
$1 \leqslant D \leqslant 255$
$0 \leqslant W \leqslant 255$



Nota 2: initial value of G0 can either be 1 or 0 (depending on mask option 3.4).

Pulse burst generation

This mode outputs on G0 a specified number of pulses of a specified width. The number of pulses is specified by the contents of register RW, while the pulse width is specified by the contents of register RD. The pulse burst is generated each time STA is set to a « 1 ». CD is automatically loaded from register RD, at each transition of G0. After the W pulses are transmitted, counter CW is automatically loaded from register RW, and STA is reset. Setting STA to 1 will repeat the previous sequence.

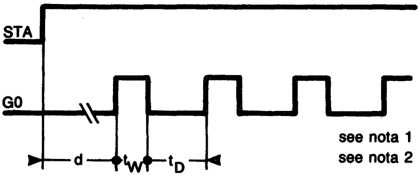


t_D = (D + 1). t_{CLKA}
t_{CLKA} = Block A clock period
N = (W + 1)
D = Content of register RD
W = Content of register RW
d: delay (see nota 1)
0 ⩽ D,W ⩽ 255

Nota 1: If counter CD is not loaded with a defined value before start up, d will be $t_{CLKA} \leqslant d \leqslant 255. t_{CLKA}$. Note that this undefined value of d is equivalent to an undefined delay in outputing the waveform. In order to control this delay time d, user should load counter CD (in addition to register RD).

Duty-cycle generation

This mode generates a rectangular waveform on G0. Pulse width is specified by the contents of register RW, inter-pulse length is specified by the content of register RD (be carefull with I.V: see nota 2)

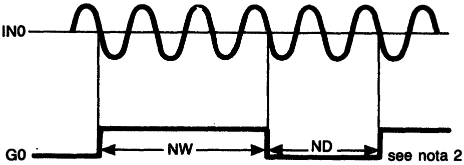


t_D = (D + 1). t_{CLKA} W = Content of register RW
t_W = (W + 1). t_{CLKA} d: delay (see nota 1)
D = Content of register RD 0 ⩽ D,W⩽ 255
t_{CLKA} = Block A clock period

Analog period counter

This mode generates a rectangular waveform on G0. The pulse width is specified by the contents of register RW, and inter-pulse length is specified by the contents of register RD.

The counter is not controlled by the clock of block A, but decrements each IN0 falling edge. G0 transition is synchronised on SK. Maximum delay between IN0 = 0V and G0 transition is therefore less than equal to one SK period (4 µs ⩽ t_{SK} ⩽ 10 µs).



ND = (D + 1)
NW = (W + 1)
Where W = Contents of register RW
D = Contents of register RD
0 ⩽ D,W ⩽ 255

Remark: analog signal frequency should not exceed $1/2tc$.
$F_{IN0} \leqslant 1/2tc$.

BLOCK B MODE DESCRIPTION

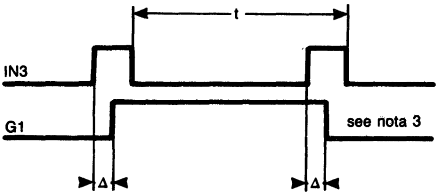
Remark: all the following timings are drawn in the case where I.V (initial value) fo G1 is 0.

Measurement mode

This mode is mask-option programmable: period measurement mode, or duty-cycle measurement mode.

Period measurement mode: This mode measures the period of an external waveform on IN3.

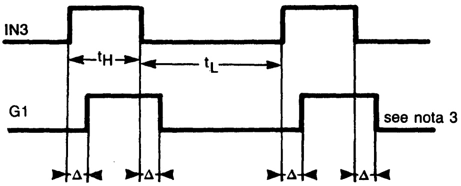
On the high to low transition, counter CB is transferred to register RB, and then cleared. As the counter is counting down from zero, the value read from the register is a two's complement value. The register can be read at any time, even when being changed: the data is always valid.



t = (N + 1) t_{CLKB}
Where t_{CLKB} = Block B clock period
N = Two's complement value of register RB
0 ⩽ N ⩽ 2^{16}-1
Δ = 1,5 x t_{CLK}
t_{CLK} = System clock period

The output G1 toggles each time a counter to register transfer is performed.

Duty-cycle measurement mode: From an external signal on IN3, the counter CB starts, counting the pulse width high. On the falling edge of IN3, CB is transferred to the register RB, and then cleared. Then, counter CB starts counting the IN3 inter-pulse length. On rising edge, CB is transfered to RB, CB is cleared, and duty cycle measurement starts again. G1 output toggles each time a counter to register transfer is performed. The user has to read the register contents before it is changed by the next measurement. This can be done by an interrupt, or by monitoring IN3 or G1.



t_H = (NH + 1). t_{CLKB}
t_L = (NL + 1). t_{CLKB}
t_{CLKB} = Block B clock perdiod
Δ = 1,5 x t_{CLK}

NL, NH = Two's complement value of register RB contents
0 ⩽ NL, NH ⩽ 2^{16}-1

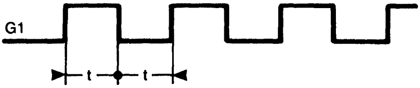
Nota 3: initial value of G1 can either be 1 or 0 (depending on mask option 35)

External event counter
In this mode, counter CB is an external events (or pulses) counter. There is no automatic clear of the counter: it counts down from whatever state it was. Thus, to ease the reading of the information, the counter should be preloaded. Preloading the counter with all zeroes will give the two's complement of the count, preloading the counter with all ones will give the one's complement of the count. The counter decrements on the falling edge of IN3 pin. The counter, counts down and eventually rolls over from 0000 to FFFF (Hex).

The register RB acts as a « capture-latch », being loaded by the counter's value every instruction cycle. Thus, it enables a simple monitoring of the count.

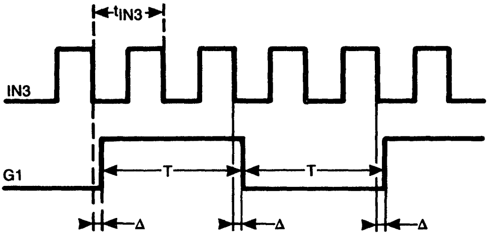
G1 toggles every time counter CB counts through « 0 ».

Frequency generation
This mode generates a square wave on G1. The period is determined by the contents of register RB. Using the autoreload procedure, the counter is only loaded when it counts down through zero. Therefore, it may be necessary to initially load the counter (see nota 1). IN3 has no effect in this mode.



t = (N + 1) t_{CLKB}; Period = 2 (N + 1) t_{CLKB}
Where N = Content of counter RB
0 ⩽ N ⩽ 2^{16}-1
t_{CLKB} = Block B clock period

Frequency divider
This mode generates a square wave on G1. The period is determined by the contents of register RB, and by the external frequency applied on IN3 pin. The counter decrements on the high to low transition of IN3 pin. Using the autoreload procedure, the counter is only loaded when it counts down through zero. Therefore, it may be necessary to initially load the counter.



Δ = 2,5 t_{CLK}
t = (N + 1) t_{IN3} ; Period = 2 (N + 1) t_{IN3}
Where N = contents of counter RB
O ⩽ N ⩽ 2^{16}-1
t_{CLK} = System clock period

Symbol	Definition
Internal Architecture Symbols	
A	4-bit Accumulator
B	8-bit RAM address register
Br	Upper 4 bits of B (register address)
Bd	Lower 4 bits of B (digit address)
C	1-bit Carry register
D	4-bit Data output port
EN	4-bit Enable register
G	4-bit General purpose I/O port
IL	One 1-bit (IN$_3$) latch
IN	4-bit input port
L	8-bit 3-state I/O port
M	4-bit contents of RAM addressed by B
PC	11-bit ROM address program counter
Q	8-bit latch for L port
SA, SB, SC	11-bit 3-level subroutine stack
SIO	4 bit Shift register
SK	Logic-controlled clock output
SKL	1-bit lach for SK output
T	8-bit timer

Instruction operand symbols	
d	4-bit operand field, 0-15 binary (RAM digit select)
r	3-bit operand field, 0-7 (RAM register select)
a	11-bit operand field, 0-2047 (1023)
y	4-bit Operand field, 0-15 (immediate data)
RAM(x)	RAM addressed by variable x
ROM(x)	ROM addressed by variable x

Operational Symbols	
+	Plus
—	Minus
→	Replaces
↔	is exchanged with
=	is equal to
$\bar{A}$	one's complement of A
⊕	exclusive-or
:	range of values

TS94100 INSTRUCTION SET

Mnemonic	Operand	Hx Code	Machine Language Code (Binary)	Data Flow	Skip Conditions	Description
ARITHMETIC INSTRUCTIONS						
ASC		30	0 0 1 1 | 0 0 0 0	A+C+RAM(B) → A Carry → C	Carry	Add with carry, Skip on Carry
ADD		31	0 0 1 1 | 0 0 0 1	A+RAM(B)→ A	None	Add RAM to A
ADT		4A	0 1 0 0 | 1 0 1 0	A+10$_{10}$→ A	None	Add TEN to A
AISC	y	5—	0 1 0 1 | y	A+y → A	Carry	Add Immediate Skip on Carry (y ≠ 0)
CASC		10	0 0 0 1 | 0 0 0 0	$\bar{A}$+RAM(B)+C → A Carry → C	Carry	Complement and Add with Carry, Skip on Carry
CLRA		00	0 0 0 0 | 0 0 0 0	0 → A	None	Clear A
COMP		40	0 1 0 0 | 0 0 0 0	$\bar{A}$ → A	None	Ones complement of A to A
NOP		44	0 1 0 0 | 0 1 0 0	None	None	No Operation
RC		32	0 0 1 1 | 0 0 1 0	"0" → C	none	Reset C
SC		22	0 0 1 0 | 0 0 1 0	"1" → C	None	Set C
XOR		02	0 0 0 0 | 0 0 1 0	A⊕ RAM(B) → A	None	Exclusive-OR RAM with A

INSTRUCTION SET (continued)

MEMORY REFERENCE INSTRUCTIONS

Mnemonic	Operand	Hx Code	Machine Language Code (Binary)	Data Flow	Skip Conditions	Description
CAMT		33 3F	0011 0011 0011 1111	A → $T_{7:4}$ RAM(B) → $T_{3:0}$	None	Copy A, RAM to T
CTMA		33 2F	0011 0011 0010 1111	$T_{7:4}$ → RAM(B) $T_{3:0}$ → A	None	Copy T to RAM, A
CAMQ		33 3C	0011 0011 0011 1100	A → $Q_{7:4}$ RAM(B) → $Q_{3:0}$	None	Copy A, RAM to Q
CQMA		33 2C	0011 0011 0010 1100	$Q_{7:4}$ → RAM(B) $Q_{3:0}$ → A	None	Copy Q to RAM, A
LD	r	—5	00 r 0101 (r = 0,3)	RAM(B) → A Br ⊕ r → Br	None	Load RAM into A Exclusive-OR Br with r
LDD	r.d	23	0010 0011 0 r d	RAM(r,d) → A	None	Load A with RAM pointed to directly by r, d
LQID		BF	1011 1111	ROM($PC_{10:8}$,A,M) → Q	None	Load Q indirect (Note 3)
RMB	0 1 2 3	40 45 42 43	0100 1100 0100 0101 0100 0010 0100 0011	0 → $RAM(B)_0$ 0 → $RAM(B)_1$ 0 → $RAM(B)_2$ 0 → $RAM(B)_3$	None	Reset RAM Bit
SMB	0 1 2 3	4D 47 46 4B	0100 1101 0100 0111 0100 0110 0100 1011	1 → $RAM(B)_0$ 1 → $RAM(B)_1$ 1 → $RAM(B)_2$ 1 → $RAM(B)_3$	None	Set RAM Bit
STII	y	7—	0111 y	y → RAM(B) Bd + 1 → Bd	None	Store Memory immediate and increment Bd
TCRA		23 8B	0010 0011 1000 1011	CD → RD 1111 → A CW → RW	None	Transfer Counter to Register (Timer A) (Note 8)
TCRB		23 9B	0010 0011 1001 1011	CB → RB 1111 → A	None	Transfer Counter to Register (Timer B) (Note 8)
TMB		33 35	0011 0011 0011 0101	$\overline{Br3}$ → B_r3	None	One's complement of B_r3 to B_r3 (Toggle memory bank)
TRCA		23 8A	0010 0011 1000 1010	RD → CD 1111 → A RW → CW	None	Transfer Register to Counter (Timer A) (Note 8)
TRCB		23 9A	0010 0011 1001 1010	RB → CB 1111 → A	None	Transfer Register to Counter (Timer B) (Note 8)
X	r	—6	00 r 0110 (r = 0,3)	RAM(B) ↔ A Br ⊕ r → Br	None	Exchange RAM with A Exclusive OR Br with r
XAD	rd	23	0010 0011 1 r d	RAM(r,d) ↔ A	None	Exchange A with RAM pointed to directly by r d
XDS	r	—7	00 r 0111 (r = 0,3)	RAM(B) ↔ A Bd-1 → Bd Br ⊕ r → Br	Bd decrements past 0	Exchange RAM with A and Decrement Bd . Exclusive Or Br with r
XIS	r	—4	00 r 0100 (r = 0,3)	RAM(B) ↔ A Bd + 1 → Bd Br ⊕ r → Br	Bd increments past 15	Exchange RAM with A and increment Bd Exclusive Or Br with r

INSTRUCTION SET (continued)

Mnemonic	Operand	Hx Code	Machine Language Code (Binary)	Data Flow	Skip Conditions	Description
TRANSFER OF CONTROL INSTRUCTIONS						
JID		FF	1 1 1 1 | 1 1 1 1	ROM($PC_{10:8}$A,M) → $PC_{7:0}$	None	Jump Indirect (Note 3)
JMP	a	6 - - -	0 1 1 0 | 0 | $a_{10:8}$ $a_{7:0}$	a → PC	None	Jump
JP	a	- - - -	1 | $a_{6:0}$ (pages 2, 3, only) or 1 1 | $a_{5:0}$ (all other pages)	a → $PC_{6:0}$ a → $PC_{5:0}$	None	Jump within Page (Note 4)
JSRP	a	- -	1 0 | $a_{5:0}$	PC+1 → SA → SB → SC 00010 → $PC_{10:6}$ a → $PC_{5:0}$	None	Jump to Subroutine Page (Note 5)
JSR	a	6 -	0 1 1 0 | 1 | $a_{10:8}$ $a_{7:0}$	PC+1 → SA → SB → SC a → PC	None	Jump to Subroutine
RET		48	0 1 0 0 | 1 0 0 0	SC → SB → SA → PC	None	Return from Subroutine
RETSK		49	0 1 0 0 | 1 0 0 1	SC → SB → SA → PC	Always Skip on Return	Return from Subroutine then Skip
REGISTER REFERENCE INSTRUCTIONS						
CAB		50	0 1 0 1 | 0 0 0 0	A → Bd	None	Copy A to Bd
CBA		4E	0 1 0 0 | 1 1 1 0	Bd → A	None	Copy Bd to A
LBI	r,d	- - 33 - -	0 0 | r | (d-1) (r=0:3) (d=0,9:15) or 0 0 1 1 | 0 0 1 1 1 | r | d (any r, any d)	r,d → B	Skip until not a LBI	Load B immediate with r,d (Note 6)
LEI	y	33 6 -	0 0 1 1 | 0 0 1 1 0 1 1 0 | y	y → EN	None	Load EN Immediate (Note 7)
XABR		12	0 0 0 1 | 0 0 1 0	A↔ Br 0 → A_3	None	Exchange A with Br
TEST INSTRUCTIONS						
SKC		20	0 0 1 0 | 0 0 0 0		C="1"	Skip if C is True
SKE		21	0 0 1 0 | 0 0 0 1		A=RAM(B)	Skip if A Equals RAM
SKGZ		33 21	0 0 1 1 | 0 0 1 1 0 0 1 0 | 0 0 0 1		$G_{3:0}$=0	Skip if G is Zero (all 4 bits)
SKGBZ		33	0 0 1 1 | 0 0 1 1	1st byte		Skip if G Bit is Zero
	0	01	0 0 0 0 | 0 0 0 1	2nd byte	G_0=0	
	1	11	0 0 0 1 | 0 0 0 1		G_1=0	
	2	03	0 0 0 0 | 0 0 1 1		G_2=0	
	3	13	0 0 0 1 | 0 0 1 1		G_3=0	

INSTRUCTION SET (continued)

Mnemonic	Operand	Hx Code	Machine Language Code (Binary)	Data Flow	Skip Conditions	Description
SKMBZ	0	01	0000 0001		RAM $(B)_0 = 0$	Skip if RAM Bit is Zero
	1	11	0001 0001		RAM $(B)_1 = 0$	
	2	03	0000 0011		RAM $(B)_2 = 0$	
	3	13	0001 0011		RAM $(B)_3 = 0$	
SKT		41	0100 0001		A time-base counter carry has occured since last test	Skip on Timer (Note 3)

INPUT/OUTPUT INSTRUCTIONS

ING		33 2A	0011 0011 0010 1010	G → A	None	Input G Ports to A (Note 9)
ININ		33 28	0011 0011 0010 1000	IN3, IN2, IN1, "0"→A	None	Input IN inputs to A
INIL		33 29	0011 0011 0010 1001	IL_3, CKO, "0", "0" → A	None	Input IL_3 Latch to A (Note 3)
INL		33 2E	0011 0011 0010 1110	$L_{7:4}$→ RAM(B) $L_{3:0}$→ A	None	Input L Ports to RAM, A
OBD		33 3E	0011 0011 0011 1110	Bd → D	None	Output Bd to D Outputs
OGI	y	33 5 -	0011 0011 0101 y	Y2→G2 Y3→G3	None	Output to G Ports Immediate
OMG		33 3A	0011 0011 0011 1010	$RAM(B)_2 \rightarrow G_2$ $RAM(B)_3 \rightarrow G_3$	None	Output RAM to G Pórts
XAS		4F	0100 1111	A ↔ SIO, C → SKL	None	Exchange A with SIO (Note 3)

Note 1 : All subscripts for alphabetical symbols indicate bit numbers unless explicitly defined (e.g., Br and Bd are explicitly defined). Bits are numbered 0 to N where 0 signifies the least significant bit (low-order, right-most bit). For example, A_3 indicates the most significant (left-most) bit of the 4-bit A register.

Note 3 : For additional on the operation of the XAS, JID, LQID, INIL, and SKT instructions, see below.

Note 4 : The JP instruction allows a jump, while in subroutine page 2 or 3, to any ROM location within the two-page boundary of pages 2 or 3. The JP instruction, otherwise, permits a jump to a ROM location within the current 64-word page. JP may not jump to the last word of a page.

Note 5 : A JSRP transfers program control to subroutine page 2 (0010 is loaded into the upper 4 bits of P). A JSRP may not be used when in pages 2 or 3. JSRP may not jump to the last word in page 2.

Note 6 : LBI is a single-byte instruction if d=0, 9, 10, 11, 12, 13, 14, or 15. The machine code for the lower 4 bits equals the binary value of the "d" data minus 1, e.g., to load the lower four bits of B (Bd) with the value 9(1001$_2$), the lower 4 bits of the LBI instruction equal 8 (1000$_2$). To load 0, the lower 4 bits of the LBI instruction should equal 15 (1111$_2$).

Note 7 : Machine code for operand field y for LEI instruction should equal the binary value to be latched into EN, where a "1" or "0" in each bit of EN corresponds with the selection or deselection of a particular function associated with each bit. (See Functional Description, EN Register).

Note 8: All transfer instructions (TRCA, TCRA, TRCB, TCRB) must be selected when $B_r3 = 1$. If $B_r3 = 0$, these transfer instructions will be decoded as XAD instruction.

Note 9: G0 and G1 can be used as inputs only with options 34.1 and 35.2.

DESCRIPTION OF SELECTED INSTRUCTIONS

XAS Instruction

XAS (Exchange A with SIO) copies C to the SKL latch and exchanges the accumulator with the 4-bit contents of the SIO serial-in/serial-out shift register. An XAS instruction can be performed once every 4 instruction cycles to effect a continuous data stream.

LQID Instruction

LQID (Load Q Indirect) loads the 8-bit Q register with the contents of ROM pointed to by the 11-bit word PC$_{10}$: PC$_8$, A,M. LQID can be used for table lookup or code conversion such as BCD to seven-segment. The LQID instruction « pushes » the stack (PC + 1 → SA → SB → SC) and replaces the least significant 8 bits of the PC as follows: A → PC$_{7:4}$, RAM$_{(B)}$ → PC$_{3:0}$, leaving PC$_{10}$, PC$_9$ and PC$_8$ unchanged. The ROM data pointed to by the new address is fetched and loaded into the Q latches. Next, the stack is « popped » (SC → SB → SA → PC), restoring the saved value of PC to continue sequential program execution. Since LQID pushes SB into SC, the previous contents of SC are lost.

Note: LQID uses 2 instruction cycles if executed, one if skipped.

JID Instruction

JID (Jump Indirect) is an indirect addressing instruction, transferring program control to a new ROM location pointed to indirectly by A and M. It loads the lower 8 bits of the ROM address register PC with the contents of ROM addressed by the 11-bit word, PC$_{10:8}$, A,M. PC$_{10}$, PC$_9$ and PC$_8$ are not affected by JID.

Note: JID uses 2 instruction cycles if executed, one if skipped.

SKT Instruction

The SKT (Skip On Timer) instruction tests the state of the T counter overflow latch (see internal logic, above), executing the next program instruction if the latch is not set. If the latch has been set since the previous test, the next program instruction is skipped and the latch is reset. The features associated with this instruction allow the processor to generate its own time-base for real-time processing, rather than relying on an external input signal.

SKT can be used (when the option selected for CPU interrupt source is the timer overflow) during the interrupt service routine to clear the overflow latch.

INIL Instruction

INIL (Input IL Latch to A) inputs latch IL3, CKO and 0 into A. The IL3 latch is set if a low-going pulse (« 1 » to « 0 ») has occured on the IN3 input since the last INIL instruction, provided the input pulse stays low for at least two instruction cycles. Execution of an INIL inputs IL 3 into A3, and resets this latch to allow it to respond to subsequent low-going pulses on the IN3. If CKO is mask programmed as a general purpose input, an INIL will input the state of CKO into A2. If CKO has not been so programmed, a « 1 » will be placed in A2. A1 and A0 are cleared. IL latch is cleared on reset.

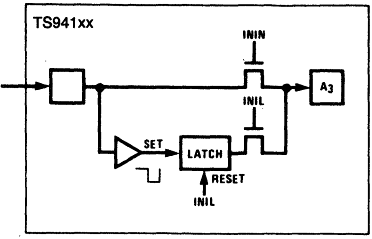


FIGURE 14: INIL HARDWARE IMPLEMENTATION

RET and RETSK Instructions

When a RET instruction is executed at the end of the interrupt subroutine, it restores the state of bit Br3 that was saved at the beginning of the interrupt subroutine. When used at the end of a software subroutine, a RET instruction will clear bit Br3. The RETSK instruction never modifies bit Br3.

Instruction set notes

- The first word of a program (ROM address 0) must be a CLRA (Clear A) instruction.
- Although skipped instructions are not executed, they are still fetched from the program memory. Thus program paths take the same number of cycles whether instructions are skipped or executed except for JID, and LQID.
- The ROM is organized into pages of 64 words each. The Program Counter is a 11-bit binary counter, and will count through page boundaries. If a JP is the last word of a page, it operates as if it were in the next page. For example, a JP Located in the last word of a page will jump to a location in the next page. JID or LQID located in the last word of every fourth page (i.e. hex address 0FF, 1FF, 2FF, 3FF, 4FF, etc.) will access data in the next group of four pages.

Note: TS94120 needs only 10 bits to address its ROM. Therefore, the eleventh bit (10) is ignored.

MAXIMUM RATINGS

RATING	SYMBOL	VALUE	UNIT
Input voltage IN0 SI	VI	— —0.8 to +10 —0.5 to +14	V
Voltage on any other Pin		—0.5 to +10	
Ambient operating temperature TS94100CP TS94100TP	T_A	0 to +70 —40 to +85	°C °C
Ambient storage temperature	T_{stg}	—65 to + 150	°C
Lead temperature (soldering 10s)		300	°C
Power dissipation T_A = +25°C T_A = +70°C	P_D	1 550	W mW
Total source current		36	mA
Total sink current		200	mA

D.C. AND OPERATING CHARACTERISTICS

T_A = 0°C to +70°C V_{CC} = 4.5V to 5.5V (Unless otherwise specifed)

Characteristic	Symbol	Min	Typ	Max	Unit
Power supply ripple (peak to peak) - Note A	V_{ppl}	—	—	0.5	V
Operating supply current (all inputs set to 1) (all outputs set to 1) (all inputs set to 1) (all outputs set to 0)	I_{CC}	— —	— —	15 15.5	mA

INPUT

Characteristic	Symbol	Min	Typ	Max	Unit
CKI input level Crystal or RC Logic high Logic low External clock Logic high Logic low	V_{IH} V_{IL} V_{IH} V_{IL}	0.5V_{CC} — 0.7V_{CC} —	— — — —	— 1 — 0.2	V V
$\overline{\text{RESET}}$ and WATCHDOG (D3) SCHMITT trigger input level Logic high Logic low	V_{IH} V_{IL}	0.7V_{CC} —	— —	— 0.6	V
IN0 offset voltage (note B, Fig. 15)		—	—	150	mV
Input level on all other inputs With TTL option selected Logic high Logic low With high voltage option selected Logic high Logic low	V_{IH} V_{IL} V_{IH} V_{IL}	2 — 3.6 —	— — — —	— 0.8 — 1.2	V V V V
Input capacitance	C_I	—	—	7	pF
High impedance input leakage	I_I	—2	—	+2	µA

OUTPUT

Characteristic	Symbol	Min	Typ	Max	Unit
TTL load Logic high (I_{OH} = —100µA) Logic low (I_{OL} = 1.6 mA)	V_{OH} V_{OL}	2.4 —	— —	— 0.4	V V
CMOS load Logic high (I_{OH} = —10µA) Logic low (I_{OL} = +10µA)	V_{OH} V_{OL}	0.7V_{CC} —	— —	— 0.2	V V

OUPUT SINK CURRENT

Characteristic	Symbol	Min	Typ	Max	Unit
SO,SK and L0-7 outputs with TTL option					
(V_{CC} = 5.5V, V_{OL} = 0.4V)	I_{OL}	2	4.6	10	mA
(V_{CC} = 4.5V, V_{OL} = 0.4V)	I_{OL}	1.6	4	9	mA
G1-G3, D0-D3, L0-L7 outputs with very high current option					
(V_{CC} = 5.5V, V_{OL} = 1.0V)	I_{OL}	17	28	40	mA
(V_{CC} = 4.5V, V_{OL} = 1.0V)	I_{OL}	15	25	40	mA
G0 output (Triac driver)					
V_{CC} = 5.5V, V_{OL} = 2.0V	I_{OL}	50	70	90	mA
V_{CC} = 4.5V, V_{OL} = 1.0V	I_{OL}	28	40	46	mA
CKI (Single-pin RC oscillator)					
V_{CC} = 4.5V, V_{IH} = 3.5V	I_{IH}	2	4,6	10	mA

OUTPUT SOURCE CURRENT

G2, G3, D0-D3 standard output and SO, SK output with TTL option					
V_{CC} = 5.5V, V_{OH} = 2.0V	I_{OH}	—250	—700	—1900	μA
V_{CC} = 4.5V, V_{OH} = 2.0V	I_{OH}	—100	—350	—1000	μA
SO, SK outputs with high current option, G0, G1 and L0-L7 outputs					
V_{CC} = 5.5V, V_{OH} = 2.0V	I_{OH}	—1.2	—2.8	—7	mA
V_{CC} = 4.5V, V_{OH} = 1.0V	I_{OH}	—1.2	—2.8	—7	mA
Input load source current					
V_{CC} = 5.5V, V_{IL} = 0V	I_{IL}	—11	—45	—150	μA
V_{CC} = 4.5V, V_{IL} = 0V	I_{IL}	—10	—45	—150	μA

Characteristic	Symbol	Min	Typ	Max	Unit
3-State Output leakage current (Available on L only)	I_L	—	—5	+5	μA
Total Sink Current allowed					
All outputs combined	ΣI_{sink}	—	—	200	mA
D, G Ports		—	—	120	
L Port		—	—	160	
each L0-L7, G1-G3, D0-D3 pin		—	—	30	
G0 pin		—	—	60	
Each other pin		—	—	5	
Total Source Current allowed					
All I/O combined	ΣI_{source}	—	—	36	mA
Each L0-L7, G0, G1, SO, SK pin		—	—	4	
All other pins combined		—	—	4	

Note A: V_{CC} change must be less than 0.5V in a 1ms period to maintain proper operations.

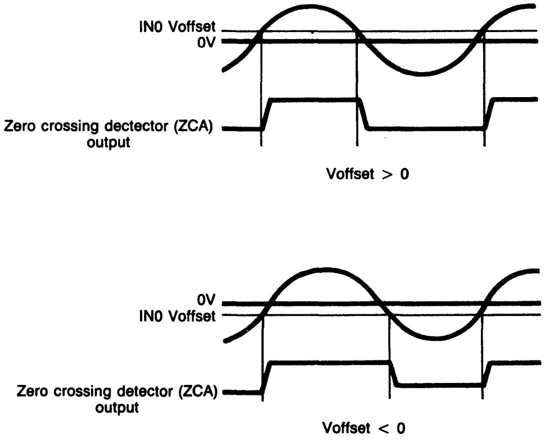
Note B: IN0 Offset Voltage is the absolute value of the difference between the voltage at IN0 and GROUND.

MCU AND DUAL TIMER ELECTRICAL CHARACTERISTICS
0°C ⩽ T_A ⩽ 70°C ; 4.5V ⩽ V_{CC} ⩽ 5.5V (Unless otherwise specified)

Characteristic		Symbol	Min	Typ	Max	Unit
Instruction cycle time						
XTAL (Fig. 16)	+ 20 (option 4.2)	t_c	4	—	11	µs
	+ 10 (option 4.1)	t_c	4	—	11	µs
	+ 5 (option 4.0)	t_c	5	—	11	µs
RC Oscillator	+ 10 (option 4.1)	t_c	4	—	11	µs
External clock	+ 10 (option 4.1)					
Rise time, fall time on CKI input (External waveform generator on CKI)	+ 10 mode	t_r, t_f	—	—	50	ns
CKI duty cycle (Fig. 17)			0.4	—	0.6	
Input set up time	All inputs	$t_{set\ up}$	t_c/10 + 1.3	—	—	µs
Input hold time	All inputs	t_{hold}	100	—	—	ns
Output propagation delay (Load = one TTL load)						
SK (EN0 = 0, SK = 1)		t_{s1}, t_{s0}	—	—	2	µs
SK as an output (SK = 0)		t_{pd1}, t_{pd0}	$-t_c/10$	—	0.6	µs
SO as an output (EN0, EN3) = (00, 10, 11)		t_{pd1}; t_{pd0}	$-t_c/10$	—	0.6	µs
SO as SIO output (EN0, EN3) = 01		t_{pd1}, t_{pd0}	$t_c/10$	—	$t_c/5$ + 0.6	µs
All other outputs		t_{pd1}, t_{pd0}	$-t_c/10$	—	1.0	µs
G0, G1 rise time		t_{LH}	—	—	1.0	µs
G0, G1 fall time		t_{HL}	—	—	200	ns

Dual timer

System clock (Fig. 19)		f_{CLK}	450	—	1250	kHz
Timer A and timer B Prescaler	+ 1	f_{CLKA}, f_{CLKB}	450	—	1250	kHz
Prescaler	+ 5	f_{CLKA}, f_{CLKB}	90	—	250	kHz
Prescaler	+ 20	f_{CLKA}, f_{CLKB}	22,5	—	62,5	kHz
Prescaler	+ 40	f_{CLKA}, f_{CLKB}	11250	—	31250	Hz
IN0, IN3 INPUT						
All modes		f_{IN0}, f_{IN3}	0	—	$f_{CLK}/2$	—
IN0 in analog period counter mode			0	—	$1/2t_c$	—



Voffset > 0

Voffset < 0

FIGURE 15: ZERO CROSSING DETECTOR OUTPUT

Characteristic	Symbol	Min	Typ	Max	Unit
External C value	C	250	—	—	pF
External R value (C = 250pF)	R	4	—	8	kΩ
External C values					
f_{CKI} = 5 MHz	C1	—	15	—	pF
	C2	—	15	—	pF
f_{CKI} = 455 kHz	C1	—	30	—	pF
	C2	—	30	—	pF

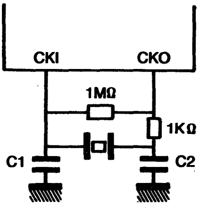


FIGURE 16

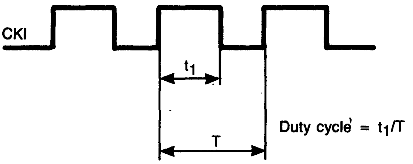
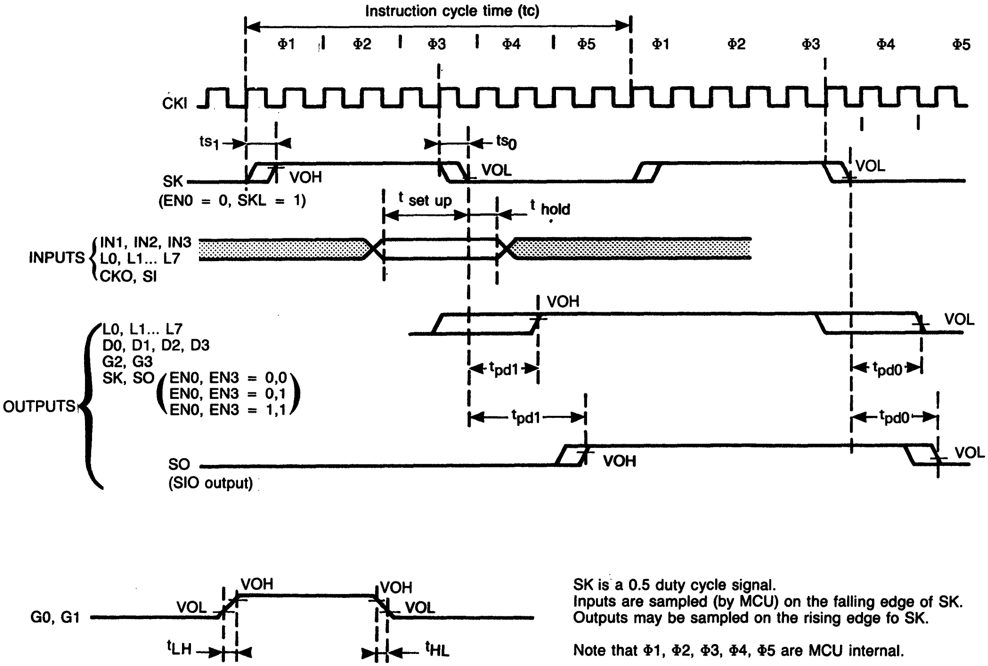


FIGURE 17



SK is a 0.5 duty cycle signal.
Inputs are sampled (by MCU) on the falling edge of SK.
Outputs may be sampled on the rising edge fo SK.

Note that Φ1, Φ2, Φ3, Φ4, Φ5 are MCU internal.

FIGURE 18: I/O TIMINGS (÷ 10 MODE)

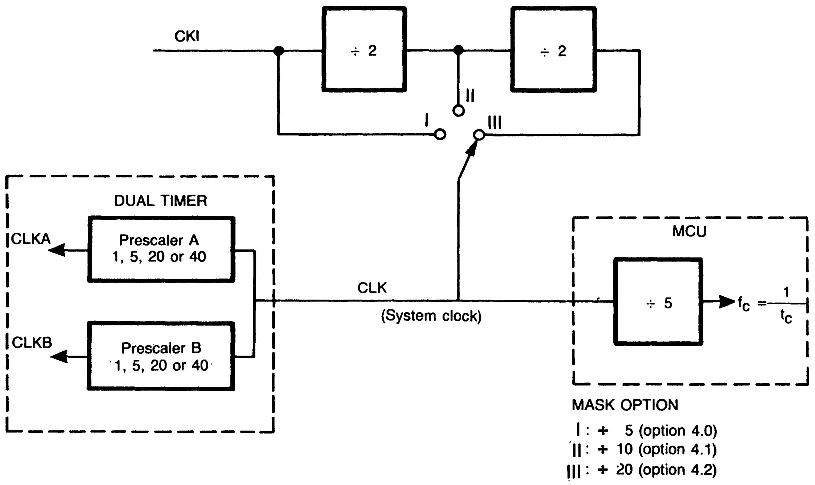


FIGURE 19: PRESCALERS MASK-OPTIONS

I/O OPTIONS

TS94100 outputs have the following optional configurations illustrated by figure 20:

a. General purpose - an enhancement-mode device to ground in conjunction with a depletion-mode paralleled by an enhancement-mode device to Vcc.
On G2-G3 and SO-SK with TTL option, this configuration is compatible with TTL and CMOS input requirements. On G1 and SO-SK with high current option, it allows fast rising and falling edges when driving capacitive loads. On G0, the enhancement-mode device to ground meets the typical current sinking requirements to directly trigger a triac.

b. Open-drain - an enhancement mode device to ground only, allowing external pull-up, as required by the user's application. Available on SO, SK, D, G2-G3 outputs.

c. 3-state general purpose - same as **a.** but may be disabled.
The sinking devices meet the typical current sinking requirements of a LED display. These devices may be turned off under program control (see functional description, EN register) in a high-impedance state to provide required LED segments blanking for a multiplexed display. Available on L outputs only.

d. Open-drain L - same as **b.** but may be disabled - available on L only.

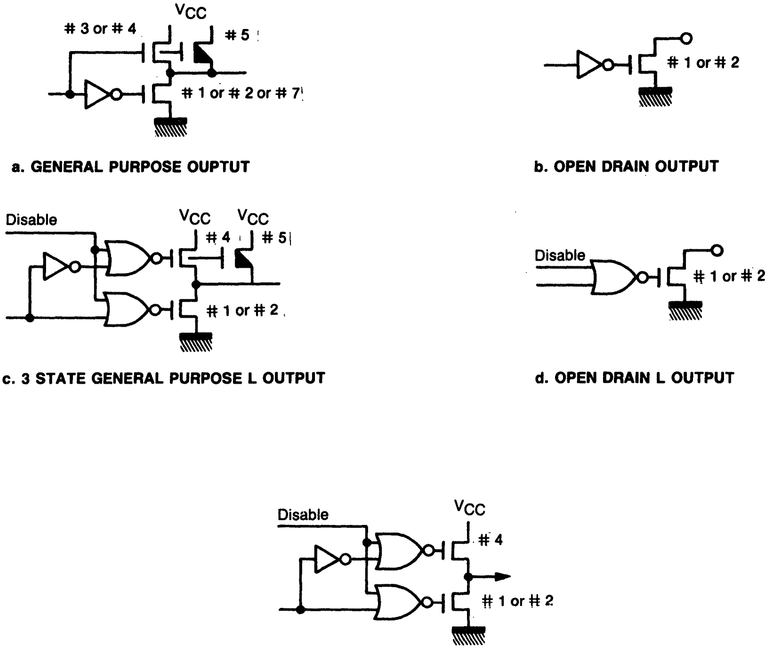
e. 3-state push-pull - an enhancement-mode device to ground and V_{CC}. Available on L outputs only.

TS94100 inputs have the following optional configurations:

f. An on-chip depletion load device to Vcc.

g. A Hi-Z input which must be driven to a « 1 » or « 0 » by external components.

OUTPUT CONFIGURATIONS



a. GENERAL PURPOSE OUPTUT

b. OPEN DRAIN OUTPUT

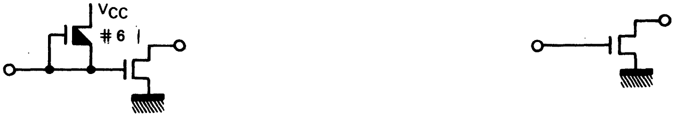
c. 3 STATE GENERAL PURPOSE L OUTPUT

d. OPEN DRAIN L OUTPUT

e. 3-STATE PUSH-PULL L OUTPUT

FIGURE 20

INPUT CONFIGURATIONS



f. INTPUT WITH LOAD

g. Hi-Z INPUT

FIGURE 21

The above input and output configurations share common enhancement-mode and depletion-mode devices. Specifically, all configurations use one or more of 7 devices (numbered # 1 to # 7 respectively).

The SO, SK, D, G2-G3 outputs can be configured as shown in **a, b.**

The G0-G1 are configured as general purpose output, as shown in **a.**

Note that when inputing data to the G ports, the G outputs should be set to a « 1 ». The L outputs can be configured as shown in **c, d, e.**

An important point to remember if using configuration **c** with the L drivers is that even when the L drivers are disabled, the depletion load device will source a small amount of current (see figure 20 device 5) ; however, when the L lines are used as inputs, the disabled depletion device cannot source a sufficient current to pull up an external input to a logic 1.

IN0 is configured as an Hi-Z input.

Output	Output scheme	Mask option	Transistor type					
			1	2	3	4	5	7
SO, SK	a	opt. 9-0	•		•		•	
	a	opt. 9-1	•			•	•	
	b	opt. 9-2	•					
G2, G3	a	opt. 20-0		•	•		•	
D0, D1, D2, D3	b	opt. 20-1		•				
G0	a					•	•	•
G1	a			•		•	•	
L0, L1... L7	c	opt. 11-0	•			•	•	
	c	opt. 11-1		•		•	•	
	d	opt. 11-2	•					
	d	opt. 11-3		•				
	e	opt. 11-4	•			•		
	e	opt. 11-5		•		•		

OPTION LIST

OPTION 1: CKO Pin configuration
0: clock generator output to crystal/resonator
1: general purpose input, load device to Vcc
2: general purpose input, high-Z

OPTION 2: CKO intput levels
0: standard TTL input levels (0 = 0.8V, 1 = 2.0V)
1 : higher voltage input levels (0 = 1.2V, 1 = 3.6V)

OPTION 3: CKI input configuration
0: single pin RC controlled oscillator (option 4 must be + 10)
1: crystal input
2: external clock (option 4 must be + 10)

OPTION 4: CKI divider
0: CKI frequency divided by 5
1: CKI frequency divided by 10
2: CKI frequency divided by 20

OPTION 5: Reset pin configuration
0: load device to Vcc
1: high-Z input

OPTION 6: IN input configuration (IN1-IN3)
0: load device to Vcc
1: high-Z input

OPTION 7: IN input levels (IN1-IN3)
0: standard TTL input levels (0 = 0.8V, 1 = 2.0V)
1: higher voltage input levels (0 = 1.2V, 1 = 3.6V)
* **Note:** IN0 has no option

OPTION 8: SI input levels
0: standard TTL input levels (0 = 0.8V, 1 = 2.0V)
1: higher voltage input levels (0 = 1.2V, 1 = 3.6V)

OPTION 9: SO output configuration
0: general purpose TTL output
1: general purpose high current output
2: open drain output

OPTION 10: SK output configuration
same as OPTION 9

OPTION 11: L0 output driver
0: 3-state general purpose TTL output
1: 3-state general purpose very high current output
2: open drain TTL output
3: open drain very high current output
4: 3-state push-pull TTL output
5: 3-state push-pull very high current output

OPTION 12: L1 output driver
same as OPTION 11

OPTION 13: L2 output driver
same as OPTION 11

OPTION 14: L3 output driver
same as OPTION 11

OPTION 15: L4 output driver
same as OPTION 11

OPTION 16: L5 output driver
same as OPTION 11

OPTION 17: L6 output driver
same as OPTION 11

OPTION 18: L7 output driver
same as OPTION 11

OPTION 19: L input level L0-L7
0: standard TTL input levels (0 = 0.8V, 1 = 2.0V)
1: higher voltage input levels (0 = 1.2V, 1 = 3.6V)

OPTION 20: G2 output driver
0: general purpose output
1: open drain output

OPTION 21: G3 output driver
same as OPTION 20

OPTION 22: G input level (G0-G3)
0: standard TTL input levels (0 = 0.8V, 1 = 2V)
1: higher voltage input levels (0 = 1.2V, 1 = 3.6V)

OPTION 23: DO output configuration
same as OPTION 20

OPTION 24: D1 output configuration
same as OPTION 20

OPTION 25: D2 output configuration
same as OPTION 20

OPTION 26: D3 output configuration
same as OPTION 20

OPTION 27: CPU interrupt source
0: external interrupt
1: timer overflow

OPTION 28: timer input
0: prescaler output
1: event counter

OPTION 29: watchdog
0: no watchdog
1: D3 pin as watchdog input (OPTION 26 must be open drain)

OPTION 30: Dual timer block A clock (CLKA)
0: system clock (CLK) divided by 1
1: system clock (CLK) divided by 5
2: system clock (CLK) divided by 20
3: system clock (CLK) divided by 40

OPTION 31: Dual timer block B clock (CLKB)
0: system clock (CLK) divided by 1
1: system clock (CLK) divided by 5
2: system clock (CLK) divided by 20
3: system clock (CLK) divided by 40

OPTION 32: Dual timer interrupt source (block A)
0: zero crossing detector output falling edge (ZCA)
1: zero crossing detector output rising edge and falling edge (ZCA)

OPTION 33: Mode selection for MRB0 = 0 and MRB1 = 0
0: period measurement
1: duty cycle measurement

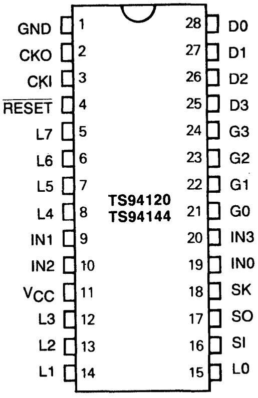
OPTION 34: G0 output for STA = 0
0: logical 0
1: logical 1

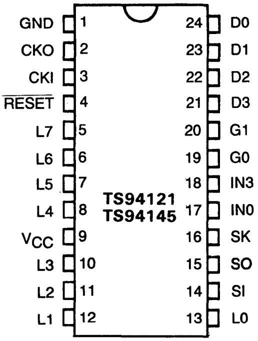
OPTION 35: G1 output for STB = 0 or when G1 is disabled (see CSB on fig. 14)
0: logical 0
1: logical 1

PIN ASSIGNMENTS

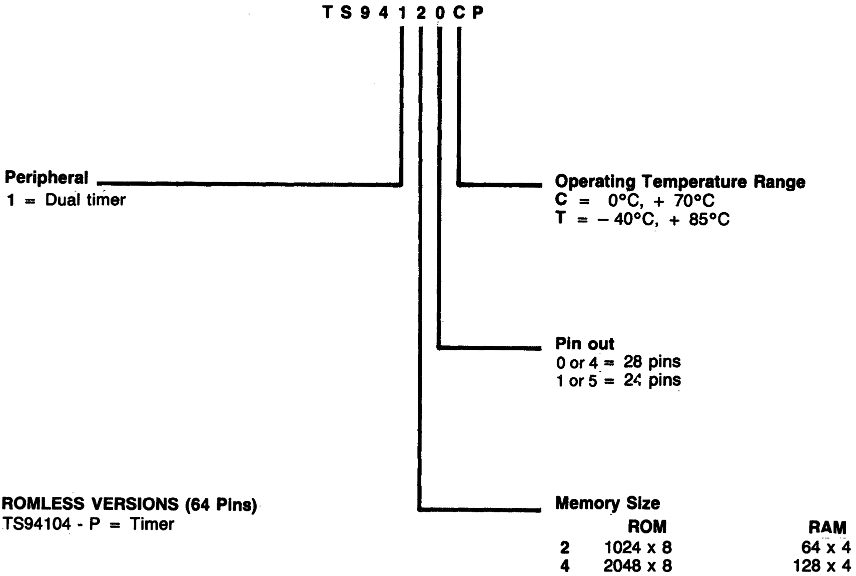
PIN NAMES

L7 - L0	8 bi-directionnal I/O ports with 3-state output
G3 - G2	2 bi-directionnal I/O ports
G0 - G1	2 dual timer peripheral outputs or 2 general purpose inputs
IN0	Dual timer peripheral analog input (Zero crossing detector input)
IN1-IN2	2 general purpose inputs
IN3	General purpose or dual timer peripheral input
SI	Serial input
SO	Serial output (or general purpose output)
D0 - D2	3 general purpose outputs
D3	General purpose output (or watch dog input)
SK	Logic controlled clock (or general purpose output)
CKI	Oscillator input
CKO	Oscillator output (or general purpose input)
$\overline{\text{RESET}}$	RESET input
Vcc	Power supply
GND	Ground





ORDERING INFORMATION



TS94120CP

Peripheral
1 = Dual timer

Operating Temperature Range
C = 0°C, + 70°C
T = – 40°C, + 85°C

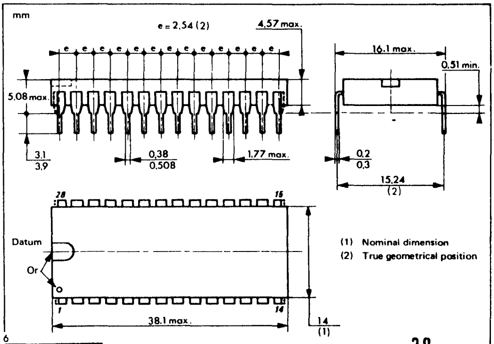
Pin out
0 or 4 = 28 pins
1 or 5 = 24 pins

ROMLESS VERSIONS (64 Pins)
TS94104 - P = Timer

Memory Size

	ROM	RAM
2	1024 x 8	64 x 4
4	2048 x 8	128 x 4

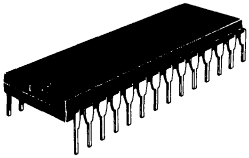
PHYSICAL DIMENSIONS



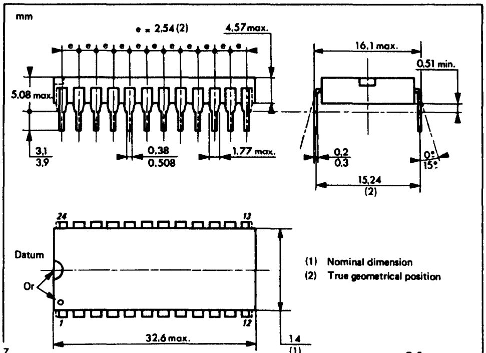
28 pins

DIN				
A51E CEI	D.A.T.A.	JEDEC	F 144 SITELESC	CB-132

CB-132



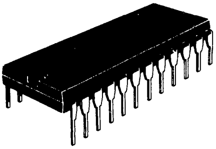
P SUFFIX
PLASTIC PACKAGE



24 pins

DIN				
A51D CEI	D.A.T.A.	JEDEC	F 118 SITELESC	CB-68

CB-68

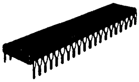


P SUFFIX
PLASTIC PACKAGE

These specifications are subject to change without notice.
Please inquire with our sales offices about the availability of the different products.

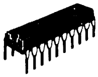
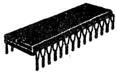
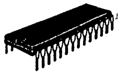
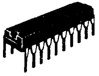
8-bit microcomputers

EF6801 FAMILY

Function	Part number	Characteristic		Package	Page
HIGH PERFORMANCE MCUs	EF6801	Dedicated MCUs for applications where high computing facilities are required such as personal/home computers, injection/trip computers, data transmission computers. • 8 X 8 multiply instruction • 29 I/O lines and 2 control lines • 16-bit programmable timer • Single-chip or expanded operation up to 64 Kbytes addressing space • Serial Communication Interface (SCI) • Emulation and development on DEVICE®	• RAM : 128 bytes (64 standby) • ROM : 2 Kbytes	 DIL40	2-7
	EF6803		• RAM : 128 bytes (64 standby)		2-7
	EF681U4*		• RAM : 192 bytes (32 standby) • ROM 4 Kbytes	 PLCC44	2-47
	EF6803U4*		• RAM : 192 bytes (32 standby)		2-47

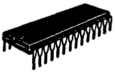
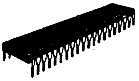
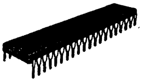
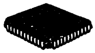
* DIL only

EF6804 FAMILY

Function	Part number	Characteristic		Package	Page
LOW-END MCU	EF6804J2	The highest improvement in cost reduction for low-end applications An MCU at a TTL/LS price level. • 12 bidirectional I/O lines (8 LED compatible) • Self check mode • 8-bit timer with 7-bit software programmable prescaler • Emulation and development on DEVICE®	• RAM : 32 bytes • ROM : 1012 bytes	 DIL20	2-91
LOW COST MCU	EF6804P2	High on-chip feature integration well suited for additional 4-bit application extensions. • 20 bidirectional I/O lines (8 LED compatible) • Self check mode • 8-bit timer with 7-bit software programmable prescaler • Emulation and development on DEVICE®	• RAM : 32 bytes • ROM : 1020 bytes	 DIL28 PLCC28	2-135
LOW POWER MCUs	EF68HC04P3	Dedicated to power and data saving applications or requiring protection against mains failures. • Pin to pin and software compatible with EF6804P2 • Self check mode • 8-bit timer with 7-bit software programmable prescaler • Emulation and development on DEVICE®	• RAM : 124 bytes • ROM : 2 Kbytes • 20 I/O lines	 DIL28 PLCC28	2-181
	TS68HC04J3		• RAM : 124 bytes • ROM : 2 Kbytes • 12 I/O lines	 DIL20	2-179

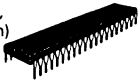
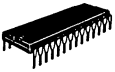
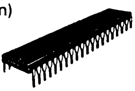
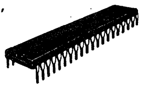
DEVICE® is THOMSON SEMICONDUCTEURS' development/emulation tool.

EF6805 FAMILY

Function	Part number	Characteristic		Package	Page
PROCESS CONTROL MCUs	EF6805P2	General purpose MCUs covering a wide range of applications such as electronic ignition, keyboard encoding, home appliances, electronic games,...	• RAM : 64 bytes • ROM : 1 Kbyte 20 I/O lines (8 LED compatible)	 DIL28	2-225
	EF6805P6	• Self check mode • 8-bit timer with 7-bit programmable prescaler • Emulation and development on DEVICE®	• RAM : 64 bytes • ROM : 1796 Kbytes • 20 I/O lines (8 LED compatible)	 PLCC28	2-249
	EF6805U2		• RAM : 64 bytes • ROM : 2 Kbytes • 32 I/O lines : - 24 bidirectional (8 LED compatible) - 8 input only	 DIL40	2-329
	EF6805U3		• RAM : 112 bytes • ROM : 3776 bytes • 32 I/O lines : - 24 bidirectional (8 LED compatible) - 8 input only	 PLCC44	2-355
A/D CONVERSION MCUs	EF6805R2	Dedicated MCUs for industrial control and applications where low cost analog signal computing is required such as automotive motor control.	• RAM : 64 bytes • ROM : 2 Kbytes • Mask programmable prescaler	 DIL40	2-273
	EF6805R3	• 32 I/O lines : - 24 bidirectional (8 LED compatible) - 8 input only • Self check mode • 8-bit timer with 7-bit programmable prescaler • A/D converter with 8-bit conversion and 4 multiplexed analog inputs • Emulation and development on DEVICE®	• RAM : 112 bytes • ROM : 3776 bytes • Software programmable prescaler	 PLCC44	2-301

DEVICE® is THOMSON SEMICONDUCTEURS' development/emulation tool.

MK3870 FAMILY

Function	Part number	Characteristic		Package	Page
LOW POWER MCU	MK3870	General purpose single chip microcomputer • Programmable binary timer • External interrupt • Crystal, LC, RC or external time base	• RAM : 64 bytes scratchpad, 64 bytes executable (option) • ROM : 2 K or 4 Kbytes • 32 TTL compatible I/O	 DIL40	2-399
LOW COST MCU	MK2870	28 pin version of MK3870, ideal for applications where PC board space is premium • Programmable binary timer • External interrupt • Crystal, LC, RC or external time base	• RAM : 64 bytes • ROM : 1 Kbyte • 20 TTL compatible I/O	 DIL28	2-381
SERIAL I/O PORT MCU	MK3873	Single chip microcomputer which introduces a major addition to the 3870 microcomputer family, a serial input/output port • Programmable binary timer • Vectored interrupts • External interrupt • Crystal, LC, RC or external time base	• RAM : 64 bytes scratchpad, 64 bytes executable (option) • ROM : 2 Kbyte • 29 TTL compatible parallel I/O	 DIL40	2-427
LOW POWER STANDBY RAM MCU	MK3875	Single chip microcomputer which offers a low power standby mode of operation as an addition to the 3870 family • Programmable binary timer • External interrupt • Crystal, LC, RC or external time base	• RAM : 64 bytes scratchpad, 64 Bytes executable • ROM : 4 Kbytes • 30 TTL compatible I/O	 DIL40	2-453
PIGGY-BACK EPROM MCUs	MK38P70	EPROM version of 3870 family which aliminates the need for emulator board products. Can be used in prototype systems in order to test design concepts in field service before commiting to high volume production • Piggyback PROM package • Accepts 24 pin or 28 pin EPROM memories • Full emulation of 3870 family	• Identical pinout as MK3870	PIGGY-BACK	2 399
	MK38P73		• Identical pinout as MK3873		2-427
	MK38P75		• Identical pinout as MK3875		2-453

THOMSON COMPONENTS SEMICONDUCTEURS

EF6801 • EF6803

MICROCOMPUTER/MICROPROCESSOR (MCU/MPU)

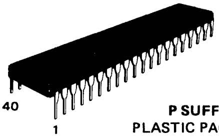
The EF6801 is an 8-bit single-chip microcomputer unit (MCU) which significantly enhances the capabilities of the 6800 family of parts. It includes an upgraded 6800 microprocessor unit (MPU) with upward-source and object-code compatibility. Execution times of key instructions have been improved and several new instructions have been added including an unsigned multiply. The MCU can function as a monolithic microcomputer or can be expanded to a 64K byte address space. It is TTL compatible and requires one + 5-volt power supply. On-chip resources include 2048 bytes of ROM, 128 bytes of RAM, a Serial Communications Interface (SCI), parallel I/O, and a three function Programmable Timer. The EF6803 can be considered as an EF6801 operating in Modes 2 or 3. EF6801 MCU Family features include :

- Enhanced EF6800 Instruction Set
- 8 x 8 Multiply Instruction
- Serial Communications Interface (SCI)
- Upward Source and Object Code Compatibility with the 6800
- 16-Bit Three-Function Programmable Timer
- Single-Chip or Expanded Operation to 64K Byte Address Space
- Bus Compatibility with the 6800 Family
- 2048 Bytes of ROM (EF6801)
- 128 Bytes of RAM
- 64 Bytes of RAM Retainable During Powerdown
- 29 Parallel I/O and Two Handshake Control Lines
- Internal Clock Generator with Divide-by-Four Output.
- Complete Development System Support on DEVICE®.
- –40°C to + 85°C Temperature range
- –40°C to + 105°C Temperature range

DEVICE® is THOMSON SEMICONDUCTEURS' development/emulation tool.

HMOS

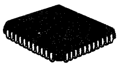
CASES



P SUFFIX
PLASTIC PACKAGE

ALSO AVAILABLE
C SUFFIX
CERAMIC PACKAGE

CB-521



FN SUFFIX
PLCC 44

PIN ASSIGMENT

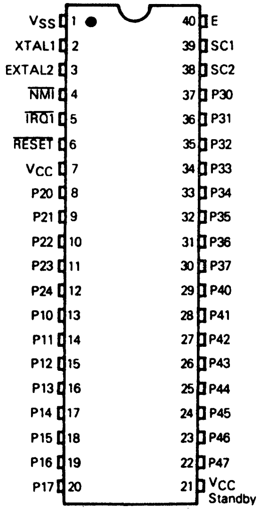
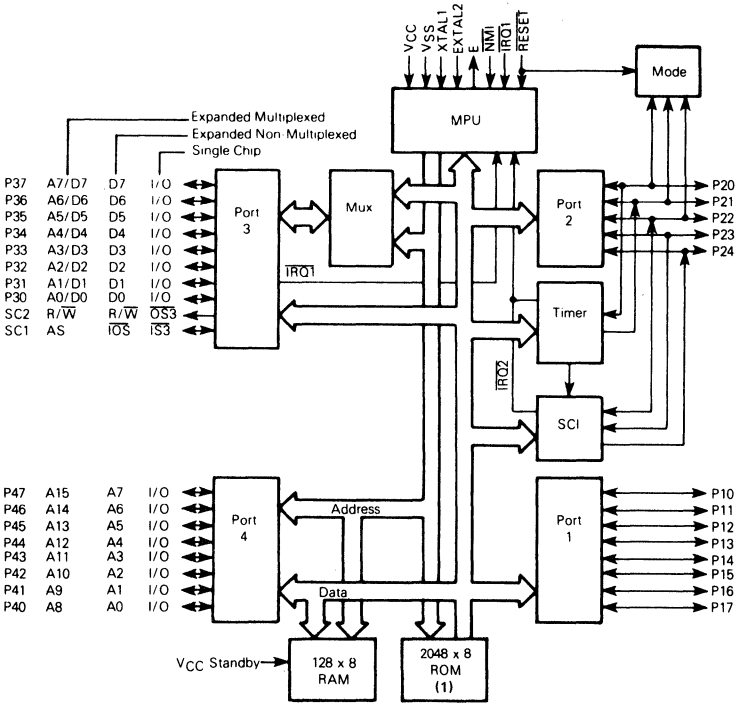


FIGURE 1 – 6801/6803 BLOCK DIAGRAM



(1) No functioning ROM in EF6803

POWER CONSIDERATIONS

The average chip-junction temperature, T_J, in °C can be obtained from:

$$T_J = T_A + (P_D \bullet \theta_{JA}) \qquad (1)$$

Where:

T_A ≡ Ambient Temperature, °C

θ_{JA} ≡ Package Thermal Resistance, Junction-to-Ambient, °C/W

P_D ≡ $P_{INT} + P_{PORT}$

P_{INT} ≡ $I_{CC} \times V_{CC}$, Watts – Chip Internal Power

P_{PORT} ≡ Port Power Dissipation, Watts – User Determined

For most applications $P_{PORT} \ll P_{INT}$ and can be neglected. P_{PORT} may become significant if the device is configured to drive Darlington bases or sink LED loads.

An approximate relationship between P_D and T_J (if P_{PORT} is neglected) is:

$$P_D = K \div (T_J + 273°C) \qquad (2)$$

Solving equations 1 and 2 for K gives:

$$K = P_D \bullet (T_A + 273°C) + \theta_{JA} \bullet P_D^2 \qquad (3)$$

Where K is a constant pertaining to the particular part. K can be determined from equation 3 by measuring P_D (at equilibrium) for a known T_A. Using this value of K the values of P_D and T_J can be obtained by solving equations (1) and (2) iteratively for any value of T_A.

MAXIMUM RATINGS

Rating	Symbol	Value	Unit
Supply Voltage	V_{CC}	– 0.3 to + 7.0	V
Input Voltage	V_{in}	– 0.3 to + 7.0	V
Operating Temperature Range EF6801/03, EF6801/03-1, EF68A01/03 EF68B01/03 EF6801/03, EF6801/03-1 : V suffix EF6801/03, EF6801/03-1 : A suffix	T_A	T_L to T_H 0 to 70 – 40 to 85 – 40 to 105	°C
Storage Temperature Range	T_{stg}	– 55 to + 150	°C

This device contains circuitry to protect the inputs against damage due to high static voltages or electric fields; however, it is advised that normal precautions be taken to avoid application of any voltage higher than maximum rated voltages to this high-impedance circuit. For proper operation it is recommended that V_{in} and V_{out} be constrained to the range $V_{SS} \leq (V_{in}$ or $V_{out}) \leq V_{CC}$. Input protection is enhanced by connecting unused inputs to either V_{DD} or V_{SS}.

THERMAL CHARACTERISTICS

Characteristic	Symbol	Value	Rating
Thermal Resistance Plastic PLCC	θ_{JA}	 50 100	°C/W

CONTROL TIMING (V_{CC} = 5.0 V ± 5%, V_{SS} = 0, T_A = 0 to 70°C)

Characteristic	Symbol	EF6801		EF6801-1		EF68A01		EF68B01		Unit
		Min	Max	Min	Max	Min	Max	Min	Max	
Frequency of Operation	f_o	0.5	1.0	0.5	1.25	0.5	1.5	0.5	2.0	MHz
Crystal Frequency	f_{XTAL}	2.0	4.0	2.0	5.0	2.0	6.0	2.0	8.0	MHz
External Oscillator Frequency	$4f_o$	2.0	4.0	2.0	5.0	2.0	6.0	2.0	8.0	MHz
Crystal Oscillator Start Up Time	t_{rc}	–	100	–	100	–	100	–	100	ms
Processor Control Setup Time	t_{PCS}	200	–	170	–	140	–	110	–	ns

DC ELECTRICAL CHARACTERISTICS (V_{CC} = 5.0 Vdc ± 5%, V_{SS} = 0, T_A = T_L to T_H, unless otherwise noted)

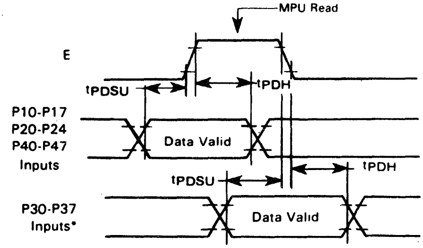
Characteristic		Symbol	EF6801/03 0°C to + 70°C		EF6801/03 –40°C to + 85°C		EF6801/03 –40°C to + 105°C		Unit
			Min	Max	Min	Max	Min	Max	
Input High Voltage	$\overline{\text{RESET}}$		V_{SS} + 4.0	V_{CC}	V_{SS} + 4.0	V_{CC}	V_{SS} + 4.0	V_{CC}	
	Other Inputs	V_{IH}	V_{SS} + 2.0	V_{CC}	V_{SS} + 2.2	V_{CC}	V_{SS} + 2.2	V_{CC}	V
Input Low Voltage	All Inputs	V_{IL}	V_{SS} – 0.3	V_{SS} + 0.8	V_{SS} – 0.3	V_{SS} + 0.8	V_{SS} – 0.3	V_{SS} + 0.8	V
Input Load Current	Port 4		–	0.5	–	0.8	–	0.8	
(V_{in} = 0 to 2.4 V)	SCI	I_{in}	–	0.8	–	1.0	–	1.0	mA
Input Leakage Current (V_{in} = 0 to 5.25 V)	$\overline{\text{NMI}}$, $\overline{\text{IRQ1}}$, $\overline{\text{RESET}}$	I_{in}	–	2.5	–	5.0	–	5.0	μA
Hi-Z (Off State) Input Current (V_{in} = 0.5 to 2.4 V)	Ports 1, 2, and 3	I_{TSI}	–	10	–	20	–	20	μA
Output High Voltage (I_{Load} = – 65 μA, V_{CC} = Min)*	E, Port 4, SC1, SC2	V_{OH}	V_{SS} + 2.4	–	V_{SS} + 2.4	–	V_{SS} + 2.4	–	V
(I_{Load} = – 100 μA, V_{CC} = Min)	Other Outputs		V_{SS} + 2.4	–	V_{SS} + 2.4	–	V_{SS} + 2.4	–	
Output Low Voltage (I_{Load} = 2.0 mA, V_{CC} = Min)	All Outputs	V_{OL}	–	V_{SS} + 0.5	–	V_{SS} + 0.6	–	V_{SS} + 0.6	V
Darlington Drive Current (V_O = 1.5 V)	Port 1	I_{OH}	1.0	4.0	1.0	5.0	1.0	5.0	mA
Internal Power Dissipation (Measured at T_A = T_L in Steady-State Operation)		P_{INT}	–	1200	–	1500	–	1500	mW
Input Capacitance	Port 3, Port 4, SCI	C_{in}	–	12.5	–	12.5	–	12.5	pF ·
(V_{in} = 0, T_A = 25°C, f_o = 1.0 MHz)	Other Inputs		–	10	–	10	–	10	
V_{CC} Standby	Powerdown	V_{SBB}	4.0	5.25	4.0	5.25	4.0	5.25	
	Powerup	V_{SB}	4.75	5.25	4.75	5.25	4.75	5.25	V
Standby Current	Powerdown	I_{SBB}	–	6.0	–	8.0	–	8.0	mA

* Negotiable to – 100 μA (for further information contact the factory)

PERIPHERAL PORT TIMING (Refer to Figures 2-5)

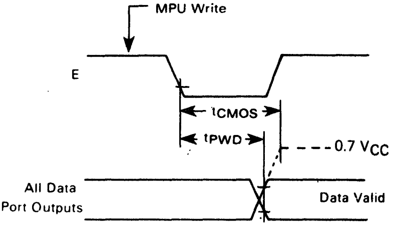
Characteristics	Symbol	EF6801 EF6803		EF6801-1 EF6803-1		EF68A01 EF68A03		EF68B01 EF68B03		Unit
		Min	Max	Min	Max	Min	Max	Min	Max	
Peripheral Data Setup Time	t_{PDSU}	200	–	200	–	150	–	100	–	ns
Peripheral Data Hold Time	t_{PDH}	200	–	200	–	150	–	100	–	ns
Delay Time, Enable Positive Transition to $\overline{OS3}$ Negative Transition	t_{OSD1}	–	350	–	350	–	300	–	250	ns
Delay Time, Enable Positive Transition to $\overline{OS3}$ Positive Transition	t_{OSD2}	–	350	–	350	–	300	–	250	ns
Delay Time, Enable Negative Transition to Peripheral Data Valid	t_{PWD}	–	350	–	350	–	300	–	250	ns
Delay Time, Enable Negative Transition to Peripheral CMOS Data Valid	t_{CMOS}	–	2.0	–	2.0	–	2.0	–	2.0	µs
Input Strobe Pulse Width	t_{PWIS}	200	–	200	–	150	–	100	–	ns
Input Data Hold Time	t_{IH}	50	–	50	–	40	–	30	–	ns
Input Data Setup Time	t_{IS}	20	–	20	–	20	–	20	–	ns

FIGURE 2 – DATA SETUP AND HOLD TIMES (MPU READ)



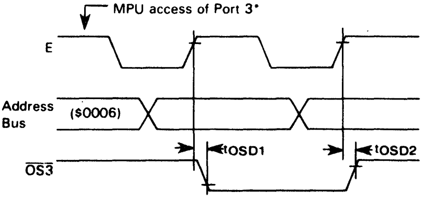
*Port 3 Non-Latched Operation (LATCH ENABLE = 0)

FIGURE 3 – DATA SETUP AND HOLD TIMES (MPU WRITE)



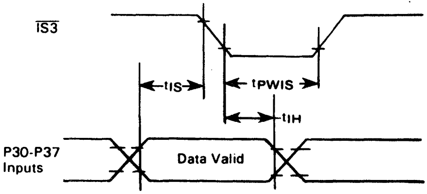
NOTES:
1 10 k Pullup resistor required for Port 2 to reach 0.7 V_{CC}
2 Not applicable to P21
3 Port 4 cannot be pulled above V_{CC}

FIGURE 4 – PORT 3 OUTPUT STROBE TIMING (EF6801 SINGLE-CHIP MODE)



*Access matches Output Strobe Select (OSS = 0, a read; OSS = 1, a write)

FIGURE 5 – PORT 3 LATCH TIMING (EF6801 SINGLE-CHIP MODE)



NOTE: Timing measurements are referenced to and from a low voltage of 0.8 volt and a high voltage of 2.0 volts, unless otherwise noted.

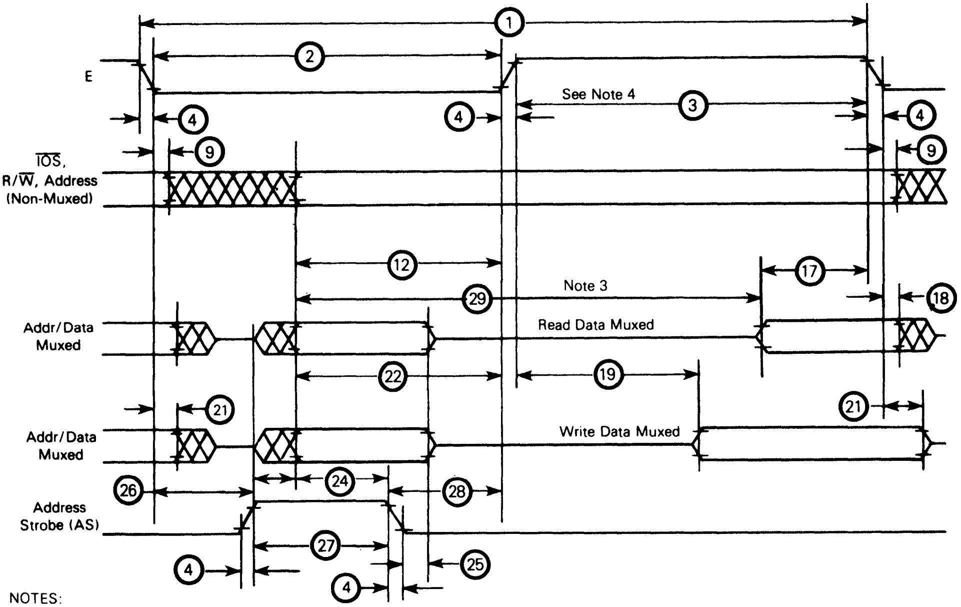
BUS TIMING (See Notes 1 and 2)

Ident. Number	Characteristics	Symbol	EF6801 EF6803		EF6801-1 EF6803-1		EF68A01 EF68A03		EF68B01 EF68B03		Unit
			Min	Max	Min	Max	Min	Max	Min	Max	
1	Cycle Time	t_{cyc}	1.0	2.0	0.8	2.0	0.667	2.0	0.5	2.0	μs
2	Pulse Width, E Low	PW_{EL}	430	1000	360	1000	300	1000	210	1000	ns
3	Pulse Width, E High	PW_{EH}	450	1000	360	1000	300	1000	220	1000	ns
4	Clock Rise and Fall Time	t_r, t_f	–	25	–	25	–	25	–	20	ns
9	Address Hold Time	t_{AH}	20	–	20	–	20	–	10	–	ns
12	Non-Muxed Address Valid Time to E*	t_{AV}	200	–	150	–	115	–	70	–	ns
17	Read Data Setup Time	t_{DSR}	80	–	70	–	60	–	40	–	ns
18	Read Data Hold Time	t_{DHR}	10	–	10	–	10	–	10	–	ns
19	Write Data Delay Time	t_{DDW}	–	225	–	200	–	170	–	120	ns
21	Write Data Hold Time	t_{DHW}	20	–	20	–	20	–	10	–	ns
22	Muxed Address Valid Time to E Rise*	t_{AVM}	200	–	150	–	115	–	80	–	ns
24	Muxed Address Valid Time to AS Fall*	t_{ASL}	60	–	50	–	40	–	20	–	ns
25	Muxed Address Hold Time	t_{AHL}	20	–	20	–	20	–	10	–	ns
26	Delay Time, E to AS Rise*	t_{ASD}	90**	–	70**	–	60**	–	45**	–	ns
27	Pulse Width, AS High*	PW_{ASH}	220	–	170	–	140	–	110	–	ns
28	Delay Time, AS to E Rise*	t_{ASED}	90	–	70	–	60	–	45	–	ns
29	Usable Access Time*	t_{ACC}	595	–	465	–	380	–	270	–	ns

*At specified cycle time.

**t_{ASD} parameters listed assume external TTL clock drive with 50 % ±5 % duty cycle. Devices driven by an external TTL clock with 50 % ± 1 % duty cycle or which use a crystal have the following t_{ASD} specifications : 100 ns min. (1.0 MHz devices), 80 ns min. (1.25 MHz devices), 65 ns min. (1.5 MHz devices), 50 ns min. (2.0 MHz devices).

FIGURE 6 – BUS TIMING



NOTES:

1. Voltage levels shown are $V_L \leq 0.5$ V, $V_H \geq 2.4$ V, unless otherwise specified.
2. Measurement points shown are 0.8 V and 2.0 V, unless otherwise specified.
3. Usable access time is computed by: 12+3–17+4.
4. Memory devices should be enabled only during E high to avoid Port 3 bus contention.

FIGURE 7 – CMOS LOAD

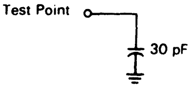
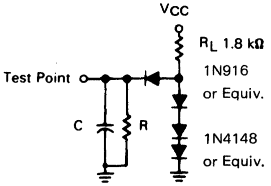


FIGURE 8 – TIMING TEST LOAD PORTS 1, 2, 3, 4



C = 90 pF for P30-P37, P40-P47, E, SC1, SC2
= 30 pF for P10-P17, P20-P24
R = 37 kΩ for P40-P47, E, SC1, SC2
= 24 kΩ for P10-P17, P20-P24
= 24 kΩ for P30-P37

INTRODUCTION

The EF6801 is an 8-bit monolithic microcomputer which can be configured to function in a wide variety of applications. The facility which provides this extraordinary flexibility is its ability to be hardware programmed into eight different operating modes. The operating mode controls the configuration of 18 of the 40 MCU pins, available on-chip resources, memory map, location (internal or external) of interrupt vectors, and type of external bus. The configuration of the remaining 22 pins is not dependent on the operating mode.

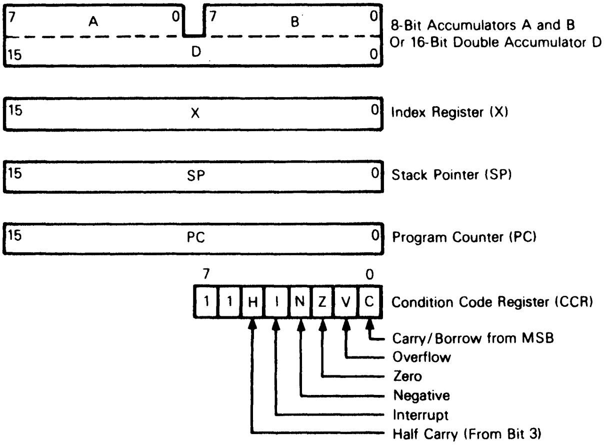
Twenty-nine pins are organized as three 8-bit ports and one 5-bit port. Each port consists of at least a Data Register and a write-only Data Direction Register. The Data Direction Register is used to define whether corresponding bits in the Data Register are configured as an input (clear) or output (set).

The term "port," by itself, refers to all of the hardware associated with the port. When the port is used as a "data port" or "I/O port," it is controlled by the port Data Direction Register and the programmer has direct access to the port pins using the port Data Register. Port pins are labled as Pij where i identifies one of four ports and j indicates the particular bit.

The Microprocessor Unit (MPU) is an enhanced EF6800 MPU with additional capabilities and greater throughput. It is upward source and object code compatible with the EF6800. The programming model is depicted in Figure 9, where Accumulator D is a concatenation of Accumulators A and B. A list of new operations added to the 6800 instruction set are shown in Table 1.

The EF6803 can be considered an EF6801 that operates in Modes 2 and 3 only.

FIGURE 9 – PROGRAMMING MODEL



OPERATING MODES

The EF6801 provides eight different operating modes (Modes 0 through 7), the EF6803 provides two operating modes (Modes 2 and 3). The operating modes are hardware selectable and determine the device memory map, the configuration of Port 3, Port 4, SC1, SC2, and the physical location of the interrupt vectors.

FUNDAMENTAL MODES

The eight operating modes can be grouped into three fundamental modes which refer to the type of bus it supports: Single Chip, Expanded Non-Multiplexed, and Expanded Multiplexed. Single chip modes include 4 and 7, Expanded Non-Multiplexed is Mode 5 and the remaining five are Expanded Multiplexed modes. Table 2 summarizes the characteristics of the operating modes.

EF6801 Single-Chip Modes (4,7)

In the Single-Chip Mode, the four MCU ports are configured as parallel input/output data ports, as shown in Figure 10. The MCU functions as a monolithic microcomputer in these two modes without external address or data buses. A maximum of 29 I/O lines and two Port 3 control lines are provided. Peripherals or another MCU can be interfaced to Port 3 in a loosely coupled dual processor configuration, as shown in Figure 11.

TABLE 1 — NEW INSTRUCTIONS

Instruction	Description
ABX	Unsigned addition of Accumulator B to Index Register
ADDD	Adds (without carry) the double accumulator to memory and leaves the sum in the double accumulator
ASLD or LSLD	Shifts the double accumulator left (towards MSB) one bit; the LSB is cleared and the MSB is shifted into the C-bit
BHS	Branch if Higher or Same; unsigned conditional branch (same as BCC)
BLO	Branch if Lower; Unsigned conditional branch (same as BCS)
BRN	Branch Never
JSR	Additional addressing mode: direct
LDD	Loads double accumulator from memory
LSL	Shifts memory or accumulator left (towards MSB) one bit; the LSB is cleared and the MSB is shifted into the C-bit (same as ASL)
LSRD	Shifts the double accumulator right (towards LSB) one bit; the MSB is cleared and the LSB is shifted into the C-bit
MUL	Unsigned multiply; multiplies the two accumulators and leaves the product in the double accumulator
PSHX	Pushes the Index Register to stack
PULX	Pulls the Index Register from stack
STD	Stores the double accumulator to memory
SUBD	Subtracts memory from the double accumulator and leaves the difference in the double accumulator
CPX	Internal processing modified to permit its use with any conditional branch instruction

In Single-Chip Test Mode (4), the RAM responds to $XX80 through $XXFF and the ROM is removed from the internal address map. A test program must first be loaded into the RAM using modes 0, 1, 2, or 6. If the MCU is Reset and then programmed into Mode 4, execution will begin at $XXFE:XXFF. Mode 5 can be irreversibly entered from Mode 4 without asserting $\overline{\text{RESET}}$ by setting bit 5 of the Port 2 Data Register. This mode is used primarily to test Ports 3 and 4 in the Single-Chip and Non-Multiplexed Modes.

EF6801 Expanded Non-Multiplexed Mode (5)

A modest amount of external memory space is provided in the Expanded Non-Multiplexed Mode while significant on-chip resources are retained. Port 3 functions as an 8-bit bidirectional data bus and Port 4 is configured initially as an input data port. Any combination of the eight least-significant address lines may be obtained by writing to the Port 4 Data Direction Register. Stated alternatively, any combination of A0 to A7 may be provided while retaining the remainder as input data lines. Internal pullup resistors pull the Port 4 lines high until the port is configured.

Figure 12 illustrates a typical system configuration in the Expanded Non-Multiplexed Mode. The MCU interfaces directly with 6800 family parts and can access 256 bytes of external address space at $100 through $1FF. IOS provides an address decode of external memory ($100-$1FF) and can be used as a memory page select or chip select line.

TABLE 2 – SUMMARY OF EF6801/03 OPERATING MODES

Common to all Modes: Reserved Register Area Port 1 Port 2 Programmable Timer Serial Communications Interface
Single Chip Mode 7 128 bytes of RAM; 2048 bytes of ROM Port 3 is a parallel I/O port with two control lines Port 4 is a parallel I/O port SC1 is Input Strobe 3 ($\overline{\text{IS3}}$) SC2 is Output Strobe 3 ($\overline{\text{OS3}}$)
Expanded Non-Multiplexed Mode 5 128 bytes of RAM; 2048 bytes of ROM 256 bytes of external memory space Port 3 is an 8-bit data bus Port 4 is an input port/address bus SC1 is Input/Output Select ($\overline{\text{IOS}}$) SC2 is Read/Write (R/$\overline{\text{W}}$)
Expanded Multiplexed Modes 1, 2, 3, 6* Four memory space options (64K address space): (1) No internal RAM or ROM (Mode 3) (2) Internal RAM, no ROM (Mode 2) (3) Internal RAM and ROM (Mode 1) (4) Internal RAM, ROM with partial address bus (Mode 6) Port 3 is a multiplexed address/data bus Port 4 is an address bus (inputs/address in Mode 6) SC1 is Address Strobe (AS) SC2 is Read/Write (R/$\overline{\text{W}}$)
Test Modes 0 and 4 Expanded Multiplexed Test Mode 0 May be used to test RAM and ROM Single Chip and Non-Multiplexed Test Mode 4 (1) May be changed to Mode 5 without going through Reset (2) May be used to test Ports 3 and 4 as I/O ports

*The EF6803 operates only in modes 2 and 3

FIGURE 10 – SINGLE-CHIP MODE

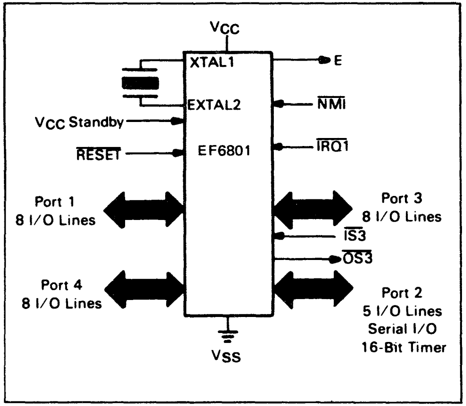


FIGURE 11 – SINGLE-CHIP DUAL PROCESSOR CONFIGURATION

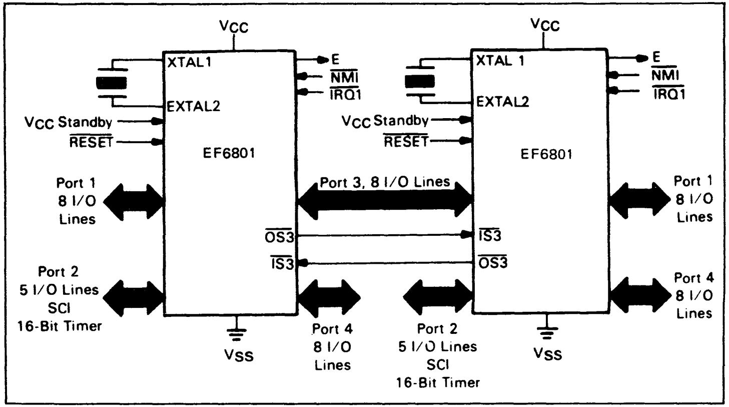
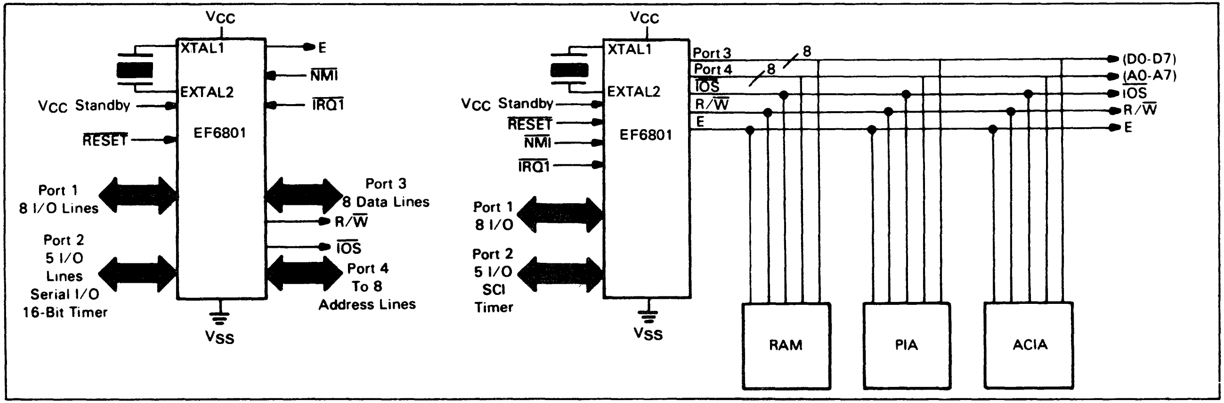


FIGURE 12 – EXPANDED NON-MULTIPLEXED CONFIGURATION



Expanded-Multiplexed Modes (0, 1, 2, 3, 6)

A 64K byte memory space is provided in the expanded multiplexed modes. In each of the expanded multiplexed modes Port 3 functions as a time multiplexed address/data bus with address valid on the negative edge of Address Strobe (AS), and data valid while E is high. In Modes 0 to 3, Port 4 provides address lines A8 to A15. In Mode 6, however, Port 4 initially is configured at $\overline{\text{RESET}}$ as an input data port. The port 4 Data Direction Register can then be changed to provide any combination of address lines, A8 to A15. Stated alternatively, any subset of A8 to A15 can be provided while retaining the remaining port 4 lines as input data lines. Internal pullup resistors pull the Port 4 lines high until software configures the port.

In Mode 0, the Reset vector is external for the first two E-cycles after the positive edge of $\overline{\text{RESET}}$, and internal thereafter. In addition, the internal and external data buses are connected so there must be no memory map overlap in order to avoid potential bus conflicts. Mode 0 is used primarily to verify the ROM pattern and monitor the internal data bus with the automated test equipment.

Only the EF6801 can operate in each of the expanded multiplexed modes. The EF6803 operates only in Modes 2 and 3.

Figure 13 depicts a typical configuration for the Expanded-Multiplexed Modes. Address Strobe can be used to control a transparent D-type latch to capture addresses A0-A7, as shown in Figure 14. This allows Port 3 to function as a Data Bus when E is high.

PROGRAMMING THE MODE

The operating mode is determined at $\overline{\text{RESET}}$ by the levels asserted on P22, P21, and P20. These levels are latched into PC2, PC1, and PC0 of the program control register on the positive edge of $\overline{\text{RESET}}$. The operating mode may be read from the Port 2 Data Register as shown below, and programming levels and timing must be met as shown in Figure 15. A brief outline of the operating modes is shown in Table 3.

PORT 2 DATA REGISTER

7	6	5	4	3	2	1	0	
PC2	PC1	PC0	P24	P23	P22	P21	P20	$0003

Circuitry to provide the programming levels is dependent primarily on the normal system usage of the three pins. If configured as outputs, the circuit shown in Figure 16 may be used; otherwise, three-state buffers can be used to provide isolation while programming the mode.

TABLE 3 — MODE SELECTION SUMMARY

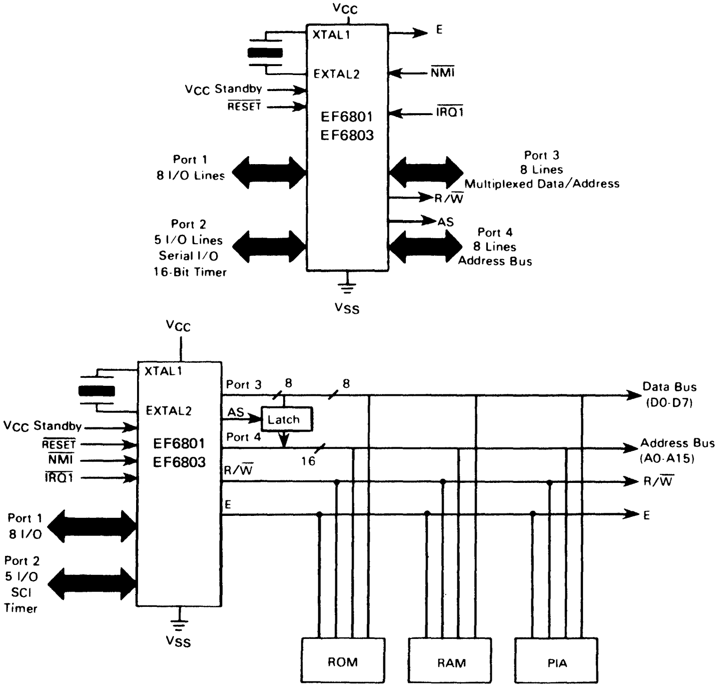
Mode*	P22 PC2	P21 PC1	P20 PC0	ROM	RAM	Interrupt Vectors	Bus Mode	Operating Mode
7	H	H	H	I	I	I	I	Single Chip
6	H	H	L	I	I	I	MUX[5, 6]	Multiplexed/Partial Decode
5	H	L	H	I	I	I	NMUX[5, 6]	Non-Multiplexed/Partial Decode
4	H	L	L	I[2]	I[1]	I	I	Single Chip Test
3	L	H	H	E	E	E	MUX[4]	Multiplexed/No RAM or ROM
2	L	H	L	E	I	E	MUX[4]	Multiplexed/RAM
1	L	L	H	I	I	E	MUX[4]	Multiplexed/RAM & ROM
0	L	L	L	I	I	I[3]	MUX[4]	Multiplexed Test

Legend:
I — Internal
E — External
MUX — Multiplexed
NMUX — Non-Multiplexed
L — Logic "0"
H — Logic "1"

Notes:
[1] Internal RAM is addressed at $XX80
[2] Internal ROM is disabled
[3] $\overline{\text{RESET}}$ vector is external for 2 cycles after $\overline{\text{RESET}}$ goes high
[4] Addresses associated with Ports 3 and 4 are considered external in Modes 0, 1, 2, and 3
[5] Addresses associated with Port 3 are considered external in Modes 5 and 6
[6] Port 4 default is user data input; address output is optional by writing to Port 4 Data Direction Register

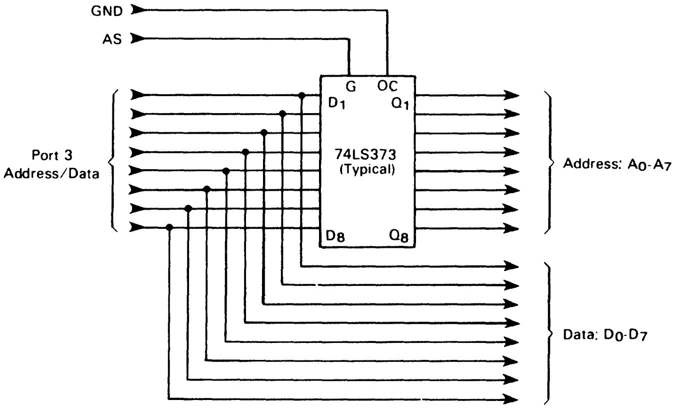
*The EF6803 operates only in Modes 2 and 3

FIGURE 13 – EXPANDED MULTIPLEXED CONFIGURATION



NOTE: To avoid data bus (Port 3) contention in the expanded multiplexed modes, memory devices should be enabled only during E high time.

FIGURE 14 – TYPICAL LATCH ARRANGEMENT



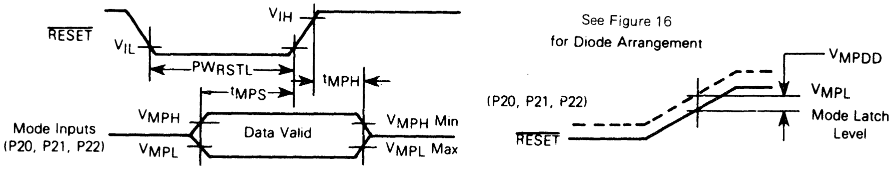


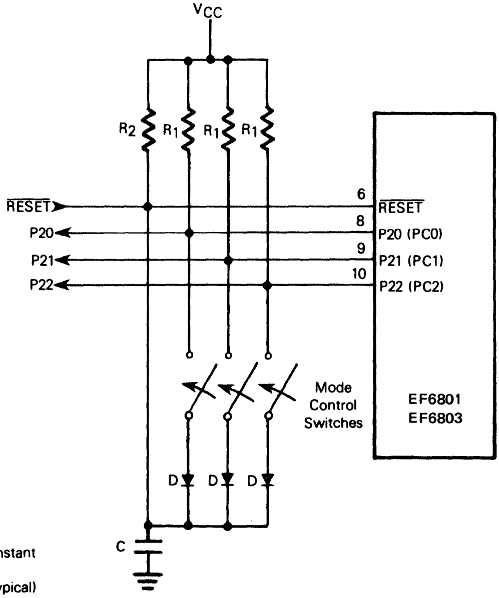
FIGURE 15 – MODE PROGRAMMING TIMING

MODE PROGRAMMING (Refer to Figure 15)

Characteristic	Symbol	Min	Max	Unit
Mode Programming Input Voltage Low*	V_{MPL}	–	1.8	V
Mode Programming Input Voltage High	V_{MPH}	4.0	–	V
Mode Programming Diode Differential (If Diodes are Used)	V_{MPDD}	0.6	–	V
$\overline{\text{RESET}}$ Low Pulse Width	PW_{RSTL}	3.0	–	E-Cycles
Mode Programming Setup Time	t_{MPS}	2.0	–	E-Cycles
Mode Programming Hold Time $\overline{\text{RESET}}$ Rise Time≥1 µs $\overline{\text{RESET}}$ Rise Time<1 µs	t_{MPH}	0 100	– –	ns

*For T_A=–40°C to + 105°C, V_{MPL}= 1.7 V.

FIGURE 16 – TYPICAL MODE PROGRAMMING CIRCUIT

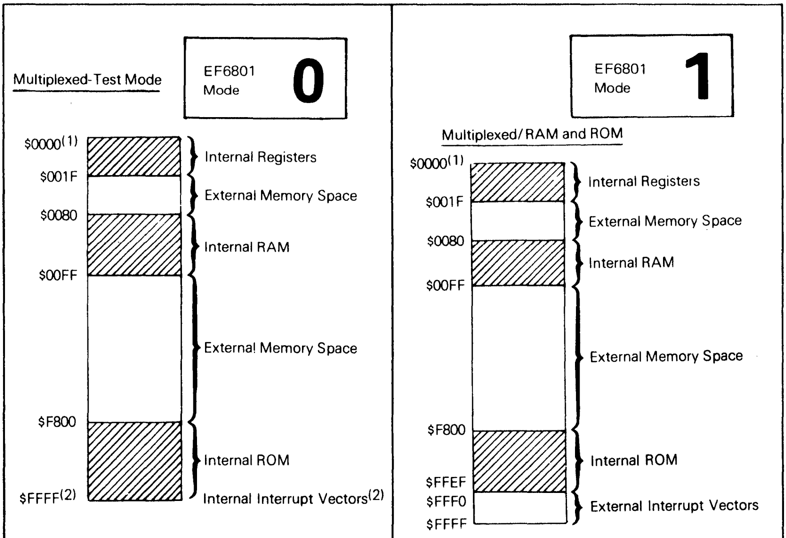


Notes:
1. Mode 7 as shown
2. R_2•C = Reset time constant
3. R_1 = 10 k (typical)
4. D = 1N914, 1N4001 (typical)
5. Diode V_f should not exceed V_{MPDD} min.

MEMORY MAPS

The 6801 Family can provide up to 64K byte address space depending on the operating mode. A memory map for each operating mode is shown in Figure 17. The first 32 locations of each map are reserved for the internal register area, as shown in Table 4, with exceptions as indicated.

FIGURE 17 – EF6801/03 MEMORY MAPS (Sheet 1 of 3)



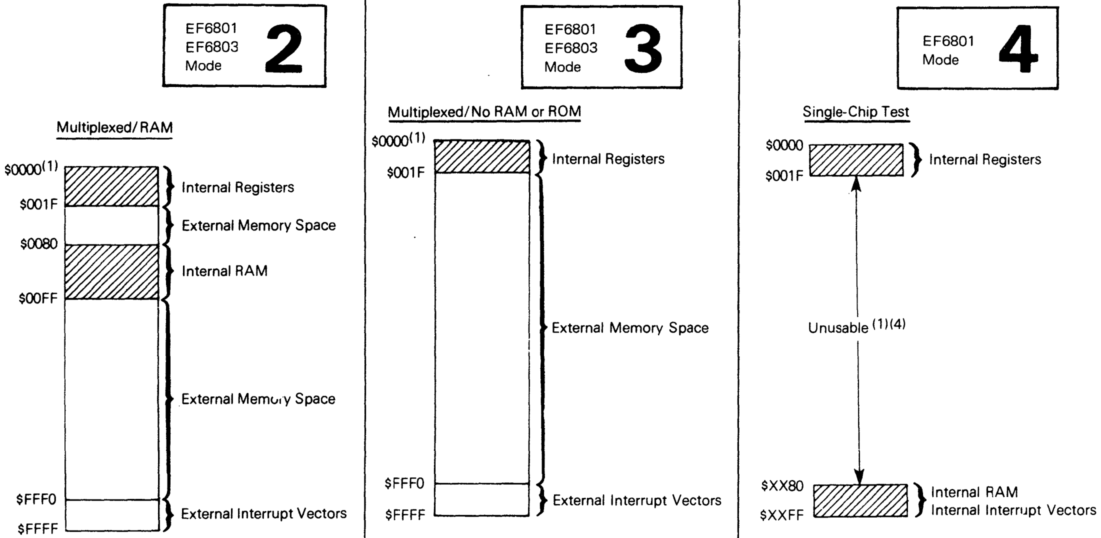
NOTES:

1) Excludes the following addresses which may be used externally: $04, $05, $06, $07, and $0F.
2) Addresses $FFFE and $FFFF are considered external if accessed within two cycles after a positive edge of $\overline{\text{RESET}}$ and internal at all other times.
3) After two MPU cycles, there must be no overlapping of internal and external memory spaces to avoid driving the data bus with more than one device.
4) This mode is the only mode which may be used to examine the interrupt vectors in internal ROM using an external $\overline{\text{RESET}}$ vector.

NOTES:

1) Excludes the following addresses which may be used externally: $04, $05, $06, $07, and $0F.
2) Internal ROM addresses $FFF0 to $FFFF are not usable.

FIGURE 17 – EF6801/03 MEMORY MAPS (Sheet 2 of 3)



NOTES:

1) Excludes the following addresses which may be used externally: $04, $05, $06, $07, and $0F.

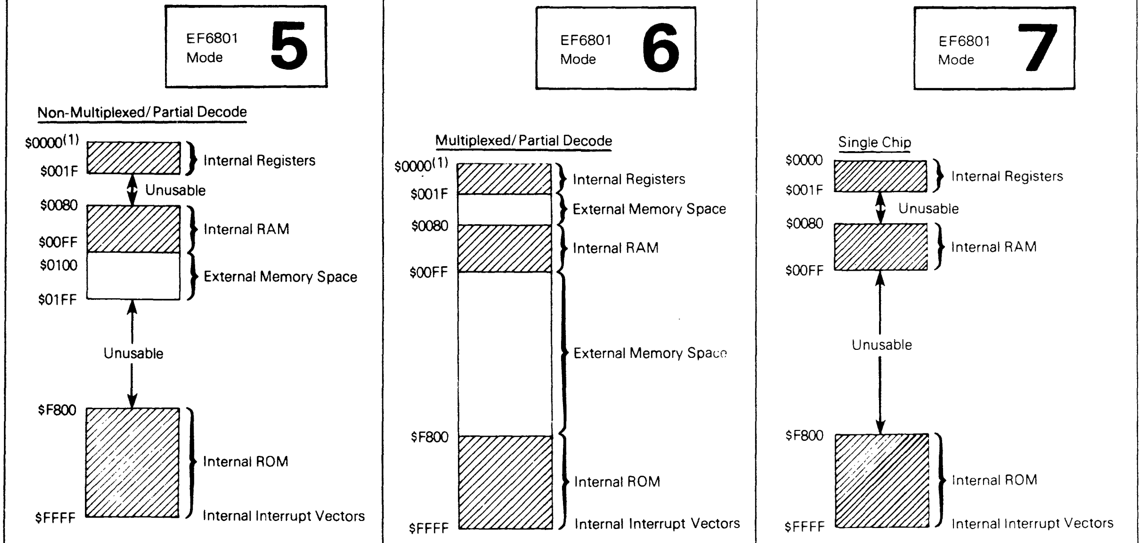
NOTES:

1) Excludes the following addresses which may be used externally: $04, $05, $06, $07, and $0F.

NOTES:

1) The internal ROM is disabled.
2) Mode 4 may be changed to Mode 5 without having to assert $\overline{\text{RESET}}$ by writing a one into the PCO bit of the port 2 data register.
3) Addresses A8 to A15 are treated as "don't cares" to decode internal RAM.
4) Internal RAM will appear at $XX80 to $XXFF.

FIGURE 17 – EF6801/03 MEMORY MAPS (Sheet 3 of 3)



NOTES:

1) Excludes the following addresses which may **not** be used externally: \$04, \$06, and \$0F (no $\overline{\text{IOS}}$).
2) This mode may be entered without going through $\overline{\text{RESET}}$ by using mode 4 and subsequently writing a one into the PCO bit of the port 2 data register.
3) Address lines A0 to A7 will not contain addresses until the data direction register for port 4 has been written with ones in the appropriate bits. These address lines will assert ones until made outputs by writing the data direction register.

NOTES:

1) Excludes the following addresses which may be used externally: \$04, \$06, and \$0F.
2) Address lines A8-A15 will not contain addresses until the data direction register for port 4 has been written with ones in the appropriate bits. These address lines will assert ones until made outputs by writing the data direction register.

EF6801/03 INTERRUPTS

The 6801 Family supports two types of interrupt requests: maskable and non-maskable. A Non-Maskable Interrupt ($\overline{\text{NMI}}$) is always recognized and acted upon at the completion of the current instruction. Maskable interrupts are controlled by the Condition Code Register I-bit and by individual enable bits. The I-bit controls all maskable interrupts. Of the maskable interrupts, there are two types: $\overline{\text{IRQ1}}$ and $\overline{\text{IRQ2}}$. The Programmable Timer and Serial Communications Interface use an internal $\overline{\text{IRQ2}}$ interrupt line, as shown in Figure 1. External devices (and IS3) use $\overline{\text{IRQ1}}$. An $\overline{\text{IRQ1}}$ interrupt is serviced before $\overline{\text{IRQ2}}$ if both are pending.

All IRQ2 interrupts use hardware prioritized vectors. The single SCI interrupt and three timer interrupts are serviced in a prioritized order and each is vectored to a separate location. All interrupt vector locations are shown in Table 5.

The Interrupt flowchart is depicted in Figure 18 and is common to every interrupt excluding reset. During interrupt servicing the Program Counter, Index Register, A Accumulator, B Accumulator, and Condition Code Register are pushed to the stack. The I-bit is set to inhibit maskable interrupts and a vector is fetched corresponding to the current highest priority interrupt. The vector is transferred to the Program Counter and instruction execution is resumed. Interrupt and $\overline{\text{RESET}}$ timing are illustrated in Figures 19 and 20.

FUNCTIONAL PIN DESCRIPTIONS

V_{CC} AND V_{SS}

V_{CC} and V_{SS} provide power to a large portion of the MCU. The power supply should provide +5 volts (±5%) to V_{CC}, and V_{SS} should be tied to ground. Total power dissipation (including V_{CC} Standby), will not exceed P_D milliwatts.

V_{CC} STANDBY

V_{CC} Standby provides power to the standby portion ($80 through $BF) of the RAM and the STBY PWR and RAME bits of the RAM Control Register. Voltage requirements depend on whether the device is in a powerup or powerdown state. In the powerup state, the power supply should provide +5 volts (±5%) and must reach V_{SB} volts before $\overline{\text{RESET}}$ reaches 4.0 volts. During powerdown, V_{CC} Standby must remain above V_{SBB} (min) to sustain the standby RAM and STBY PWR bit. While in powerdown operation, the standby current will not exceed I_{SBB}.

It is typical to power both V_{CC} and V_{CC} Standby from the same source during normal operation. A diode must be used between them to prevent supplying power to V_{CC} during powerdown operation. V_{CC} Standby should be tied to ground in Mode 3.

TABLE 4 — INTERNAL REGISTER AREA

Register	Address
Port 1 Data Direction Register***	00
Port 2 Data Direction Register***	01
Port 1 Data Register	02
Port 2 Data Register	03
Port 3 Data Direction Register***	04*
Port 4 Data Direction Register***	05**
Port 3 Data Register	06*
Port 4 Data Register	07**
Timer Control and Status Register	08
Counter (High Byte)	09
Counter (Low Byte)	0A
Output Compare Register (High Byte)	0B
Output Compare Register (Low Byte)	0C
Input Capture Register (High Byte)	0D
Input Capture Register (Low Byte)	0E
Port 3 Control and Status Register	0F*
Rate and Mode Control Register	10
Transmit/Receive Control and Status Register	11
Receive Data Register	12
Transmit Data Register	13
RAM Control Register	14
Reserved	15-1F

*External addresses in Modes 0, 1, 2, 3, 5, 6; cannot be accessed in Mode 5 (No $\overline{\text{IOS}}$)

**External addresses in Modes 0, 1, 2, 3

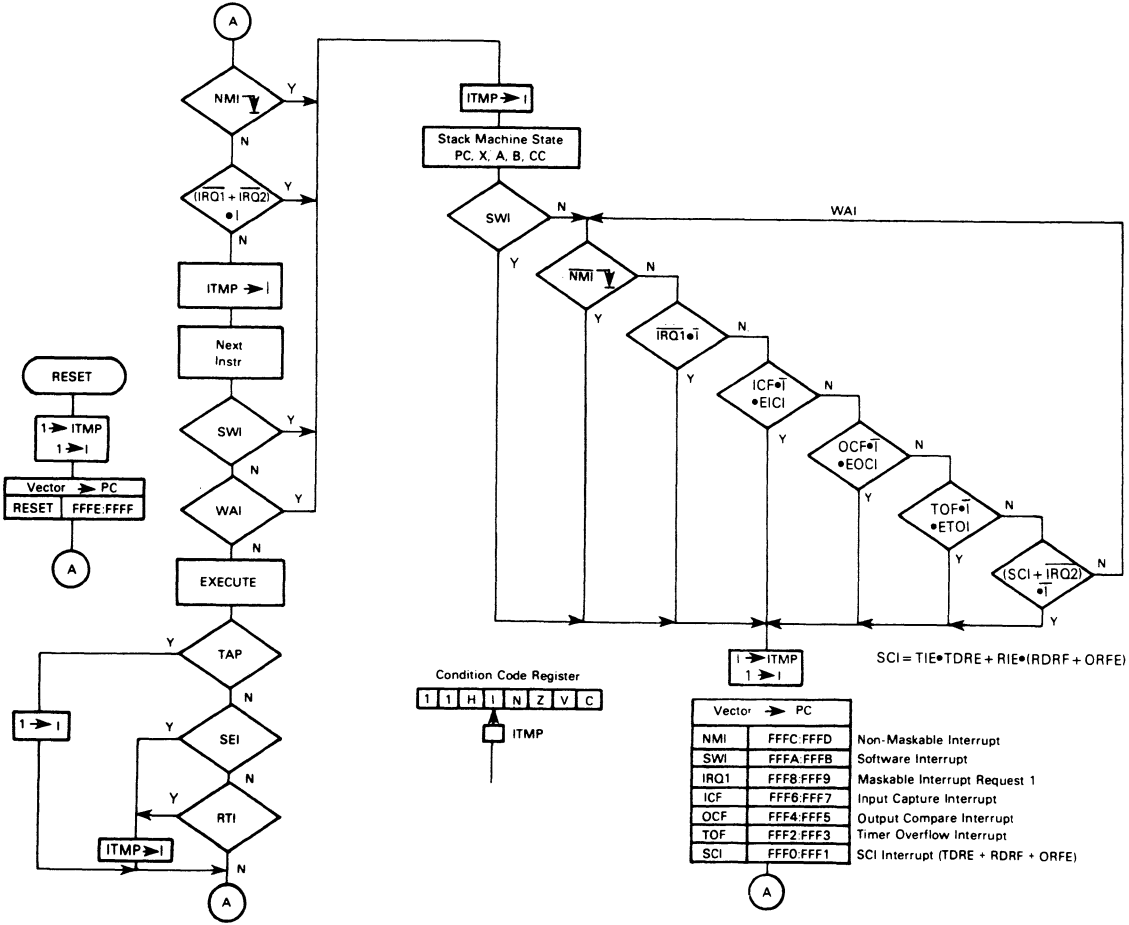
***1 = Output, 0 = Input

TABLE 5 — MCU INTERRUPT VECTOR LOCATIONS

MSB	LSB	Interrupt
FFFE	FFFF	$\overline{\text{RESET}}$
FFFC	FFFD	$\overline{\text{NMI}}$
FFFA	FFFB	Software Interrupt (SWI)
FFF8	FFF9	$\overline{\text{IRQ1}}$ (or $\overline{\text{IS3}}$)
FFF6	FFF7	ICF (Input Capture)*
FFF4	FFF5	OCF (Output Compare)*
FFF2	FFF3	TOF (Timer Overflow)*
FFF0	FFF1	SCI (RDRF+ORFE+TDRE)*

*IRQ2 Interrupt

FIGURE 18 – INTERRUPT FLOWCHART



SCI = TIE•TDRE + RIE•(RDRF + ORFE)

Vector → PC		
NMI	FFFC:FFFD	Non-Maskable Interrupt
SWI	FFFA:FFFB	Software Interrupt
IRQ1	FFF8:FFF9	Maskable Interrupt Request 1
ICF	FFF6:FFF7	Input Capture Interrupt
OCF	FFF4:FFF5	Output Compare Interrupt
TOF	FFF2:FFF3	Timer Overflow Interrupt
SCI	FFF0:FFF1	SCI Interrupt (TDRE + RDRF + ORFE)

FIGURE 19 – INTERRUPT SEQUENCE

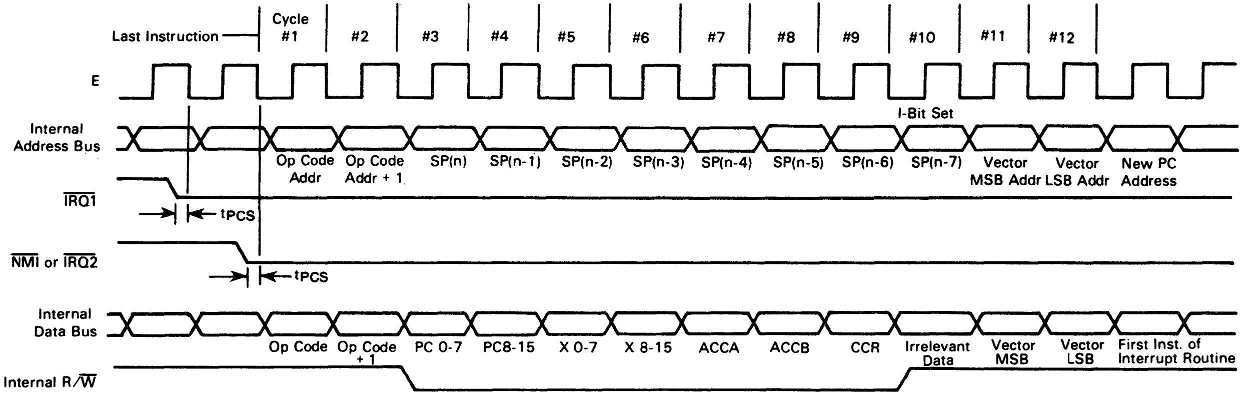
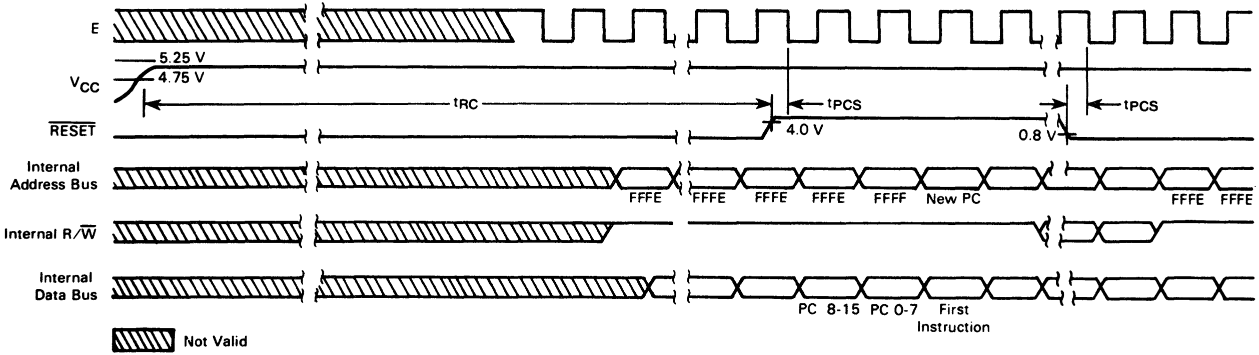


FIGURE 20 – RESET TIMING



XTAL1 AND EXTAL2

These two input pins interface either a crystal or TTL compatible clock to the MCU internal clock generator. Divide-by-four circuitry is included which allows use of the inexpensive 3.58 MHz or 4.4336 MHz Color Burst TV crystals. A 20 pF capacitor should be tied from each crystal pin to ground to ensure reliable startup and operation. Alternatively, EXTAL2 may be driven by an external TTL compatible clock at $4f_0$ with a duty cycle of 50% (±5%) with XTAL1 connected to ground.

The internal oscillator is designed to interface with an AT-cut quartz crystal resonator operated in parallel resonance mode in the frequency range specified for f_{XTAL}. The crystal should be mounted as close as possible to the input pins to minimize output distortion and startup stabilization time.* The MCU is compatible with most commercially available crystals. Nominal crystal parameters are shown in Figure 21.

$\overline{\text{RESET}}$

This input is used to reset the internal state of the device and provide an orderly startup procedure. During powerup, $\overline{\text{RESET}}$ must be held below 0.8 volts: (1) at least t_{RC} after V_{CC} reaches 4.75 volts in order to provide sufficient time for the clock generator to stabilize, and (2) until V_{CC} Standby reaches 4.75 volts. $\overline{\text{RESET}}$ must be held low at least three E-cycles if asserted during powerup operation.

E (ENABLE)

This is an output clock used primarily for bus synchronization. It is TTL compatible and is the slightly skewed divide-by-four result of the device input clock frequency. It will drive one Schottky TTL load and 90 pF, and all data given in cycles is referenced to this clock unless otherwise noted.

$\overline{\text{NMI}}$ (NON-MASKABLE INTERRUPT)

An $\overline{\text{NMI}}$ negative edge requests an MCU interrupt sequence, but the current instruction will be completed before it responds to the request. The MCU will then begin an interrupt sequence. Finally, a vector is fetched from $FFFC and $FFFD, transferred to the Program Counter and instruction execution is resumed. $\overline{\text{NMI}}$ typically requires a 3.3 kΩ (nominal) resistor to V_{CC}. There is no internal $\overline{\text{NMI}}$ pullup resistor. NMI must be held low for at least one E-cycle to be recognized under all conditions.

$\overline{\text{IRQ1}}$ (MASKABLE INTERRUPT REQUEST 1)

$\overline{\text{IRQ1}}$ is a level-sensitive input which can be used to request an interrupt sequence. The MPU will complete the current instruction before it responds to the request. If the interrupt mask bit (I-bit) in the Condition Code Register is clear, the MCU will begin an interrupt sequence. A vector is fetched from $FFF8 and $FFF9, transferred to the Program Counter, and instruction execution is resumed.

$\overline{\text{IRQ1}}$ typically requires an external 3.3 kΩ (nominal) resistor to V_{CC} for wire-OR applications. $\overline{\text{IRQ1}}$ has no internal pullup resistor.

SC1 AND SC2 (STROBE CONTROL 1 AND 2)

The function of SC1 and SC2 depends on the operating mode. SC1 is configured as an output in all modes except single chip mode, whereas SC2 is always an output. SC1 and SC2 can drive one Schottky load and 90 pF.

SC1 and SC2 In Single-Chip Mode

In Single-Chip Mode, SC1 and SC2 are configured as an input and output, respectively, and both function as Port 3 control lines. SC1 functions as $\overline{\text{IS3}}$ and can be used to indicate that Port 3 input data is ready or output data has been accepted. Three options associated with $\overline{\text{IS3}}$ are controlled by Port 3 Control and Status Register and are discussed in the Port 3 description. If unused, $\overline{\text{IS3}}$ can remain unconnected.

SC2 is configured as $\overline{\text{OS3}}$ and can be used to strobe output data or acknowledge input data. It is controlled by Output Strobe Select (OSS) in the Port 3 Control and Status Register. The strobe is generated by a read (OSS = 0) or write (OSS = 1) to the Port 3 Data Register. $\overline{\text{OS3}}$ timing is shown in Figure 4.

SC1 And SC2 In Expanded Non-Multiplexed Mode

In the Expanded Non-Multiplexed Mode, both SC1 and SC2 are configured as outputs. SC1 functions as Input/Output Select ($\overline{\text{IOS}}$) and is asserted only when $0100 through $01FF is sensed on the internal address bus.

SC2 is configured as Read/Write and is used to control the direction of data bus transfers. An MPU read is enabled when Read/Write and E are high.

SC1 And SC2 In Expanded Multiplexed Mode

In the Expanded Multiplexed Modes, both SC1 and SC2 are configured as outputs. SC1 functions as Address Strobe and can be used to demultiplex the eight least significant addresses and the data bus. A latch controlled by Address Strobe captures address on the negative edge, as shown in Figure 14.

SC2 is configured as Read/Write and is used to control the direction of data bus transfers. An MPU read is enabled when Read/Write and E are high.

P10-P17 (PORT 1)

Port 1 is a mode independent 8-bit I/O port with each line an input or output as defined by the Port 1 Data Direction Register. The TTL compatible three-state output buffers can drive one Schottky TTL load and 30 pF, Darlington transistors, or CMOS devices using external pullup resistors. It is configured as a data input port by $\overline{\text{RESET}}$. Unused lines can remain unconnected.

P20-P24 (PORT 2)

Port 2 is a mode-independent, 5-bit, multipurpose I/O port. The voltage levels present on P20, P21, and P22 on the rising edge of $\overline{\text{RESET}}$ determine the operating mode of the MCU. The entire port is then configured as a data input port. The Port 2 lines can be selectively configured as data output lines by setting the appropriate bits in the Port 2 Data Direction Register. The Port 2 Data Register is used to move data through the port. However, if P21 is configured as an output, it will be tied to the timer Output Compare function and cannot be used to provide output from the Port 2 Data Register.

Port 2 can also be used to provide an interface for the Serial Communications Interface and the timer Input Edge function. These configurations are described in the Programmable Timer and Serial Communications Interface (SCI) section.

The Port 2 three-state, TTL-compatible output buffers are capable of driving one Schottky TTL load and 30 pF, or CMOS devices using external pullup resistors.

PORT 2 DATA REGISTER

7	6	5	4	3	2	1	0	
PC2	PC1	PC0	P24	P23	P22	P21	P20	$0003

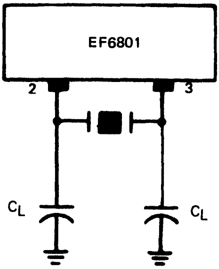
FIGURE 21 – 6801 FAMILY OSCILLATOR CHARACTERISTICS

(a) Nominal Recommended Crystal Parameters

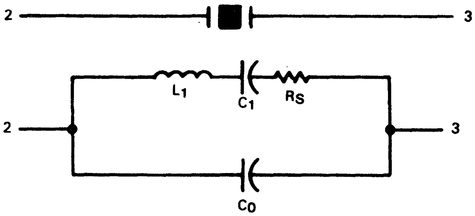
Nominal Crystal Parameters*

	3.58 MHz	4.00 MHz	5.0 MHz	6.0 MHz	8.0 MHz
RS	60 Ω	50 Ω	30-50 Ω	30-50 Ω	20-40 Ω
C_0	3.5 pF	6.5 pF	4-6 pF	4-6 pF	4-6 pF
C_1	0.015 pF	0.025 pF	0.01-0.02 pF	0.01-0.02 pF	0.01-0.02 pF
Q	>40 K	>30 K	>20 K	>20 K	>20 K

*NOTE: These are representative AT-cut crystal parameters only. Crystals of other types of cut may also be used.

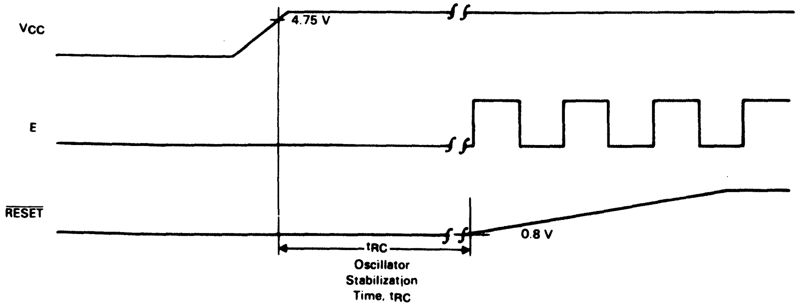


C_L = 20 pF (typical)



Equivalent Circuit

(b) Oscillator Stabilization Time (t_{RC})



P30-P37 (PORT 3)

Port 3 can be configured as an I/O port, a bidirectional 8-bit data bus, or a multiplexed address/data bus depending on the operating mode. The TTL compatible three-state output buffers can drive one Schottky TTL load and 90 pF. Unused lines can remain unconnected.

Port 3 In Single-Chip Mode

Port 3 is an 8-bit I/O port in the Single-Chip Mode, with each line configured by the Port 3 Data Direction Register. There are also two lines, $\overline{\text{IS3}}$ and $\overline{\text{OS3}}$, which can be used to control Port 3 data transfers.

Three Port 3 options are controlled by the Port 3 Control and Status Register and are available only in Single-Chip Mode: (1) Port 3 input data can be latched using $\overline{\text{IS3}}$ as a control signal, (2) $\overline{\text{OS3}}$ can be generated by either an MPU read or write to the Port 3 Data Register, and (3) an $\overline{\text{IRQ1}}$ interrupt can be enabled by an $\overline{\text{IS3}}$ negative edge. Port 3 latch timing is shown in Figure 5.

PORT 3 CONTROL AND STATUS REGISTER

7	6	5	4	3	2	1	0	
IS3 Flag	IS3 IRQ1 Enable	X	OSS	Latch Enable	X	X	X	$000F

Bit 0-2 Not used.

Bit 3 LATCH ENABLE. This bit controls the input latch for Port 3. If set, input data is latched by an $\overline{\text{IS3}}$ negative edge. The latch is transparent after a read of the Port 3 Data Register. LATCH ENALBLE is cleared during reset.

Bit 4 OSS (Output Strobe Select). This bit determines whether $\overline{\text{OS3}}$ will be generated by a read or write of the Port 3 Data Register. When clear, the strobe is generated by a read; when set, it is generated by a write. OSS is cleared during reset.

Bit 5 Not used.

Bit 6 IS3 IRQ1 ENABLE. When set, an $\overline{\text{IRQ1}}$ interrupt will be enabled whenever IS3 FLAG is set; when clear, the interrupt is inhibited. This bit is cleared during reset.

Bit 7 IS3 FLAG. This read-only status bit is set by an $\overline{\text{IS3}}$ negative edge. It is cleared by a read of the Port 3 Control and Status Register (with IS3 FLAG set) followed by a read or write to the Port 3 Data Register or during reset.

Port 3 In Expanded Non-Multiplexed Mode

Port 3 is configured as a bidirectional data bus (D7-D0) in the Expanded Non-Multiplexed Mode. The direction of data transfers is controlled by Read/Write (SC2). Data is clocked by E (Enable).

Port 3 In Expanded Multiplexed Mode

Port 3 is configured as a time multiplexed address (A0-A7) and data bus (D7-D0) in the Expanded Multiplexed Modes, where Address Strobe (AS) can be used to demultiplex the two buses. Port 3 is held in a high impedance state between valid address and data to prevent bus conflicts.

P40-P47 (PORT 4)

Port 4 is configured as an 8-bit I/O port, as address outputs, or as data inputs depending on the operating mode. Port 4 can drive one Schottky TTL load and 90 pF and is the only port with internal pullup resistors. Unused lines can remain unconnected.

Port 4 In Single-Chip Mode

In Single-Chip Mode, Port 4 functions as an 8-bit I/O port with each line configured by the Port 4 Data Direction Register. Internal pullup resistors allow the port to directly interface with CMOS at 5 volt levels. External pullup resistors to more than 5 volts, however, cannot be used.

Port 4 In Expanded Non-Multiplexed Mode

Port 4 is configured from reset as an 8-bit input port, where the Port 4 Data Direction Register can be written to provide any or all of eight address lines, A0 to A7. Internal pullup resistors pull the lines high until the Port 4 Data Direction Register is configured.

Port 4 In Expanded Multiplexed Mode

In all Expanded Multiplexed modes except Mode 6, Port 4 functions as half of the address bus and provides A8 to A15. In Mode 6, the port is configured from reset as an 8-bit parallel input port, where the Port 4 Data Direction Register can be written to provide any or all of upper address lines A8 to A15. Internal pullup resistors pull the lines high until the Port 4 Data Direction Register is configured, where bit 0 controls A8.

RESIDENT MEMORY

The EF6801 provides 2048 bytes of on-board ROM and 128 bytes of on-board RAM.

One half of the RAM is powered through the V_{CC} standby pin and is maintainable during V_{CC} powerdown. This standby portion of the RAM consists of 64 bytes located from $80 through $BF.

Power must be supplied to V_{CC} standby if the internal RAM is to be used regardless of whether standby power operation is anticipated.

The RAM is controlled by the RAM Control Register.

RAM CONTROL REGISTER ($14)

The RAM Control Register includes two bits which can be used to control RAM accesses and determine the adequacy of the standby power source during powerdown operation. It is intended that RAME be cleared and STBY PWR be set as part of a powerdown procedure.

RAM CONTROL REGISTER

7	6	5	4	3	2	1	0
STBY PWR	RAME	X	X	X	X	X	X

Bit 0-5	Not used.
Bit 6 RAME	RAM Enable. This read/write bit can be used to remove the entire RAM from the internal memory map. RAME is set (enabled) during reset provided standby power is available on the positive edge of $\overline{\text{RESET}}$. If RAME is clear, any access to a RAM address is external. If RAME is set and not in mode 3, the RAM is included in the internal map.
Bit 7 STBY PWR	Standby Power. This bit is a read/write status bit which, when once set, remains set as long as V_{CC} standby remains above V_{SBB} (minimum). As long as this bit is set following a period of standby operation, the standby power supply has adequately preserved the data in the standby RAM. If this bit is cleared during a period of standby operation, it indicates that V_{CC} standby had fallen to a level sufficiently below V_{SBB} (minimum) to suspect that data in the standby RAM is not valid. This bit can be set only by software and is not affected during reset.

PROGRAMMABLE TIMER

The programmable timer can be used to perform input waveform measurements while independently generating an output waveform. Pulse widths can vary from several microseconds to many seconds. A block diagram of the timer is shown in Figure 22.

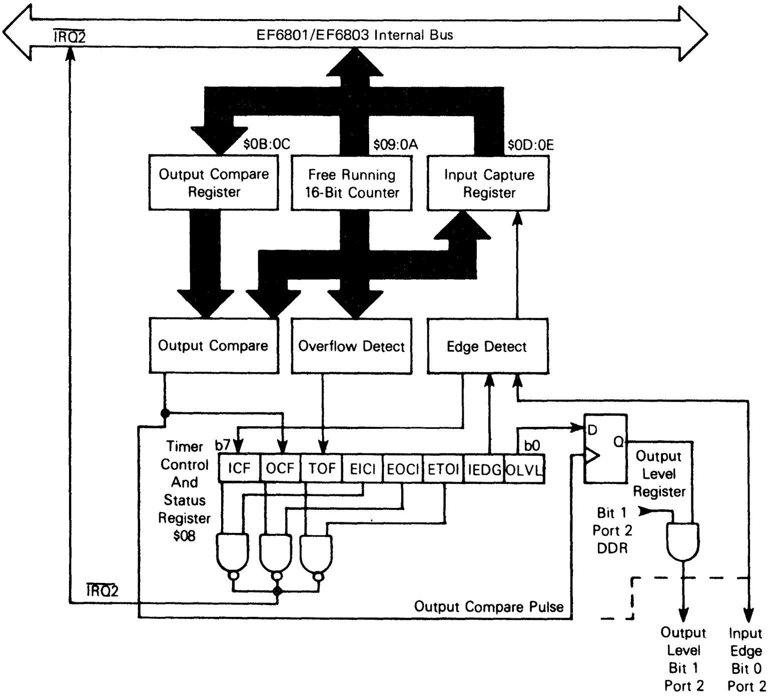
COUNTER ($09:0A)

The key timer element is a 16-bit free-running counter which is incremented by E (enable). It is cleared during reset and is read-only with one exception: a write to the counter ($09) will preset it to $FFF8. This feature, intended for testing, can disturb serial operations because the counter provides the SCI internal bit rate clock. TOF is set whenever the counter contains all ones.

OUTPUT COMPARE REGISTER ($0B:0C)

The output compare register is a 16-bit read/write register used to control an output waveform or provide an arbitrary timeout flag. It is compared with the free-running counter on each E cycle. When a match occurs, OCF is set and OLVL is clocked to an output level register. If port 2, bit 1, is configured as an output, OLVL will appear at P21 and the output compare register and OLVL can then be changed for the next

FIGURE 22 – BLOCK DIAGRAM OF PROGRAMMABLE TIMER



and OLVL is clocked to an output level register. If Port 2, bit 1, is configured as an output, OLVL will appear at P21 and the Output Compare Register and OLVL can then be changed for the next compare. The function is inhibited for one cycle after a write to its high byte ($0B) to ensure a valid compare. The Output Compare Register is set to $FFFF at RESET.

INPUT CAPTURE REGISTER ($0D:0E)

The Input Capture Register is a 16-bit read-only register used to store the free-running counter when a "proper" input transition occurs as defined by IEDG. Port 2, bit 0 should be configured as an input, but the edge detect circuit always senses P20 even when configured as an output. An input capture can occur independently of ICF: the register always contains the most current value. Counter transfer is inhibited, however, between accesses of a double byte MPU read. The input pulse width must be at least two E-cycles to ensure an input capture under all conditions.

TIMER CONTROL AND STATUS REGISTER ($08)

The Timer Control and Status Register (TCSR) is an 8-bit register of which all bits are readable, while only bits 0-4 can be written. The three most significant bits provide the timer status and indicate if:

- a proper level transition has been detected,
- a match has occured between the free-running counter and the output compare register, and
- the free-running counter has overflowed.

Each of the three events can generate an IRQ2 interrupt and is controlled by an individual enable bit in the TCSR.

TIMER CONTROL AND STATUS REGISTER (TCSR)

7	6	5	4	3	2	1	0	
ICF	OCF	TOF	EICI	EOCI	ETOI	IEDG	OLVL	$0008

Bit 0 OLVL — Output level. OLVL is clocked to the output level register by a successful output compare and will appear at P21 if Bit 1 of the Port 2 Data Direction Register is set. It is cleared during reset.

Bit 1 EIDG — Input Edge. IEDG is cleared during reset and controls which level transition will trigger a counter transfer to the Input Capture Register:

IEDG = 0 Transfer on a negative-edge

IEDG = 1 Transfer on a positive-edge.

Bit 2 ETOI — Enable Timer Overflow Interrupt. When set, an IRQ2 interrupt is enabled for a timer overflow; when clear, the interrupt is inhibited. It is cleared during reset.

Bit 3 EOCI — Enable Output Compare Interrupt. When set, an IRQ2 interrupt is enabled for an output compare; when clear, the interrupt is inhibited. It is cleared during reset.

Bit 4 EICI — Enable Input Capture Interrupt. When set, an IRQ2 interrupt is enabled for an input capture; when clear, the interrupt is inhibited. It is cleared during reset.

Bit 5 TOF — Timer Overflow Flag. TOF is set when the counter contains all 1's. It is cleared by reading the TCSR (with TOF set) then reading the counter high byte ($09), or during reset.

Bit 6 OCF — Output Compare Flag. OCF is set when the Output Compare Register matches the free-running counter. It is cleared by reading the TCSR (with OCF set) and then writing to the Output Compare Register ($0B or $0C), or during reset.

Bit 7 ICF — Input Capture Flag. ICF is set to indicate a proper level transition; it is cleared by reading the TCSR (with ICF set) and then the Input Capture Register High Byte ($0D), or during reset.

SERIAL COMMUNICATIONS INTERFACE (SCI)

A full-duplex asynchronous Serial Communications Interface (SCI) is provided with two data formats and a variety of rates. The SCI transmitter and receiver are functionally independent, but use the same data format and bit rate. Serial data formats include standard mark/space (NRZ) and Bi-phase and both provide one start bit, eight data bits, and one stop bit. "Baud" and "bit rate" are used synonymously in the following description.

WAKE-UP FEATURE

In a typical serial loop multi-processor configuration, the software protocol will usually identify the addressee(s) at the beginning of the message. In order to permit uninterested MPU's to ignore the remainder of the message, a wake-up feature is included whereby all further SCI receiver flag (and interrupt) processing can be inhibited until its data line goes idle. An SCI receiver is re-enabled by an idle string of ten consecutive 1's or during reset. Software must provide for the required idle string between consecutive messages and prevent it within messages.

PROGRAMMABLE OPTIONS

The following features of the SCI are programmable:

- format: standard mark/space (NRZ) or Bi-phase
- clock: external or internal bit rate clock
- Baud: one of 4 per E-clock frequency, or external clock (× 8 desired baud)
- wake-up feature: enabled or disabled
- interrupt requests: enabled individually for transmitter and receiver
- clock output: internal bit rate clock enabled or disabled to P22

SERIAL COMMUNICATIONS REGISTERS

The Serial Communications Interface includes four addressable registers as depicted in Figure 23. It is controlled by the Rate and Mode Control Register and the Transmit/Receive Control and Status Register. Data is transmitted and received utilizing a write-only Transmit Register and a read-only Receive Register. The shift registers are not accessible to software.

Rate and Mode Control Register (RMCR) ($10)

The Rate and Mode Control Register controls the SCI bit rate, format, clock source, and under certain conditions, the configuration of P22. The register consists of four write-only bits which are cleared during reset. The two least significant bits control the bit rate of the internal clock and the remaining two bits control the format and clock source.

RATE AND MODE CONTROL REGISTER (RMCR)

7	6	5	4	3	2	1	0	
X	X	X	X	CC1	CC0	SS1	SS0	$0010

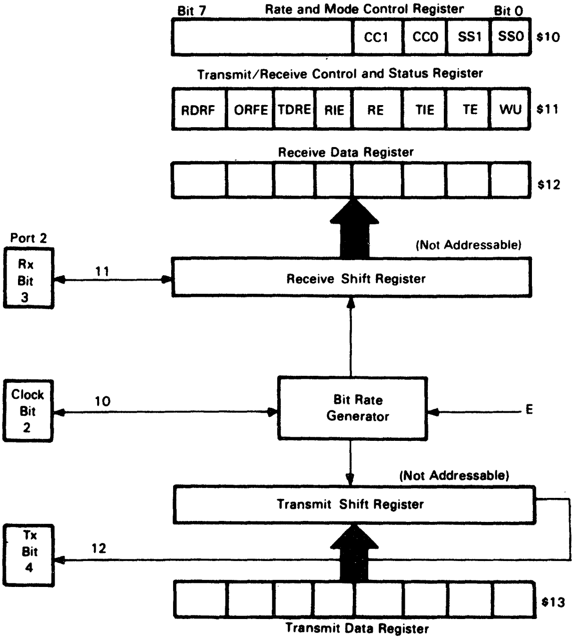
Bit 1:Bit 0 SS1:SS0 Speed Select. These two bits select the Baud rate when using the internal clock. Four rates may be selected which are a function of the MCU input frequency. Table 6 lists bit time and rates for three selected MCU frequencies.

Bit 3:Bit 2 CC1:CC0 Clock Control and Format Select. These two bits control the format and select the serial clock source. If CC1 is set, the DDR value for P22 is forced to the complement of CC0 and cannot be altered until CC1 is cleared. If CC1 is cleared after having been set, its DDR value is unchanged. Table 7 defines the formats, clock source, and use of P22.

If both CC1 and CC0 are set, an external TTL compatible clock must be connected to P22 at eight times (8X) the desired bit rate, but not greater than E, with a duty cycle of 50% (±10%). If CC1:CC0=10, the internal bit rate clock is provided at P22 regardless of the values for TE or RE.

NOTE: The source of SCI internal bit rate clock is the timer free running counter. An MPU write to the counter can disturb serial operations.

FIGURE 23 – SCI REGISTERS



Transmit/Receive Control And Status Register (TRCSR) ($11)

The Transmit/Receive Control and Status Register controls the transmitter, receiver, wake-up feature, and two individual interrupts and monitors the status of serial operations. All eight bits are readable while bits 0 to 4 are also writable. The register is initialized to $20 by $\overline{\text{RESET}}$.

TRANSMIT/RECEIVE CONTROL AND STATUS REGISTER (TRCSR)

7	6	5	4	3	2	1	0	
RDRF	ORFE	TDRE	RIE	RE	TIE	TE	WU	$0011

Bit 0 WU — "Wake-up" on Idle Line. When set, WU enables the wake-up function; it is cleared by ten consecutive 1's or during reset. WU will not set if the line is idle.

Bit 1 TE — Transmit Enable. When set, P24 DDR bit is set, cannot be changed, and will remain set if TE is subsequently cleared. When TE is changed from clear to set, the transmitter is connected to P24 and a preamble of nine consecutive 1's is transmitted. TE is cleared during reset.

Bit 2 TIE — Transmit Interrupt Enable. When set, an $\overline{\text{IRQ2}}$ interrupt is enabled when TDRE is set; when clear, the interrupt is inhibited. TE is cleared during reset.

Bit 3 RE — Receive Enable. When set, the P23 DDR bit is cleared, cannot be changed, and will remain clear if RE is subsequently cleared. While RE is set, the SCI receiver is enabled. RE is cleared during reset.

Bit 4 RIE — Receiver Interrupt Enable. When set, an $\overline{\text{IRQ2}}$ interrupt is enabled when RDRF and/or ORFE is set; when clear, the interrupt is inhibited. RIE is cleared during reset.

Bit 5 TDRE — Transmit Data Register Empty. TDRE is set when the Transmit Data Register is transferred to the output serial shift register or during reset. It is cleared by reading the TRCSR (with TDRE set) and then writing to the Transmit Data Register. Additional data will be transmitted only if TDRE has been cleared.

Bit 6 ORFE — Overrun Framing Error. If set, ORFE indicates either an overrun or framing error. An overrun is a new byte ready to transfer to the Receiver Data Register with RDRF still set. A receiver framing error has occurred when the byte boundaries of the bit stream are not synchronized to the bit counter. An overrun can be distinguished from a framing error by the state of RDRF: if RDRF is set, then an overrun has occurred; otherwise a framing error has been detected. Data is not transferred to the Receive Data Register in an overrun condition. Unframed data causing a framing error is transferred to the Receive Data Register. However, subsequent data transfer is blocked until the framing error flag is cleared.* ORFE is cleared by reading the 1RCSR (with ORFE set) then the Receive Data Register, or during reset.

Bit 7 RDRF — Receive Data Register Full. RDRF is set when the input serial shift register is transferred to the Receive Data Register. It is cleared by reading the TRCSR (with RDRF set), and then the Receive Data Register, or during reset.

TABLE 6 — SCI BIT TIMES AND RATES

SS1:SS0		$4f_o$→	2.4576 MHz	4.0 Mhz	4.9152 MHz
		E	614.4 kHz	1.0 MHz	1.2288 MHz
0	0	÷16	26 μs/38,400 Baud	16 μs/62,500 Baud	13.0 μs/76,800 Baud
0	1	÷128	208 μs/4,800 baud	128 μs/7812.5 Baud	104.2 μs/9,600 Baud
1	0	÷1024	1.67 ms/600 Baud$	1.024 ms/976.6 Baud	833.3 μs/1,200 Baud
1	1	÷4096	6.67 ms/150 Baud	4.096 ms/244.1 Baud	3.33 ms/300 Baud
*External (P22)			13.0 μs/76,800 Baud	8.0 μs/125,000 Baud	6.5 μs/153,600 Baud

*Using maximum clock rate

TABLE 7 — SCI FORMAT AND CLOCK SOURCE CONTROL

CC1:CC0	Format	Clock Source	Port 2 Bit 2
00	Bi-Phase	Internal	Not Used
01	NRZ	Internal	Not Used
10	NRZ	Internal	Output
11	NRZ	External	Input

SERIAL OPERATIONS

The SCI is initialized by writing control bytes first to the Rate and Mode Control Register and then to the Transmit/Receive Control and Status Register. When TE is set, the output of the transmit serial shift register is connected to P24 and serial output is initiated by transmitting a 9-bit preamble of 1's.

At this point one of two situations exist: 1) if the Transmit Data Register is empty (TDRE = 1), a continuous string of 1's will be sent indicating an idle line, or 2) if a byte has been written to the Transmit-Data Register (TDRE = 0), it will be transferred to the output serial shift register (synchronized with the bit rate clock), TDRE will be set, and transmission will begin.

The start bit (0), eight data bits (beginning with bit 0) and a stop bit (1), will be transmitted. If TDRE is still set when the next byte transfer should occur, 1's will be sent until more data is provided. In Bi-phase format, the output toggles at the start of each bit and at half-bit time when a "1" is sent. Receive operation is controlled by RE which configures P23 as an input and enables the receiver. SCI data formats are illustrated in Figure 24.

INSTRUCTION SET

The EF6801/03 is upward source and object code compatible with the EF6800. Execution times of key instructions have been reduced and several new instructions have been added, including a hardware multiply. A list of new operations added to the EF6800 instruction set is shown in Table 1.

In addition, two new special opcodes, 4E and 5E, are provided for test purposes. These opcodes force the Program Counter to increment like a 16-bit counter, causing address lines used in the expanded modes to increment until the device is reset. These opcodes have no mnemonics.

The coding of the first (or only) byte corresponding to an executable instruction is sufficient to identify the instruction and the addressing mode. The hexadecimal equivalents of the binary codes, which result from the translation of the 82 instructions in all valid modes of addressing, are shown in Table 8. There are 220 valid machine codes, 34 unassigned codes, and 2 codes reserved for test purposes.

PROGRAMMING MODEL

A programming model for the EF6801/03 is shown in Figure 10. Accumulator A can be concatenated with accumulator B and jointly referred to as accumulator D where A is the most significant byte. Any operation which modifies the double accumulator will also modify accumulator A and/or B. Other registers are defined as follows:

Program Counter – The program counter is a 16-bit register which always points to the next instruction.

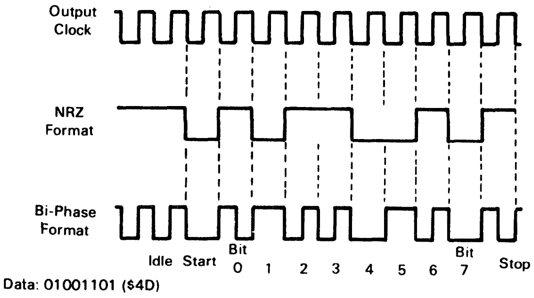
Stack Pointer – The stack pointer is a 16-bit register which contains the address of the next available location in a pushdown/pullup (LIFO) queue. The stack resides in random access memory at a location defined by the programmer.

Index Register – The Index Register is a 16-bit register which can be used to store data or provide an address for the indexed mode of addressing.

Accumulators – The MPU contains two 8-bit accumulators, A and B, which are used to store operands and results from the arithmetic logic unit (ALU). They can also be concatenated and referred to as the D (double) accumulator.

Condition Code Registers – The condition code register indicates the results of an instruction and includes the following five condition bits: Negative (N), Zero (Z), Overflow (V), Carry/Borrow from MSB (C), and Half Carry from bit 3 (H). These bits are testable by the conditional branch instructions. Bit 4 is the interrupt mask (I-bit) and inhibits all maskable interrupts when set. The two unused bits, B6 and B7, are read as ones.

FIGURE 24 – SCI DATA FORMATS



ADDRESSING MODES

Six addressing modes can be used to reference memory. A summary of addressing modes for all instructions is presented in Tables 9, 10, 11, and 12, where execution times are provided in E-cycles. Instruction execution times are summarized in Table 13. With an input frequency of 4 MHz, E-cycles are equivalent to microseconds. A cycle-by-cycle description of bus activity for each instruction is provided in Table 14 and a description of selected instructions is shown in Figure 25.

Immediate Addressing – The operand or "immediate byte(s)" is contained in the following byte(s) of the instruction where the number of bytes matches the size of the register. These are two or three byte instructions.

Direct Addressing – The least significant byte of the operand address is contained in the second byte of the instruction and the most significant byte is assumed to be $00. Direct addressing allows the user to access $00 through $FF using two byte instructions and execution time is reduced by eliminating the additional memory access. In most applications, the 256-byte area is reserved for frequently referenced data.

Extended Addressing – The second and third bytes of the instruction contain the absolute address of the operand. These are three byte instrutions.

Indexed Addressing – The unsigned offset contained in the second byte of the instruction is added with carry to the Index Register and used to reference memory without changing the Index Register. These are two byte instructions.

Inherent Addressing – The operand(s) are registers and no memory reference is required. These are single byte instructions.

Relative Addressing – Relative addressing is used only for branch instructions. If the branch condition is true, the Program Counter is overwritten with the sum of a signed single byte displacement in the second byte of the instruction and the current Program Counter. This provides a branch range of – 126 to 129 bytes from the first byte of the instruction. These are two byte instructions.

TABLE 8 – CPU INSTRUCTION MAP

OP	MNEM	MODE	~	#	OP	MNEM	MODE	~	#	OP	MNEM	MODE	~	#	OP	MNEM	MODE	~	#	OP	MNEM	MODE	~	#
00	•				34	DES	INHER	3	1	68	ASL	INDXD	6	2	9C	CPX	DIR	5	2	D0	SUBB	DIR	3	2
01	NOP	INHER	2	1	35	TXS		3	1	69	ROL		6	2	9D	JSR		5	2	D1	CMPB		3	2
02	•				36	PSHA		3	1	6A	DEC		6	2	9E	LDS		4	2	D2	SBCB		3	2
03	•				37	PSHB		3	1	6B	•				9F	STS	DIR	4	2	D3	ADDD		5	2
04	LSRD		3	1	38	PULX		5	1	6C	INC		6	2	40	SUBA	INDXD	4	2	D4	ANDB		3	2
05	ASLD		3	1	39	RTS		5	1	6D	TST		6	2	A1	CMPA		4	2	D5	BITB		3	2
06	TAP		2	1	3A	ABX		3	1	6E	JMP		3	2	A2	SBCA		4	2	D6	LDAB		3	2
07	TPA		2	1	3B	RTI		10	1	6F	CLR	INDXD	6	2	A3	SUBD		6	2	D7	STAB		3	2
08	INX		3	1	3C	PSHX		4	1	70	NEG	EXTND	6	3	A4	ANDA		4	2	D8	EORB		3	2
09	DEX		3	1	3D	MUL		10	1	71	•				A5	BITA		4	2	D9	ADCB		3	2
0A	CLV		2	1	3E	WAI		9	1	72	•				A6	LDAA		4	2	DA	ORAB		3	2
0B	SEV		2	1	3F	SWI		12	1	73	COM		6	3	A7	STAA		4	2	DB	ADDB		3	2
0C	CLC		2	1	40	NEGA		2	1	74	LSR		6	3	A8	EORA		4	2	DC	LDD		4	2
0D	SEC		2	1	41	•				75	•				A9	ADCA		4	2	DD	STD		4	2
0E	CLI		2	1	42	•				76	ROR		6	3	AA	ORAA		4	2	DE	LDX		4	2
0F	SEI		2	1	43	COMA		2	1	77	ASR		6	3	AB	ADDA		4	2	DF	STX	DIR	4	2
10	SBA		2	1	44	LSRA		2	1	78	ASL		6	3	AC	CPX		6	2	E0	SUBB	INDXD	4	2
11	CBA		2	1	45	•				79	ROL		6	3	AD	JSR		6	2	E1	CMPB		4	2
12	•				46	RORA		2	1	7A	DEC		6	3	AE	LDS		5	2	E2	SBCB		4	2
13	•				47	ASRA		2	1	7B	•				AF	STS	INDXD	5	2	E3	ADDD		6	2
14	•				48	ASLA		2	1	7C	INC		6	3	B0	SUBA	EXTND	4	3	E4	ANDB		4	2
15	•				49	ROLA		2	1	7D	TST		6	3	B1	CMPA		4	3	E5	BITB		4	2
16	TAB		2	1	4A	DECA		2	1	7E	JMP		3	3	B2	SBCA		4	3	E6	LDAB		4	2
17	TBA		2	1	4B	•				7F	CLR	EXTND	6	3	B3	SUBD		6	3	E7	STAB		4	2
18	•				4C	INCA		2	1	80	SUBA	IMMED	2	2	B4	ANDA		4	3	E8	EORB		4	2
19	DAA	INHER	2	1	4D	TSTA		2	1	81	CMPA		2	2	B5	BITA		4	3	E9	ADCB		4	2
1A	•				4E	T				82	SBCA		2	2	B6	LDAA		4	3	EA	ORAB		4	2
1B	ABA	INHER	2	1	4F	CLRA		2	1	83	SUBD		4	3	B7	STAA		4	3	EB	ADDB		4	2
1C	•				50	NEGB		2	1	84	ANDA		2	2	B8	EORA		4	3	EC	LDD		5	2
1D	•				51	•				85	BITA		2	2	B9	ADCA		4	3	ED	STD		5	2
1E	•				52	•				86	LDAA		2	2	BA	ORAA		4	3	EE	LDX		5	2
1F	•				53	COMB		2	1	87	•				BB	ADDA		4	3	EF	STX	INDXD	5	2
20	BRA	REL	3	2	54	LSRB		2	1	88	EORA		2	2	BC	CPX		6	3	F0	SUBB	EXTND	4	3
21	BRN		3	2	55	•				89	ADCA		2	2	BD	JSR		6	3	F1	CMPB		4	3
22	BHI		3	2	56	RORB		2	1	8A	ORAA		2	2	BE	LDS		5	3	F2	SBCB		4	3
23	BLS		3	2	57	ASRB		2	1	8B	ADDA		2	2	BF	STS	EXTND	5	3	F3	ADDD		6	3
24	BCC		3	2	58	ASLB		2	1	8C	CPX	IMMED	4	3	C0	SUBB	IMMED	2	2	F4	ANDB		4	3
25	BCS		3	2	59	ROLB		2	1	8D	BSR	REL	6	2	C1	CMPB		2	2	F5	BITB		4	3
26	BNE		3	2	5A	DECB		2	1	8E	LDS	IMMED	3	3	C2	SBCB		2	2	F6	LDAB		4	3
27	BEQ		3	2	5B	•				8F	•				C3	ADDD		4	3	F7	STAB		4	3
28	BVC		3	2	5C	INCB		2	1	90	SUBA	DIR	3	2	C4	ANDB		2	2	F8	EORB		4	3
29	BVS		3	2	5D	TSTB		2	1	91	CMPA		3	2	C5	BITB		2	2	F9	ADCB		4	3
2A	BPL		3	2	5E	T				92	SBCA		3	2	C6	LDAB		2	2	FA	ORAB		4	3
2B	BMI		3	2	5F	CLRB	INHER	2	1	93	SUBD		5	2	C7	•				FB	ADDB		4	3
2C	BGE		3	2	60	NEG	INDXD	6	2	94	ANDA		3	2	C8	EORB		2	2	FC	LDD		5	3
2D	BLT		3	2	61	•				95	BITA		3	2	C9	ADCB		2	2	FD	STD		5	3
2E	BGT		3	2	62	•				96	LDAA		3	2	CA	ORAB		2	2	FE	LDX		5	3
2F	BLE	REL	3	2	63	COM		6	2	97	STAA		3	2	CB	ADDB		2	2	FF	STX	EXTND	5	3
30	TSX	INHER	3	1	64	LSR		6	2	98	EORA		3	2	CC	LDD		3	3					
31	INS		3	1	65	•				99	ADCA		3	2	CD	•								
32	PULA		4	1	66	ROR		6	2	9A	ORAA		3	2	CD	LDX	IMMED	3	3					
33	PULB		4	1	67	ASR	INDXD	6	2	9B	ADDA		3	2	CF	•								

*UNDEFINED OP CODE

NOTES:

1. Addressing Modes

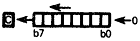
   INHER = Inherent　INDXD = Indexed　IMMED = Immediate
   REL = Relative　EXTND = Extended　DIR = Direct

2. Unassigned opcodes are indicated by "*" and should not be executed.
3. Codes marked by "T" force the PC to function as a 16-bit counter.

TABLE 9 — INDEX REGISTER AND STACK MANIPULATION INSTRUCTIONS

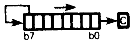
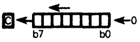
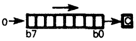
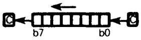
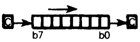
Pointer Operations	Mnemonic	Immed			Direct			Index			Extnd			Inherent			Boolean/ Arithmetic Operation	Condition Codes					
																		5	4	3	2	1	0
		OP	~	#	OP	~	#	OP	~	#	OP	~	#	OP	~	#		H	I	N	Z	V	C
Compare Index Reg	CPX	8C	4	3	9C	5	2	AC	6	2	BC	6	3				X - M : M + 1	●	●	↕	↕	↕	↕
Decrement Index Reg	DEX													09	3	1	X - 1 → X	●	●	●	↕	●	●
Decrement Stack Pntr	DES													34	3	1	SP - 1 → SP	●	●	●	●	●	●
Increment Index Reg	INX													08	3	1	X + 1 → X	●	●	●	↕	●	●
Increment Stack Pntr	INS													31	3	1	1 SP + 1 → SP	●	●	●	●	●	●
Load Index Reg	LDX	CE	3	3	DE	4	2	EE	5	2	FE	5	3				M → X_H, (M + 1) → X_L	●	●	↕	↕	R	●
Load Stack Pntr	LDS	8E	3	3	9E	4	2	AE	5	2	BE	5	3				M → SP_H, (M + 1) → SP_L	●	●	↕	↕	R	●
Store Index Reg	STX				DF	4	2	EF	5	2	FF	5	3				X_H → M, X_L → (M + 1)	●	●	↕	↕	R	●
Store Stack Pntr	STS				9F	4	2	AF	5	2	BF	5	3				SP_H → M, SP_L → (M + 1)	●	●	↕	↕	R	●
Index Reg → Stack Pntr	TXS													35	3	1	X - 1 → SP	●	●	●	●	●	●
Stack Pntr → Index Reg	TSX													30	3	1	SP + 1 → X	●	●	●	●	●	●
Add	ABX													3A	3	1	B + X → X	●	●	●	●	●	●
Push Data	PSHX													3C	4	1	X_L → M_{SP}, SP - 1 → SP X_H → M_{SP}, SP - 1 → SP	●	●	●	●	●	●
Pull Data	PULX													38	5	1	SP + 1 → SP, M_{SP} → X_H SP + 1 → SP, M_{SP} → X_L	●	●	●	●	●	●

TABLE 10 — ACCUMULATOR AND MEMORY INSTRUCTIONS

Accumulator and Memory Operations	MNE	Immed			Direct			Index			Extend			Inher			Boolean Expression	Condition Codes					
		Op	~	#	Op	~	#	Op	~	#	Op	~	#	Op	~	#		H	I	N	Z	V	C
Add Acmltrs	ABA													1B	2	1	A + B → A	↕	●	↕	↕	↕	↕
Add B to X	ABX													3A	3	1	00:B + X → X	●	●	●	●	●	●
Add with Carry	ADCA	89	2	2	99	3	2	A9	4	2	B9	4	3				A + M + C → A	↕	●	↕	↕	↕	↕
	ADCB	C9	2	2	D9	3	2	E9	4	2	F9	4	3				B + M + C → B	↕	●	↕	↕	↕	↕
Add	ADDA	8B	2	2	9B	3	2	AB	4	2	BB	4	3				A + M → A	↕	●	↕	↕	↕	↕
	ADDB	CB	2	2	DB	3	2	EB	4	2	FB	4	3				B + M → A	↕	●	↕	↕	↕	↕
Add Double	ADDD	C3	4	3	D3	5	2	E3	6	2	F3	6	3				D + M:M + 1 → D	●	●	↕	↕	↕	↕
And	ANDA	84	2	2	94	3	2	A4	4	2	B4	4	3				A · M → A	●	●	↕	↕	R	●
	ANDB	C4	2	2	D4	3	2	E4	4	2	F4	4	3				B · M → B	●	●	↕	↕	R	●
Shift Left, Arithmetic	ASL							68	6	2	78	6	3					●	●	↕	↕	↕	↕
	ASLA													48	2	1		●	●	↕	↕	↕	↕
	ASLB													58	2	1		●	●	↕	↕	↕	↕

— Continued —

TABLE 10 – ACCUMULATOR AND MEMORY INSTRUCTIONS (CONTINUED)

Accumulator and Memory Operations	MNE	Immed			Direct			Index			Extend			Inher			Boolean Expression	Condition Codes					
		Op	~	#	Op	~	#	Op	~	#	Op	~	#	Op	~	#		H	I	N	Z	V	C
Shift Left Dbl	ASLD													05	3	1		●	●	↕	↕	↕	↕
Shift Right, Arithmetic	ASR							67	6	2	77	6	3					●	●	↕	↕	↕	↕
	ASRA													47	2	1		●	●	↕	↕	↕	↕
	ASRB													57	2	1		●	●	↕	↕	↕	↕
Bit Test	BITA	85	2	2	95	3	2	A5	4	2	B5	4	3				A · M	●	●	↕	↕	R	●
	BITB	C5	2	2	D5	3	2	E5	4	2	F5	4	3				B · M	●	●	↕	↕	R	●
Compare Acmltrs	CBA													11	2	1	A - B	●	●	↕	↕	↕	↕
Clear	CLR							6F	6	2	7F	6	3				00 → M	●	●	R	S	R	R
	CLRA													4F	2	1	00 → A	●	●	R	S	R	R
	CLRB													5F	2	1	00 → B	●	●	R	S	R	R
Compare	CMPA	81	2	2	91	3	2	A1	4	2	B1	4	3				A - M	●	●	↕	↕	↕	↕
	CMPB	C1	2	2	D1	3	2	E1	4	2	F1	4	3				B - M	●	●	↕	↕	↕	↕
1's Complement	COM							63	6	2	73	6	3				$\overline{M}$ → M	●	●	↕	↕	R	S
	COMA													43	2	1	$\overline{A}$ → A	●	●	↕	↕	R	S
	COMB													53	2	1	$\overline{B}$ → B	●	●	↕	↕	R	S
Decimal Adj, A	DAA													19	2	1	Adj binary sum to BCD	●	●	↕	↕	↕	↕
Decrement	DEC							6A	6	2	7A	6	3				M - 1 → M	●	●	↕	↕	↕	●
	DECA													4A	2	1	A - 1 → A	●	●	↕	↕	↕	●
	DECB													5A	2	1	B - 1 → B	●	●	↕	↕	↕	●
Exclusive OR	EORA	88	2	2	98	3	2	A8	4	2	B8	4	3				A ⊕ M → A	●	●	↕	↕	R	●
	EORB	C8	2	2	D8	3	2	E8	4	2	F8	4	3				B ⊕ M → B	●	●	↕	↕	R	●
Increment	INC							6C	6	2	7C	6	3				M + 1 → M	●	●	↕	↕	↕	●
	INCA													4C	2	1	A + 1 → A	●	●	↕	↕	↕	●
	INCB													5C	2	1	B + 1 → B	●	●	↕	↕	↕	●
Load Acmltrs	LDAA	86	2	2	96	3	2	A6	4	2	B6	4	3				M → A	●	●	↕	↕	R	●
	LDAB	C6	2	2	D6	3	2	E6	4	2	F6	4	3				M → B	●	●	↕	↕	R	●
Load Double	LDD	CC	3	3	DC	4	2	EC	5	2	FC	5	3				M:M + 1 → D	●	●	↕	↕	R	●
Logical Shift, Left	LSL							68	6	2	78	6	3					●	●	↕	↕	↕	↕
	LSLA													48	2	1		●	●	↕	↕	↕	↕
	LSLB													58	2	1		●	●	↕	↕	↕	↕
	LSLD													05	3	1		●	●	↕	↕	↕	↕
Shift Right, Logical	LSR							64	6	2	74	6	3					●	●	R	↕	↕	↕
	LSRA													44	2	1		●	●	R	↕	↕	↕
	LSRB													54	2	1		●	●	R	↕	↕	↕
	LSRD													04	3	1		●	●	R	↕	↕	↕
Multiply	MUL													3D	10	1	A X B → D	●	●	●	●	●	↕
2's Complement (Negate)	NEG							60	6	2	70	6	3				00 - M → M	●	●	↕	↕	↕	↕
	NEGA													40	2	1	00 - A → A	●	●	↕	↕	↕	↕
	NEGB													50	2	1	00 - B → B	●	●	↕	↕	↕	↕
No Operation	NOP													01	2	1	PC + 1 → PC	●	●	●	●	●	●
Inclusive OR	ORAA	8A	2	2	9A	3	2	AA	4	2	BA	4	3				A + M → A	●	●	↕	↕	R	●
	ORAB	CA	2	2	DA	3	2	EA	4	2	FA	4	3				B + M → B	●	●	↕	↕	R	●
Push Data	PSHA													36	3	1	A → Stack	●	●	●	●	●	●
	PSHB													37	3	1	B → Stack	●	●	●	●	●	●
Pull Data	PULA													32	4	1	Stack → A	●	●	●	●	●	●
	PULB													33	4	1	Stack → B	●	●	●	●	●	●
Rotate Left	ROL							69	6	2	79	6	3					●	●	↕	↕	↕	↕
	ROLA													49	2	1		●	●	↕	↕	↕	↕
	ROLB													59	2	1		●	●	↕	↕	↕	↕
Rotate Right	ROR							66	6	2	76	6	3					●	●	↕	↕	↕	↕
	RORA													46	2	1		●	●	↕	↕	↕	↕
	RORB													56	2	1		●	●	↕	↕	↕	↕
Subtract Acmltr	SBA													10	2	1	A - B → A	●	●	↕	↕	↕	↕
Subtract with Carry	SBCA	82	2	2	92	3	2	A2	4	2	B2	4	3				A - M - C → A	●	●	↕	↕	↕	↕
	SBCB	C2	2	2	D2	3	2	E2	4	2	F2	4	3				B - M - C → B	●	●	↕	↕	↕	↕
Store Acmltrs	STAA				97	3	2	A7	4	2	B7	4	3				A → M	●	●	↕	↕	R	●
	STAB				D7	3	2	E7	4	2	F7	4	3				B → M	●	●	↕	↕	R	●
	STD				DD	4	2	ED	5	2	FD	5	3				D → M:M + 1	●	●	↕	↕	R	●
Subtract	SUBA	80	2	2	90	3	2	A0	4	2	B0	4	3				A - M → A	●	●	↕	↕	↕	↕
	SUBB	C0	2	2	D0	3	2	E0	4	2	F0	4	3				B - M → B	●	●	↕	↕	↕	↕
Subtract Double	SUBD	83	4	3	93	5	2	A3	6	2	B3	6	3				D - M:M + 1 → D	●	●	↕	↕	↕	↕
Transfer Acmltr	TAB													16	2	1	A → B	●	●	↕	↕	R	●
	TBA													17	2	1	B → A	●	●	↕	↕	R	●
Test, Zero or Minus	TST							6D	6	2	7D	6	3				M - 00	●	●	↕	↕	R	R
	TSTA													4D	2	1	A - 00	●	●	↕	↕	R	R
	TSTB													5D	2	1	B - 00	●	●	↕	↕	R	R

The Condition Code Register notes are listed after Table 12.

TABLE 11 – JUMP AND BRANCH INSTRUCTIONS

		Direct			Relative			Index			Extnd			Inherent				Cond. Code Reg.					
																		5	4	3	2	1	0
Operations	Mnemonic	OP	~	#	OP	~	#	OP	~	#	OP	~	#	OP	~	#	Branch Test	H	I	N	Z	V	C
Branch Always	BRA				20	3	2										None	●	●	●	●	●	●
Branch Never	BRN				21	3	2										None	●	●	●	●	●	●
Branch If Carry Clear	BCC				24	3	2										C = 0	●	●	●	●	●	●
Branch If Carry Set	BCS				25	3	2										C = 1	●	●	●	●	●	●
Branch If = Zero	BEQ				27	3	2										Z = 1	●	●	●	●	●	●
Branch If ≥ Zero	BGE				2C	3	2										N⊕V = 0	●	●	●	●	●	●
Branch If > Zero	BGT				2E	3	2										Z + (N⊕V) = 0	●	●	●	●	●	●
Branch If Higher	BHI				22	3	2										C + Z = 0	●	●	●	●	●	●
Branch If Higher or Same	BHS				24	3	2										C = 0	●	●	●	●	●	●
Branch If ≤ Zero	BLE				2F	3	2										Z + (N⊕V) = 1	●	●	●	●	●	●
Branch If Carry Set	BLO				25	3	2										C = 1	●	●	●	●	●	●
Branch If Lower Or Same	BLS				23	3	2										C + Z = 1	●	●	●	●	●	●
Branch If < Zero	BLT				2D	3	2										N⊕V = 1	●	●	●	●	●	●
Branch If Minus	BMI				2B	3	2										N = 1	●	●	●	●	●	●
Branch If Not Equal Zero	BNE				26	3	2										Z = 0	●	●	●	●	●	●
Branch If Overflow Clear	BVC				28	3	2										V = 0	●	●	●	●	●	●
Branch If Overflow Set	BVS				29	3	2										V = 1	●	●	●	●	●	●
Branch If Plus	BPL				2A	3	2										N = 0	●	●	●	●	●	●
Branch To Subroutine	BSR				8D	6	2										See Special Operations - Figure 26	●	●	●	●	●	●
Jump	JMP							6E	3	2	7E	3	3					●	●	●	●	●	●
Jump To Subroutine	JSR	9D	5	2				AD	6	2	BD	6	3					●	●	●	●	●	●
No Operation	NOP													01	2	1		●	●	●	●	●	●
Return From Interrupt	RTI													3B	10	1	See Special Operations - Figure 26	↕	↕	↕	↕	↕	↕
Return From Subroutine	RTS													39	5	1		●	●	●	●	●	●
Software Interrupt	SWI													3F	12	1		●	S	●	●	●	●
Wait For Interrupt	WAI													3E	9	1		●	●	●	●	●	●

TABLE 12 – CONDITION CODE REGISTER MANIPULATION INSTRUCTIONS

	Inherent					Cond. Code Reg.					
						5	4	3	2	1	0
Operations	Mnemonic	OP	~	#	Boolean Operation	H	I	N	Z	V	C
Clear Carry	CLC	0C	2	1	0 → C	●	●	●	●	●	R
Clear Interrupt Mask	CLI	0E	2	1	0 → I	●	R	●	●	●	●
Clear Overflow	CLV	0A	2	1	0 → V	●	●	●	●	R	●
Set Carry	SEC	0D	2	1	1 → C	●	●	●	●	●	S
Set Interrupt Mask	SEI	0F	2	1	1 → I	●	S	●	●	●	●
Set Overflow	SEV	0B	2	1	1 → V	●	●	●	●	S	●
Accumulator A → CCR	TAP	06	2	1	A → CCR	↕	↕	↕	↕	↕	↕
CCR → Accumulator A	TPA	07	2	1	CCR → A	●	●	●	●	●	●

LEGEND

OP Operation Code (Hexadecimal)
~ Number of MPU Cycles
M_{SP} Contents of memory location pointed to by Stack Pointer
# Number of Program Bytes
+ Arithmetic Plus
- Arithmetic Minus
● Boolean AND
X Arithmetic Multiply
+ Boolean Inclusive OR
⊕ Boolean Exclusive OR
$\overline{M}$ Complement of M
→ Transfer Into
0 Bit = Zero
00 Byte = Zero

CONDITION CODE SYMBOLS

H Half-carry from bit 3
I Interrupt mask
N Negative (sign bit)
Z Zero (byte)
V Overflow, 2's complement
C Carry/Borrow from MSB
R Reset Always
S Set Always
↕ Affected
● Not Affected

TABLE 13 — INSTRUCTION EXECUTION TIMES IN E-CYCLES

	ADDRESSING MODE					
	Immediate	Direct	Extended	Indexed	Inherent	Relative
ABA	●	●	●	●	2	●
ABX	●	●	●	●	3	●
ADC	2	3	4	4	●	●
ADD	2	3	4	4	●	●
ADDD	4	5	6	6	●	●
AND	2	3	4	4	●	●
ASL	●	●	6	6	2	●
ASLD	●	●	●	●	3	●
ASR	●	●	6	6	2	●
BCC	●	●	●	●	●	3
BCS	●	●	●	●	●	3
BEQ	●	●	●	●	●	3
BGE	●	●	●	●	●	3
BGT	●	●	●	●	●	3
BHI	●	●	●	●	●	3
BHS	●	●	●	●	●	3
BIT	2	3	4	4	●	●
BLE	●	●	●	●	●	3
BLO	●	●	●	●	●	3
BLS	●	●	●	●	●	3
BLT	●	●	●	●	●	3
BMI	●	●	●	●	●	3
BNE	●	●	●	●	●	3
BPL	●	●	●	●	●	3
BRA	●	●	●	●	●	3
BRN	●	●	●	●	●	3
BSR	●	●	●	●	●	6
BVC	●	●	●	●	●	3
BVS	●	●	●	●	●	3
CBA	●	●	●	●	2	●
CLC	●	●	●	●	2	●
CLI	●	●	●	●	2	●
CLR	●	●	6	6	2	●
CLV	●	●	●	●	2	●
CMP	2	3	4	4	●	●
COM	●	●	6	6	2	●
CPX	4	5	6	6	●	●
DAA	●	●	●	●	2	●
DEC	●	●	6	6	2	●
DES	●	●	●	●	3	●
DEX	●	●	●	●	3	●
EOR	2	3	4	4	●	●
INC	●	●	6	6	●	●
INS	●	●	●	●	3	●
INX	●	●	●	●	3	●
JMP	●	●	3	3	●	●
JSR	●	5	6	6	●	●
LDA	2	3	4	4	●	●
LDD	3	4	5	5	●	●
LDS	3	4	5	5	●	●
LDX	3	4	5	5	●	●
LSL	●	●	6	6	2	●
LSLD	●	●	●	●	3	●
LSR	●	●	6	6	2	●
LSRD	●	●	●	●	3	●
MUL	●	●	●	●	10	●
NEG	●	●	6	6	2	●
NOP	●	●	●	●	2	●
ORA	2	3	4	4	●	●
PSH	●	●	●	●	3	●
PSHX	●	●	●	●	4	●
PUL	●	●	●	●	4	●
PULX	●	●	●	●	5	●
ROL	●	●	6	6	2	●
ROR	●	●	6	6	2	●
RTI	●	●	●	●	10	●
RTS	●	●	●	●	5	●
SBA	●	●	●	●	2	●
SBC	2	3	4	4	●	●
SEC	●	●	●	●	2	●
SEI	●	●	●	●	2	●
SEV	●	●	●	●	2	●
STA	●	3	4	4	●	●
STD	●	4	5	5	●	●
STS	●	4	5	5	●	●
STX	●	4	5	5	●	●
SUB	2	3	4	4	●	●
SUBD	4	5	6	6	●	●
SWI	●	●	●	●	12	●
TAB	●	●	●	●	2	●
TAP	●	●	●	●	2	●
TBA	●	●	●	●	2	●
TPA	●	●	●	●	2	●
TST	●	●	6	6	2	●
TSX	●	●	●	●	3	●
TXS	●	●	●	●	3	●
WAI	●	●	●	●	9	●

SUMMARY OF CYCLE-BY-CYCLE OPERATION

Table 14 provides a detailed description of the information present on the address bus, data bus, and the read/write (R/$\overline{W}$) line during each cycle of each instruction.

The information is useful in comparing actual with expected results during debug of both software and hardware as the program is executed. The information is categorized in groups according to addressing mode and number of cycles per instruction. In general, instructions with the same addressing mode and number of cycles execute in the same manner. Exceptions are indicated in the table.

Note that during MPU reads of internal locations, the resultant value will not appear on the external data bus except in mode 0. "High order" byte refers to the most-significant byte of a 16-bit value.

TABLE 14 – CYCLE-BY-CYCLE OPERATION (Sheet 1 of 5)

Address Mode and Instructions	Cycles	Cycle #	Address Bus	R/$\overline{W}$ Line	Data Bus
IMMEDIATE					
ADC EOR ADD LDA AND ORA BIT SBC CMP SUB	2	1 2	Opcode Address Opcode Address + 1	1 1	Opcode Operand Data
LDS LDX LDD	3	1 2 3	Opcode Address Opcode Address + 1 Opcode Address + 2	1 1 1	Opcode Operand Data (High Order Byte) Operand Data (Low Order Byte)
CPX SUBD ADDD	4	1 2 3 4	Opcode Address Opcode Address + 1 Opcode Address + 2 Address Bus FFFF	1 1 1 1	Opcode Operand Data (High Order Byte) Operand Data (Low Order Byte) Low Byte of Restart Vector
DIRECT					
ADC EOR ADD LDA AND ORA BIT SBC CMP SUB	3	1 2 3	Opcode Address Opcode Address + 1 Address of Operand	1 1 1	Opcode Address of Operand Operand Data
STA	3	1 2 3	Opcode Address Opcode Address + 1 Destination Address	1 1 0	Opcode Destination Address Data from Accumulator
LDS LDX LDD	4	1 2 3 4	Opcode Address Opcode Address + 1 Address of Operand Operand Address + 1	1 1 1 1	Opcode Address of Operand Operand Data (High Order Byte) Operand Data (Low Order Byte)
STS STX STD	4	1 2 3 4	Opcode Address Opcode Address + 1 Address of Operand Address of Operand + 1	1 1 0 0	Opcode Address of Operand Register Data (High Order Byte) Register Data (Low Order Byte)
CPX SUBD ADDD	5	1 2 3 4 5	Opcode Address Opcode Address + 1 Operand Address Operand Address + 1 Address Bus FFFF	1 1 1 1 1	Opcode Address of Operand Operand Data (High Order Byte) Operand Data (Low Order Byte) Low Byte of Restart Vector
JSR	5	1 2 3 4 5	Opcode Address Opcode Address + 1 Subroutine Address Stack Pointer Stack Pointer – 1	1 1 1 0 0	Opcode Irrelevant Data First Subroutine Opcode Return Address (Low Order Byte) Return Address (High Order Byte)

TABLE 14 — CYCLE-BY-CYCLE OPERATION (Sheet 2 of 5)

Address Mode and Instructions	Cycles	Cycle #	Address Bus	$R/\overline{W}$ Line	Data Bus
EXTENDED					
JMP	3	1	Opcode Address	1	Opcode
		2	Opcode Address + 1	1	Jump Address (High Order Byte)
		3	Opcode Address + 2	1	Jump Address (Low Order Byte)
ADC EOR	4	1	Opcode Address	1	Opcode
ADD LDA		2	Opcode Address + 1	1	Address of Operand
AND ORA		3	Opcode Address + 2	1	Address of Operand (Low Order Byte)
BIT SBC		4	Address of Operand	1	Operand Data
CMP SUB					
STA	4	1	Opcode Address	1	Opcode
		2	Opcode Address + 1	1	Destination Address (High Order Byte)
		3	Opcode Address + 2	1	Destination Address (Low Order Byte)
		4	Operand Destination Address	0	Data from Accumulator
LDS	5	1	Opcode Address	1	Opcode
LDX		2	Opcode Address + 1	1	Address of Operand (High Order Byte)
LDD		3	Opcode Address + 2	1	Address of Operand (Low Order Byte)
		4	Address of Operand	1	Operand Data (High Order Byte)
		5	Address of Operand + 1	1	Operand Data (Low Order Byte)
STS	5	1	Opcode Address	1	Opcode
STX		2	Opcode Address + 1	1	Address of Operand (High Order Byte)
STD		3	Opcode Address + 2	1	Address of Operand (Low Order Byte)
		4	Address of Operand	0	Operand Data (High Order Byte)
		5	Address of Operand + 1	0	Operand Data (Low Order Byte)
ASL LSR	6	1	Opcode Address	1	Opcode
ASR NEG		2	Opcode Address + 1	1	Address of Operand (High Order Byte)
CLR ROL		3	Opcode Address + 2	1	Address of Operand (Low Order Byte)
COM ROR		4	Address of Operand	1	Current Operand Data
DEC TST*		5	Address Bus FFFF	1	Low Byte of Restart Vector
INC		6	Address of Operand	0	New Operand Data
CPX	6	1	Opcode Address	1	Opcode
SUBD		2	Opcode Address + 1	1	Operand Address (High Order Byte)
ADDD		3	Opcode Address + 2	1	Operand Address (Low Order Byte)
		4	Operand Address	1	Operand Data (High Order Byte)
		5	Operand Address + 1	1	Operand Data (Low Order Byte)
		6	Address Bus FFFF	1	Low Byte of Restart Vector
JSR	6	1	Opcode Address	1	Opcode
		2	Opcode Address + 1	1	Address of Subroutine (High Order Byte)
		3	Opcode Address + 2	1	Address of Subroutine (Low Order Byte)
		4	Subroutine Starting Address	1	Opcode of Next Instruction
		5	Stack Pointer	0	Return Address (Low Order Byte)
		6	Stack Pointer − 1	0	Return Address (High Order Byte)

*TST does not perform the write cycle during the sixth cycle. The sixth cycle is another address bus = $FFFF.

TABLE 14 — CYCLE-BY-CYCLE OPERATION (Sheet 3 of 5)

Address Mode and Instructions	Cycles	Cycle #	Address Bus	R/$\overline{W}$ Line	Data Bus
INDEXED					
JMP	3	1	Opcode Address	1	Opcode
		2	Opcode Address + 1	1	Offset
		3	Address Bus FFFF	1	Low Byte of Restart Vector
ADC EOR	4	1	Opcode Address	1	Opcode
ADD LDA		2	Opcode Address + 1	1	Offset
AND ORA		3	Address Bus FFFF	1	Low Byte of Restart Vector
BIT SBC		4	Index Register Plus Offset	1	Operand Data
CMP SUB					
STA	4	1	Opcode Address	1	Opcode
		2	Opcode Address + 1	1	Offset
		3	Address Bus FFFF	1	Low Byte of Restart Vector
		4	Index Register Plus Offset	0	Operand Data
LDS	5	1	Opcode Address	1	Opcode
LDX		2	Opcode Address + 1	1	Offset
LDD		3	Address Bus FFFF	1	Low Byte of Restart Vector
		4	Index Register Plus Offset	1	Operand Data (High Order Byte)
		5	Index Register Plus Offset + 1	1	Operand Data (Low Order Byte)
STS	5	1	Opcode Address	1	Opcode
STX		2	Opcode Address + 1	1	Offset
STD		3	Address Bus FFFF	1	Low Byte of Restart Vector
		4	Index Register Plus Offset	0	Operand Data (High Order Byte)
		5	Index Register Plus Offset + 1	0	Operand Data (Low Order Byte)
ASL LSR	6	1	Opcode Address	1	Opcode
ASR NEG		2	Opcode Address + 1	1	Offset
CLR ROL		3	Address Bus FFFF	1	Low Byte of Restart Vector
COM ROR		4	Index Register Plus Offset	1	Current Operand Data
DEC TST*		5	Address Bus FFFF	1	Low Byte of Restart Vector
INC		6	Index Register Plus Offset	0	New Operand Data
CPX	6	1	Opcode Address	1	Opcode
SUBD		2	Opcode Address + 1	1	Offset
ADDD		3	Address Bus FFFF	1	Low Byte of Restart Vector
		4	Index Register + Offset	1	Operand Data (High Order Byte)
		5	Index Register + Offset + 1	1	Operand Data (Low Order Byte)
		6	Address Bus FFFF		Low Byte of Restart Vector
JSR	6	1	Opcode Address	1	Opcode
		2	Opcode Address + 1	1	Offset
		3	Address Bus FFFF	1	Low Byte of Restart Vector
		4	Index Register + Offset	1	First Subroutine Opcode
		5	Stack Pointer	0	Return Address (Low Order Byte)
		6	Stack Pointer − 1	0	Return Address (High Order Byte)

*TST does not perform the write cycle during the sixth cycle. The sixth cycle is another address bus = $FFFF.

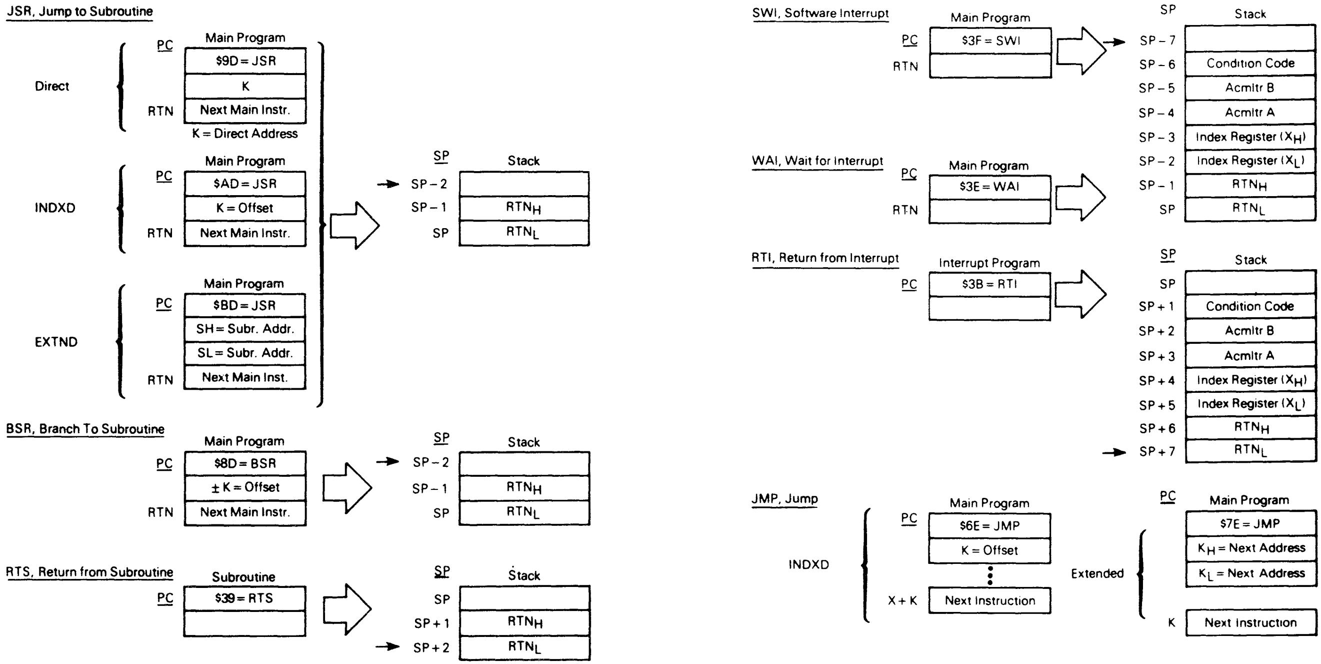
TABLE 14 – CYCLE-BY-CYCLE OPERATION (Sheet 4 of 5)

Address Mode and Instructions	Cycles	Cycle #	Address Bus	R/$\overline{W}$ Line	Data Bus
INHERENT					
ABA DAA SEC	2	1	Opcode Address	1	Opcode
ASL DEC SEI		2	Opcode Address + 1	1	Opcode of Next Instruction
ASR INC SEV					
CBA LSR TAB					
CLC NEG TAP					
CLI NOP TBA					
CLR ROL TPA					
CLV ROR TST					
COM SBA					
ABX	3	1	Opcode Address	1	Opcode
		2	Opcode Address + 1	1	Irrelevant Data
		3	Address Bus FFFF	1	Low Byte of Restart Vector
ASLD	3	1	Opcode Address	1	Opcode
LSRD		2	Opcode Address + 1	1	Irrelevant Data
		3	Address Bus FFFF	1	Low Byte of Restart Vector
DES	3	1	Opcode Address	1	Opcode
INS		2	Opcode Address + 1	1	Opcode of Next Instruction
		3	Previous Stack Pointer Contents	1	Irrelevant Data
INX	3	1	Opcode Address	1	Opcode
DEX		2	Opcode Address + 1	1	Opcode of Next Instruction
		3	Address Bus FFFF	1	Low Byte of Restart Vector
PSHA	3	1	Opcode Address	1	Opcode
PSHB		2	Opcode Address + 1	1	Opcode of Next Instruction
		3	Stack Pointer	0	Accumulator Data
TSX	3	1	Opcode Address	1	Opcode
		2	Opcode Address + 1	1	Opcode of Next Instruction
		3	Stack Pointer	1	Irrelevant Data
TXS	3	1	Opcode Address	1	Opcode
		2	Opcode Address + 1	1	Opcode of Next Instruction
		3	Address Bus FFFF	1	Low Byte of Restart Vector
PULA	4	1	Opcode Address	1	Opcode
PULB		2	Opcode Address + 1	1	Opcode of Next Instruction
		3	Stack Pointer	1	Irrelevant Data
		4	Stack Pointer + 1	1	Operand Data from Stack
PSHX	4	1	Opcode Address	1	Opcode
		2	Opcode Address + 1	1	Irrelevant Data
		3	Stack Pointer	0	Index Register (Low Order Byte)
		4	Stack Pointer – 1	0	Index Register (High Order Byte)
PULX	5	1	Opcode Address	1	Opcode
		2	Opcode Address + 1	1	Irrelevant Data
		3	Stack Pointer	1	Irrelevant Data
		4	Stack Pointer + 1	1	Index Register (High Order Byte)
		5	Stack Pointer + 2	1	Index Register (Low Order Byte)
RTS	5	1	Opcode Address	1	Opcode
		2	Opcode Address + 1	1	Irrelevant Data
		3	Stack Pointer	1	Irrelevant Data
		4	Stack Pointer + 1	1	Address of Next Instruction (High Order Byte)
		5	Stack Pointer + 2	1	Address of Next Instruction (Low Order Byte)
WAI	9	1	Opcode Address	1	Opcode
		2	Opcode Address + 1	1	Opcode of Next Instruction
		3	Stack Pointer	0	Return Address (Low Order Byte)
		4	Stack Pointer – 1	0	Return Address (High Order Byte)
		5	Stack Pointer – 2	0	Index Register (Low Order Byte)
		6	Stack Pointer – 3	0	Index Register (High Order Byte)
		7	Stack Pointer – 4	0	Contents of Accumulator A
		8	Stack Pointer – 5	0	Contents of Accumulator B
		9	Stack Pointer – 6	0	Contents of Condition Code Register

TABLE 14 – CYCLE-BY-CYCLE OPERATION (Sheet 5 of 5)

Address Mode and Instructions	Cycles	Cycle #	Address Bus	R/$\overline{W}$ Line	Data Bus
INHERENT					
MUL	10	1	Opcode Address	1	Opcode
		2	Opcode Address + 1	1	Irrelevant Data
		3	Address Bus FFFF	1	Low Byte of Restart Vector
		4	Address Bus FFFF	1	Low Byte of Restart Vector
		5	Address Bus FFFF	1	Low Byte of Restart Vector
		6	Address Bus FFFF	1	Low Byte of Restart Vector
		7	Address Bus FFFF	1	Low Byte of Restart Vector
		8	Address Bus FFFF	1	Low Byte of Restart Vector
		9	Address Bus FFFF	1	Low Byte of Restart Vector
		10	Address Bus FFFF	1	Low Byte of Restart Vector
RTI	10	1	Opcode Address	1	Opcode
		2	Opcode Address + 1	1	Irrelevant Data
		3	Stack Pointer	1	Irrelevant Data
		4	Stack Pointer + 1	1	Contents of Condition Code Register from Stack
		5	Stack Pointer + 2	1	Contents of Accumulator B from Stack
		6	Stack Pointer + 3	1	Contents of Accumulator A from Stack
		7	Stack Pointer + 4	1	Index Register from Stack (High Order Byte)
		8	Stack Pointer + 5	1	Index Register from Stack (Low Order Byte)
		9	Stack Pointer + 6	1	Next Instruction Address from Stack (High Order Byte)
		10	Stack Pointer + 7	1	Next Instruction Address from Stack (Low Order Byte)
SWI	12	1	Opcode Address	1	Opcode
		2	Opcode Address + 1	1	Irrelevant Data
		3	Stack Pointer	0	Return Address (Low Order Byte)
		4	Stack Pointer − 1	0	Return Address (High Order Byte)
		5	Stack Pointer − 2	0	Index Register (Low Ordᵉr Byte)
		6	Stack Pointer − 3	0	Index Register (High Order Byte)
		7	Stack Pointer − 4	0	Contents of Accumulator A
		8	Stack Pointer − 5	0	Contents of Accumulator B
		9	Stack Pointer − 6	0	Contents of Condition Code Register
		10	Stack Pointer − 7	1	Irrelevant Data
		11	Vector Address FFFA (Hex)	1	Address of Subroutine (High Order Byte)
		12	Vector Address FFFB (Hex)	1	Address of Subroutine (Low Order Byte)
RELATIVE					
BCC BHT BNE BLO BCS BLE BPL BHS BEQ BLS BRA BRN BGE BLT BVC BGT BMI BVS	3	1	Opcode Address	1	Opcode
		2	Opcode Address + 1	1	Branch Offset
		3	Address Buss FFFF	1	Low Byte of Restart Vector
BSR	6	1	Opcode Address	1	Opcode
		2	Opcode Address + 1	1	Branch Offset
		3	Address Bus FFFF	1	Low Byte of Restart Vector
		4	Subroutine Starting Address	1	Opcode of Next Instruction
		5	Stack Pointer	0	Return Address (Low Order Byte)
		6	Stack Pointer − 1	0	Return Address (High Order Byte)

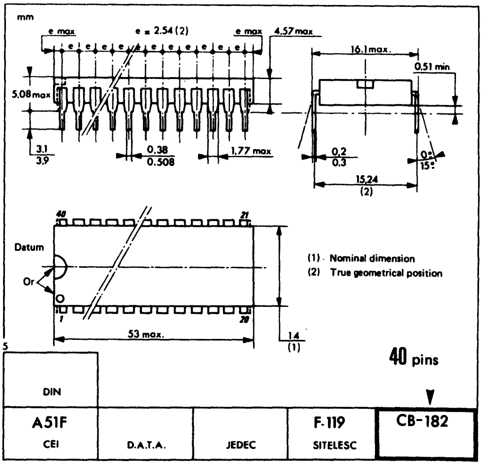
FIGURE 25 – SPECIAL OPERATIONS



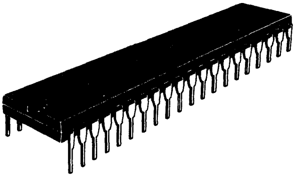
Legend:

RTN = Address of next instruction in Main Program to be executed upon return from subroutine
RTN$_H$ = Most significant byte of Return Address
RTN$_L$ = Least significant byte of Return Address
→ = Stack Pointer After Execution
K = 8-bit Unsigned Value

PHYSICAL DIMENSIONS



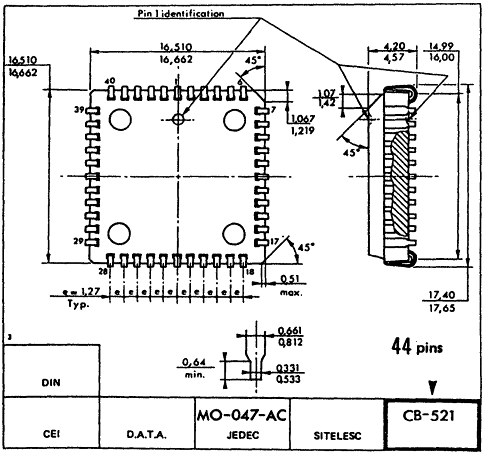
CB-182



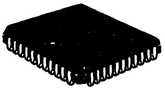
P SUFFIX
PLASTIC PACKAGE

ALSO AVAILABLE

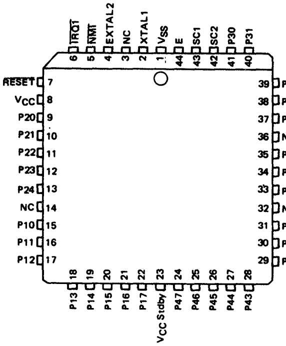
C SUFFIX
CERAMIC PACKAGE



CB-521



FN SUFFIX
PLCC 44



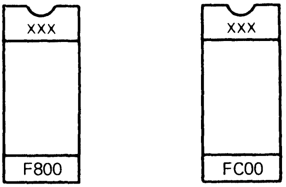
ORDERING INFORMATION

The information required when ordering a custom MCU is listed below. The ROM program may be transmitted to THOMSON SEMICONDUCTEURS on EPROM(s) or an EFDOS/MDOS* disk file.

To initiate a ROM pattern for the MCU, it is necessary to first contact your local THOMSON SEMICONDUCTEURS representative or distributor.

EPROMs

Two 2708 or one 2716 type EPROMs, programmed with the customer program (positive logic sense for address and data), may be submitted for pattern generation. The EPROM must be clearly marked to indicate which EPROM corresponds to which address space. The recommended marking procedure is illustrated below :



XXX = Customer ID

After the EPROM(s) are marked, they should be placed in conductive IC carriers and securely packed. Do not use styrofoam.

VERIFICATION MEDIA

All original pattern media (EPROMs or floppy disk) are filed for contractual purposes and are not returned. A computer listing of the ROM code will be generated and returned along with a listing verification form. The listing should be thoroughly checked and the verification form completed, signed, and returned to THOMSON SEMICONDUCTEURS. The signed verification form constitutes the contractual agreement for creation of the customer mask. If desired, THOMSON SEMICONDUCTEURS will program on blank EPROM from the data file used to create the custom mask and aid in the verification process.

ROM VERIFICATION UNITS (RVUs)

Ten MCUs containing the customer's ROM pattern will be sent for program verification. These units will have been made using the custom mask but are for the purpose of ROM verification only. For expediency they are usually unmarked, packaged in ceramic, and tested only at room temperature and 5 volts. These RVUs are included in the mask charge and are not production parts. The RVUs are thus not guaranteed by THOMSON SEMICONDUCTEURS. Quality Assurance, and should be discarded after verification is completed.

FLEXIBLE DISKS

The disk media submitted must be single-sided, EFDOS/MDOS* compatible floppies.

The customer must write the binary file name and company name on the disk with a felt-tip-pen. The minimum EFDOS/MDOS* system files, as well as the absolute binary object file (Filename .LO type of file) from the 6801 cross assembler, must be on the disk. An object file made from a memory dump using the ROLLOUT command is also acceptable. Consider submitting a source listing as well as the following files : filename .LX (DEVICE/EXORciser loadable format) and filename .SA (ASCII Source Code). These files will of course be kept confidential and are used 1) to speed up the process in-house if any problems arise, and 2) to speed up the user-to-factory interface if the user finds any software errors and needs assistance quickly from THOMSON SEMICONDUCTEURS factory representatives.

EFDOS is THOMSON SEMICONDUCTEURS' Disk Operating System available on development systems such as DEVICE,...

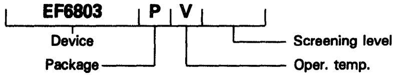
MDOS* is MOTOROLA's Disk Operating System available on development systems such as EXORciser,...

*Requires prior factory approval.

Whenever ordering a custom MCU is required, please contact your local THOMSON SEMICONDUCTEURS representative or THOMSON SEMICONDUCTEURS distributor and/or complete and send the attached "MCU customer ordering sheet" to your local THOMSON SEMICONDUCTEURS representative.

EXORciser is a registered trade mark of MOTOROLA Inc.

ORDERING INFORMATION



The table below horizontally shows all available suffix combinations for package, operating temperature and screening level. Other possibilities on request.

DEVICE		PACKAGE					OPER. TEMP.			SCREENING LEVEL			
		C	J	P	E	FN	L*	V	M	Std	D	G/B	B/B
1.0 MHz	EF6801/03			●		●	●			●			
				●				●		●			
	EF6803	●						●		●			
1.25 MHz	EF6801/03-1			●		●	●			●			
				●				●		●			
	EF6803-1	●						●		●			
1.5 MHz	EF68A01/03			●			●			●			
	EF68A03	●						●		●			
2.0 MHz	EF68B01/03			●			●			●			

Examples : EF6801P, EF6801FN, EF6801PV, EF6803CV

Package : **C :** Ceramic DIL, **J :** Cerdip DIL, **P :** Plastic DIL, **E :** LCCC, **FN :** PLCC.
Oper. temp. : **L* :** 0°C to +70°C, **V :** −40°C to +85°C, **M :** −55°C to +125°C, ***** : may be omitted.
Screening level : **Std :** (no-end suffix), **D :** NFC 96883 level D,
G/B : NFC 96883 level G, **B/B :** NFC 96883 level B and MIL-STD-883C level B.

These specifications are subjet to change without notice.
Please inquire with our sales offices about the availability of the different products.

Printed in France

Ref.DSEF680103

THOMSON COMPONENTS SEMICONDUCTEURS

EF6801U4 • EF6803U4

MICROCOMPUTER/MICROPROCESSOR (MCU/MPU)

ADVANCE INFORMATION

The EF6801U4 is an 8-bit single-chip microcomputer unit (MCU) which enhances the capabilities of the EF6801 and significantly enhances the capabilities of the EF6800 Family of parts. It includes an EF6801 microprocessor unit (MPU) with direct object-code compatibility and upward object-code compatibility with the EF6800. Execution times of key instructions have been improved over the EF6800 and the new instructions found on the EF6801 are included. The MCU can function as a monolithic microcomputer or can be expanded to a 64K byte address space. It is TTL compatible and requires one +5-volt power supply. On-chip resources include 4096 bytes of ROM, 192 bytes of RAM, a serial communications interface (SCI), parallel I/O, and a 16-bit six-function programmable timer. The EF6803U4 can be considered as an EF6801U4 operating in modes 2 or 3 ; i.e., those that do not use internal ROM.

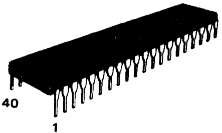
- Enhanced EF6800 Instruction Set
- Upward Source and Object Code Compatibility with the EF6800 and EF6801
- Bus Compatibility with the EF6800 Family
- 8 x 8 Multiply Instruction
- Single-Chip or Expanded Operation to 64K Byte Address Space
- Internal Clock Generator with Divide-by-Four Output
- Serial Communications Interface (SCI)
- 16-Bit Six-Function Programmable Timer
- Three Output Compare Functions
- Two Input Capture Functions
- Counter Alternate Address
- 4096 Bytes of ROM (EF6801U4)
- 192 Bytes of RAM
- 32 Bytes of RAM Retainable During Power Down
- 29 Parallel I/O and Two Handshake Control Lines
- $\overline{\text{NMI}}$ Inhibited Until Stack Load
- Complete Development System Support on DEVICE®.
- −40°C to 85°C Temperature Range

DEVICE® is THOMSON SEMICONDUCTEURS' development/emulation tool.

HMOS

CASES

CB-182

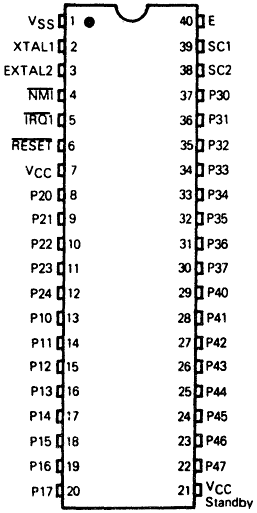


P SUFFIX
PLASTIC PACKAGE

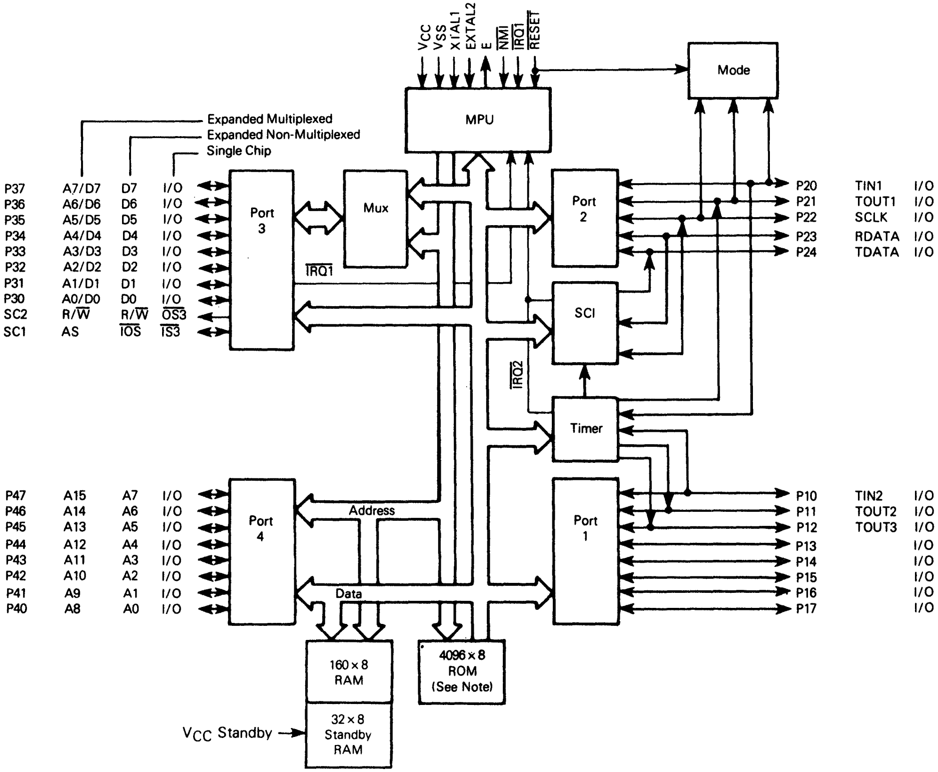
CB-521

FN SUFFIX
PLCC 44

PIN ASSIGNMENT



EF6801U4 MICROCOMPUTER FAMILY BLOCK DIAGRAM



NOTE : No functioning ROM in EF6803U4.

MAXIMUM RATINGS

Rating	Symbol	Value	Unit
Supply Voltage	V_{CC}	−0.3 to +7.0	V
Input Voltage	V_{in}	−0.3 to +7.0	V
Operating Temperature Range	T_A	T_H to T_L	°C
EF6801/03U4, EF6801/03U4-1, EF68A01/03U4		0 to 70	
EF6801/03U4, EF6801/03U4-1 : V suffix		−40 to 85	
Storage Temperature Range	T_{stg}	−55 to +150	°C

This device contains circuitry to protect the inputs against damage due to high static voltages or electric fields; however, it is advised that normal precautions be taken to avoid application of any voltage higher than maximum rated voltages to this high-impedance circuit. For proper operation it is recommended that V_{in} and V_{out} be constrained to the range $V_{SS} \leq (V_{in}$ or $V_{out}) \leq V_{CC}$. Input protection is enhanced by connecting unused inputs to either V_{DD} or V_{SS}.

THERMAL CHARACTERISTICS

Characteristic	Symbol	Value	Rating
Thermal Resistance			
Plastic	θ_{JA}	50	°C/W
PLCC		100	

POWER CONSIDERATIONS

The average chip-junction temperature, T_J, in °C can be obtained from:

$$T_J = T_A + (P_D \bullet \theta_{JA}) \qquad (1)$$

Where:

$T_A \equiv$ Ambient Temperature, °C

$\theta_{JA} \equiv$ Package Thermal Resistance, Junction-to-Ambient, °C/W

$P_D \equiv P_{INT} + P_{PORT}$

$P_{INT} \equiv I_{CC} \times V_{CC}$, Watts – Chip Internal Power

$P_{PORT} \equiv$ Port Power Dissipation, Watts – User Determined

For most applications $P_{PORT} \ll P_{INT}$ and can be neglected. P_{PORT} may become significant if the device is configured to drive Darlington bases or sink LED loads.

An approximate relationship between P_D and T_J (if P_{PORT} is neglected) is:

$$P_D = K \div (T_J + 273°C) \qquad (2)$$

Solving equations 1 and 2 for K gives:

$$K = P_D \bullet (T_A + 273°C) + \theta_{JA} \bullet P_D^2 \qquad (3)$$

Where K is a constant pertaining to the particular part. K can be determined from equation 3 by measuring P_D (at equilibrium) for a known T_A. Using this value of K the values of P_D and T_J can be obtained by solving equations (1) and (2) iteratively for any value of T_A.

CONTROL TIMING (V_{CC}=5.0 V ±5%, V_{SS}=0, T_A=0 to 70°C)

Characteristic	Symbol	EF6801U4 EF6803U4		EF6801U4-1 EF6803U4-1		EF68A01U4 EF68A03U4		Unit
		Min	Max	Min	Max	Min	Max	
Frequency of Operation	f_o	0.5	1.0	0.5	1.25	0.5	1.5	MHz
Crystal Frequency	f_{XTAL}	2.0	4.0	2.0	5.0	2.0	6.0	MHz
External Oscillator Frequency	$4 f_o$	2.0	4.0	2.0	5.0	2.0	6.0	MHz
Crystal Oscillator Startup Time	t_{rc}	–	100	–	100	–	100	ms
Processor Control Setup Time	t_{PCS}	200	–	170	–	140	–	ns

DC ELECTRICAL CHARACTERISTICS (V_{CC} = 5.0 Vdc ±5%, V_{SS} = 0, $T_A = T_L$ to T_H unless otherwise noted)

Characteristic		Symbol	EF6801/03U4-A01/03U4 0° to + 70°C		EF6801U4/6803U4 - 40° to + 85°C		Unit
			Min	Max	Min	Max	
Input High Voltage	$\overline{RESET}$ Other Inputs*	V_{IH}	$V_{SS}+4.0$ $V_{SS}+2.0$	V_{CC} V_{CC}	$V_{SS}+4.0$ $V_{SS}+2.2$	V_{CC} V_{CC}	V
Input Low Voltage	All Inputs*	V_{IL}	$V_{SS}-0.3$	$V_{SS}+0.8$	$V_{SS}-0.3$	$V_{SS}+0.8$	V
Input Load Current (V_{in} = 0 to 2.4 V)	Port 4 SCI	I_{in}	– –	0.5 0.8	– –	0.8 1.0	mA
Input Leakage Current (V_{in} = 0 to 5.5 V)	$\overline{NMI}$, $\overline{IRQ1}$, $\overline{RESET}$	I_{in}	–	2.5	–	5.0	μA
Hi-Z (Off-State) Input Current (V_{in} = 0.5 to 2.4 V)	Port 1, Port 2, Port 3	I_{TSI}	–	10	–	20	μA
Output High Voltage (I_{Load} = – 65 μA, V_{CC} = Min) (I_{Load} = – 100 μA, V_{CC} = Min)	Port 4, SC1, SC2 Other Outputs	V_{OH}	$V_{SS}+2.4$ $V_{SS}+2.4$	– –	$V_{SS}+2.4$ $V_{SS}+2.4$	– –	V
Output Low Voltage (I_{Load} = 2.0 mA, V_{CC} = Min)	All Outputs	V_{OL}	–	$V_{SS}+0.5$	–	$V_{SS}+0.6$	V
Darlington Drive Current (V_O = 1.5 V)	Port 1	I_{OH}	1.0	4.0	1.0	5.0	mA
Internal Power Dissipation (Measured at $T_A = T_L$ in Steady-State Operation)***		P_{INT}	–	1200	–	1500	mW
Input Capacitance (V_{in} = 0, T_A = 25°C, f_o = 1.0 MHz)	Port 3, Port 4, SC1 Other Inputs	C_{in}	– –	12.5 10.0	– –	12.5 10.0	pF
V_{CC} Standby	Powerdown Powerup	V_{SBB} V_{SB}	4.0 4.75	5.25 5.25	4.0 4.75	5.25 5.25	V
Standby Current	Powerdown	I_{SBB}	–	3.0	–	3.5	mA

* Except mode programming levels; see Figure 16.

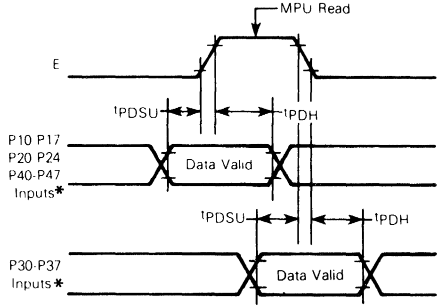
** Negotiable to – 100 μA (for further information contact the factory).

*** For the EF6801U4/EF6803U4 T_L = 0°C and for the EF6801U4/EF6803U4 : V suffix T_L = –40°C

PERIPHERAL PORT TIMING (Refer to Figures 1-4)

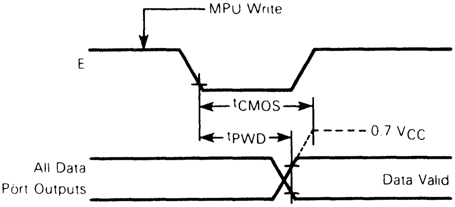
Characteristic	Symbol	EF6801/03U4 EF6801/03U4-1			EF68A01/03U4			Unit
		Min	Typ	Max	Min	Typ	Max	
Peripheral Data Setup Time	t_{PDSU}	200	–	–	150	–	–	ns
Peripheral Data Hold Time	t_{PDH}	200	–	–	150	–	–	ns
Delay Time, Enable Positive Transition to $\overline{OS3}$ Negative Transition	t_{OSD1}	–	–	350	–	–	300	ns
Delay Time, Enable Positive Transition to $\overline{OS3}$ Positive Transition	t_{OSD2}	–	–	350	–	–	300	ns
Delay Time, Enable Negative Transition to Peripheral Data Valid Port 1 Port 2, 3, 4	t_{PWD}	– –	– –	350 350	– –	– –	300 300	ns
Delay Time, Enable Negative Transition to Peripheral CMOS Data Valid	t_{CMOS}	–	–	2.0	–	–	2.0	μs
Input Strobe Pulse Width	t_{PWIS}	200	–	–	150	–	–	ns
Input Data Hold Time	t_{IH}	50	–	–	40	–	–	ns
Input Data Setup Time	t_{IS}	20	–	–	20	–	–	ns

FIGURE 1 — DATA SETUP AND HOLD TIMES (MPU READ)



*Port 3 non-latched operation (Latch enable = 0)

FIGURE 2 — DATA SETUP AND HOLD TIMES (MPU WRITE)



NOTES:

1. 10 k pullup resistor required for port 2 to reach 0.7 V_{CC}
2. Not applicable to P21
3. Port 4 cannot be pulled above V_{CC}

FIGURE 3 - PORT 3 OUTPUT STROBE TIMING (EF6801U4 SINGLE-CHIP MODE)

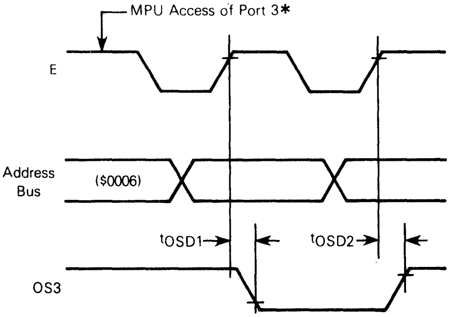
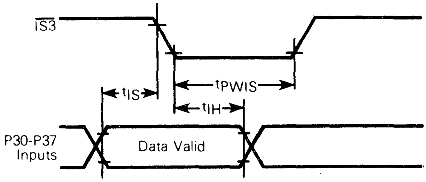


FIGURE 4 - PORT 3 LATCH TIMING (EF6801U4 SINGLE-CHIP MODE)



*Access matches output strobe select (OSS = 0, a read; OSS = 1, a write)

NOTE: Timing measurements are referenced to and from a low voltage of 0.8 volt and a high voltage of 2.0 volts, unless otherwise noted.

FIGURE 5 — CMOS LOAD

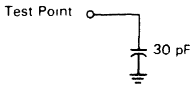
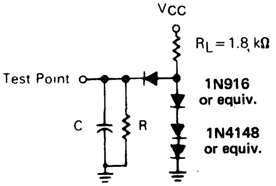


FIGURE 6 — TIMING TEST LOAD PORTS 1, 2, 3, AND 4



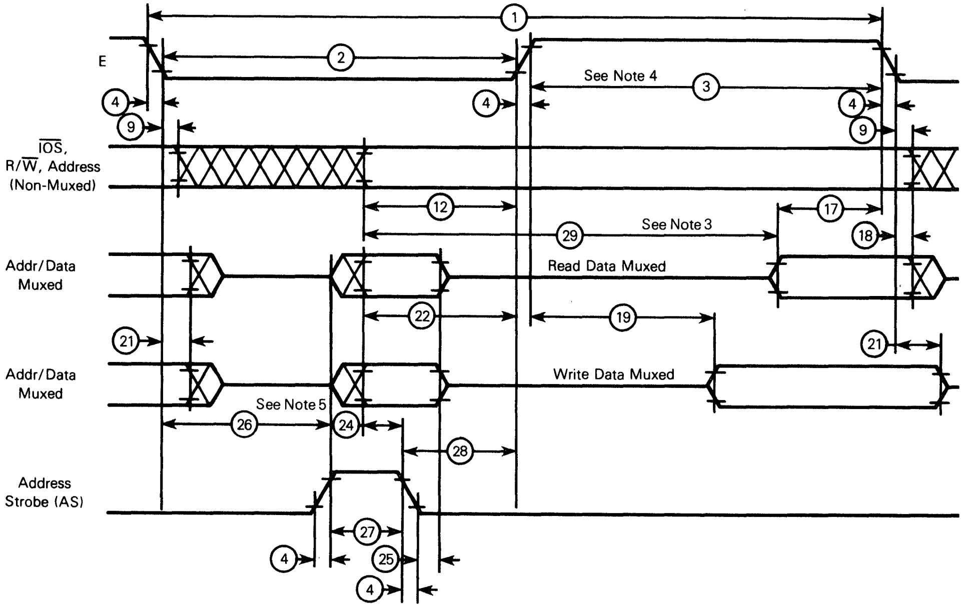
C = 90 pF for P30-P37, P40-P47, E, SC1, SC2
= 30 pF for P10-P17, P20-P24
R = 37 kΩ for P40-P47, SC1, SC2
= 24 kΩ for P10-P17, P20-P24
= 24 kΩ for P30-P37, E

BUS TIMING (See Notes 1 and 2, and Figure 7)

Ident. Number	Characteristics	Symbol	EF6801U4 EF6803 U4		EF6801U4-1 EF6803U4-1		EF68A01U4 EF68A03U4		Unit
			Min	Max	Min	Max	Min	Max	
1	Cycle Time	t_{cyc}	1.0	2.0	0.8	2.0	0.66	2.0	µs
2	Pulse Width, E Low	PW_{EL}	430	1000	360	1000	300	1000	ns
3	Pulse Width, E High	PW_{EH}	450	1000	360	1000	300	1000	ns
4	Clock Rise and Fall Time	t_r, t_f	–	25	–	25	–	25	ns
9	Address Hold Time	t_{AH}	20	–	20	–	20	–	ns
12	Non-Muxed Address Valid Time to E*	t_{AV}	200	–	150	–	115	–	ns
17	Read Data Setup Time	t_{DSR}	80	–	70	–	60	–	ns
18	Read Data Hold Time	t_{DHR}	10	–	10	–	10	–	ns
19	Write Data Delay Time	t_{DDW}	–	225	–	200	–	160	ns
21	Write Data Hold Time	t_{DHW}	20	–	20	–	20	–	ns
22	Muxed Address Valid Time to E Rise*	t_{AVM}	160	–	120	–	100	–	ns
24	Muxed Address Valid Time to AS Fall*	t_{ASL}	40	–	30	–	30	–	ns
25	Muxed Address Hold Time	t_{AHL}	20	–	20	–	20	–	ns
26	Delay Time, E to AS Rise*	t_{ASD}	200	–	170	–	130	–	ns
27	Pulse Width, AS High*	PW_{ASH}	100	–	80	–	60	–	ns
28	Delay Time, AS to E Rise*	t_{ASED}	90	–	70	–	60	–	ns
29	Usable Access Time*(See Note 3)	t_{ACC}	555	–	435	–	385	–	ns

* At specified cycle time.

FIGURE 7 – BUS TIMING



NOTES:

1. Voltage levels shown are $V_L \leq 0.5$ V, $V_H \geq 2.4$ V, unless otherwise specified.
2. Measurement points shown are 0.8 V and 2.0 V, unless otherwise specified.
3. Usable access time is computed by 22+3−17+4.
4. Memory devices should be enabled only during E high to avoid port 3 bus contention.
5. Item 26 is different from the EF6801 but it is upward compatible.

INTRODUCTION

The EF6801U4 is an 8-bit monolithic microcomputer which can be configured to function in a wide variety of applications. The facility which provides this extraordinary flexibility is its ability to be hardware programmed into eight different operating modes. The operating mode controls the configuration of 18 of the 40 MCU pins, available on-chip resources, memory map, location (internal or external) of interrupt vectors, and type of external bus. The configuration of the remaining 22 pins is not dependent on the operating mode.

Twenty-nine pins are organized as three 8-bit ports and one 5-bit port. Each port consists of at least a data register and a write-only data direction register. The data direction register is used to define whether corresponding bits in the data register are configured as an input (clear) or output (set).

The term "port" by itself refers to all of the hardware associated with the port. When the port is used as a "data port" or "I/O port," it is controlled by the port data direction register and the programmer has direct access to the port pins using the port data register. Port pins are labeled as Pij where i identifies one of four ports and j indicates the particular bit.

The microprocessor unit (MPU) is an enhanced EF6800 MPU with additional capabilities and greater throughput. It is upward source and object code compatible with the EF6800 and the EF6801. The programming model is depicted in Figure 8 where accumulator D is a concatenation of accumulators A and B. A list of new operations added to the EF6800 instruction set are shown in Table 1.

The EF6803U4 can be considered an EF6801U4 that operates in modes 2 and 3 only.

FIGURE 8 – PROGRAMMING MODEL

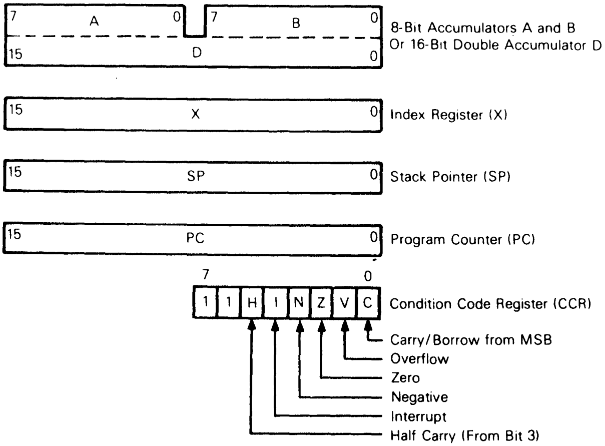


TABLE 1 – NEW INSTRUCTIONS

Instruction	Description
ABX	Unsigned addition of accumulator B to index register
ADDD	Adds (without carry) the double accumulator to memory and leaves the sum in the double accumulator
ASLD or LSLD	Shifts the double accumulator left (towards MSB) one bit, the LSB is cleared, and the MSB is shifted into the C bit
BHS	Branch if higher or same, unsigned conditional branch (same as BCC)
BLO	Branch if lower, unsigned conditional branch (same as BCS)
BRN	Branch never
JSR	Additional addressing mode direct
LDD	Loads double accumulator from memory
LSL	Shifts memory or accumulator left (towards MSB) one bit, the LSB is cleared, and the MSB is shifted into the C bit (same as ASL)
LSRD	Shifts the double accumulator right (towards LSB) one bit, the MSB is cleared, and the LSB is shifted into the C bit
MUL	Unsigned multiply, multiplies the two accumulators and leaves the product in the double accumulator
PSHX	Pushes the index register to stack
PULX	Pulls the index register from stack
STD	Stores the double accumulator to memory
SUBD	Subtracts memory from the double accumulator and leaves the difference in the double accumulator
CPX	Internal processing modified to permit its use with any conditional branch instruction

OPERATING MODES

The EF6801U4 provides seven different operating modes (modes 0 through 3 and 5 through 7) and the EF6803U4 provides two operating modes (modes 2 and 3). The operating modes are hardware selectable and determine the device memory map, the configuration of port 3, port 4, SC1, SC2, and the physical location of the interrupt vectors.

FUNDAMENTAL MODES

The seven operating modes (0-3, 5-7) can be grouped into three fundamental modes which refer to the type of bus it supports: single chip, expanded non-multiplexed, and expanded multiplexed. Single chip is mode 7, expanded non-multiplexed is mode 5, and the remaining 5 are expanded multiplexed modes. Table 2 summarizes the characteristics of the operating modes.

EF6801U4 SINGLE-CHIP MODE (7) -- In the single-chip mode, the four MCU ports are configured as parallel input/output data ports, as shown in Figure 9. The MCU functions as a monolithic microcomputer in this mode without external address or data buses. A maximum of 29 I/O lines and two port 3 control lines are provided. Peripherals or another MCU can be interfaced to port 3 in a loosely coupled dual-processor configuration, as shown in Figure 10.

TABLE 2 – SUMMARY OF EF6801U4/EF6803U4 OPERATING MODES

Single-Chip (Mode 7)
192 bytes of RAM, 4096 bytes of ROM
Port 3 is a parallel I/O port with two control lines
Port 4 is a parallel I/O port

Expanded Non-Multiplexed (Mode 5)
192 bytes of RAM, 4096 bytes of ROM
256 bytes of external memory space
Port 3 is an 8-bit data bus
Port 4 is an input port/address bus

Expanded Multiplexed (Modes 0, 1, 2, 3, 6*)
Four memory space options (total 64K address space)
(1) Internal RAM and ROM with partial address bus (mode 1)
(2) Internal RAM, no ROM (mode 2)
(3) Extended addressing of internal I/O and RAM
(4) Internal RAM and ROM with partial address bus (mode 6)
Port 3 is multiplexed address/data bus
Port 4 is address bus (inputs/address in mode 6)
Test mode (mode 0):
May be used to test internal RAM and ROM
May be used to test ports 3 and 4 as I/O ports by writing into mode 7
Only modes 5, 6, and 7 can be irreversibly entered from mode 0

Resources Common to All Modes
Reserved register area
Port 1 input/output operation
Port 2 input/output operation
Timer operation
Serial communications interface operation

***The EF6803U4 operates only in modes 2 and 3.**

FIGURE 9 – SINGLE-CHIP MODE

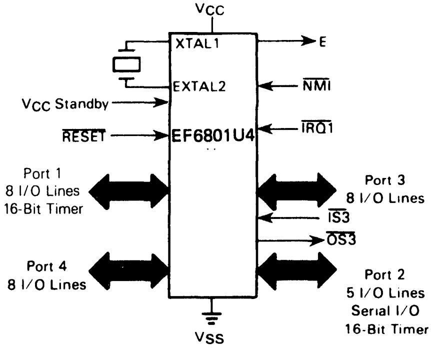
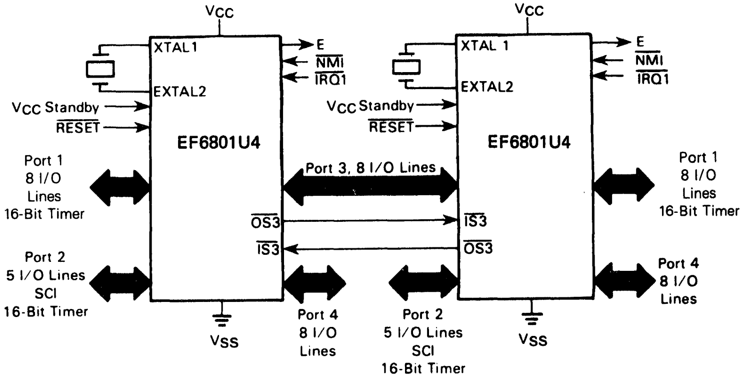


FIGURE 10 – SINGLE-CHIP DUAL PROCESSOR CONFIGURATION



EF6801U4 EXPANDED NON-MULTIPLEXED MODE (5) – A modest amount of external memory space is provided in the expanded non-multiplexed mode while significant on-chip resources are retained. Port 3 functions as an 8-bit bidirectional data bus and port 4 is configured initially as an input data port. Any combination of the eight least significant address lines may be obtained by writing to the port 4 data direction register. Stated alternatively, any combination of A0 to A7 may be provided while retaining the remainder as input data lines. Internal pullup resistors pull the port 4 lines high until the port is configured.

Figure 11 illustrates a typical system configuration in the expanded non-multiplexed mode. The MCU interfaces directly with EF6800 family parts and can access 256 bytes of external address space at $100 through $1FF. IOS provides an address decode of external memory ($100-$1FF) and can be used as a memory-page select or chip-select line.

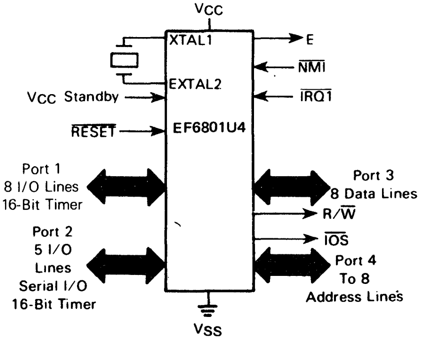
EXPANDED MULTIPLEXED MODES (0, 1, 2, 3, 6) – A 64K byte memory space is provided in the expanded multiplexed modes. In each of the expanded multiplexed modes, port 3 functions as a time multiplexed address/data bus with address valid on the negative edge of address strobe (AS) and data valid while E is high. In modes 0, 2, and 3, port 4 provides address lines A8 to A15. In modes 1 and 6, however, port 4 initially is configured at reset as an input data port. The port 4 data direction register can then be changed to provide any combination of address lines A8 to A15. Stated alternatively, any subset of A8 to A15 can be provided while retaining the remaining port 4 lines as input data lines. Internal pullup resistors pull the port 4 lines high until software configures the port. In mode 1, the internal pullup resistors will hold the upper address lines high producing a value of $FFXX for a reset vector. A simple method of getting the desired address lines configured as outputs is to have an external EPROM not fully decoded so it appears at two address locations (i.e., $FXXX and $BXXX). Then, when the reset vector appears as $FFFE, the EPROM will be accessed and can point to an address in the top $100 bytes of the internal or external ROM/EPROM that will configure port 4 as desired.

In mode 0, the reset and interrupt vectors are located at $BFF0-$BFFF. In addition, the internal and external data buses are connected, so there must be no memory map overlap in order to avoid potential bus conflicts. By writing the PC0-PC2 bits in the port 2 data register, modes 5, 6, and 7 can be irreversibly entered from mode 0. Mode 0 is used primarily to verify the ROM pattern and monitor the internal data bus with the automated test equipment.

Only the EF6801U4 can operate in each of the expanded multiplexed modes. The EF6803U4 operates only in modes 2 and 3.

Figure 12 depicts a typical configuration for the expanded multiplexed modes. Address strobe can be used to control a transparent D-type latch to capture addresses A0-A7, as shown in Figure 13. This allows port 3 to function as a data bus when E is high.

FIGURE 11 – EXPANDED NON-MULTIPLEXED CONFIGURATION



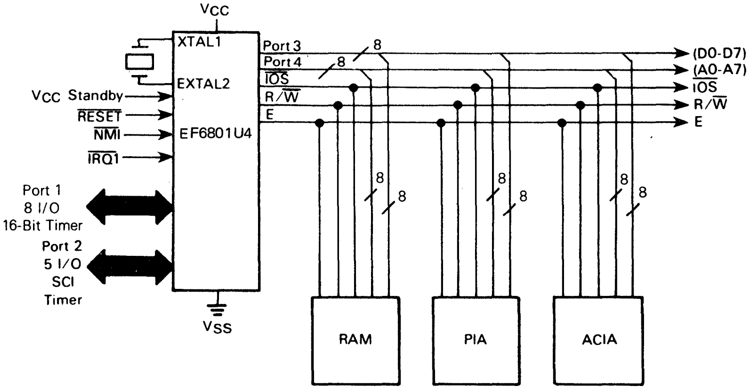
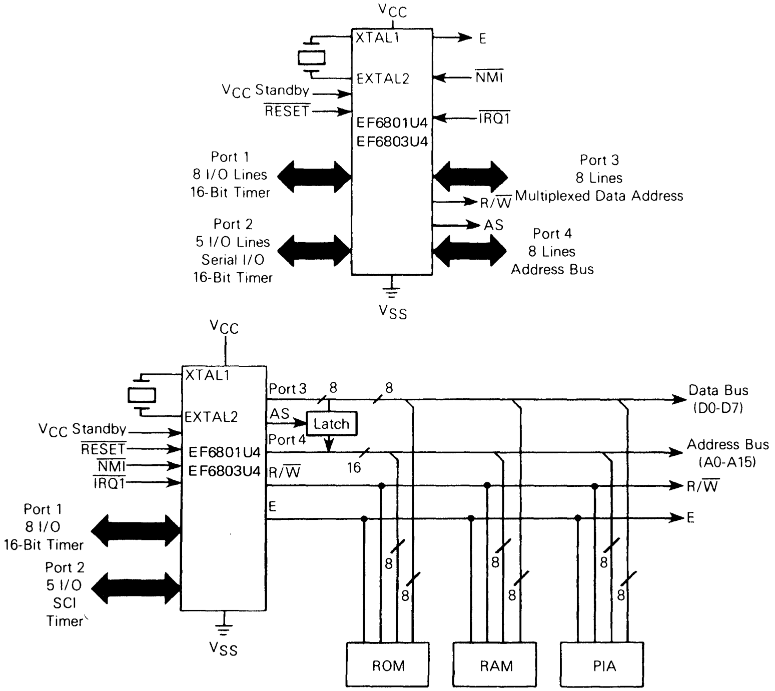
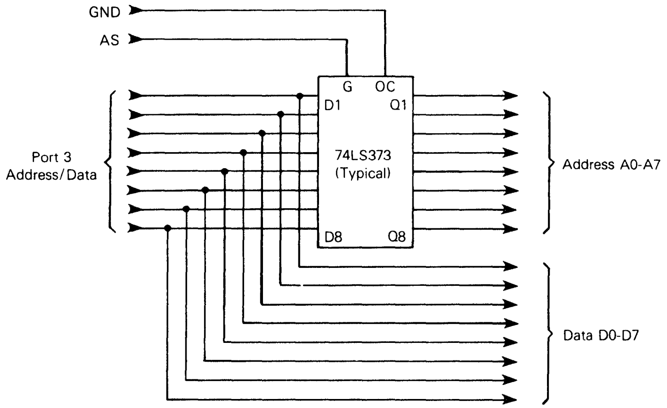


FIGURE 12 – EXPANDED MULTIPLEXED CONFIGURATION



NOTE: To avoid data bus (port 3) contention in the expanded multiplexed modes, memory devices should be enabled only during E high time.

FIGURE 13 – TYPICAL LATCH ARRANGEMENT



PROGRAMMING THE MODE

The operating mode is determined at $\overline{\text{RESET}}$ by the levels asserted on P22, P21, and P20. These levels are latched into PC2, PC1, and PC0 of the program control register on the positive edge of $\overline{\text{RESET}}$. The operating mode may be read from the port 2 data register, as shown below, and programming levels and timing must be met as shown in Figure 14. A brief outline of the operating modes is shown in Table 3.

PORT 2 DATA REGISTER

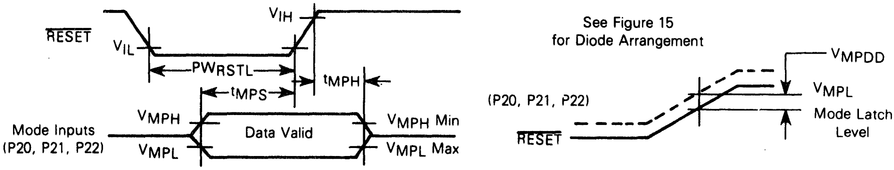
7	6	5	4	3	2	1	0	
PC2	PC1	PC0	P24	P23	P22	P21	P20	$03

Circuitry to provide the programming levels is dependent primarily on the normal system usage of the three pins. If configured as outputs, the circuit shown in Figure 15 may be used; otherwise, three-state buffers can be used to provide isolation while programming the mode.

MEMORY MAPS

The EF6801U4/EF6803U4 can provide up to 64K byte address space depending on the operating mode. A memory map for each operating mode is shown in Figure 16. The first 32 locations of each map are reserved for the internal register area, as shown in Table 4, with exceptions as indicated.

FIGURE 14 — MODE PROGRAMMING TIMING



MODE PROGRAMMING (Refer to Figure 14)

Characteristic	Symbol	Min	Max	Unit
Mode Programming Input Voltage Low	V_{MPL}	–	1.8	V
Mode Programming Input Voltage High	V_{MPH}	4.0	–	V
Mode Programming Diode Differential (If Diodes are Used)	V_{MPDD}	0.6	–	V
$\overline{\text{RESET}}$ Low Pulse Width	PW_{RSTL}	3.0	–	E Cycles
Mode Programming Setup Time	t_{MPS}	2.0	–	E Cycles
Mode Programming Hold Time $\overline{\text{RESET}}$ Rise Time ≥ 1 μs $\overline{\text{RESET}}$ Rise Time < 1 μs	t_{MPH}	 0 100	 – –	ns

TABLE 3 — MODE SELECTION SUMMARY

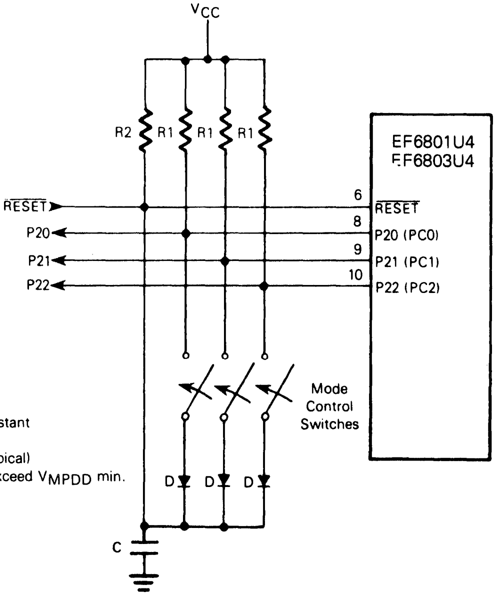
Mode*	P22 PC2	P21 PC1	P20 PC0	ROM	RAM	Interrupt Vectors	Bus Mode	Operating Mode
7	H	H	H	I	I	I	I	Single Chip
6	H	H	L	I	I	I	MUX(2, 3)	Multiplexed/Partial Decode
5	H	L	H	I	I	I	NMUX(2, 3)	Non-Multiplexed/Partial Decode
4	H	L	L	–	–	–	–	Undefined(4)
3	L	H	H	E	I	E	MUX(1, 5)	Multiplexed/RAM
2	L	H	L	E	I	E	MUX(1)	Multiplexed/RAM
1	L	L	H	I	I	E	MUX(1, 3)	Multiplexed/RAM and ROM
0	L	L	L	I	I	E	MUX(1)	Multiplexed Test

LEGEND
I – Internal
E – External
MUX – Multiplexed
NMUX – Non-Multiplexed
L – Logic "0"
H – Logic "1"

NOTES:
1. Addresses associated with ports 3 and 4 are considered external in modes 0, 1, 2, and 3.
2. Addresses associated with port 3 are considered external in modes 5 and 6.
3. Port 4 default is user data input; address output is optional by writing to port 4 data direction register.
4. Mode 4 is a non-user mode and should not be used as an operating mode.
5. Mode 3 has the internal RAM and internal registers relocated at $D000-$D0FF.

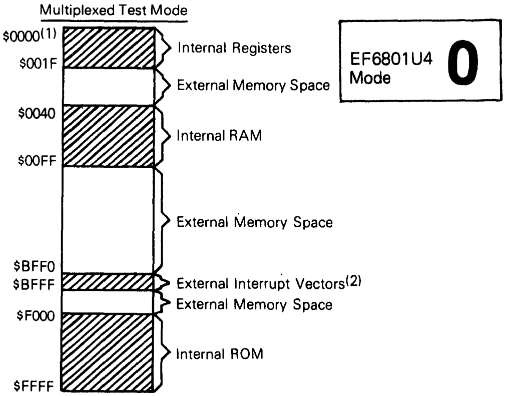
* The EF6803U4 operates only in modes 2 and 3.

FIGURE 15 – TYPICAL MODE PROGRAMMING CIRCUIT



NOTES:
1. Mode 7 as shown
2. R2•C = Reset time constant
3. R1 = 10 k (typical)
4. D = 1N914, 1N4001 (typical)
5. Diode V_f should not exceed V_{MPDD} min.

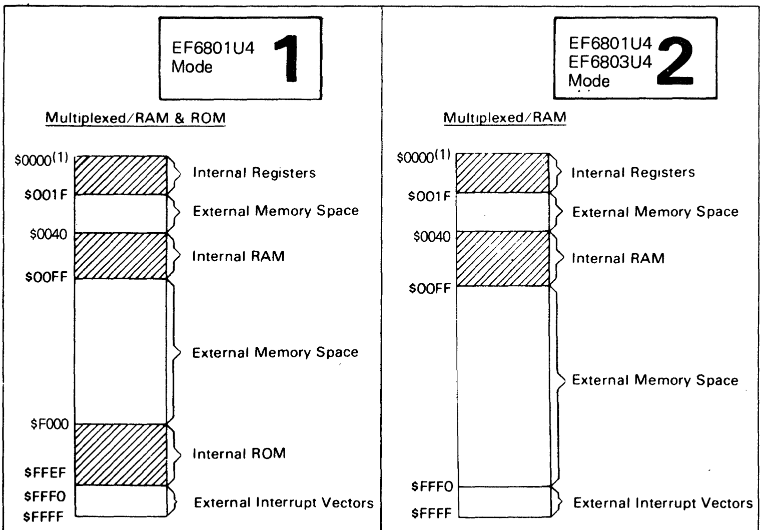
FIGURE 16 – EF6801U4/EF6803U4 MEMORY MAPS (Sheet 1 of 4)



NOTES:

1) Excludes the following addresses which may be used externally: $04, $05, $06, $07, and $0F.
2) The interrupt vectors are at $BFF0-$BFFF.
3) There must be no overlapping of internal and external memory spaces to avoid driving the data bus with more than one device.
4) This mode is the only mode which may be used to examine the entire ROM using an external RESET vector.
5) Modes 5-7 can be irreversibly entered from mode 0 by writing to the PC0-PC2 bits of the port 2 data register.

FIGURE 16 – EF6801U4/EF6803U4 MEMORY MAPS (Sheet 2 of 4)



NOTES:

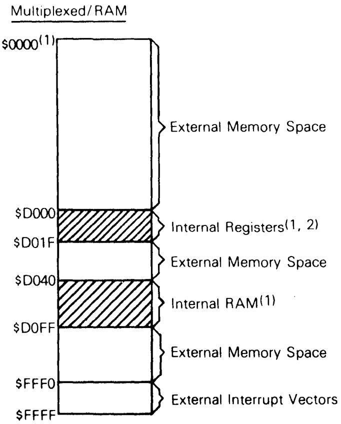
1) Excludes the following addresses which may be used externally: $05 and $07.
2) Internal ROM addresses $FFF0 to $FFFF are not usable.
3) Address lines A8-A15 will not contain addresses until the data direction register for port 4 has been written with "1s" in the appropriate bits. These address lines will assert "1s" until made outputs by writing the data direction register.

NOTE:

1) Excludes the following addresses which may be used externally: $04, $05, $06, $07, and $0F.

FIGURE 16 – EF6801U4/EF6803U4 MEMORY MAPS (Sheet 3 of 4)

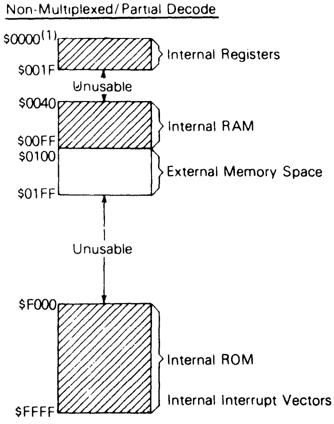
EF6801U4
EF6803U4
Mode **3**



NOTES:

1) Relocating the internal registers and the internal RAM to high memory allows the processor to make use of direct addressing.
2) Excludes the following addresses which may be used externally: $D004, $D005, $D006, $D007, and $D00F.

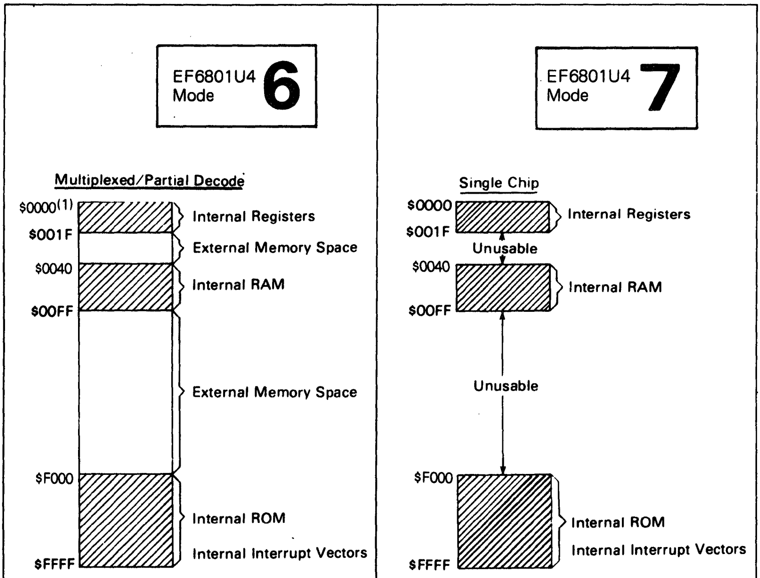
EF6801U4
Mode **5**



NOTES:

1) Excludes the following addresses which may not be used externally: $04, $06, and $0F (no $\overline{IOS}$).
2) Address lines A0 to A7 will not contain addresses until the data direction register for port 4 has been written with "1s" in the appropriate bits. These address lines will assert "1s" until made outputs by writing the data direction register.

FIGURE 16 – EF6801U4/EF6803U4 MEMORY MAPS (Sheet 4 of 4)



NOTES:

1) Excludes the following addresses which may be used externally: $04, $06, $0F.
2) Address lines A8-A15 will not contain addresses until the data direction register for port 4 has been written with "1s" in the appropriate bits. These address lines will assert "1s" until made outputs by writing the data direction register.

TABLE 4 – INTERNAL REGISTER AREA

Register	Address	
	Other Modes	Mode 3
Port 1 Data Direction Register* * *	0000	D000
Port 2 Data Direction Register* * *	0001	D001
Port 1 Data Register	0002	D002
Port 2 Data Register	0003	D003
Port 3 Data Direction Register* * *	0004*	D004*
Port 4 Data Direction Register* * *	0005* *	D005* *
Port 3 Data Register	0006*	D006*
Port 4 Data Register	0007* *	D007* *
Timer Control and Status Register	0008	D008
Counter (High Byte)	0009	D009
Counter (Low Byte)	000A	D00A
Output Compare Register (High Byte)	000B	D00B
Output Compare Register (Low Byte)	000C	D00C
Input Capture Register (High Byte)	000D	D00D
Input Capture Register (Low Byte)	000E	D00E
Port 3 Control and Status Register	000F*	D00F*
Rate and Mode Control Register	0010	D010
Transmit/Receive Control and Status Register	0011	D011
Receive Data Register	0012	D012
Transmit Data Register	0013	D013
RAM Control Register	0014	D014
Counter Alternate Address (High Byte)	0015	D015
Counter Alternate Address (Low Byte)	0016	D016
Timer Control Register 1	0017	D017
Timer Control Register 2	0018	D018
Timer Status Register	0019	D019
Output Compare Register 2 (High Byte)	001A	D01A
Output Compare Register 2 (Low Byte)	001B	D01B
Output Compare Register 3 (High Byte)	001C	D01C
Output Compare Register 3 (Low Byte)	001D	D01D
Input Capture Register 2 (High Byte)	001E	D01E
Input Capture Register 2 (Low Byte)	001F	D01F

*External addresses in modes 0, 1, 2, 3, 5, and 6 cannot be accessed in mode 5 (no IOS).

* *External Addresses in Modes 0, 2, and 3.

* * *1 = Output, 0 = Input

EF6801U4/EF6803U4 INTERRUPTS

The EF6801 Family supports two types of interrupt requests: maskable and non-maskable. A non-maskable interrupt ($\overline{\text{NMI}}$) is always recognized and acted upon at the completion of the current instruction. Maskable interrupts are controlled by the condition code register I bit and by individual enable bits. The I bit controls all maskable interrupts. Of the maskable interrupts, there are two types: $\overline{\text{IRQ1}}$ and $\overline{\text{IRQ2}}$. The programmable timer and serial communications interface use an internal $\overline{\text{IRQ2}}$ interrupt line, as shown in the block diagram. External devices and IS3 use $\overline{\text{IRQ1}}$. An $\overline{\text{IRQ1}}$ interrupt is serviced before $\overline{\text{IRQ2}}$ if both are pending.

NOTE

After reset, an $\overline{\text{NMI}}$ will not be serviced until the first program load of the stack pointer. Any $\overline{\text{NMI}}$ generated before this load will be remembered by the processor and serviced subsequent to the stack pointer load.

All $\overline{\text{IRQ2}}$ interrupts use hardware prioritized vectors. The single SCI interrupt and three timer interrupts are serviced in a prioritized order and each is vectored to a separate location. All interrupt vector locations are shown in Table 5. In mode 0, reset and interrupt vectors are defined as $BFF0-$BFFF.

The interrupt flowchart is depicted in Figure 17 and is common to every interrupt excluding reset. During interrupt servicing, the program counter, index register, A accumulator, B accumulator, and condition code register are pushed to the stack. The I bit is set to inhibit maskable interrupts and a vector is fetched corresponding to the current highest priority interrupt. The vector is transferred to the program counter and instruction execution is resumed. Interrupt and $\overline{\text{RESET}}$ timing are illustrated in Figures 18 and 19.

TABLE 5 – MCU INTERRUPT VECTOR LOCATIONS

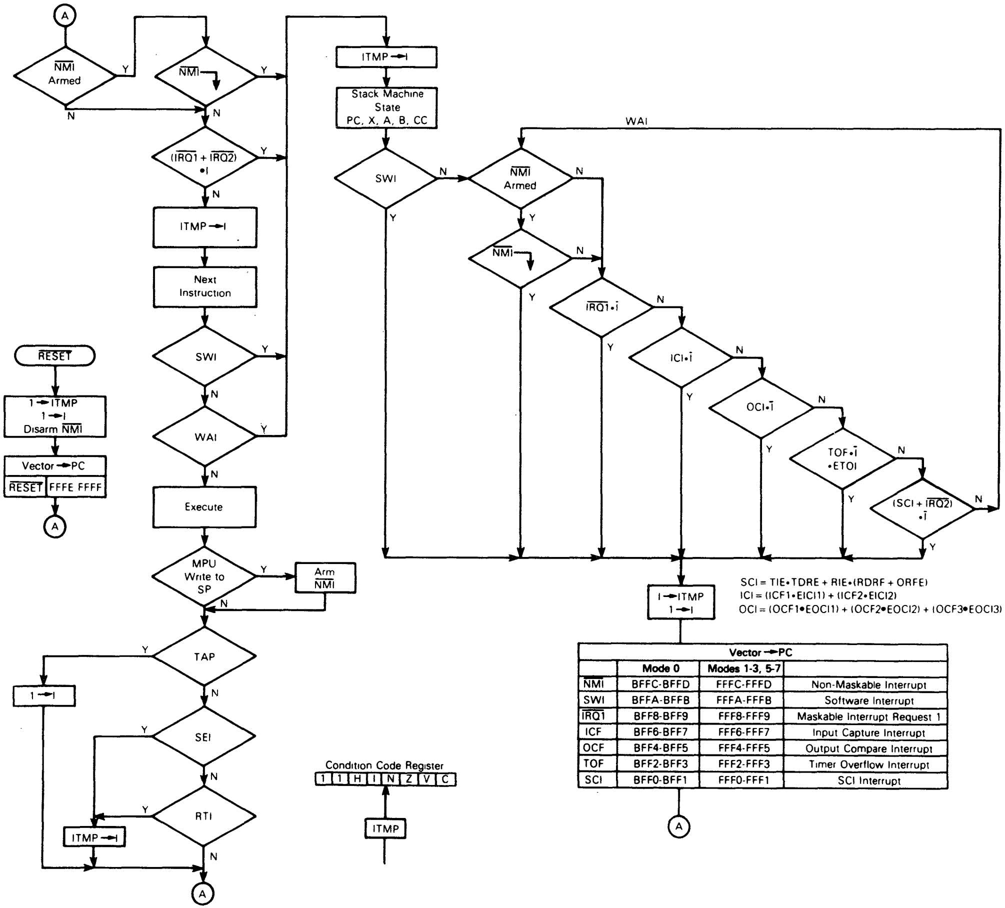
Mode 0		Modes 1-3, 5-7		Interrupt* * *
MSB	LSB	MSB	LSB	
BFFE	BFFF	FFFE	FFFF	$\overline{\text{RESET}}$
BFFC	BFFD	FFFC	FFFD	Non-Maskable Interrupt* *
BFFA	BFFB	FFFA	FFFB	Software Interrupt
BFF8	BFF9	FFF8	FFF9	Maskable Interrupt Request 1
BFF6	BFF7	FFF6	FFF7	Input Capture Flag*
BFF4	BFF5	FFF4	FFF5	Output Compare Flag*
BFF2	BFF3	FFF2	FFF3	Timer Overflow Flag*
BFF0	BFF1	FFF0	FFF1	Serial Communications Interface*

*$\overline{\text{IRQ2}}$ interrupt

* *$\overline{\text{NMI}}$ must be armed (by accessng stack pointer) before an $\overline{\text{NMI}}$ is executed.

* * * Mode 4 interrupt vectors are undefined.

FIGURE 17 — INTERRUPT FLOWCHART



SCI = TIE•TDRE + RIE•(RDRF + ORFE)
ICI = (ICF1•EICI1) + (ICF2•EICI2)
OCI = (OCF1•EOCI1) + (OCF2•EOCI2) + (OCF3•EOCI3)

Vector → PC			
	Mode 0	Modes 1-3, 5-7	
NMI	BFFC-BFFD	FFFC-FFFD	Non-Maskable Interrupt
SWI	BFFA-BFFB	FFFA-FFFB	Software Interrupt
IRQ1	BFF8-BFF9	FFF8-FFF9	Maskable Interrupt Request 1
ICF	BFF6-BFF7	FFF6-FFF7	Input Capture Interrupt
OCF	BFF4-BFF5	FFF4-FFF5	Output Compare Interrupt
TOF	BFF2-BFF3	FFF2-FFF3	Timer Overflow Interrupt
SCI	BFF0-BFF1	FFF0-FFF1	SCI Interrupt

FIGURE 18 — INTERRUPT SEQUENCE

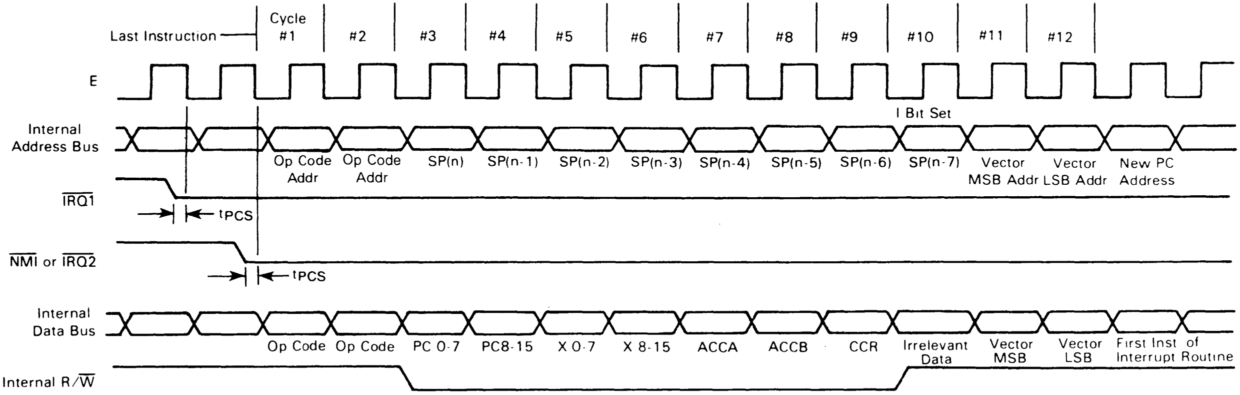
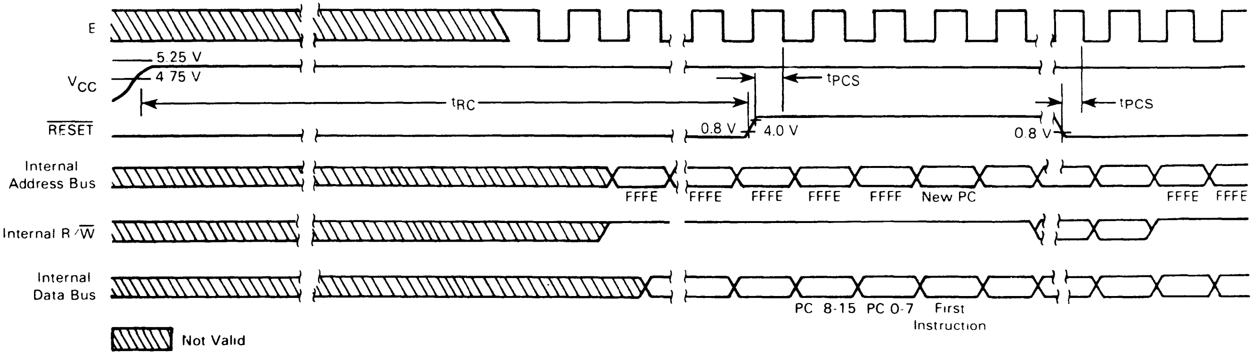


FIGURE 19 — RESET TIMING



FUNCTIONAL PIN DESCRIPTIONS

V_{CC} AND V_{SS}

V_{CC} and V_{SS} provide power to a large portion of the MCU. The power supply should provide +5 volts (±5%) to V_{CC} and V_{SS} should be tied to ground. Total power dissipation (including V_{CC} standby) will not exceed P_D milliwatts.

V_{CC} STANDBY

V_{CC} standby provides power to the standby portion ($40 through $5F in all modes except mode 3 which is $D040 through $D05F) of the RAM and the STBY PWR and RAME bits of the RAM control register. Voltage requirements depend on whether the device is in a power-up or power-down state. In the power-up state, the power supply should provide +5 volts (±5%) and must reach V_{SB} volts before $\overline{\text{RESET}}$ reaches 4.0 volts. During power down, V_{CC} standby must remain above V_{SBB} (minimum) to sustain the standby RAM and STBY PWR bit. While in power-down operation, the standby current will not exceed I_{SBB}.

It is typical to power both V_{CC} and V_{CC} standby from the same source during normal operation. A diode must be used between them to prevent supplying power to V_{CC} during power-down operation.

XTAL1 AND EXTAL 2

These two input pins interface either a crystal or TTL-compatible clock to the MCU internal clock generator. Divide-by-four circuitry is included which allows use of the inexpensive 3.58 MHz or 4.4336 MHz color burst TV crystals. A 20 pF capacitor should be tied from each crystal pin to ground to ensure reliable startup and operation. Alternatively, EXTAL2 may be driven by an external TTL-compatible clock at 4 f_o with a duty cycle of 50% (±5%) with XTAL1 connected ground.

The internal oscillator is designed to interface with an AT-cut quartz crystal resonator operated in parallel resonance mode in the frequency range specified for f_{XTAL}. The crystal should be mounted as close as possible to the input pins to minimize output distortion and startup stabilization time. The MCU is compatible with most commercially available crystals. Nominal crystal parameters are shown in Figure 20.

$\overline{\text{RESET}}$

This input is used to reset the internal state of the device and provide an orderly startup procedure. During power up, $\overline{\text{RESET}}$ must be held below 0.8 volt : (1) at least t_{RC} after V_{CC} reaches 4.75 volts in order to provide sufficient time for the clock generator to stabilize, and (2) until V_{CC} standby reaches 4.75 volts. $\overline{\text{RESET}}$ must be held low at least three E cycles if asserted during power-up operation.

E (ENABLE)

This is an output clock used primarily for bus synchronization. It is TTL compatible and is the slightly skewed divide-by-four result of the device input clock frequency. It will drive one Schottky TTL load and 90 pF, and all data given in cycles is referenced to this clock unless otherwise noted.

$\overline{\text{NMI}}$ (NON-MASKABLE INTERRUPT)

An $\overline{\text{NMI}}$ negative edge requests an MCU interrupt sequence, but the current instruction will be completed before it responds to the request. The MCU will then begin an interrupt sequence. Finally, a vector is fetched from $FFFC and $FFFD ($BFFC and $BFFD in mode 0), transferred to the program counter, and instruction execution is resumed. $\overline{\text{NMI}}$ typically requires a 3.3 kΩ (nominal) resistor to V_{CC}. There is no internal $\overline{\text{NMI}}$ pullup resistor. $\overline{\text{NMI}}$ must be held low for at least one E cycle to be recognized under all conditions.

NOTE

After reset, an $\overline{\text{NMI}}$ will not be serviced until the first program load of the stack pointer. Any $\overline{\text{NMI}}$ generated before this load will remain pending by the processor.

$\overline{\text{IRQ1}}$ (MASKABLE INTERRUPT REQUEST 1)

$\overline{\text{IRQ1}}$ is a level-sensitive input which can be used to request an interrupt sequence. The MPU will complete the current instruction before it responds to the request. If the interrupt mask bit (I bit) in the condition code register is clear, the MCU will begin an interrupt sequence. A vector is fetched from $FFF8 and $FFF9 ($BFF8 and $BFF9 in mode 0), transferred to the program counter, and instruction execution is resumed.

$\overline{\text{IRQ1}}$ typically requires an external 3.3 kΩ (nominal) resistor to V_{CC} for wire-OR applications. $\overline{\text{IRQ1}}$ has no internal pullup resistor.

SC1 AND SC2 (STROBE CONTROL 1 AND 2)

The function of SC1 and SC2 depends on the operating mode. SC1 is configured as an output in all modes except single-chip mode, whereas SC2 is always an output. SC1 and SC2 can drive one Schottky load and 90 pF.

SC1 AND SC2 IN SINGLE-CHIP MODE – In single-chip mode, SC1 and SC2 are configured as an input and output, respectively, and both function as port 3 control lines. SC1 functions as IS3 and can be used to indicate that port 3 input data is ready or output data has been accepted. Three options associated with IS3 are controlled by the port 3 control and status register and are discussed in the port 3 description; refer to **P30-P37 (PORT 3).** If unused, IS3 can remain unconnected.

SC2 is configured as OS3 and can be used to strobe output data or acknowledge input data. It is controlled by output strobe select (OSS) in the port 3 control and status register. The strobe is generated by a read (OSS = 0) or write (OSS = 1) to the port 3 data register. OS3 timing is shown in Figure 3.

SC1 AND SC2 IN EXPANDED NON-MULTIPLEXED MODE – In the expanded non-multiplexed mode, both SC1 and SC2 are configured as outputs. SC1 functions as input/output select (IOS) and is asserted only when $0100 through $01FF is sensed on the internal address bus.

SC2 is configured as read/write and is used to control the direction of data bus transfers. An MPU read is enabled when read/write and E are high.

SC1 AND SC2 IN EXPANDED MULTIPLEXED MODE – In the expanded multiplexed modes, both SC1 and SC2 are configured as outputs. SC1 functions as address strobe and can be used to demultiplex the eight least significant addresses and the data bus. A latch controlled by address strobe captures the lower address on the negative edge, as shown in Figure 13.

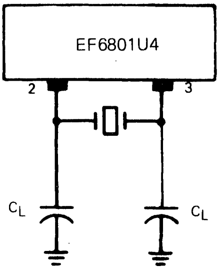
FIGURE 20 – EF6801U4/EF6803U4 FAMILY OSCILLATOR CHARACTERISTICS

(a) Nominal Recommended Crystal Parameters

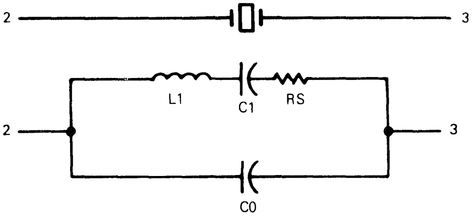
Nominal Crystal Parameters*

	3.58 MHz	4.00 MHz	5.0 MHz
RS	60 Ω	50 Ω	30-50 Ω
C0	3.5 pF	6.5 pF	4-6 pF
C1	0.015 pF	0.025 pF	0.01-0.02 pF
Q	>40 K	>30 K	>20 K

*NOTE: These are representative AT-cut crystal parameters only. Crystals of other types of cut may also be used.

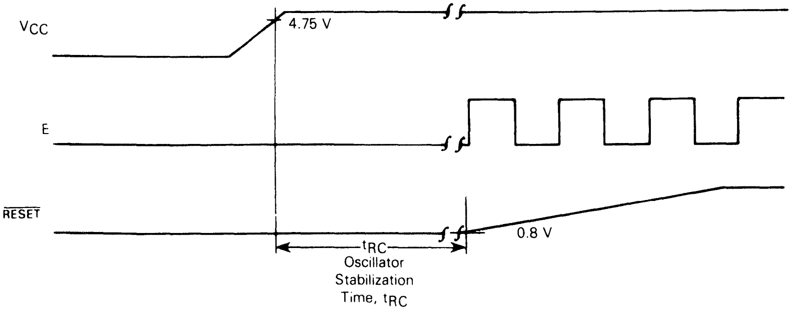


C_L = 20 pF (typical)



Equivalent Circuit

(b) Oscillator Stabilization Time (t_{RC})



SC2 is configured as read/write and is used to control the direction of data bus transfers. An MPU read is enabled when read/write and E are high.

P10-P17 (PORT 1)

Port 1 is a mode independent 8-bit I/O and timer port. Each line can be configured as either an input or output as defined by the port 1 data direction register. Port 1 bits 0, 1, and 2 (P10, P11, and P12) can also be used to exercise one input edge function and two output compare functions of the timer. The TTL compatible three-state buffers can drive one Schottky TTL load and 30 pF, Darlington transistors, or CMOS devices using external pullup resistors. It is configured as a data input port during $\overline{\text{RESET}}$. Unused pins can remain unconnected.

P20-P24 (PORT 2)

Port 2 is a mode-independent, 5-bit, multipurpose I/O port. The voltage levels present on P20, P21, and P22 on the rising edge of $\overline{\text{RESET}}$ determine the operating mode of the MCU. The entire port is then configured as a data input port. The port 2 lines can be selectively configured as data output lines by setting the appropriate bits in the port 2 data direction register. The port 2 data register is used to move data through the port. However, if P21 is configured as an output, it is tied to the timer output compare 1 function and cannot be used to provide output from the port 2 data register unless output enable 1 (OE1) is cleared in timer control register 1.

Port 2 can also be used to provide an interface for the serial communications interface and the timer input edge function. These configurations are described in **SERIAL COMMUNICATIONS INTERFACE** and **PROGRAMMABLE TIMER.**

The port 2 three-state TTL-compatible output buffers are capable of driving one Schottky TTL load and 30 pF, or CMOS devices using external pullup resistors.

PORT 2 DATA REGISTER

7	6	5	4	3	2	1	0	
PC2	PC1	PC0	P24	P23	P22	P21	P20	$03

P30-P37 (PORT 3)

Port 3 can be configured as an I/O port, a bidirectional 8-bit data bus, or a multiplexed address/data bus depending on the operating mode. The TTL compatible three-state output buffers can drive one Schottky TTL load and 90 pF. Unused lines can remain unconnected.

PORT 3 IN SINGLE-CHIP MODE – Port 3 is an 8-bit I/O port in the single-chip mode with each line configured by the port 3 data direction register. There are also two lines, IS3 and OS3, which can be used to control port 3 data transfers.

Three port 3 options are controlled by the port 3 control and status register and are available only in single-chip mode: 1) port 3 input data can be latched using IS3 as a control signal, 2) OS3 can be generated by either an MPU read or write to the port 3 data register, and 3) an $\overline{\text{IRQ1}}$ interrupt can be enabled by an IS3 negative edge. Port 3 latch timing is shown in Figure 4.

PORT 3 CONTROL AND STATUS REGISTER

7	6	5	4	3	2	1	0	
IS3 Flag	IS3 $\overline{\text{IRQ1}}$	X	OSS	Latch Enable	X	X	X	$0F

Bits 0-2 Not used.

Bit 3 **Latch Enable** – This bit controls the input latch for port 3. If set, input data is latched by an IS3 negative edge. The latch is transparent after a read of the port 3 data register. Latch enable is cleared during reset.

Bit 4 **OSS (Output Strobe Select)** – This bit determines whether OS3 will be generated by a read or write of the port 3 data register. When clear, the strobe is generated by a read; when set, it is generated by a write. OSS is cleared during reset.

Bit 5 Not used.

Bit 6 **IS3 $\overline{\text{IRQ1}}$ Enable** – When set, an $\overline{\text{IRQ1}}$ interrupt will be enabled whenever the IS3 flag is set; when clear, the interrupt is inhibited. This bit is cleared during reset.

Bit 7 **IS3 Flag** – This read-only status bit is set by an IS3 negative edge. It is cleared by a read of the port 3 data register or during reset.

PORT 3 IN EXPANDED NON-MULTIPLEXED MODE – Port 3 is configured as a bidirectional data bus (D7-D0) in the expanded non-multiplexed mode. The direction of data transfers is controlled by read/write (SC2). Data is clocked by E (enable).

PORT 3 IN EXPANDED MULTIPLEXED MODE – Port 3 is configured as a time multiplexed address (A7-A0) and data bus (D7-D0) in the expanded multiplexed mode where address strobe (AS) can be used to demultiplex the two buses. Port 3 is held in a high-impedance state between valid address and data to prevent bus conflicts.

P40-P47 (PORT 4)

Port 4 is configured as an 8-bit I/O port, as address outputs, or as data inputs depending on the operating mode. Port 4 can drive one Schottky TTL load and 90 pF, and is the only port with internal pullup resistors. Unused lines can remain unconnected.

PORT 4 IN SINGLE-CHIP MODE – In single-chip mode, port 4 functions as an 8-bit I/O port with each line configured by the port 4 data direction register. Internal pullup resistors allow the port to directly interface with CMOS at 5-volt levels. External pullup resistors to more than 5 volts, however, cannot be used.

PORT 4 IN EXPANDED NON-MULTIPLEXED MODE – Port 4 is configured from reset as an 8-bit input port where the port 4 data direction register can be written to provide any or all of eight address lines A0 to A7. Internal pullup resistors pull the lines high until the port 4 data direction register is configured.

PORT 4 IN EXPANDED MULTIPLEXED MODE – In all expanded multiplexed modes except modes 1 and 6, port 4 functions as half of the address bus and provides A8 to A15. In modes 1 and 6, the port is configured from reset as an 8-bit parallel input port where the port 4 data direction register can be written to provide any or all of upper address lines A8 to A15. Internal pullup resistors pull the lines high until the port 4 data direction register is configured where bit 0 controls A8.

RESIDENT MEMORY

The EF6801U4 provides 4096 bytes of on-chip ROM and 192 bytes of on-chip RAM.

Thirty-two bytes of the RAM are powered through the V_{CC} standby pin and are maintainable during V_{CC} power down. This standby portion of the RAM consists of 32 bytes located from $40 through $5F in all modes except mode 3 which is $D040 through $D05F.

Power must be supplied to V_{CC} standby if the internal RAM is to be used regardless of whether standby power operation is anticipated.

The RAM is controlled by the RAM control register.

RAM CONTROL REGISTER ($14)

The RAM control register includes two bits which can be used to control RAM accesses and determine the adequacy of the standby power source during power-down operation. It is intended that RAME be cleared and STBY PWR be set as part of a power-down procedure.

RAM CONTROL REGISTER

7	6	5	4	3	2	1	0	
STBY PWR	RAME	X	X	X	X	X	X	$14

Bits 0-5 Not used.

Bit 6 **RAM Enable** – This read/write bit can be used to remove the entire RAM from the internal memory map. RAME is set (enabled) during reset provided standby power is available on the positive edge of $\overline{\text{RESET}}$. If RAME is clear, any access to a RAM address is external. If RAME is set, the RAM is included in the internal map.

Bit 7 **Standby Power** – This bit is a read/write status bit which when cleared indicates that V_{CC} standby has decreased sufficiently below V_{SBB} (minimum) to make data in the standby RAM suspect. It can be set only by software and is not affected during reset.

PROGRAMMABLE TIMER

The programmable timer can be used to perform measurements on two separate input waveforms while independently generating three output waveforms. Pulse widths can vary from several microseconds to many seconds. A block diagram of the timer is shown in Figure 21.

COUNTER ($09:0A), ($15, $16)

The key timer element is a 16-bit free-running counter which is incremented by E (enable). It is cleared during reset and is read-only with one exception: in mode 0 a write to the counter ($09) will configure it to $FFF8. This feature, intended for testing, can disturb serial operations because the counter provides the SCI internal bit rate clock. The TOF is set whenever the counter contains all ones. If ETOI is set, an interrupt will occur when the TOF is set. The counter may also be read as $15 and $16 to avoid inadvertently clearing the TOF.

OUTPUT COMPARE REGISTERS ($0B:0C), ($1A:1B), ($1C:1D)

The three output compare registers are 16-bit read/write registers, each used to control an output waveform or provide an arbitrary time-out flag. They are compared with the free-running counter during the negative half of each E cycle. When a match occurs, the corresponding output compare flag (OCF) is set and the corresponding output level (OLVL) is clocked to an output level register. If both the corresponding output enable bit and data direction register bit are set, the value represented in the output level register will appear on the corresponding port pin. The appropriate OLVL bit can then be changed for the next compare.

The function is inhibited for one cycle after a write to its high byte ($0B, $1A, or $1C) to ensure a valid compare after a double byte write. Writes can be made to either byte of the output compare register without affecting the other byte. The OLVL value will be clocked out independently of whether the OCF had previously been cleared. The output compare registers are set to $FFFF during reset.

INPUT CAPTURE REGISTERS ($0D:0E), ($1E:1F)

The two input capture registers are 16-bit read-only registers used to store the free-running counter when a "proper" input transition occurs as defined by the corresponding input edge bit (IEDG1 or IEDG2). The input pin's data direction register should be configured as an input, but the edge detect circuit always senses P10 and P20 even when configured as an output. The counter value will be latched into the input capture registers on the second negative edge of the E clock following the transition.

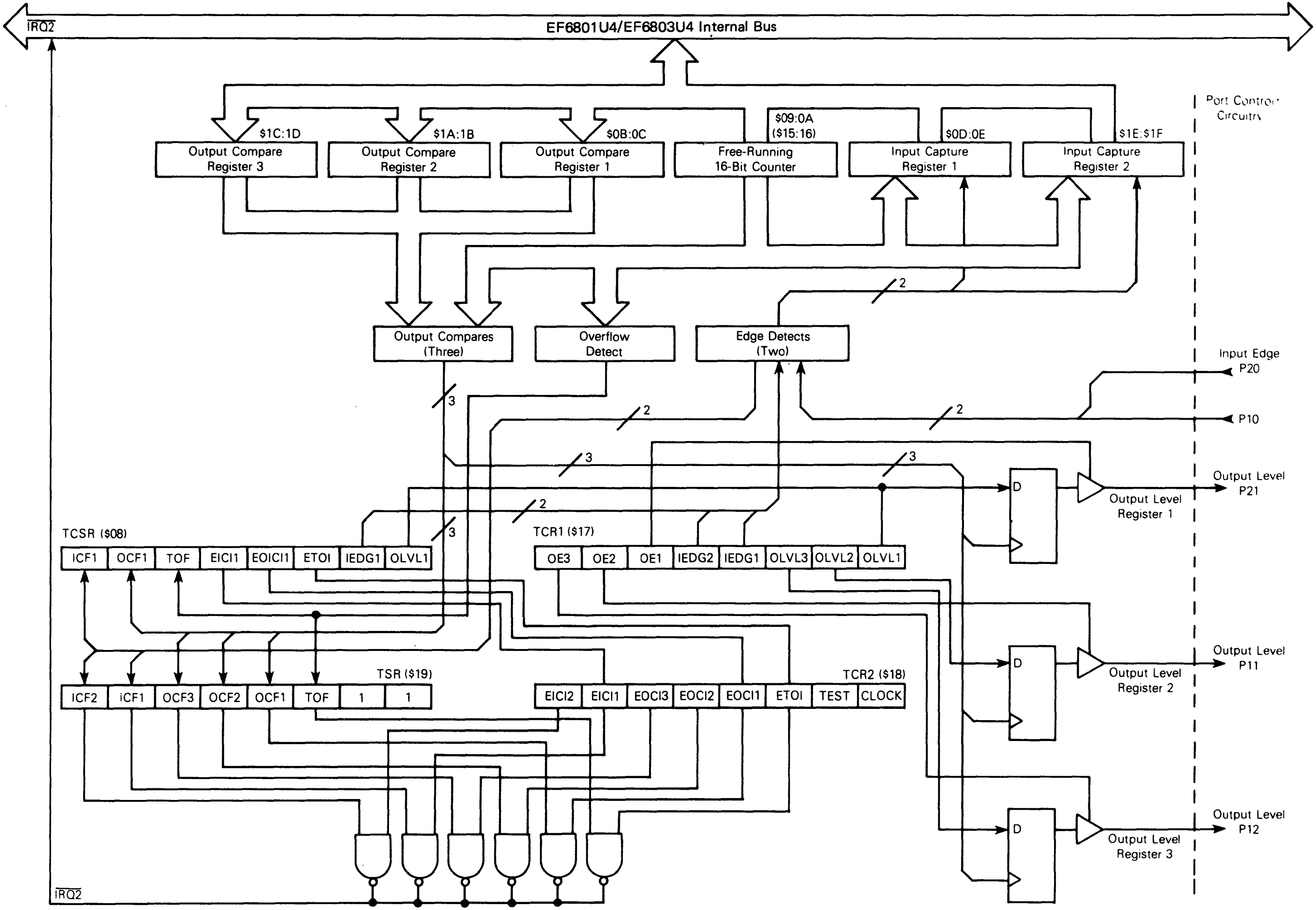
An input capture can occur independently of ICF; the register always contains the most current value. Counter transfer is inhibited, however, between accesses of a double byte MPU read. The input pulse width must be at least two E cycles to ensure an input capture under all conditions.

TIMER CONTROL AND STATUS REGISTERS

Four registers are used to provide the EF6801U4/EF6803U4 with control and status information about the three output compare functions, the timer overflow function, and the two input edge functions of the timer. They are:

Timer Control and Status Register (TCSR)
Timer Control Register 1 (TCR1)
Timer Control Register 2 (TCR2)
Timer Status Register (TSR)

FIGURE 21 — BLOCK DIAGRAM OF PROGRAMMABLE TIMER



TIMER CONTROL AND STATUS REGISTER (TCSR) ($08) – The timer control and status register is an 8-bit register of which all bits are readable, while only bits 0-4 can be written. All the bits in this register are also accessible through the two timer control registers and the timer status register. The three most significant bits provide the timer status and indicate if:

1. a proper level transition has been detected at P20,
2. a match has occurred between the free-running counter and output compare register 1, or
3. the free-running counter has overflowed.

Each of the three events can generate an $\overline{\text{IRQ2}}$ interrupt and is controlled by an individual enable bit in the TCSR.

TIMER CONTROL AND STATUS REGISTER

7	6	5	4	3	2	1	0	
ICF1	OCF1	TOF	EICI1	EOCI1	ETOI	IEDG1	OLVL1	$08

Bit 0 **Output Level 1** – OLVL1 is clocked to output level register 1 by a successful output compare and will appear at P21 if bit 1 of the port 2 data direction register is set and the OE1 control bit in timer control register 1 is set. OLVL1 and output level register 1 are cleared during reset. Refer to **TIMER CONTROL REGISTER 1 (TCR1) ($17)**.

Bit 1 **Input Edge 1** – IEDG1 is cleared during reset and controls which level transition on P20 will trigger a counter transfer to input capture register 1:
IEDG1 = 0 transfer on a negative-edge
IEDG1 = 1 transfer on a positive-edge
Refer to **TIMER CONTROL REGISTER 1 (TCR1) ($17)**.

Bit 2 **Enable Timer Overflow Interrupt** – When set, an $\overline{\text{IRQ2}}$ interrupt will be generated when the timer overflow flag is set; when clear, the interrupt is inhibited. ETOI is cleared during reset. Refer to **TIMER CONTROL REGISTER 2 (TCR2) ($18)**.

Bit 3 **Enable Output Compare Interrupt 1** – When set, an $\overline{\text{IRQ2}}$ interrupt will be generated when output compare flag 1 is set; when clear, the interrupt is inhibited. EOCI1 is cleared during reset. Refer to **TIMER CONTROL REGISTER 2 (TCR2) ($18)**.

Bit 4 **Enable Input Capture Interrupt 1** – When set, an $\overline{\text{IRQ2}}$ interrupt will be generated when input capture flag 1 is set; when clear, the interrupt is inhibited. EICI1 is cleared during reset. Refer to **TIMER CONTROL REGISTER 2 (TCR2) ($18).**

Bit 5 **Timer Overflow Flag** – The TOF is set when the counter contains all ones ($FFFF). It is cleared by reading the TCSR or the TSR (with TOF set) and the counter high byte ($09), or during reset. Refer to **TIMER STATUS REGISTER (TSR) ($19)**.

Bit 6 **Output Compare Flag 1** – OCF1 is set when output compare register 1 matches the free-running counter. OCF1 is cleared by reading the TCSR or the TSR (with OCF1 set) and then writing to output compare register 1 ($0B or $0C), or during reset. Refer to **TIMER STATUS REGISTER (TSR) ($19)**.

Bit 7 **Input Capture Flag** – ICF1 is set to indicate that a proper level transition has occurred; it is cleared by reading the TCSR or the TSR (with ICF1 set) and the input capture register 1 high byte ($0D), or during reset. Refer to **TIMER STATUS REGISTER (TSR) ($19)**.

TIMER CONTROL REGISTER 1 (TCR1) ($17) – Timer control register 1 is an 8-bit read/write register which contains the control bits for interfacing the output compare and input capture registers to the corresponding I/O pins.

TIMER CONTROL REGISTER 1

7	6	5	4	3	2	1	0	
OE3	OE2	OE1	IEDG2	IEDG1	OLVL3	OLVL2	OLVL1	$17

Bit 0 **Output Level 1** – OLVL1 is clocked to output level register 1 by a successful output compare and will appear at P21 if bit 1 of the port 2 data direction register is set and the OE1 control bit is set. OLVL1 and output level register 1 are cleared during reset. Refer to **TIMER CONTROL AND STATUS REGISTER (TCSR) ($08)**.

Bit 1 **Output Level 2** – OLVL2 is clocked to output level register 2 by a successful output compare and will appear at P11 if bit 1 of port 1 data direction register is set and the OE2 control bit is set. OLVL2 and output level register 2 are cleared during reset.

Bit 2 **Output Level 3** – OLVL3 is clocked to output level register 3 by a successful output compare and will appear at P12 if bit 2 of port 1 data direction register is set and the OE3 control bit is set. OLVL3 and output level register 3 are cleared during reset.

Bit 3 **Input Edge 1** – IEDG1 is cleared during reset and controls which level transition on P20 will trigger a counter transfer to input capture register 1.
IEDG1 = 0 transfer on a negative-edge
IEDG1 = 1 transfer on a positive-edge
Refer to **TIMER CONTROL AND STATUS REGISTER (TCSR) ($08)**.

Bit 4 **Input Edge 2** – IEDG2 is cleared during reset and controls which level transition on P10 will trigger a counter transfer to input capture register 2.
IEDG2 = 0 transfer on a negative-edge
IEDG2 = 1 transfer on a positive-edge

Bit 5 **Output Enable 1** – OE1 is set during reset and enables the contents of output level register 1 to be connected to P21 when bit 1 of port 2 data direction register is set.
OE1 = 0 port 2 bit 1 data register output
OE1 = 1 output level register 1

Bit 6 **Output Enable 2** – OE2 is cleared during reset and enables the contents of output level register 2 to be connected to P11 when bit 1 of port 1 data direction register is set.
OE2 = 0 port 1 bit 1 data register output
OE2 = 1 output level register 2

Bit 7 **Output Enable 3** – OE3 is cleared during reset and enables the contents of output level register 3 to be connected to P12 when bit 2 of port 1 data direction register is set

OE3 = 0 port 1 bit 2 data register output
OE3 = 1 output level register 3

TIMER CONTROL REGISTER 2 (TCR2) ($18) – Timer control register 2 is an 8-bit read/write register (except bits 0 and 1) which enable the interrupts associated with the free-running counter, the output compare registers, and the input capture registers. In test mode 0, two more bits (clock and test) are available for checking the timer.

TIMER CONTROL REGISTER 2
(Non-Test Modes)

7	6	5	4	3	2	1	0	
EICI2	EICI1	EOCI3	EOCI2	EOCI1	ETOI	1	1	$18

Bits 0-1 **Read-Only Bits** – When read, these bits return a value of 1. Refer to **TIMER CONTROL REGISTER 2 (Test Mode)**.

Bit 2 **Enable Timer Overflow Interrupt** – When set, an $\overline{\text{IRQ2}}$ interrupt will be generated when the timer overflow flag is set; when clear, the interrupt is inhibited. ETOI is cleared during reset. Refer to **TIMER CONTROL AND STATUS REGISTER (TCSR) ($08)**.

Bit 3 **Enable Output Compare Interrupt 1** – When set, an $\overline{\text{IRQ2}}$ interrupt will be generated when the output compare flag 1 is set; when clear, the interrupt is inhibited. EOCI1 is cleared during reset. Refer to **TIMER CONTROL AND STATUS REGISTER (TCSR) ($08)**.

Bit 4 **Enable Output Compare Interrupt 2** – When set, an $\overline{\text{IRQ2}}$ interrupt will be generated when the output compare flag 2 is set; when clear, the interrupt is inhibited. EOCI2 is cleared during reset.

Bit 5 **Enable Output Compare Interrupt 3** – When set, an $\overline{\text{IRQ2}}$ interrupt will be generated when the output compare flag 3 is set; when clear, the interrupt is inhibited. EOCI3 is cleared during reset.

Bit 6 **Enable Input Capture Interrupt 1** – When set, an $\overline{\text{IRQ2}}$ interrupt will be generated when the input capture flag 1 is set; when clear, the interrupt is inhibited. EICI1 is cleared during reset. Refer to **TIMER CONTROL AND STATUS REGISTER (TCSR) ($08)**.

Bit 7 **Enable Input Capture Interrupt 2** – When set, an $\overline{\text{IRQ2}}$ interrupt will be generated when the input capture flag 2 is set; when clear, the interrupt is inhibited. EICI2 is cleared during reset.

The timer test bits (test and clock) allow the free-running counter to be tested as two separate 8-bit counters to speed testing.

TIMER CONTROL REGISTER 2
(Test Mode)

7	6	5	4	3	2	1	0	
EICI2	EICI1	EOCI3	EOCI2	EOCI1	ETOI	TEST	CLOCK	$18

Bit 0 **CLOCK** – The CLOCK control bit selects which half of the 16-bit free-running counter (MSB or LSB) should be clocked with E. The CLOCK bit is a read/write bit only in mode 0 and is set during reset.

CLOCK = 0 – Only the eight most significant bits of the free-running counter run with TEST = 0.
CLOCK = 1 – Only the eight least significant bits of the free-running counter run when TEST = 0.

Bit 1 **TEST** – the TEST control bit enables the timer test mode. TEST is a read/write bit in mode 0 and is set during reset.

TEST = 0 – Timer test mode enabled:
a) The timer LSB latch is transparent which allows the LSB to be read independently of the MSB.
b) Either the MSB or the LSB of the timer is clocked by E, as defined by the CLOCK bit.
TEST = 1 – Timer test mode disabled.

Bits 2-7 See **TIMER CONTROL REGISTER 2 (Non-Test Modes)**. (These bits function the same as in the non-test modes.)

TIMER STATUS REGISTER (TSR) ($19) – The timer status register is an 8-bit read-only register which contains the flags associated with the free-running counter, the output compare registers, and the input capture registers.

TIMER STATUS REGISTER

7	6	5	4	3	2	1	0	
ICF2	ICF1	OCF3	OCF2	OCF1	TOF	1	1	$19

Bits 0-1 Not used.

Bit 2 **Timer Overflow Flag** – The TOF is set when the counter contains all ones ($FFFF). It is cleared by reading the TSR or the TCSR (with TOF set) and then the counter high byte ($09), or during reset. Refer to **TIMER CONTROL AND STATUS REGISTER (TCSR) ($08)**.

Bit 3 **Output Compare Flag 1** – OCF1 is set when output compare register 1 matches the free-running counter. OCF1 is cleared by reading the TSR or the TCSR (with OCF1 set) and then writing to output compare register 1 ($0B or $0C), or during reset. Refer to **TIMER CONTROL AND STATUS REGISTER (TCSR) ($08)**.

Bit 4 **Output Compare Flag 2** – OCF2 is set when output compare register 2 matches the free-running counter. OCF2 is cleared by reading the TSR (with OCF2 set) and then writing to output compare register 2 ($1A or $1B), or during reset.

Bit 5 **Output Compare Flag 3** – OCF3 is set when output compare register 3 matches the free-running counter. OCF3 is cleared by reading the TSR (with OCF3 set) and then writing to output compare register 3 ($1C or $1D), or during reset.

Bit 6 **Input Capture Flag 1** – ICF1 is set to indicate that a proper level transition has occurred; it is cleared by reading the TSR or the TCSR (with ICF1 set) and the input capture register 1 high byte ($0D), or during reset. Refer to **TIMER CONTROL AND STATUS REGISTER (TCSR) ($08)**.

Bit 7 **Input Capture Flag 2** – ICF2 is set to indicate that a proper level transition has occurred; it is cleared by reading the TSR (with ICF2 set) and the input capture register 2 high byte ($1E), or during reset.

SERIAL COMMUNICATIONS INTERFACE

A full-duplex asynchronous serial communications interface (SCI) is provided with two data formats and a variety of rates. The SCI transmitter and receiver are functionally independent but use the same data format and bit rate. Serial data formats include standard mark/space (NRZ) and bi-phase and both provide one start bit, eight data bits, and one stop bit. "Baud" and "bit rate" are used synonymously in the following description.

WAKE-UP FEATURE

In a typical serial loop multiprocessor configuration, the software protocol will usually identify the addressee(s) at the beginning of the message. In order to permit uninterested MPUs to ignore the remainder of the message, wake-up feature is included whereby all further SCI receiver flag (and interrupt) processing can be inhibited until its data line goes idle. An SCI receiver is re-enabled by an idle string of ten consecutive ones or during reset. Software must provide for the required idle string between consecutive messages and prevent it within messages.

PROGRAMMABLE OPTIONS

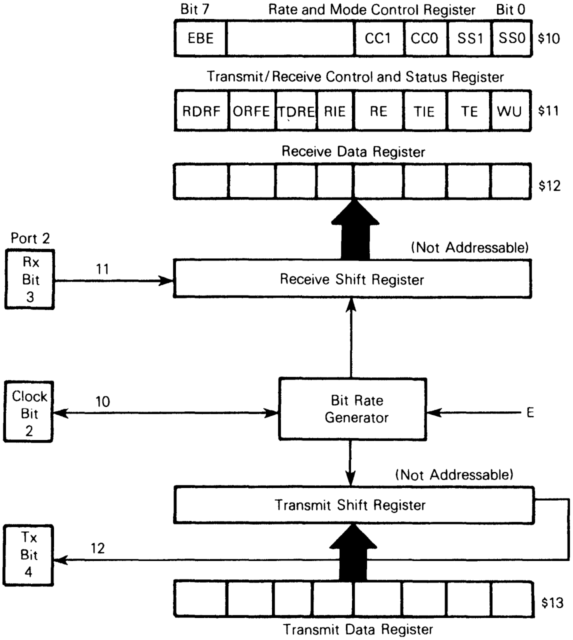
The following features of the SCI are programmable:

- Format: standard mark/space (NRZ) or bi-phase
- Clock: external or internal bit rate clock
- Baud: one of eight per E clock frequency or external clock (× 8 desired baud)
- Wake-Up Feature: enabled or disabled
- Interrupt Requests: enabled individually for transmitter and receiver
- Clock Output: internal bit rate clock enabled or disabled to P22

SERIAL COMMUNICATIONS REGISTERS

The serial communications interface includes four addressable registers as depicted in Figure 22. It is controlled by the rate and mode control register and the transmit/receive control and status register. Data is transmitted and received utilizing a write-only transmit register and a read-only receive register. The shift registers are not accessible to software.

FIGURE 22 – SCI REGISTERS



RATE AND MODE CONTROL REGISTER (RMCR) ($10) – The rate and mode control register controls the SCI bit rate, format, clock source, and under certain conditions, the configuration of P22. The register consists of five write-only bits which are cleared during reset. The two least significant bits in conjunction with bit 7 control the bit rate of the internal clock and the remaining two bits control the format and clock source.

RATE AND MODE CONTROL REGISTER

7	6	5	4	3	2	1	0	
EBE	X	X	X	CC1	CC0	SS1	SS0	$10

Bit 1:Bit 0 **SS1:SS0 Speed Select** – These two bits select the baud when using the internal clock. Eight rates may be selected (in conjunction with bit 7) which are a function of the MCU input frequency. Table 6 lists bit time and rates for three selected MCU frequencies.

Bit 3:Bit 2 **CC1:CC0 Clock Control and Format Select** – These two bits control the format and select the serial clock source. If CC1 is set, the DDR value for P22 is forced to the complement of CC0 and cannot be altered until CC1 is cleared. If CC1 is cleared after having been set, its DDR value is unchanged. Table 7 defines the formats, clock source, and use of P22.

Bits 4-6 Not used.

Bit 7 **EBE Enhanced Baud Enable** – EBE selects the standard EF6801 baud rates when clear and the additional baud rates when set (Table 6). This bit is cleared by reset and is a write-only control bit.

EBE=0 standard EF6801 baud rates
EBE=1 additional baud rates

If both CC1 and CC0 are set, an external TTL-compatible clock must be connected to P22 at eight times (8×) the desired bit rate, but not greater than E, with a duty cycle of 50% (±10%). If CC1:CC0=10, the internal bit rate clock is provided at P22 regardless of the values for TE or RE.

NOTE

The source of SCI internal bit rate clock is the timer free-running counter. An MPU write to the counter in mode 0 can disturb serial operations.

TABLE 6 – SCI BIT TIMES AND RATES

EBE	SS1:SS0		$4 f_0 \rightarrow$	2.4576 MHz		4.0 MHz		4.9152 MHz	
				614.4 kHz		1.0 MHz		1.2288 MHz	
			E	Baud	Time	Baud	Time	Baud	Time
0	0	0	÷16	38400.0	26 μs	62500.0	16.0 μs	76800.0	13.0 μs
0	0	1	÷128	4800.0	208.3 μs	7812.5	128.0 μs	9600.0	104.2 μs
0	1	0	÷1024	600.0	1.67 ms	976.6	1.024 ms	1200.0	833.3 μs
0	1	1	÷4096	150.0	6.67 ms	244.1	4.096 ms	300.0	3.33 ms
1	0	0	÷64	9600.0	104.2 μs	15625.0	64 μs	19200.0	52.0 μs
1	0	1	÷256	2400.0	416.6 μs	3906.3	256 μs	4800.0	208.3 μs
1	1	0	÷512	1200.0	833.3 μs	1953.1	512 μs	2400.0	416.6 μs
1	1	1	÷2048	300.0	3.33 ms	488.3	2.05 ms	600.0	1.67 ms
External (P22)*				76800.0	13.0 μs	125000.0	8.0 μs	153600.0	6.5 μs

* Using maximum clock rate

TABLE 7 – SCI FORMAT AND CLOCK SOURCE CONTROL

CC1:CC0	Format	Clock Source	Port 2 Bit 2
00	Bi-Phase	Internal	Not Used
01	NRZ	Internal	Not Used
10	NRZ	Internal	Output
11	NRZ	External	Input

TRANSMIT/RECEIVE CONTROL AND STATUS REGISTER (TRCSR) ($11) – The transmit/receive control and status register controls the transmitter, receiver, wake-up feature, and two individual interrupts, and monitors the status of serial operations. All eight bits are readable while bits 0 to 4 are also writable. The register is initialized to $20 by $\overline{\text{RESET}}$.

TRANSMIT/RECEIVE CONTROL AND STATUS REGISTER

7	6	5	4	3	2	1	0	
RDRF	ORFE	TDRE	RIE	RE	TIE	TE	WU	$11

Bit 0 **"Wake-Up" on Idle Line** – When set, WU enables the wake-up function; it is cleared by ten consecutive ones or during reset. WU will not be set if the line is idle. Refer to **WAKE-UP FEATURE.**

Bit 1 **Transmit Enable** – When set, P24 DDR bit is set, cannot be changed, and will remain set if TE is subsequently cleared. When TE is changed from clear to set, the transmitter is connected to P24 and a preamble of nine consecutive ones is transmitted. TE is cleared during reset.

Bit 2 **Transmit Interrupt Enable** – When set, an $\overline{\text{IRQ2}}$ is set; when clear, the interrupt is inhibited. TE is cleared during reset.

Bit 3 **Receive Enable** – When set, the P23 DDR bit is cleared, cannot be changed, and will remain clear if RE is subsequently cleared. While RE is set, the SCI receiver is enabled. RE is cleared during reset.

Bit 4 **Receiver Interrupt Enable** – When set, an $\overline{\text{IRQ2}}$ interrupt is enabled when RDRF and/or ORFE is set; when clear, the interrupt is inhibited. RIE is cleared during reset.

Bit 5 **Transmit Data Register Empty** – TDRE is set when the transmit data register is transferred to the output serial shift register or during reset. It is cleared by reading the TRCSR (with TDRE set) and then writing to the transmit data register. Additional data will be transmitted only if TDRE has been cleared.

Bit 6 **Overrun Framing Error** – If set, ORFE indicates either an overrun or framing error. An overrun is a new byte ready to transfer to the receiver data register with RDRF still set. A receiver framing error has occurred when the byte boundaries of the bit stream are not synchronized to the bit counter. An overrun can be distinguished from a framing error by the state of RDRF: if RDRF is set, then an overrun has occurred; otherwise, a framing error has been detected. Data is not transferred to the receive data register in an overrun condition. Unframed data causing a framing error is transferred to the receive data register. However, subsequent data transfer is blocked until the framing error flag is cleared. ORFE is cleared by reading the TRCSR (with ORFE set) then the receive data register, or during reset.

Bit 7 **Receive Data Register Full** – RDRF is set when the input serial shift register is transferred to the receive data register, or during reset.

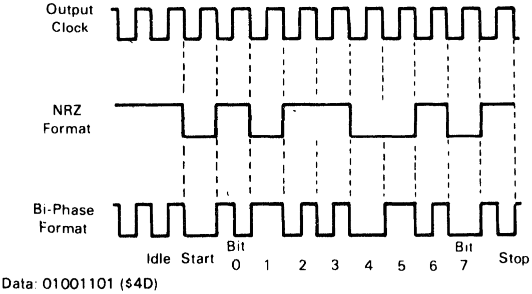
SERIAL OPERATIONS

The SCI is initialized by writing control bytes first to the rate and mode control register and then to the transmit/receive control and status register. When TE is set, the output of the transmit serial shift register is connected to P24 and serial output is initiated by transmitting a 9-bit preamble of ones.

At this point, one of two situations exists: 1) if the transmit data register is empty (TDRE=1), a continuous string of ones will be sent indicating an idle line; or 2) if a byte has been written to the transmit data register (TDRE=0), it will be transferred to the output serial shift register (synchronized with the bit rate clock), TDRE will be set, and transmission will begin.

The start bit (0), eight data bits (beginning with bit 0), and a stop bit (1) will be transmitted. If TDRE is still set when the next byte transfer occurs, ones will be sent until more data is provided. In bi-phase format, the output toggles at the start of each bit and at half-bit time when a one is sent. Receive operation is controlled by RE which configures P23 as an input and enables the receiver. SCI data formats are illustrated in Figure 23.

FIGURE 23 – SCI DATA FORMATS



INSTRUCTION SET

The EF6801U4/EF6803U4 is directly source compatible with the EF6801 and upward source and object code compatible with the EF6800. Execution times of key instructions have been reduced and several instructions have been added, including a hardware multiply. A list of new operations added to the EF6800 instruction set is shown in Table 1.

In addition, two special opcodes, 4E and 5E, are provided for test purposes. These opcodes force the program counter to increment like a 16-bit counter causing address lines used in the expanded modes to increment until the device is reset. These opcodes have no mnemonics.

The coding of the first (or only) byte corresponding to an executable instruction is sufficient to identify the instruction and the addressing mode. The hexadecimal equivalents of the binary codes, which result from the translation of the 82 instructions in all valid modes of addressing, are shown in Table 8. There are 220 valid machine codes, 34 unassigned codes, and 2 codes reserved for test purposes.

TABLE 8 – CPU INSTRUCTION MAP

OP	MNEM	MODE	~	#	OP	MNEM	MODE	~	#	OP	MNEM	MODE	~	#	OP	MNEM	MODE	~	#	OP	MNEM	MODE	~	#
00	•				34	DES	INHER	3	1	68	ASL	INDXD	6	2	9C	CPX	DIR	5	2	D0	SUBB	DIR	3	2
01	NOP	INHER	2	1	35	TXS	▲	3	1	69	ROL	▲	6	2	9D	JSR	▲	5	2	D1	CMPB	▲	3	2
02	•	▲			36	PSHA		3	1	6A	DEC		6	2	9E	LDS	▼	4	2	D2	SBCB		3	2
03	•				37	PSHB		3	1	6B	•				9F	STS	DIR	4	2	D3	ADDD		5	2
04	LSRD		3	1	38	PULX		5	1	6C	INC		6	2	A0	SUBA	INDXD	4	2	D4	ANDB		3	2
05	ASLD		3	1	39	RTS		5	1	6D	TST		6	2	A1	CMPA	▲	4	2	D5	BITB		3	2
06	TAP		2	1	3A	ABX		3	1	6E	JMP	▼	3	2	A2	SBCA		4	2	D6	LDAB		3	2
07	TPA		2	1	3B	RTI		10	1	6F	CLR	INDXD	6	2	A3	SUBD		6	2	D7	STAB		3	2
08	INX		3	1	3C	PSHX		4	1	70	NEG	EXTND	6	3	A4	ANDA		4	2	D8	EORB		3	2
09	DEX		3	1	3D	MUL		10	1	71	•	▲			A5	BITA		4	2	D9	ADCB		3	2
0A	CLV		2	1	3E	WAI		9	1	72	•				A6	LDAA		4	2	DA	ORAB		3	2
0B	SEV		2	1	3F	SWI		12	1	73	COM		6	3	A7	STAA		4	2	DB	ADDB		3	2
0C	CLC		2	1	40	NEGA		2	1	74	LSR		6	3	A8	EORA		4	2	DC	LDD		4	2
0D	SEC		2	1	41	•				75	•				A9	ADCA		4	2	DD	STD		4	2
0E	CLI		2	1	42	•				76	ROR		6	3	AA	ORAA		4	2	DE	LDX	▼	4	2
0F	SEI		2	1	43	COMA		2	1	77	ASR		6	3	AB	ADDA		4	2	DF	STX	DIR	4	2
10	SBA		2	1	44	LSRA		2	1	78	ASL		6	3	AC	CPX		6	2	E0	SUBB	INDXD	4	2
11	CBA		2	1	45	•				79	ROL		6	3	AD	JSR		6	2	E1	CMPB	▲	4	2
12	•				46	RORA		2	1	7A	DEC		6	3	AE	LDS	▼	5	2	E2	SBCB		4	2
13	•				47	ASRA		2	1	7B	•				AF	STS	INDXD	5	2	E3	ADDD		6	2
14	•				48	ASLA		2	1	7C	INC		6	3	B0	SUBA	EXTND	4	3	E4	ANDB		4	2
15	•				49	ROLA		2	1	7D	TST		6	3	B1	CMPA	▲	4	3	E5	BITB		4	2
16	TAB		2	1	4A	DECA		2	1	7E	JMP	▼	3	3	B2	SBCA		4	3	E6	LDAB		4	2
17	TBA		2	1	4B	•				7F	CLR	EXTND	6	3	B3	SUBD		6	3	E7	STAB		4	2
18	•	▼			4C	INCA		2	1	80	SUBA	IMMED	2	2	B4	ANDA		4	3	E8	EORB		4	2
19	DAA	INHER	2	1	4D	TSTA		2	1	81	CMPA	▲	2	2	B5	BITA		4	3	E9	ADCB		4	2
1A	•				4E	T				82	SBCA		2	2	B6	LDAA		4	3	EA	ORAB		4	2
1B	ABA	INHER	2	1	4F	CLRA		2	1	83	SUBD		4	3	B7	STAA		4	3	EB	ADDB		4	2
1C	•				50	NEGB		2	1	84	ANDA		2	2	B8	EORA		4	3	EC	LDD		5	2
1D	•				51	•				85	BITA		2	2	B9	ADCA		4	3	ED	STD		5	2
1E	•				52	•				86	LDAA		2	2	BA	ORAA		4	3	EE	LDX	▼	5	2
1F	•				53	COMB		2	1	87	•				BB	ADDA		4	3	EF	STX	INDXD	5	2
20	BRA	REL	3	2	54	LSRB		2	1	88	EORA		2	2	BC	CPX		6	3	F0	SUBB	EXTND	4	3
21	BRN	▲	3	2	55	•				89	ADCA		2	2	BD	JSR		6	3	F1	CMPB	▲	4	3
22	BHI		3	2	56	RORB		2	1	8A	ORAA		2	2	BE	LDS	▼	5	3	F2	SBCB		4	3
23	BLS		3	2	57	ASRB		2	1	8B	ADDA	▼	2	2	BF	STS	EXTND	5	3	F3	ADDD		6	3
24	BCC		3	2	58	ASLB		2	1	8C	CPX	IMMED	4	3	C0	SUBB	IMMED	2	2	F4	ANDB		4	3
25	BCS		3	2	59	ROLB		2	1	8D	BSR	REL	6	2	C1	CMPB	▲	2	2	F5	BITB		4	3
26	BNE		3	2	5A	DECB		2	1	8E	LDS	IMMED	3	3	C2	SBCB		2	2	F6	LDAB		4	3
27	BEQ		3	2	5B	•				8F	•				C3	ADDD		4	3	F7	STAB		4	3
28	BVC		3	2	5C	INCB		2	1	90	SUBA	DIR	3	2	C4	ANDB		2	2	F8	EORB		4	3
29	BVS		3	2	5D	TSTB		2	1	91	CMPA	▲	3	2	C5	BITB		2	2	F9	ADCB		4	3
2A	BPL		3	2	5E	T	▼			92	SBCA		3	2	C6	LDAB		2	2	FA	ORAB		4	3
2B	BMI		3	2	5F	CLRB	INHER	2	1	93	SUBD		5	2	C7	•				FB	ADDB		4	3
2C	BGE		3	2	60	NEG	INDXD	6	2	94	ANDA		3	2	C8	EORB		2	2	FC	LDD		5	3
2D	BLT		3	2	61	•	▲			95	BITA		3	2	C9	ADCB		2	2	FD	STD		5	3
2E	BGT	▼	3	2	62	•				96	LDAA		3	2	CA	ORAB		2	2	FE	LDX	▼	5	3
2F	BLE	REL	3	2	63	COM		6	2	97	STAA		3	2	CB	ADDB		2	2	FF	STX	EXTND	5	3
30	TSX	INHER	3	1	64	LSR		6	2	98	EORA		3	2	CC	LDD		3	3					
31	INS	▲	3	1	65	•				99	ADCA		3	2	CD	•	▼				* UNDEFINED OP CODE			
32	PULA		4	1	66	ROR	▼	6	2	9A	ORAA		3	2	CE	LDX	IMMED	3	3					
33	PULB	▼	4	1	67	ASR	INDXD	6	2	9B	ADDA	▼	3	2	CF	•								

NOTES:

1. Addressing Modes
   INHER≡Inherent INDXD≡Indexed IMMED≡Immediate
   REL≡Relative EXTND≡Extended DIR≡Direct
2. Unassigned opcodes are indicated by "•" and should not be executed.
3. Codes marked by "T" force the PC to function as a 16-bit counter.

PROGRAMMING MODEL

A programming model for the EF6801U4/EF6803U4 is shown in Figure 8. Accumulator A can be concatenated with accumulator B and jointly referred to as accumulator D where A is the most significant byte. Any operation which modifies the double accumulator will also modify accumulators A and/or B. Other registers are defined as follows:

PROGRAM COUNTER – The program counter is a 16-bit register which always points to the next instruction.

STACK POINTER – The stack pointer is a 16-bit register which contains the address of the next available location in a pushdown/pullup (LIFO) queue. The stack resides in random-access memory at a location defined by the programmer.

INDEX REGISTER – The index register is a 16-bit register which can be used to store data or provide an address for the indexed mode of addressing.

ACCUMULATORS – The MPU contains two 8-bit accumulators, A and B, which are used to store operands and results from the arithmetic logic unit (ALU). They can also be concatenated and referred to as the D (double) accumulator.

CONDITION CODE REGISTER – The condition code register indicates the results of an instruction and includes the following five condition bits: negative (N), zero (Z), overflow (V), carry/borrow from MSB (C), and half carry from bit 3 (H). These bits are testable by the conditional branch instructions. Bit 4 is the interrupt mask (I bit) and inhibits all maskable interrupts when set. The two unused bits, B6 and B7, are read as ones.

ADDRESSING MODES

Six addressing modes can be used to reference memory. A summary of addressing modes for all instructions is presented in Tables 9, 10, 11, and 12 where execution times are provided in E cycles. Instruction execution times are summarized in Table 13. With an input frequency of 4 MHz, one E cycle is equivalent to one microsecond. A cycle-by-cycle description of bus activity for each instruction is provided in Table 14 and descriptions of selected instructions are shown in Figure 24.

IMMEDIATE ADDRESSING – The operand or "immediate byte(s)" is contained in the following byte(s) of the instruction where the number of bytes matches the size of the register. These are two or three byte instructions.

DIRECT ADDRESSING – The least significant byte of the operand address is contained in the second byte of the instruction and the most significant byte is assumed to be $00. Direct addressing allows the user to access $00 through $FF using two byte instructions and execution time is reduced by eliminating the additional memory access. In most applications, the 256-byte area is reserved for frequently referenced data.

EXTENDED ADDRESSING – The second and third bytes of the instruction contain the absolute address of the operand. These are three byte instructions.

INDEXED ADDRESSING – The unsigned offset contained in the second byte of the instruction is added with carry to the index register and is used to reference memory without changing the index register. These are two byte instructions.

INHERENT ADDRESSING – The operand(s) is a register and no memory reference is required. These are single byte instructions.

RELATIVE ADDRESSING – Relative addressing is used only for branch instructions. If the branch condition is true, the program counter is overwritten with the sum of a signed single byte displacement in the second byte of the instruction and the current program counter. This provides a branch range of – 126 to + 129 bytes from the first byte of the instruction. These are two byte instructions.

SUMMARY OF CYCLE-BY-CYCLE OPERATION

Table 14 provides a detailed description of the information present on the address bus, data bus, and the read/write (R/$\overline{W}$) line during each cycle of each instruction.

The information is useful in comparing actual with expected results during debug of both software and hardware as the program is executed. The information is categorized in groups according to addressing mode and number of cycles per instruction. In general, instructions with the same addressing mode and number of cycles execute in the same manner. Exceptions are indicated in the table.

Note that during MPU reads of internal locations, the resultant value will not appear on the external data bus except in mode 0. "High order" byte refers to the most significant byte of a 16-bit value.

TABLE 9 – INDEX REGISTER AND STACK MANIPULATION INSTRUCTIONS

																		Condition Codes					
		Immed			Direct			Index			Extnd			Inherent			Boolean/	5	4	3	2	1	0
Pointer Operations	MNEM	Op	~	#	Op	~	#	Op	~	#	Op	~	#	Op	~	#	Arithmetic Operation	H	I	N	Z	V	C
Compare Index Register	CPX	8C	4	3	9C	5	2	AC	6	2	BC	6	3				X − M:M + 1	•	•	↕	↕	↕	↕
Decrement Index Register	DEX													09	3	1	X − 1 → X	•	•	•	↕	•	•
Decrement Stack Pointer	DES													34	3	1	SP − 1 → SP	•	•	•	•	•	•
Increment Index Register	INX													08	3	1	X + 1 → X	•	•	•	↕	•	•
Increment Stack Pointer	INS													31	3	1	1 SP + 1 → SP	•	•	•	•	•	•
Load Index Register	LDX	CE	3	3	DE	4	2	EE	5	2	FE	5	3				M → X_H,(M + 1) → X_L	•	•	↕	↕	R	•
Load Stack Pointer	LDS	8E	3	3	9E	4	2	AE	5	2	BE	5	3				M → SP_H,(M + 1) → SP_L	•	•	↕	↕	R	•
Store Index Register	STX				DF	4	2	EF	5	2	FF	5	3				X_H → M,X_L → (M + 1)	•	•	↕	↕	R	•
Store Stack Pointer	STS				9F	4	2	AF	5	2	BF	5	3				SP_H → M,SP_L → (M + 1)	•	•	↕	↕	R	•
Index Reg → Stack Pointer	TXS													35	3	1	X − 1 → SP	•	•	•	•	•	•
Stack Pntr → Index Register	TSX													30	3	1	SP + 1 → X	•	•	•	•	•	•
Add	ABX													3A	3	1	B + X → X	•	•	•	•	•	•
Push Data	PSHX													3C	4	1	X_L → M_{SP},SP − 1 → SP X_H → M_{SP},SP − 1 → SP	•	•	•	•	•	•
Pull Data	PULX													38	5	1	SP + 1 → SP,M_{SP} → X_H SP + 1 → SP,M_{SP} → X_L	•	•	•	•	•	•

TABLE 10 – ACCUMULATOR AND MEMORY INSTRUCTIONS (Sheet 1 of 2)

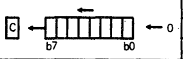
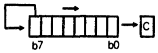
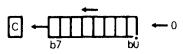
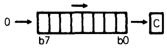
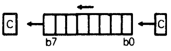
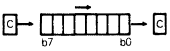
																		Condition Codes					
Accumulator and		Immed			Direct			Index			Extend			Inher			Boolean	5	4	3	2	1	0
Memory Operations	MNEM	Op	~	#	Op	~	#	Op	~	#	Op	~	#	Op	~	#	Expression	H	I	N	Z	V	C
Add Accumulators	ABA													1B	2	1	A + B → A	↕	•	↕	↕	↕	↕
Add B to X	ABX													3A	3	1	00:B + X → X	•	•	•	•	•	•
Add with Carry	ADCA	89	2	2	99	3	2	A9	4	2	B9	4	3				A + M + C → A	↕	•	↕	↕	↕	↕
	ADCB	C9	2	2	D9	3	2	E9	4	2	F9	4	3				B + M + C → B	↕	•	↕	↕	↕	↕
Add	ADDA	8B	2	2	9B	3	2	AB	4	2	BB	4	3				A + M → A	↕	•	↕	↕	↕	↕
	ADDB	CB	2	2	DB	3	2	EB	4	2	FB	4	3				B + M → A	↕	•	↕	↕	↕	↕
Add Double	ADDD	C3	4	3	D3	5	2	E3	6	2	F3	6	3				D + M:M + 1 → D	•	•	↕	↕	↕	↕
And	ANDA	84	2	2	94	3	2	A4	4	2	B4	4	3				A•M → A	•	•	↕	↕	R	•
	ANDB	C4	2	2	D4	3	2	E4	4	2	F4	4	3				B•M → B	•	•	↕	↕	R	•
Shift Left, Arithmetic	ASL							68	6	2	78	6	3					•	•	↕	↕	↕	↕
	ASLA													48	2	1		•	•	↕	↕	↕	↕
	ASLB													58	2	1		•	•	↕	↕	↕	↕
Shift Left Double	ASLD													05	3	1		•	•	↕	↕	↕	↕
Shift Right, Arithmetic	ASR							67	6	2	77	6	3					•	•	↕	↕	↕	↕
	ASRA													47	2	1		•	•	↕	↕	↕	↕
	ASRB													57	2	1		•	•	↕	↕	↕	↕
Bit Test	BITA	85	2	2	95	3	2	A5	4	2	B5	4	3				A•M	•	•	↕	↕	R	•
	BITB	C5	2	2	D5	3	2	E5	4	2	F5	4	3				B•M	•	•	↕	↕	R	•
Compare Accumulators	CBA													11	2	1	A − B	•	•	↕	↕	↕	↕
Clear	CLR							6F	6	2	7F	6	3				00 → M	•	•	R	S	R	R
	CLRA													4F	2	1	00 → A	•	•	R	S	R	R
	CLRB													5F	2	1	00 → B	•	•	R	S	R	R
Compare	CMPA	81	2	2	91	3	2	A1	4	2	B1	4	3				A − M	•	•	↕	↕	↕	↕
	CMPB	C1	2	2	D1	3	2	E1	4	2	F1	4	3				B − M	•	•	↕	↕	↕	↕
1's Complement	COM							63	6	2	73	6	3				M → M	•	•	↕	↕	R	S
	COMA													43	2	1	A → A	•	•	↕	↕	R	S
	COMB													53	2	1	B → B·	•	•	↕	↕	R	S

TABLE 10 – ACCUMULATOR AND MEMORY INSTRUCTIONS (Sheet 2 of 2)

Accumulator and		Immed			Direct			Index			Extend			Inher			Boolean	Condition Codes					
																		5	4	3	2	1	0
Memory Operations	MNEM	Op	~	#	Op	~	#	Op	~	#	Op	~	#	Op	~	#	Expression	H	I	N	Z	V	C
Decimal Adjust, A	DAA													19	2	1	Adj binary sum to BCD	•	•	↕	↕	↕	↕
Decrement	DEC							6A	6	2	7A	6	3				M – 1 → M	•	•	↕	↕	↕	•
	DECA													4A	2	1	A – 1 → A	•	•	↕	↕	↕	•
	DECB													5A	2	1	B – 1 → B	•	•	↕	↕	↕	•
Exclusive OR	EORA	88	2	2	98	3	2	A8	4	2	B8	4	3				A ⊕ M → A	•	•	↕	↕	R	•
	EORB	C8	2	2	D8	3	2	E8	4	2	F8	4	3				B ⊕ M → B	•	•	↕	↕	R	•
Increment	INC							6C	6	2	7C	6	3				M + 1 → M	•	•	↕	↕	↕	•
	INCA													4C	2	1	A + 1 → A	•	•	↕	↕	↕	•
	INCB													5C	2	1	B + 1 → B	•	•	↕	↕	↕	•
Load Accumulators	LDAA	86	2	2	96	3	2	A6	4	2	B6	4	3				M → A	•	•	↕	↕	R	•
	LDAB	C6	2	2	D6	3	2	E6	4	2	F6	4	3				M → B	•	•	↕	↕	R	•
Load Double	LDD	CC	3	3	DC	4	2	EC	5	2	FC	5	3				M:M + 1 → D	•	•	↕	↕	R	•
Logical Shift, Left	LSL							68	6	2	78	6	3					•	•	↕	↕	↕	↕
	LSLA													48	2	1		•	•	↕	↕	↕	↕
	LSLB													58	2	1		•	•	↕	↕	↕	↕
	LSLD													05	3	2		•	•	↕	↕	↕	↕
Shift Right, Logical	LSR							64	6	2	74	6	3					•	•	R	↕	↕	↕
	LSRA													44	2	1		•	•	R	↕	↕	↕
	LSRB													54	2	1		•	•	R	↕	↕	↕
	LSRD													04	3	1		•	•	R	↕	↕	↕
Multiply	MUL													3D	10	1	A × B → D	•	•	•	•	•	↕
2's Complement (Negate)	NEG							60	6	2	70	6	3				00 – M → M	•	•	↕	↕	↕	↕
	NEGA													40	2	1	00 – A → A	•	•	↕	↕	↕	↕
	NEGB													50	2	1	00 – B → B	•	•	↕	↕	↕	↕
No Operation	NOP													01	2	1	PC + 1 → PC	•	•	•	•	•	•
Inclusive OR	ORAA	8A	2	2	9A	3	2	AA	4	2	BA	4	3				A + M → A	•	•	↕	↕	R	•
	ORAB	CA	2	2	DA	3	2	EA	4	2	FA	4	3				B + M → B	•	•	↕	↕	R	•
Push Data	PSHA													36	3	1	A → Stack	•	•	•	•	•	•
	PSHB													37	3	1	B → Stack	•	•	•	•	•	•
Pull Data	PULA													32	4	1	Stack → A	•	•	•	•	•	•
	PULB													33	4	1	Stack → B	•	•	•	•	•	•
Rotate Left	ROL							69	6	2	79	6	3					•	•	↕	↕	↕	↕
	ROLA													49	2	1		•	•	↕	↕	↕	↕
	ROLB													59	2	1		•	•	↕	↕	↕	↕
Rotate Right	ROR							66	6	2	76	6	3					•	•	↕	↕	↕	↕
	RORA													46	2	1		•	•	↕	↕	↕	↕
	RORB													56	2	1		•	•	↕	↕	↕	↕
Subtract Accumulator	SBA													10	2	1	A – B → A	•	•	↕	↕	↕	↕
Subtract with Carry	SBCA	82	2	2	92	3	2	A2	4	2	B2	4	3				A – M – C → A	•	•	↕	↕	↕	↕
	SBCB	C2	2	2	D2	3	2	E2	4	2	F2	4	3				B – M – C → B	•	•	↕	↕	↕	↕
Store Accumulators	STAA				97	3	2	A7	4	2	B7	4	3				A → M	•	•	↕	↕	R	•
	STAB				D7	3	2	E7	4	2	F7	4	3				B → M	•	•	↕	↕	R	•
	STD				DD	4	2	ED	5	2	FD	5	3				D → M:M + 1	•	•	↕	↕	R	•
Subtract	SUBA	80	2	2	90	3	2	A0	4	2	B0	4	3				A – M → A	•	•	↕	↕	↕	↕
	SUBB	C0	2	2	D0	3	2	E0	4	2	F0	4	3				B – M → B	•	•	↕	↕	↕	↕
Subtract Double	SUBD	83	4	3	93	5	2	A3	6	2	B3	6	3				D – M:M + 1 → D	•	•	↕	↕	↕	↕
Transfer Accumulator	TAB													16	2	1	A → B	•	•	↕	↕	R	•
	TBA													17	2	1	B → A	•	•	↕	↕	R	•
Test, Zero or Minus	TST							6D	6	2	7D	6	3				M – 00	•	•	↕	↕	R	R
	TSTA													4D	2	1	A – 00	•	•	↕	↕	R	R
	TSTB													5D	2	1	B – 00	•	•	↕	↕	R	R

The condition code register notes are listed after Table 12.

TABLE 11 – JUMP AND BRANCH INSTRUCTIONS

		Direct			Relative			Index			Extend			Inherent				Condition Code Reg.					
																		5	4	3	2	1	0
Operations	MNEM	Op	~	#	Op	~	#	Op	~	#	Op	~	#	Op	~	#	Branch Test	H	I	N	Z	V	C
Branch Always	BRA				20	3	2										None	•	•	•	•	•	•
Branch Never	BRN				21	3	2										None	•	•	•	•	•	•
Branch If Carry Clear	BCC				24	3	2										C=0	•	•	•	•	•	•
Branch If Carry Set	BCS				25	3	2										C=1	•	•	•	•	•	•
Branch If=Zero	BEQ				27	3	2										Z=1	•	•	•	•	•	•
Branch If ≥Zero	BGE				2C	3	2										N ⊕ V=0	•	•	•	•	•	•
Branch If >Zero	BGT				2E	3	2										Z+(N ⊕ V)=0	•	•	•	•	•	•
Branch If Higher	BHI				22	3	2										C+Z=0	•	•	•	•	•	•
Branch If Higher or Same	BHS				24	3	2										C=0	•	•	•	•	•	•
Branch If ≤Zero	BLE				2F	3	2										Z+(N ⊕ V)=1	•	•	•	•	•	•
Branch If Carry Set	BLO				25	3	2										C=1	•	•	•	•	•	•
Branch If Lower Or Same	BLS				23	3	2										C+Z=1	•	•	•	•	•	•
Branch If <Zero	BLT				2D	3	2										N ⊕ V=1	•	•	•	•	•	•
Branch If Minus	BMI				2B	3	2										N=1	•	•	•	•	•	•
Branch If Not Equal Zero	BNE				26	3	2										Z=0	•	•	•	•	•	•
Branch If Overflow Clear	BVC				28	3	2										V=0	•	•	•	•	•	•
Branch If Overflow Set	BVS				29	3	2										V=1	•	•	•	•	•	•
Branch If Plus	BPL				2A	3	2										N=0	•	•	•	•	•	•
Branch To Subroutine	BSR				8D	6	2										See Special Operations-Figure 24	•	•	•	•	•	•
Jump	JMP							6E	3	2	7E	3	3					•	•	•	•	•	•
Jump To Subroutine	JSR	9D	5	2				AD	6	2	BD	6	3					•	•	•	•	•	•
No Operation	NOP													01	2	1		•	•	•	•	•	•
Return From Interrupt	RTI													3B	10	1	See Special Operations-Figure 24	↕	↕	↕	↕	↕	↕
Return From Subroutine	RTS													39	5	1		•	•	•	•	•	•
Software Interrupt	SWI													3F	12	1		•	S	•	•	•	•
Wait For Interrupt	WAI													3E	9	1		•	•	•	•	•	•

TABLE 12 – CONDITION CODE REGISTER MANIPULATION INSTRUCTIONS

	Inherent					Condition Code Register					
						5	4	3	2	1	0
Operations	MNEM	Op	~	#	Boolean Operation	H	I	N	Z	V	C
Clear Carry	CLC	0C	2	1	0→C	•	•	•	•	•	R
Clear Interrupt Mask	CLI	0E	2	1	0→I	•	R	•	•	•	•
Clear Overflow	CLV	0A	2	1	0→V	•	•	•	•	R	•
Set Carry	SEC	0D	2	1	1→C	•	•	•	•	•	S
Set Interrupt Mask	SEI	0F	2	1	1→I	•	S	•	•	•	•
Set Overflow	SEV	0B	2	1	1→V	•	•	•	•	S	•
Accumulator A→CCR	TAP	06	2	1	A→CCR	↕	↕	↕	↕	↕	↕
CCR→Accumulator A	TPA	07	2	1	CCR→A	•	•	•	•	•	•

LEGEND

Op	Operation Code (Hexadecimal)
~	Number of MPU Cycles
M_{SP}	Contents of memory location pointed to by Stack Pointer
#	Number of Program Bytes
+	Arithmetic Plus
–	Arithmetic Minus
•	Boolean AND
X	Arithmetic Multiply
+	Boolean Inclusive OR
⊕	Boolean Exclusive OR
M	Complement of M
→	Transfer Into
0	Bit=Zero
00	Byte=Zero

CONDITION CODE SYMBOLS

H	Half-carry from bit 3
I	Interrupt mask
N	Negative (sign bit)
Z	Zero (byte)
V	Overflow, 2's complement
C	Carry/Borrow from MSB
R	Reset Always
S	Set Always
↕	Affected
•	Not Affected

TABLE 13 – INSTRUCTION EXECUTION TIMES IN E-CYCLES

	ADDRESSING MODE					
	Immediate	Direct	Extended	Indexed	Inherent	Relative
ABA	●	●	●	●	2	●
ABX	●	●	●	●	3	●
ADC	2	3	4	4	●	●
ADD	2	3	4	4	●	●
ADDD	4	5	6	6	●	●
AND	2	3	4	4	●	●
ASL	●	●	6	6	2	●
ASLD	●	●	●	●	3	●
ASR	●	●	6	6	2	●
BCC	●	●	●	●	●	3
BCS	●	●	●	●	●	3
BEQ	●	●	●	●	●	3
BGE	●	●	●	●	●	3
BGT	●	●	●	●	●	3
BHI	●	●	●	●	●	3
BHS	●	●	●	●	●	3
BIT	2	3	4	4	●	●
BLE	●	●	●	●	●	3
BLO	●	●	●	●	●	3
BLS	●	●	●	●	●	3
BLT	●	●	●	●	●	3
BMI	●	●	●	●	●	3
BNE	●	●	●	●	●	3
BPL	●	●	●	●	●	3
BRA	●	●	●	●	●	3
BRN	●	●	●	●	●	3
BSR	●	●	●	●	●	6
BVC	●	●	●	●	●	3
BVS	●	●	●	●	●	3
CBA	●	●	●	●	2	●
CLC	●	●	●	●	2	●
CLI	●	●	●	●	2	●
CLR	●	●	6	6	2	●
CLV	●	●	●	●	2	●
CMP	2	3	4	4	●	●
COM	●	●	6	6	2	●
CPX	4	5	6	6	●	●
DAA	●	●	●	●	2	●
DEC	●	●	6	6	2	●
DES	●	●	●	●	3	●
DEX	●	●	●	●	3	●
EOR	2	3	4	4	●	●
INC	●	●	6	6	●	●
INS	●	●	●	●	3	●

	ADDRESSING MODE					
	Immediate	Direct	Extended	Indexed	Inherent	Relative
INX	●	●	●	●	3	●
JMP	●	●	3	3	●	●
JSR	●	5	6	6	●	●
LDA	2	3	4	4	●	●
LDD	3	4	5	5	●	●
LDS	3	4	5	5	●	●
LDX	3	4	5	5	●	●
LSL	●	●	6	6	2	●
LSLD	●	●	●	●	3	●
LSR	●	●	6	6	2	●
LSRD	●	●	●	●	3	●
MUL	●	●	●	●	10	●
NEG	●	●	6	6	2	●
NOP	●	●	●	●	2	●
ORA	2	3	4	4	●	●
PSH	●	●	●	●	3	●
PSHX	●	●	●	●	4	●
PUL	●	●	●	●	4	●
PULX	●	●	●	●	5	●
ROL	●	●	6	6	2	●
ROR	●	●	6	6	2	●
RTI	●	●	●	●	10	●
RTS	●	●	●	●	5	●
SBA	●	●	●	●	2	●
SBC	2	3	4	4	●	●
SEC	●	●	●	●	2	●
SEI	●	●	●	●	2	●
SEV	●	●	●	●	2	●
STA	●	3	4	4	●	●
STD	●	4	5	5	●	●
STS	●	4	5	5	●	●
STX	●	4	5	5	●	●
SUB	2	3	4	4	●	●
SUBD	4	5	6	6	●	●
SWI	●	●	●	●	12	●
TAB	●	●	●	●	2	●
TAP	●	●	●	●	2	●
TBA	●	●	●	●	2	●
TPA	●	●	●	●	2	●
TST	●	●	6	6	2	●
TSX	●	●	●	●	3	●
TXS	●	●	●	●	3	●
WAI	●	●	●	●	9	●

TABLE 14 — CYCLE-BY-CYCLE OPERATION (Sheet 1 of 5)

Address Mode and Instructions	Cycles	Cycle #	Address Bus	R/$\overline{W}$ Line	Data Bus
IMMEDIATE					
ADC EOR	2	1	Opcode Address	1	Opcode
ADD LDA		2	Opcode Address + 1	1	Operand Data
AND ORA					
BIT SBC					
CMP SUB					
LDS	3	1	Opcode Address	1	Opcode
LDX		2	Opcode Address + 1	1	Operand Data (High Order Byte)
LDD		3	Opcode Address + 2	1	Operand Data (Low Order Byte)
CPX	4	1	Opcode Address	1	Opcode
SUBD		2	Opcode Address + 1	1	Operand Data (High Order Byte)
ADDD		3	Opcode Address + 2	1	Operand Data (Low Order Byte)
		4	Address Bus FFFF	1	Low Byte of Restart Vector
DIRECT					
ADC EOR	3	1	Opcode Address	1	Opcode
ADD LDA		2	Opcode Address + 1	1	Address of Operand
AND ORA		3	Address of Operand	1	Operand Data
BIT SBC					
CMP SUB					
STA	3	1	Opcode Address	1	Opcode
		2	Opcode Address + 1	1	Destination Address
		3	Destination Address	0	Data from Accumulator
LDS	4	1	Opcode Address	1	Opcode
LDX		2	Opcode Address + 1	1	Address of Operand
LDD		3	Address of Operand	1	Operand Data (High Order Byte)
		4	Operand Address + 1	1	Operand Data (Low Order Byte)
STS	4	1	Opcode Address	1	Opcode
STX		2	Opcode Address + 1	1	Address of Operand
STD		3	Address of Operand	0	Register Data (High Order Byte)
		4	Address of Operand + 1	0	Register Data (Low Order Byte)
CPX	5	1	Opcode Address	1	Opcode
SUBD		2	Opcode Address + 1	1	Address of Operand
ADDD		3	Operand Address	1	Operand Data (High Order Byte)
		4	Operand Address + 1	1	Operand Data (Low Order Byte)
		5	Address Bus FFFF	1	Low Byte of Restart Vector
JSR	5	1	Opcode Address	1	Opcode
		2	Opcode Address + 1	1	Irrelevant Data
		3	Subroutine Address	1	First Subroutine Opcode
		4	Stack Pointer	0	Return Address (Low Order Byte)
		5	Stack Pointer − 1	0	Return Address (High Order Byte)

TABLE 14 – CYCLE-BY-CYCLE OPERATION (Sheet 2 of 5)

Address Mode and Instructions	Cycles	Cycle #	Address Bus	R/$\overline{W}$ Line	Data Bus
EXTENDED					
JMP	3	1	Opcode Address	1	Opcode
		2	Opcode Address + 1	1	Jump Address (High Order Byte)
		3	Opcode Address + 2	1	Jump Address (Low Order Byte)
ADC EOR	4	1	Opcode Address	1	Opcode
ADD LDA		2	Opcode Address + 1	1	Address of Operand
AND ORA		3	Opcode·Address + 2	1	Address of Operand (Low Order Byte)
BIT SBC		4	Address of Operand	1	Operand Data
CMP SUB					
STA	4	1	Opcode Address	1	Opcode
		2	Opcode Address + 1	1	Destination Address (High Order Byte)
		3	Opcode Address + 2	1	Destination Address (Low Order Byte)
		4	Operand Destination Address	0	Data from Accumulator
LDS	5	1	Opcode Address	1	Opcode
LDX		2	Opcode Address + 1	1	Address of Operand (High Order Byte)
LDD		3	Opcode Address + 2	1	Address of Operand (Low Order Byte)
		4	Address of Operand	1	Operand Data (High Order Byte)
		5	Address of Operand + 1	1	Operand Data (Low Order Byte)
STS	5	1	Opcode Address	1	Opcode
STX		2	Opcode Address + 1	1	Address of Operand (High Order Byte)
STD		3	Opcode Address + 2	1	Address of Operand (Low Order Byte)
		4	Address of Operand	0	Operand Data (High Order Byte)
		5	Address of Operand + 1	0	Operand Data (Low Order Byte)
ASL LSR	6	1	Opcode Address	1	Opcode
ASR NEG		2	Opcode Address + 1	1	Address of Operand (High Order Byte)
CLR ROL		3	Opcode Address + 2	1	Address of Operand (Low Order Byte)
COM ROR		4	Address of Operand	1	Current Operand Data
DEC TST*		5	Address Bus FFFF	1	Low Byte of Restart Vector
INC		6	Address of Operand	0	New Operand Data
CPX	6	1	Opcode Address	1	Opcode
SUBD		2	Opcode Address + 1	1	Operand Address (High Order Byte)
ADDD		3	Opcode Address + 2	1	Operand Address (Low Order Byte)
		4	Operand Address	1	Operand Data (High Order Byte)
		5	Operand Address + 1	1	Operand Data (Low Order Byte)
		6	Address Bus FFFF	1	Low Byte of Restart Vector
JSR	6	1	Opcode Address	1	Opcode
		2	Opcode Address + 1	1	Address of Subroutine (High Order Byte)
		3	Opcode Address + 2	1	Address of Subroutine (Low Order Byte)
		4	Subroutine Starting Address	1	Opcode of Next Instruction
		5	Stack Pointer	0	Return Address (Low Order Byte)
		6	Stack Pointer – 1	0	Return Address (High Order Byte)

*TST does not perform the write cycle during the sixth cycle. The sixth cycle is another address bus = $FFFF.

TABLE 14 – CYCLE-BY-CYCLE OPERATION (Sheet 3 of 5)

Address Mode and Instructions	Cycles	Cycle #	Address Bus	$R/\overline{W}$ Line	Data Bus
INDEXED					
JMP	3	1	Opcode Address	1	Opcode
		2	Opcode Address + 1	1	Offset
		3	Address Bus FFFF	1	Low Byte of Restart Vector
ADC EOR	4	1	Opcode Address	1	Opcode
ADD LDA		2	Opcode Address + 1	1	Offset
AND ORA		3	Address Bus FFFF	1	Low Byte of Restart Vector
BIT SBC		4	Index Register Plus Offset	1	Operand Data
CMP SUB					
STA	4	1	Opcode Address	1	Opcode
		2	Opcode Address + 1	1	Offset
		3	Address Bus FFFF	1	Low Byte of Restart Vector
		4	Index Register Plus Offset	0	Operand Data
LDS	5	1	Opcode Address	1	Opcode
LDX		2	Opcode Address + 1	1	Offset
LDD		3	Address Bus FFFF	1	Low Byte of Restart Vector
		4	Index Register Plus Offset	1	Operand Data (High Order Byte)
		5	Index Register Plus Offset + 1	1	Operand Data (Low Order Byte)
STS	5	1	Opcode Address	1	Opcode
STX		2	Opcode Address + 1	1	Offset
STD		3	Address Bus FFFF	1	Low Byte of Restart Vector
		4	Index Register Plus Offset	0	Operand Data (High Order Byte)
		5	Index Register Plus Offset + 1	0	Operand Data (Low Order Byte)
ASL LSR	6	1	Opcode Address	1	Opcode
ASR NEG		2	Opcode Address + 1	1	Offset
CLR ROL		3	Address Bus FFFF	1	Low Byte of Restart Vector
COM ROR		4	Index Register Plus Offset	1	Current Operand Data
DEC TST*		5	Address Bus FFFF	1	Low Byte of Restart Vector
INC		6	Index Register Plus Offset	0	New Operand Data
CPX	6	1	Opcode Address	1	Opcode
SUBD		2	Opcode Address + 1	1	Offset
ADDD		3	Address Bus FFFF	1	Low Byte of Restart Vector
		4	Index Register + Offset	1	Operand Data (High Order Byte)
		5	Index Register + Offset + 1	1	Operand Data (Low Order Byte)
		6	Address Bus FFFF		Low Byte of Restart Vector
JSR	6	1	Opcode Address	1	Opcode
		2	Opcode Address + 1	1	Offset
		3	Address Bus FFFF	1	Low Byte of Restart Vector
		4	Index Register + Offset	1	First Subroutine Opcode
		5	Stack Pointer	0	Return Address (Low Order Byte)
		6	Stack Pointer – 1	0	Return Address (High Order Byte)

*TST does not perform the write cycle during the sixth cycle. The sixth cycle is another address bus = $FFFF.

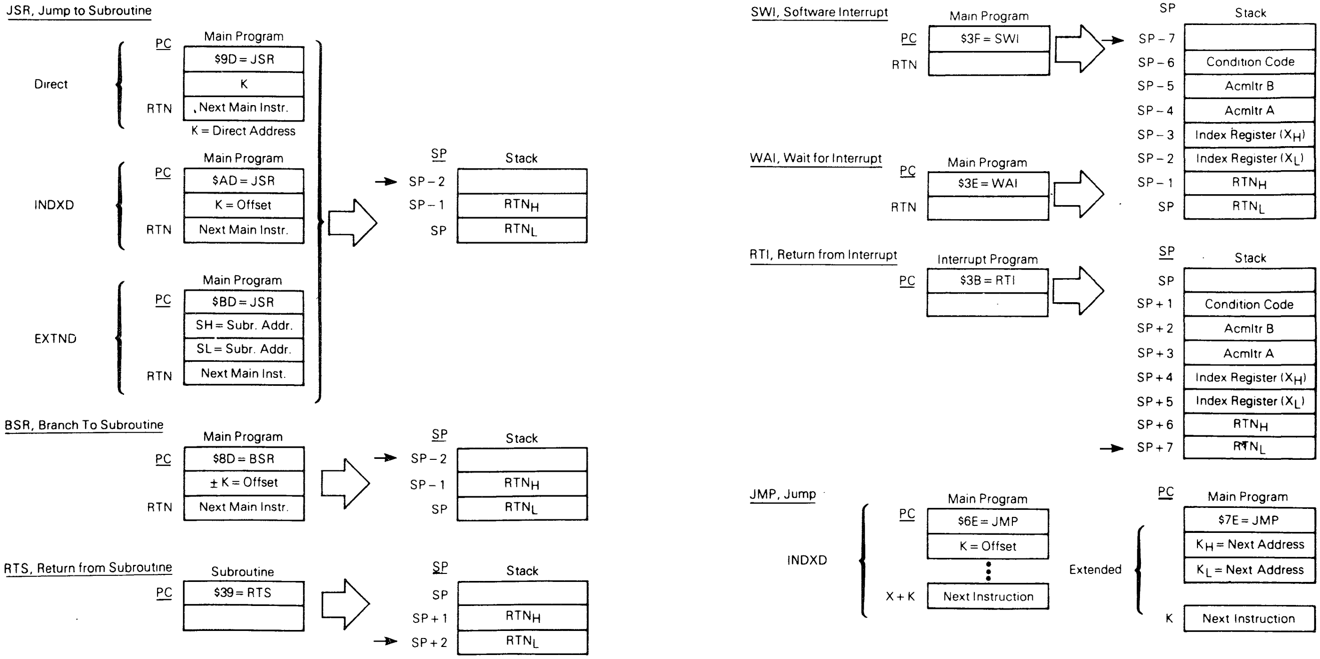
TABLE 14 – CYCLE-BY-CYCLE OPERATION (Sheet 4 of 5)

Address Mode and Instructions	Cycles	Cycle #	Address Bus	R/$\overline{W}$ Line	Data Bus
INHERENT					
ABA DAA SEC	2	1	Opcode Address	1	Opcode
ASL DEC SEI		2	Opcode Address + 1	1	Opcode of Next Instruction
ASR INC SEV					
CBA LSR TAB					
CLC NEG TAP					
CLI NOP TBA					
CLR ROL TPA					
CLV ROR TST					
COM SBA					
ABX	3	1	Opcode Address	1	Opcode
		2	Opcode Address + 1	1	Irrelevant Data
		3	Address Bus FFFF	1	Low Byte of Restart Vector
ASLD LSRD	3	1	Opcode Address	1	Opcode
		2	Opcode Address + 1	1	Irrelevant Data
		3	Address Bus FFFF	1	Low Byte of Restart Vector
DES INS	3	1	Opcode Address	1	Opcode
		2	Opcode Address + 1	1	Opcode of Next Instruction
		3	Previous Stack Pointer Contents	1	Irrelevant Data
INX DEX	3	1	Opcode Address	1	Opcode
		2	Opcode Address + 1	1	Opcode of Next Instruction
		3	Address Bus FFFF	1	Low Byte of Restart Vector
PSHA PSHB	3	1	Opcode Address	1	Opcode
		2	Opcode Address + 1	1	Opcode of Next Instruction
		3	Stack Pointer	0	Accumulator Data
TSX	3	1	Opcode Address	1	Opcode
		2	Opcode Address + 1	1	Opcode of Next Instruction
		3	Stack Pointer	1	Irrelevant Data
TXS	3	1	Opcode Address	1	Opcode
		2	Opcode Address + 1	1	Opcode of Next Instruction
		3	Address Bus FFFF	1	Low Byte of Restart Vector
PULA PULB	4	1	Opcode Address	1	Opcode
		2	Opcode Address + 1	1	Opcode of Next Instruction
		3	Stack Pointer	1	Irrelevant Data
		4	Stack Pointer + 1	1	Operand Data from Stack
PSHX	4	1	Opcode Address	1	Opcode
		2	Opcode Address + 1	1	Irrelevant Data
		3	Stack Pointer	0	Index Register (Low Order Byte)
		4	Stack Pointer – 1	0	Index Register (High Order Byte)
PULX	5	1	Opcode Address	1	Opcode
		2	Opcode Address + 1	1	Irrelevant Data
		3	Stack Pointer	1	Irrelevant Data
		4	Stack Pointer + 1	1	Index Register (High Order Byte)
		5	Stack Pointer + 2	1	Index Register (Low Order Byte)
RTS	5	1	Opcode Address	1	Opcode
		2	Opcode Address + 1	1	Irrelevant Data
		3	Stack Pointer	1	Irrelevant Data
		4	Stack Pointer + 1	1	Address of Next Instruction (High Order Byte)
		5	Stack Pointer + 2	1	Address of Next Instruction (Low Order Byte)
WAI	9	1	Opcode Address	1	Opcode
		2	Opcode Address + 1	1	Opcode of Next Instruction
		3	Stack Pointer	0	Return Address (Low Order Byte)
		4	Stack Pointer – 1	0	Return Address (High Order Byte)
		5	Stack Pointer – 2	0	Index Register (Low Order Byte)
		6	Stack Pointer – 3	0	Index Register (High Order Byte)
		7	Stack Pointer – 4	0	Contents of Accumulator A
		8	Stack Pointer – 5	0	Contents of Accumulator B
		9	Stack Pointer – 6	0	Contents of Condition Code Register

TABLE 14 – CYCLE-BY-CYCLE OPERATION (Sheet 5 of 5)

Address Mode and Instructions	Cycles	Cycle #	Address Bus	R/$\overline{W}$ Line	Data Bus
INHERENT					
MUL	10	1	Opcode Address	1	Opcode
		2	Opcode Address + 1	1	Irrelevant Data
		3	Address Bus FFFF	1	Low Byte of Restart Vector
		4	Address Bus FFFF	1	Low Byte of Restart Vector
		5	Address Bus FFFF	1	Low Byte of Restart Vector
		6	Address Bus FFFF	1	Low Byte of Restart Vector
		7	Address Bus FFFF	1	Low Byte of Restart Vector
		8	Address Bus FFFF	1	Low Byte of Restart Vector
		9	Address Bus FFFF	1	Low Byte of Restart Vector
		10	Address Bus FFFF	1	Low Byte of Restart Vector
RTI	10	1	Opcode Address	1	Opcode
		2	Opcode Address + 1	1	Irrelevant Data
		3	Stack Pointer	1	Irrelevant Data
		4	Stack Pointer + 1	1	Contents of Condition Code Register from Stack
		5	Stack Pointer + 2	1	Contents of Accumulator B from Stack
		6	Stack Pointer + 3	1	Contents of Accumulator A from Stack
		7	Stack Pointer + 4	1	Index Register from Stack (High Order Byte)
		8	Stack Pointer + 5	1	Index Register from Stack (Low Order Byte)
		9	Stack Pointer + 6	1	Next Instruction Address from Stack (High Order Byte)
		10	Stack Pointer + 7	1	Next Instruction Address from Stack (Low Order Byte)
SWI	12	1	Opcode Address	1	Opcode
		2	Opcode Address + 1	1	Irrelevant Data
		3	Stack Pointer	0	Return Address (Low Order Byte)
		4	Stack Pointer – 1	0	Return Address (High Order Byte)
		5	Stack Pointer – 2	0	Index Register (Low Order Byte)
		6	Stack Pointer – 3	0	Index Register (High Order Byte)
		7	Stack Pointer – 4	0	Contents of Accumulator A
		8	Stack Pointer – 5	0	Contents of Accumulator B
		9	Stack Pointer – 6	0	Contents of Condition Code Register
		10	Stack Pointer – 7	1	Irrelevant Data
		11	Vector Address FFFA (Hex)	1	Address of Subroutine (High Order Byte)
		12	Vector Address FFFB (Hex)	1	Address of Subroutine (Low Order Byte)
RELATIVE					
BCC BHT BNE BLO	3	1	Opcode Address	1	Opcode
BCS BLE BPL BHS		2	Opcode Address + 1	1	Branch Offset
BEQ BLS BRA BRN		3	Address Buss FFFF	1	Low Byte of Restart Vector
BGE BLT BVC					
BGT BMI BVS					
BSR	6	1	Opcode Address	1	Opcode
		2	Opcode Address + 1	1	Branch Offset
		3	Address Bus FFFF	1	Low Byte of Restart Vector
		4	Subroutine Starting Address	1	Opcode of Next Instruction
		5	Stack Pointer	0	Return Address (Low Order Byte)
		6	Stack Pointer – 1	0	Return Address (High Order Byte)

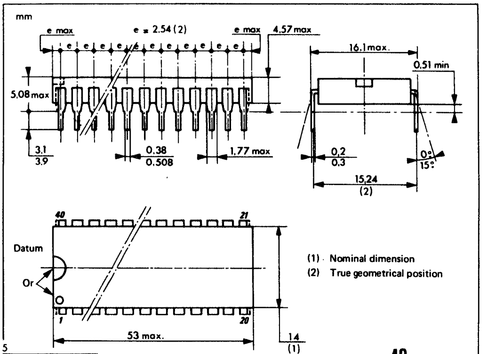
FIGURE 24 — SPECIAL OPERATIONS



Legend:

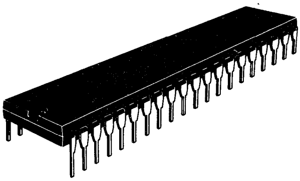
RTN = Address of next instruction in Main Program to be executed upon return from subroutine
RTN$_H$ = Most significant byte of Return Address
RTN$_L$ = Least significant byte of Return Address
→ = Stack Pointer After Execution
K = 8-bit Unsigned Value

PHYSICAL DIMENSIONS



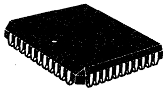
DIN				
A51F CEI	D.A.T.A.	JEDEC	F-119 SITELESC	CB-182

CB-182

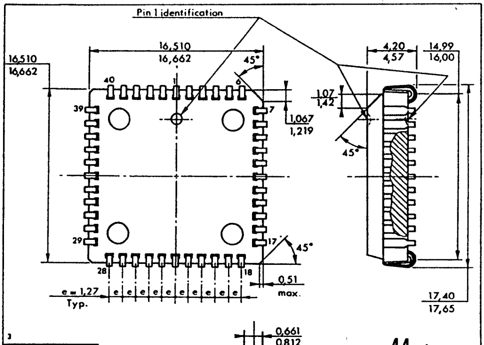


P SUFFIX
PLASTIC PACKAGE

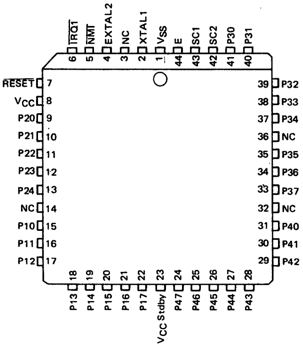
CB-521



FN SUFFIX
PLCC 44



DIN				
CEI	D.A.T.A.	MO-047-AC JEDEC	SITELESC	CB-521



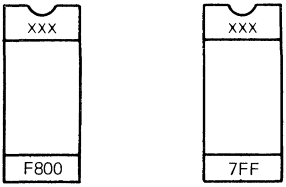
ORDERING INFORMATION

The information required when ordering a custom MCU is listed below. The ROM program may be transmitted to THOMSON SEMICONDUCTEURS on EPROM(s) or an EFDOS/MDOS* disk file.

To initiate a ROM pattern for the MCU, it is necessary to first contact your local THOMSON SEMICONDUCTEURS representative or distributor.

EPROMs

Two ET2716 or one ET2732 type EPROMs, programmed with the customer program (positive logic sense for address and data), may be submitted for pattern generation. The EPROM must be clearly marked to indicate which EPROM corresponds to which address space. The recommended marking procedure is illustrated below :



XXX = Customer ID

After the EPROM(s) are marked, they should be placed in conductive IC carriers and securely packed. Do not use styrofoam.

VERIFICATION MEDIA

All original pattern media (EPROMs or floppy disk) are filed for contractual purposes and are not returned. A computer listing of the ROM code will be generated and returned along with a listing verification form. The listing should be thoroughly checked and the verification form completed, signed, and returned to THOMSON SEMICONDUCTEURS. The signed verification form constitutes the contractual agreement for creation of the customer mask. If desired, THOMSON SEMICONDUCTEURS will program on blank EPROM from the data file used to create the custom mask and aid in the verification process.

ROM VERIFICATION UNITS (RVUs)

Ten MCUs containing the customer's ROM pattern will be sent for program verification. These units will have been made using the custom mask but are for the purpose of ROM verification only. For expediency they are usually unmarked, packaged in ceramic, and tested only at room temperature and 5 volts. These RVUs are included in the mask charge and are not production parts. The RVUs are thus not guaranteed by THOMSON SEMICONDUCTEURS. Quality Assurance, and should be discarded after verification is completed.

FLEXIBLE DISKS

The disk media submitted must be single-sided, EFDOS/MDOS* compatible floppies.

The customer must write the binary file name and company name on the disk with a felt-tip-pen. The minimum EFDOS/MDOS* system files, as well as the absolute binary object file (Filename .LO type of file) from the 6801 cross assembler, must be on the disk. An object file made from a memory dump using the ROLLOUT command is also acceptable. Consider submitting a source listing as well as the following files : filename .LX (DEVICE/EXORciser loadable format) and filename .SA (ASCII Source Code). These files will of course be kept confidential and are used 1) to speed up the process in-house if any problems arise, and 2) to speed up the user-to-factory interface if the user finds any software errors and needs assistance quickly from THOMSON SEMICONDUCTEURS factory representatives.

EFDOS is THOMSON SEMICONDUCTEURS' Disk Operating System available on development systems such as DEVICE,...

MDOS* is MOTOROLA's Disk Operating System available on development systems such as EXORciser,...

*Requires prior factory approval.

Whenever ordering a custom MCU is required, please contact your local THOMSON SEMICONDUCTEURS representative or THOMSON SEMICONDUCTEURS distributor and/or complete and send the attached "MCU customer ordering sheet" to your local THOMSON SEMICONDUCTEURS representative.

EXORciser is a registered trade mark of MOTOROLA Inc.

THOMSON
COMPONENTS
SEMICONDUCTEURS

EF6804J2

8 BIT MICROCOMPUTER

ADVANCE INFORMATION

The EF6804J2 microcomputer unit (MCU) is a member of the EF6804 Family of very low-cost single-chip microcomputers. This 8-bit microcomputer contains a CPU, on-chip CLOCK, ROM, RAM, I/O, and TIMER. It is designed for the user who needs an economical microcomputer with the proven capabilities of the EF6800-based instruction set.

The following are some of the hardware and software features of the EF6804J2 MCU.

HARDWARE FEATURES

- 5-Volt Single Supply
- 32 Bytes of RAM
- Memory Mapped I/O
- 1012 Bytes of Program ROM
- 64 Bytes of Data ROM
- 12 Bidirectional I/O Lines (Eight Lines with High Current Sink Capability)
- On-Chip Clock Generator
- Self-Test Mode
- Master Reset
- Complete Development System Support on DEVICE®
- Software Programmable 8-Bit Timer Control Register and Timer Prescaler (7 Bits, 2^n)
- Timer Pin is Programmable as Input or Output
- On-Chip Circuit for ROM Verify.

SOFTWARE FEATURES

- Similar to EF6805 HMOS Family
- Byte Efficient Instruction Set
- Easy to Program
- True Bit Manipulation
- Bit Test and Branch Instruction
- Separate Flags for Interrupt and Normal Processing
- Versatile Indirect Registers
- Conditional Branches
- Single Instruction Memory Examine/Change
- True LIFO Stack Eliminates Stack Pointer
- Nine Powerful Addressing Modes
- Any Bit in Data Space Memory May be Tested
- Any Bit in Data Space Memory Capable of Being Written to May be Set or Cleared

USER SELECTABLE OPTIONS

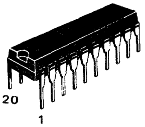
- 12 Bidirectional I/O Lines with LSTTL, LSTTL/CMOS, or Open-Drain Interface
- Crystal or Low-Cost Resistor-Capacitor Oscillator
- Mask Selectable Edge- or Level- Sensitive Interrupt Pin.

DEVICE® is THOMSON SEMICONDUCTEURS' development/emulation tool.

HMOS

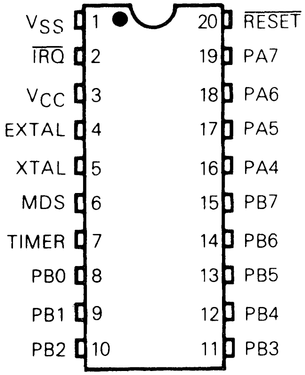
CASE

CB-194

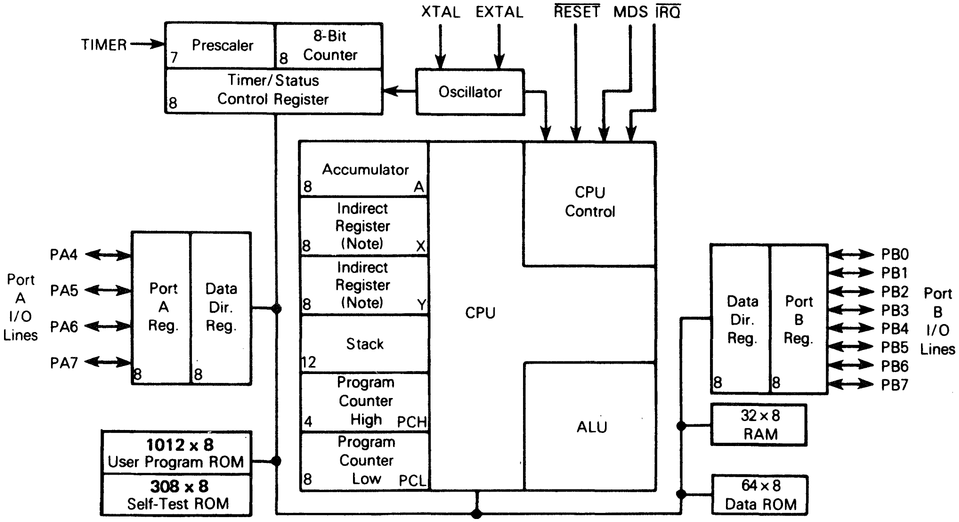


P SUFFIX
PLASTIC PACKAGE

PIN ASSIGNMENT



BLOCK DIAGRAM



NOTE: 8-Bit indirect registers X and Y, although shown as part of the CPU, are actually located in the 32 × 8 RAM at locations $80 and $81.

SECTION 2
FUNCTIONAL PIN DESCRIPTION, MEMORY, CPU, AND REGISTERS

This section provides a description of the functional pins, memory spaces, the central processing unit (CPU), and the various registers and flags.

2.1 FUNCTIONAL PIN DESCRIPTION

2.1.1 V_{CC} and V_{SS}

Power is supplied to the MCU using these two pins. V_{CC} is power and V_{SS} is the ground connection.

2.1.2 $\overline{\text{IRQ}}$

This pin provides the capability for asynchronously applying an external interrupt to the MCU. Refer to **4.1 INTERRUPT** for additional information.

2.1.3 XTAL and EXTAL

These pins provide connections to the on-chip clock oscillator circuit. A crystal, a resistor and capacitor, or an external signal, depending on the user selectable manufacturing mask option, can be connected to these pins to provide a system clock source with various stability/cost tradeoffs Lead lengths and stray capacitance on these two pins should be minimized. Refer to **4.4 INTERNAL CLOCK GENERATOR OPTIONS** for recommendations concerning these inputs.

2.1.4 TIMER

In the input mode, the timer pin is connected to the prescaler input and serves as the timer clock. In the output mode, the timer pin signals that a time out of the timer has occurred. Refer to **SECTION 3 TIMER** for additional information.

2.1.5 $\overline{\text{RESET}}$

The $\overline{\text{RESET}}$ pin is used to restart the processor of the EF6804J2 to the beginning of a program. This pin, together with the MDS pin, is also used to select the operating mode of the EF6804J2. If the MDS pin is at zero volts, the normal mode is selected and the program counter is loaded with the user restart vector. However, if the MDS pin is at +5 volts, then pins PA6 and PA7 are decoded to allow selection of the operating mode. Refer to **4.3 RESET** for additional information.

2.1.6 MDS

The MDS (mode select) pin is used to place the MCU into special operating modes. If MDS is held at +5 volts at the exit of the reset state, the decoded state of PA6 and PA7 is latched to determine the operating mode (single-chip, self-test, or ROM verify). However, if MDS is held at zero volts at the exit of the reset state, the single-chip operating mode is automatically selected (regardless of PA6 and PA7 state).

For those users familiar with the EF6801 microcomputer, mode selection is similar but much less complex in the EF6804J2. No special external diodes, switches, transistors, etc. are required in the EF6804J2.

2.1.7 Input/Output Lines (PA4-PA7, PB0-PB7)

These 12 lines are arranged into one 4-bit port (A) and one 8-bit port (B). All lines are programmable as either inputs or outputs under software control of the data direction registers. Refer to **SECTION 5 INPUT/OUTPUT PORTS** for additional information.

2.2 MEMORY

The MCU operates in three different memory spaces: program space, data space, and stack space. A representation of these memory spaces is shown in Figure 2-1. The program space (Figure 2-1a) contains all of the instructions that are to be executed, as well as the data required for the immediate addressing mode instructions, and the self-test and user vectors. The data space (Figure 2-1b) contains all of the RAM locations, plus I/O locations and some ROM used for storage of tables and constants. The stack space (Figure 2-1c) contains RAM which is used for stacking subroutine and interrupt return addresses.

The MCU is capable of addressing 4096 bytes of program space memory with its program counter and 256 bytes of data space memory with its instructions. The data space memory contains three bytes for port data registers, three bytes for port data direction registers, one byte for timer status/control, 64 bytes ROM, 32 bytes RAM (which includes two bytes for X and Y indirect registers), two bytes for timer prescaler and count registers, and one byte for the accumulator. The program space section includes 304 bytes of self-test ROM, 1008 bytes program ROM, and eight bytes of vectors for self-test and user programs.

2.3 CENTRAL PROCESSING UNIT

The CPU of the EF6804 Family is implemented independently from the I/O or memory configuration. Consequently, it can be treated as an independent central processor communicating with I/O and memory via internal addresses, data, and control buses.

(a) Program Space Memory Map

Reserved (All Ones)	$000 $ADF
Self-Test ROM	$AE0 $C0F
Program ROM	$C10 $FF7
Self-Test $\overline{IRQ}$ Vector	$FF8-$FF9
Self-Test Restart Vector	$FFA-$FFB
User $\overline{IRQ}$ Vector	$FFC-$FFD
User Restart Vector	$FFE-$FFF

(b) Data Space Memory Map

Port A Data Register				0	0	0	0	$00
Port B Data Register								$01
1	1	1	1	0	0	0	0	$02
Not Used								$03
Port A DDR				0	0	0	0	$04
Port B Data Direction Register								$05
1	1	1	1	0	0	0	0	$06
Not Used								$07 $08
Timer Status Control Register								$09
Future Expansion								$0A $1F
User Data Space ROM								$20 $5F
Future Expansion								$60 $7F
Indirect Register X								$80
Indirect Register Y								$81
Data Space RAM								$82 $9F
Future Expansion								$A0 $FC
Prescaler Register								$FD
Timer Count Register								$FE
Accumulator								$FF

(c) Stack Space Memory Map

Level 1
Level 2
Level 3
Level 4

Figure 2-1. EF6804J2 MCU Address Map

2.4 REGISTERS

The EF6804 Family CPU has four registers and two flags available to the programmer. They are shown in Figure 2-2 and are explained in the following paragraphs.

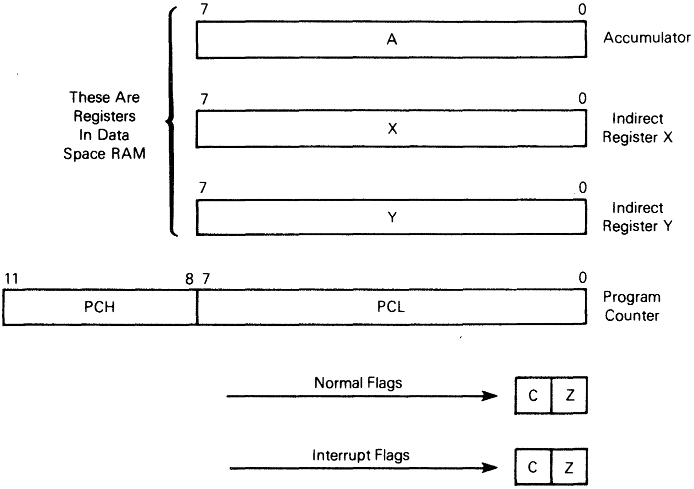


Figure 2-2. Programming Model

2.4.1 Accumulator (A)

The accumulator is an 8-bit general purpose register used in all arithmetic calculations, logical operations, and data manipulations. The accumulator is implemented as the highest RAM location ($FF) in data space and thus implies that several instructions exist which are not explicitly implemented. Refer to **6.3 IMPLIED INSTRUCTIONS** for additional information.

2.4.2 Indirect Registers (X, Y)

These two indirect registers are used to maintain pointers to other memory locations in data space. They are used in the register-indirect addressing mode, and can be accessed with the direct, indirect, short direct, or bit set/clear addressing modes. These registers are implemented as two of the 32 RAM locations ($80, $81) and as such generate implied instructions and may be manipulated in a manner similar to any RAM memory location in data space. Refer to **6.3 IMPLIED INSTRUCTIONS** for additional information.

2.4.3 Program Counter (PC)

The program counter is a 12-bit register that contains the address of the next ROM word to be used (may be opcode, operand, or address of operand). The 12-bit program counter is contained in PCL (low byte) and PCH (high nibble).

2.4.4 Flags (C, Z)

The carry (C) bit is set on a carry or a borrow out of the ALU. It is cleared if the result of an arithmetic operation does not result in a carry or a borrow. The (C) bit is also set to the value of the bit tested in a bit test instruction, and participates in the rotate left instruction.

The zero (Z) bit is set if the result of the last arithmetic or logical operation was equal to zero, otherwise it is cleared.

There are two sets of these flags, one set is for interrupt processing, the other for all other routines. When an interrupt occurs, a context switch is made from the program flags to the interrupt flags (interrupt mode). An RTI forces the context switch back to the program flags (program mode). While in either mode, only the flags for that mode are available. Further, the interrupt flags will not be cleared upon entering the interrupt mode. Instead, the flags will be as they were at the exit of the last interrupt mode. Both sets of flags are cleared by reset.

2.4.5 Stack

There is a true LIFO stack incorporated in the EF6804J2 which eliminates the need for a stack pointer. Stack space is implemented in separate RAM (12-bits wide) shown in Figure 2-1c. Whenever a subroutine call (or interrupt) occurs, the contents of the PC are shifted into the top register of the stack. At the same time (same cycle), the top register is shifted to the next level deeper. This happens to all registers with the bottom register falling out the bottom of the stack.

Whenever a subroutine or interrupt return occurs, the top register is shifted into the PC and all lower registers are shifted up one level higher. The stack RAM is four levels deep. If the stack is pulled more than four times without any pushes, the address that was stored in the bottom level will be shifted into the PC.

SECTION 3
TIMER

3.1 INTRODUCTION

A block diagram of the EF6804J2 timer circuitry is shown in Figure 3-1. The timer logic in the MCU is comprised of a simple 8-bit counter (timer count register, TCR) with a 7-bit prescaler, and a timer status/control register (TSCR). The timer count register, which may be loaded under program control, is decremented towards zero by a clock input (prescaler output). The prescaler is used to extend the maximum interval of the overall timer. The prescaler tap is selected by bits 0-2 (PS0-PS2) of the timer status/control register. Bits PS0-PS2 control the actual division of the prescaler within the range of divide-by-1 (2^0) to divide-by-128 (2^7). The timer count register (TCR) and prescaler are decremented on rising clock edges. The coding of the TCSR PS0-PS2 bits produce a division in the prescaler as shown in Table 3-1.

Table 3-1. Prescaler Coding Table

PS2	PS1	PS0	Divide By
0	0	0	1
0	0	1	2
0	1	0	4
0	1	1	8

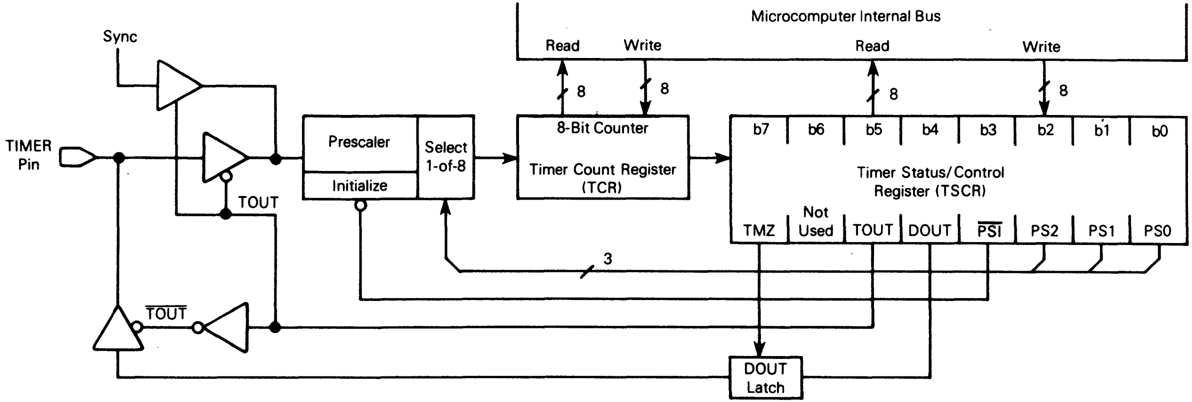
PS2	PS1	PS0	Divide By
1	0	0	16
1	0	1	32
1	1	0	64
1	1	1	128

The TIMER pin may be programmed as either an input or an output depending on the status of TOUT (TSCR bit 5). Refer to Figure 3-1. In the input mode, TOUT is a logic zero and the TIMER pin is connected directly to the prescaler input. Therefore, the timer prescaler is clocked by the signal applied from the TIMER pin. The prescaler then divides its clock input by a value determined by the coding of the TSCR bits PS0-PS2 as shown in Table 3-1. The divided prescaler output then clocks the 8-bit timer count register (TCR). When the TCR is decremented to zero, it sets the TMZ bit in the timer status/control register (TSCR). The TMZ bit can be tested under program control to perform a timer function whenever it goes high. The frequency of the external clock applied to the TIMER pin must be less than t_{byte} ($f_{osc}/48$).

In the output mode, TOUT is a logic one and the TIMER pin is connected to the DOUT latch. Therefore, the timer prescaler is clocked by the internal sync pulse (divide-by-48 of the internal oscillator). Operation is similar to that described above for the input mode. However, in the output mode, the low-to-high TMZ bit transition is used to latch the DOUT bit of the TSCR and provide it as output for the TIMER pin.

NOTE

TMZ is normally set to logic one when the timer times out (TCR decrements to $00); however, it may be set by a write of $00 to the TCR or by a write to bit 7 of the TSCR.



TIMER Pin Status

TOUT	Prescaler Clock	TIMER Pin
0	TIMER Pin	Input Mode
1	Sync	Output Mode

Figure 3-1. Timer Block Diagram

During reset, the timer count register and prescaler are set to $FF, while the timer status/control register is cleared to $00 and the DOUT LATCH (TIMER pin is in the high-impedance input mode) is forced to a logic high. The prescaler and timer count register are implemented in data space RAM locations ($FD, $FE); therefore, they are both readable and writeable. A write to either will predominate over the TCR decrement-to-$00 function; i.e., if a write and a TCR decrement-to-$00 occur simultaneously, the write will take precedence, and the TMZ bit is not set until the next timer time out.

3.2 TIMER REGISTERS

3.2.1 Timer Count Register (TCR)

7	0
MSB	LSB

TCR addr = $FE

The timer count register indicates the state of the internal 8-bit counter.

3.2.2 Timer Status/Control Register (TSCR)

7	6	5	4	3	2	1	0
TMZ	Not Used	TOUT	DOUT	$\overline{\text{PSI}}$	PS2	PS1	PS0

TSCR Address = $09

b7, TMZ Low-to-high transition indicates the timer count register has decremented to zero since the timer status/control register was last read. Cleared by a read of TSCR register if TMZ was read as a logic one.

b6 Not used.

b5, TOUT When low, this bit selects the input mode for the timer. When high, the output mode is selected.

b4, DOUT Data sent to the timer output pin when TMZ is set high (output mode only).

b3, $\overline{\text{PSI}}$ Used to initialize the prescaler and inhibit its counting while $\overline{\text{PSI}}$ = 0. The initialized value is set to $FF. The timer count register will also be inhibited (contents unchanged). When PSI = 1 the prescaler begins to count downward.

b0, b1, b2 PS0-PS1-PS2 These bits are used to select the prescaler divide-by ratio; therefore, effecting the clock input frequency to the timer count register.

3.2.3 Timer Prescaler Register

6	0
MSB	LSB

TPR Address = $FD

The timer prescaler register indicates the state of the internal 7-bit prescaler. This 7-bit prescaler divide ratio is normally determined by bits PS0-PS2 of the timer status/control register (see Table 3-1).

SECTION 4
INTERRUPT, SELF-TEST, RESET, AND INTERNAL CLOCK GENERATOR

4.1 INTERRUPT

The EF6804J2 can be interrupted by applying a logic low signal to the $\overline{\text{IRQ}}$ pin ; however, a mask option selected at the time of manufacture determines whether the negative-going edge or the actual low level is sensed to indicate an interrupt.

4.1.1 Edge-Sensitive Option

When the $\overline{\text{IRQ}}$ pin is pulled low, the internal interrupt request latch is set. Prior to each instruction fetch, the interrupt request latch is tested and, if its output is high, an interrupt sequence is initiated at the end of the current instruction (provided the interrupt mask is cleared). Figure 4-1 contains a flowchart which illustrates both the reset and interrupt sequence. The interrupt sequence consists of one cycle during which: the interrupt request latch is cleared, the interrupt mode flags are selected, the PC is saved on the stack, the interrupt mask is set, and the $\overline{\text{IRQ}}$ vector (single chip mode= $FFC/$FFD, self-test mode= $FF8/$FF9) is loaded into the PC. Internal processing of the interrupt continues until an RTI (return from interrupt) instruction is processed. During the RTI instruction, the interrupt mask is cleared and the program mode flags are selected.The next instruction of the program is then fetched and executed. Once the interrupt was initially detected and the interrupt sequence started, the interrupt request latch is cleared so that the next (second) interrupt may be detected even while the previous (first) one is being serviced. However, even though the second interrupt sets the interrupt request latch during processing of the first interrupt, the second interrupt sequence will not be initiated until completion of the interrupt service routine for the first interrupt. Completion of an interrupt service routine is always accomplished using an RTI instruction to return to the main program. The interrupt mask (which is not directly available to the programmer) is cleared during the last cycle of the RTI instruction.

4.1.2 Level-Sensitive Option

The actual operation of the level-sensitive and edge-sensitive options are similar except that the level-sensitive option does not have an interrupt request latch. With no interrupt request latch, the logic level of the $\overline{\text{IRQ}}$ pin is checked for detection of the interrupt. Also, in the interrupt sequence, there is no need to clear the interrupt request latch. These differences are illustrated in the flowchart of Figure 4-1.

4.1.3 Power Up and Timing

During the power-up sequence the interrupt mask is set to preclude any false or "ghost" interrupts from occurring. To clear the interrupt mask, the programmer should write a JSR (instead of a JMP) instruction to an initialization routine as the first instruction in a program. The initialization routine

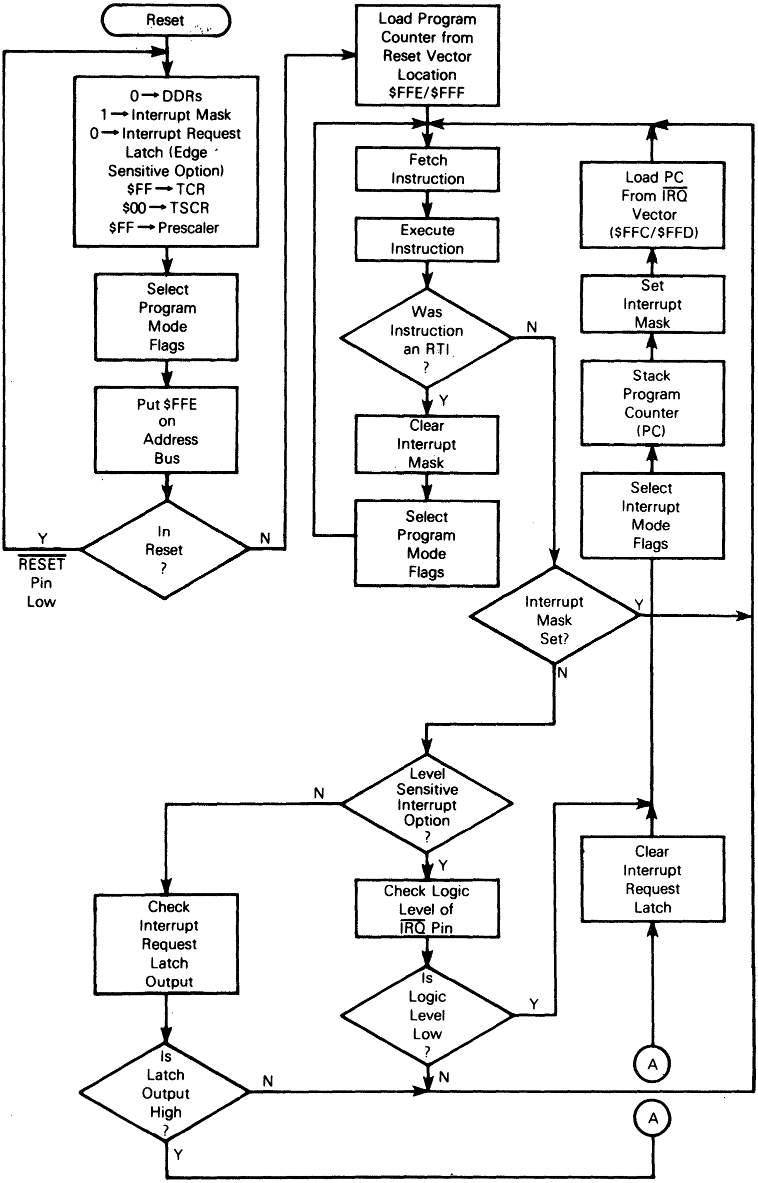

Reset
0 → DDRs
1 → Interrupt Mask
0 → Interrupt Request Latch (Edge Sensitive Option)
$FF → TCR
$00 → TSCR
$FF → Prescaler
Select Program Mode Flags
Put $FFE on Address Bus
In Reset ?
Y
RESET Pin Low
N
Load Program Counter from Reset Vector Location $FFE/$FFF
Fetch Instruction
Execute Instruction
Was Instruction an RTI ?
N
Y
Clear Interrupt Mask
Select Program Mode Flags
Load PC From IRQ Vector ($FFC/$FFD)
Set Interrupt Mask
Stack Program Counter (PC)
Select Interrupt Mode Flags
Interrupt Mask Set?
Y
N
Level Sensitive Interrupt Option ?
N
Y
Check Interrupt Request Latch Output
Check Logic Level of IRQ Pin
Clear Interrupt Request Latch
Is Logic Level Low ?
Y
N
A
Is Latch Output High ?
N
Y
A

Figure 4-1. Reset and Interrupt Processing Flowchart

should end with an RTI (instead of RTS). Maximum interrupt response time is eight machine (t_{byte}) cycles (see **4.4 INTERNAL CLOCK GENERATOR OPTIONS**). This includes five machine cycles for the longest instruction, plus one machine cycle for stacking the PC and switching flags, plus two machine cycles for synchronization of the $\overline{IRQ}$ input with the internal clock. Minimum response time is one machine cycle for stacking PC and switching flags (see **2.4.4 Flags (C, Z)**).

4.2 SELF-TEST

The EF6804J2 MCU has a unique internal ROM-based off-line self-test capability using signature analysis techniques. A test program stored in the on-chip ROM is initiated by configuring pins PA6 and PA7 during reset. The test results are sampled on a cycle-by-cycle basis by a 16-bit on-chip signature analysis register configured as a linear feedback shift register (LFSR) using the standard CCITT CRC16 polynomial. A schematic diagram of the self-test connections is shown in Figure 4-2. To perform a test of the MCU, connect it as shown in Figure 4-2a and monitor the LEDs for a 1101 ($D) pattern.

A special ROM self-test utilizing the signature analysis circuitry is also included. To initiate a test of the ROM, connect the circuit as shown in Figure 4-2b. This mode also uses the on-chip signature analysis register to verify the contents of the custom ROM by monitoring an internal bus. The "Good" LED indicates that all ROM words have been read and that the result was the correct signature.

The on-chip self-test and the ROM test are the basis of THOMSON SEMICONDUCTEURS production testing for the EF6804J2. These tests have been fault graded using statistical methods and have been found to provide high fault coverage using automatic test equipment (ATE) or the circuit of Figure 4-2.

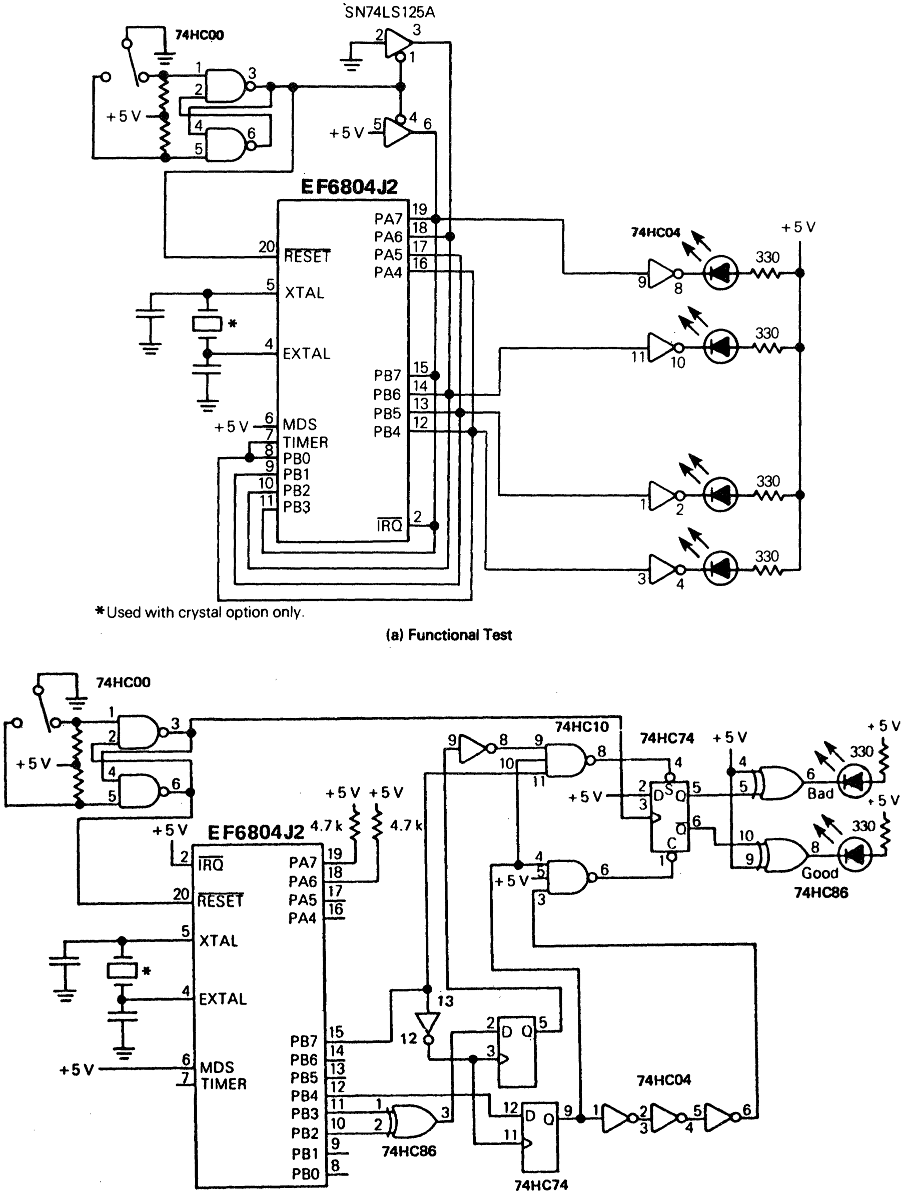
4.3 RESET

The MCU can be reset in two ways: by initial power up (see Figure 4-1) and by the external reset input ($\overline{RESET}$). During power up, a delay of t_{RHL} is needed before allowing the $\overline{RESET}$ input to go high. This time delay allows the internal clock generator to stabilize. Connecting a capacitor and resistor to the $\overline{RESET}$ input, as shown in Figure 4-3, typically provides sufficient delay.

4.4 INTERNAL CLOCK GENERATOR OPTIONS

The internal clock generator circuit is designed to require a minimum of external components. A crystal, a resistor-capacitor, or an external signal may be used to generate a system clock with various stability/cost tradeoffs. A manufacturing mask option is required to select either the crystal oscillator or the RC oscillator circuit. The different clock generator option connection methods are shown in Figure 4-4, crystal specifications and suggested PC board layouts are given in Figure 4-5, resistor-capacitor selection graph is given in Figure 4-6, and a timing diagram is illustrated in Figure 4-7. The crystal oscillator startup time is a function of many variables: crystal parameters (especially R_S), oscillator load capacitance (C_L), IC parameters, ambient temperature, and supply voltage. To ensure rapid oscillator startup, neither the crystal characteristics nor the load capacitance should exceed recommendations.

The oscillator output frequency is internally divided by four to produce the internal $\phi 1$ and $\phi 2$ clocks. The $\phi 1$ clock is divided by twelve to produce a machine byte (cycle) clock. A byte cycle is the smallest unit needed to execute any operation (i.e., increment the program counter). An instruction may need two, four, or five byte cycles to execute.



*Used with crystal option only.

(a) Functional Test

*Used with crystal option only.

(b) Simple ROM Verify Test

Figure 4-2. Self-Test Circuit

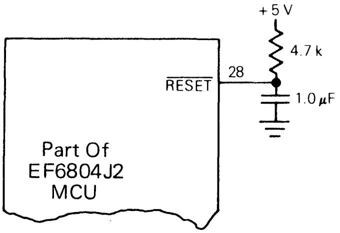


Figure 4-3. Power-Up Reset Delay Circuit

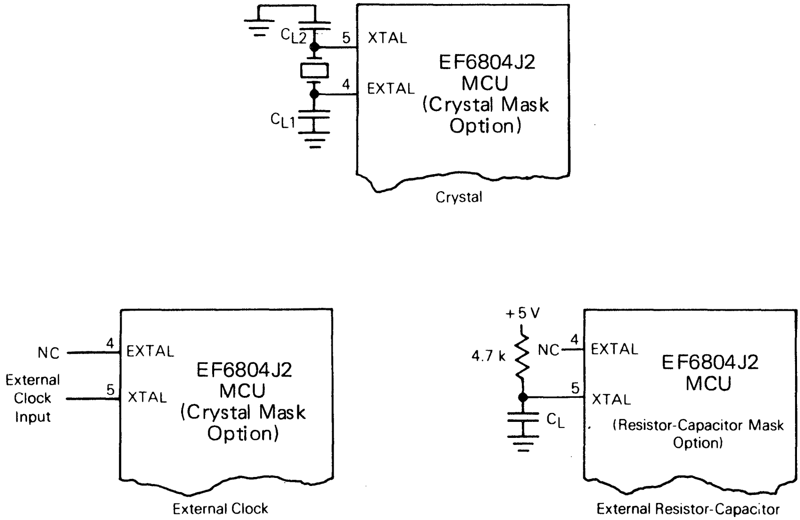
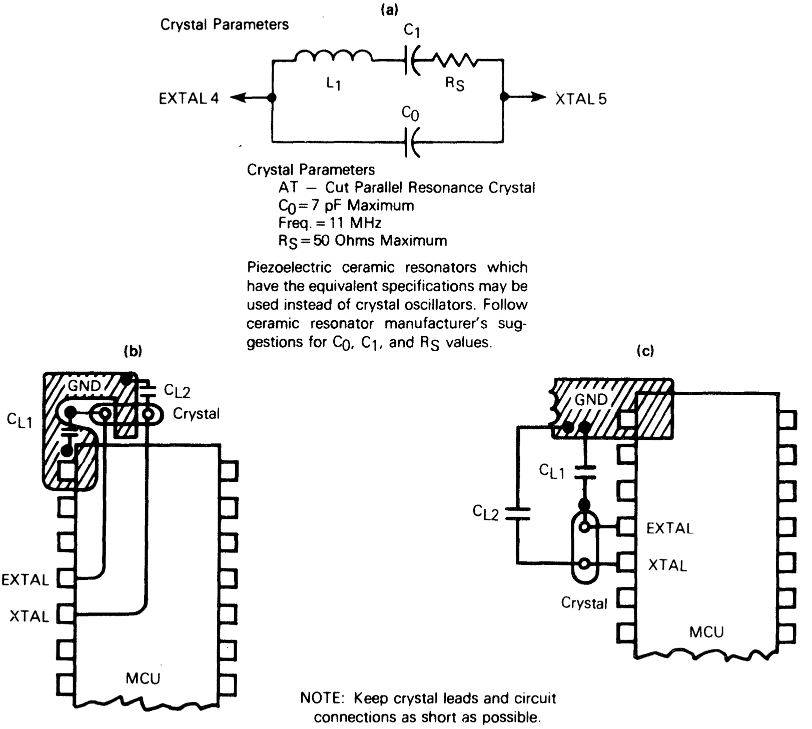


Figure 4-4. Clock Generator Options



Crystal Parameters
AT – Cut Parallel Resonance Crystal
$C_0 = 7$ pF Maximum
Freq. = 11 MHz
$R_S = 50$ Ohms Maximum

Piezoelectric ceramic resonators which have the equivalent specifications may be used instead of crystal oscillators. Follow ceramic resonator manufacturer's suggestions for C_0, C_1, and R_S values.

NOTE: Keep crystal leads and circuit connections as short as possible.

Figure 4-5. Crystal Motional Arm Parameters and Suggested PC Board Layout

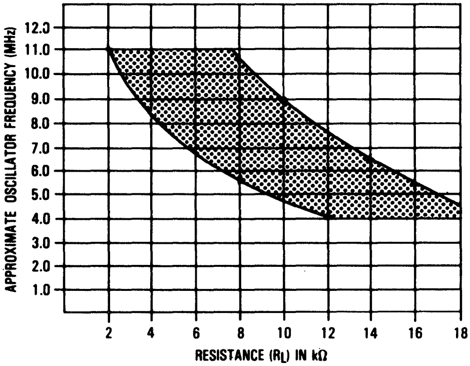
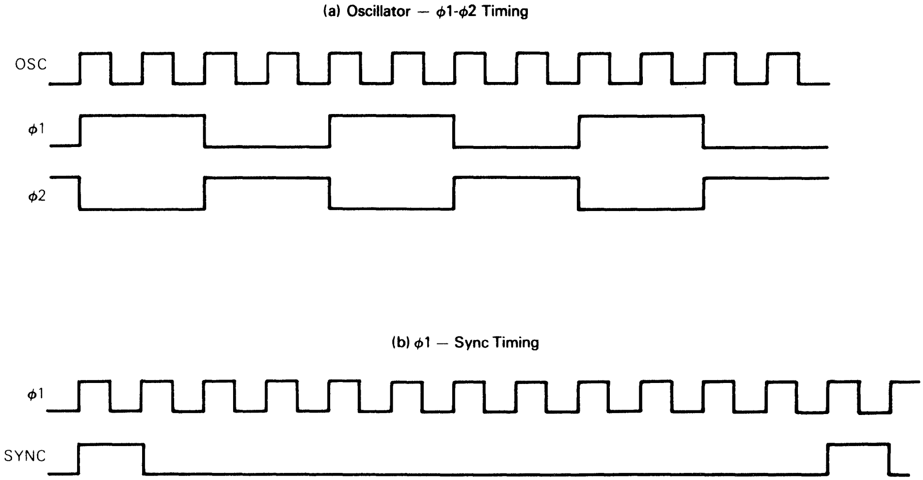


Figure 4-6. Typical Frequency Selection For Resistor-Capacitor Oscillator Option ($C_L = 17$ pF)



(a) Oscillator — ϕ1-ϕ2 Timing

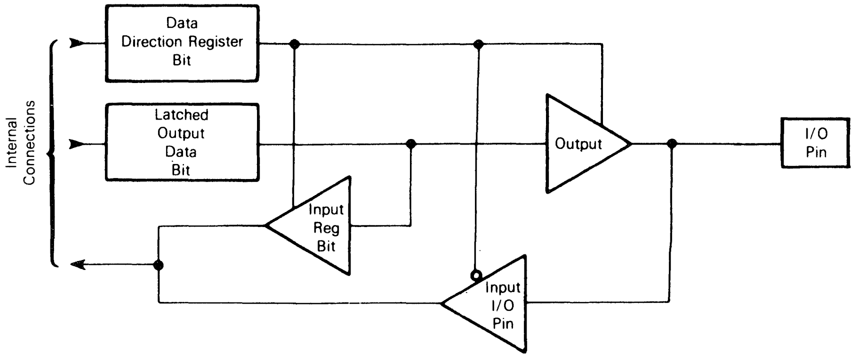
(b) ϕ1 — Sync Timing

Figure 4-7. Clock Generator Timing Diagram

SECTION 5 INPUT/OUTPUT PORTS

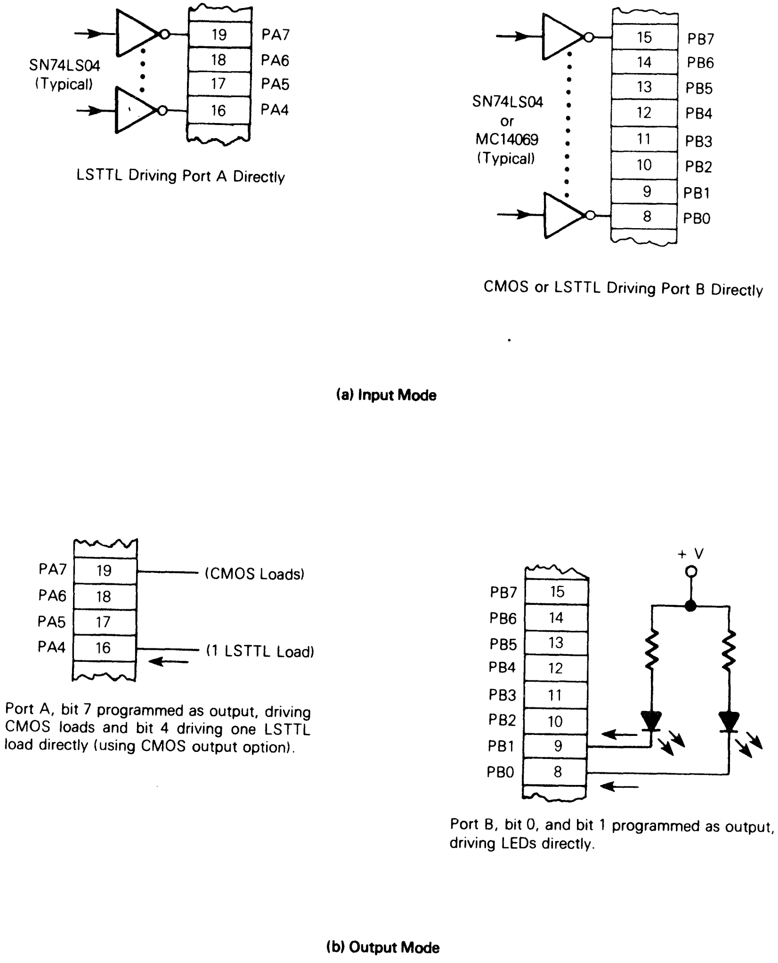
5.1 INPUT/OUTPUT

There are 12 input/output pins. All pins (port A and B) are programmable as either inputs or outputs under software control of the corresponding data direction register (DDR). The port I/O programming is accomplished by writing the corresponding bit in the port DDR to a logic one for output or a logic zero for input. On reset, all the DDRs are initialized to a logic zero state to put the ports in the input mode. The port output registers are not initialized on reset but should be initialized before changing the DDR bits to avoid undefined levels. When programmed as outputs, the latched output data is readable as input data, regardless of the logic levels at the output pin due to output loading; see Figure 5-1. All input/output pins are LSTTL compatible as both inputs and outputs. In addition, both ports may have one of two mask options: 1) internal pullup resistor for CMOS output compatibility, or 2) open drain output. The address map in Figure 2-1 gives the address of data registers and DDRs. The register configuration is discussed under the registers paragraph below and Figure 5-2 provides some examples of port connections.



Data Direction Register Bit	Output Data Bit	Output State	Input To MCU
1	0	0	0
1	1	1	1
0	X	Hi-Z	Pin

Figure 5-1. Typical I/O Port Circuitry



LSTTL Driving Port A Directly

CMOS or LSTTL Driving Port B Directly

(a) Input Mode

Port A, bit 7 programmed as output, driving CMOS loads and bit 4 driving one LSTTL load directly (using CMOS output option).

Port B, bit 0, and bit 1 programmed as output, driving LEDs directly.

(b) Output Mode

Figure 5-2. Typical Port Connections

The latched output data bit (see Figure 5-1) may always be written. Therefore, any write to a port writes to all of its data bits even though the port DDR is set to input. This may be used to initialize the data registers and avoid undefined outputs; however, care must be exercised when using read-modify-write instructions since the data read corresponds to the pin level if the DDR is an input (0) and corresponds to the latched output data when the DDR is an output (1). The 12 bidirectional lines may be configured by port to be LSTTL (standard configuration), LSTTL/CMOS (mask option), or open drain (mask option). Port B outputs are LED compatible.

NOTE

The mask option only allows changes by port. For example, if the customer wishes PA7 to be open drain, then PA4-PA7 must all be open drain.

5.2 REGISTERS

The registers described below are implemented as RAM locations and thus may be read or written.

5.2.1 Port Data Register

Port A Address = $00
Port B Address = $01

The source of data read from the port data register will be the port I/O pin or previously latched output data depending upon the contents of the corresponding data direction register (DDR). The destination of data written to the port data register will be an output data latch. If the corresponding data direction register (DDR) for the port I/O pin is programmed as an output, the data will then appear on the port pin.

5.2.2 Port Data Direction Register

Port A Address = $04
Port B Address = $05

The port DDRs configure the port pins as either inputs or outputs. Each port pin can be programmed individually to act as an input or an output. A zero in the pins corresponding bit position will program that pin as an input while a one in the pins corresponding bit position will program that pin as an output.

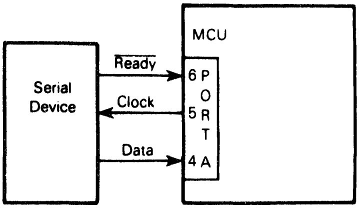
SECTION 6
SOFTWARE AND INSTRUCTION SET

6.1 SOFTWARE

6.1.1 Bit Manipulation

The EF6804J2 MCU has the ability to set or clear any register or single random access memory (RAM) writable bit with a single instruction (BSET, BCLR). Any bit in data space, including ROM, can be tested, using the BRSET and BRCLR instructions, and the program may branch as a result of its state. The carry bit is set to the value of the bit referenced by BRSET or BRCLR. A rotate instruction may then be used to accumulate serial input data in a RAM location or register. The capability to work with any bit in RAM, ROM, or I/O allows the user to have individual flags in RAM or to handle I/O bits as control lines.

The coding example in Figure 6-1 illustrates the usefulness of the bit manipulation and test instructions. Assume that the MCU is to communicate with an external serial device. The external device has a data ready signal, a data output line, and a clock line (to clock data one bit at a time, MSB first, out of the device). The MCU waits until the data is ready, clocks the external device, picks up the data in the carry flag (C bit), clears the clock line, and finally accumulates the data bit in the accumulator.



```
                    .
                    .
                    .
SELF    BRSET       6,PORTA,SELF
                    .
                    .
                    .
        BSET        5,PORTA
        BRCLR       4,PORTA,CONT
CONT    BCLR        5,PORTA
        ROLA        .
                    .
                    .
                    .
```

Figure 6-1. Bit Manipulation Example

6.1.2 Addressing Modes

The EF6804J2 MCU has nine addressing modes which are explained briefly in the following paragraphs. The EF6804J2 deals with objects in three different address spaces : program space, data space, and stack space. Program space contains the instructions which are to be executed, plus the data for immediate mode instructions. Data space contains all of the RAM locations, X and Y registers, accumulator, timer, I/O locations, and some ROM (for storage of tables and constants). Stack space contains RAM for use in stacking the return addresses for subroutines and interrupts.

The term "Effective Address" (EA) is used in describing the address modes. EA is defined as the address from which the argument for an instruction is fetched or stored.

6.1.2.1 IMMEDIATE. In the immediate addressing mode, the operand is located in program ROM and is contained in a byte following the opcode. The immediate addressing mode is used to access constants which do not change during program execution (e.g., a constant used to initialize a loop counter).

6.1.2.2 DIRECT. In the direct addressing mode, the effective address of the argument is contained in a single byte following the opcode byte. Direct addressing allows the user to directly address the 256 bytes in data space memory with a single two-byte instruction.

6.1.2.3 SHORT DIRECT. The MCU also has four locations in data space RAM ($80, $81, $82, $83) which may be used in a short-direct addressing mode. In this mode the lower two bits of the opcode determine the data space. RAM location, and the instruction is only one byte. Short direct addressing is a subset of the direct addressing mode. (The X and Y registers are at locations $80 or $81 respectively.)

6.1.2.4 EXTENDED. In the extended addressing mode, the effective address is obtained by concatenating the four least significant bits of the opcode with the byte following the opcode (12-bit address). Instructions using the extended addressing mode (JMP, JSR) are capable of branching anywhere in program space. An extended addressing mode instruction is two bytes long.

6.1.2.5 RELATIVE. The relative addressing mode is only used in conditional branch instructions. In relative addressing, the address is formed by adding the sign extended lower five bits of the opcode (the offset) to the program counter if and only if the condition is true. Otherwise, control proceeds to the next instruction. The span of relative addressing is from − 15 to + 16 from the opcode address. The programmer need not worry about calculating the correct offset when using the Motorola assembler since it calculates the proper offset and checks to see if it is within the span of the branch.

6.1.2.6 BIT SET/CLEAR. In the bit set/clear addressing mode, the bit to be set or cleared is part of the opcode, and the byte following the opcode specifies the direct address of the byte in which the specified bit is to be set or cleared. Thus, any bit in the 256 locations of data space memory, which can be written to, can be set or cleared.

6.1.2.7 BIT TEST AND BRANCH. The bit test and branch addressing mode is a combination of direct addressing and relative addressing. The bit which is to be tested is included in the opcode, and the data space address of the byte to be tested is in the single byte immediately following the opcode byte. The third byte is sign extended to twelve bits and becomes the offset added to the program counter if the condition is true. The single three-byte instruction allows the program to branch based on the condition of any bit in data space memory. The span of branching is from – 125 to + 130 from the opcode address. The state of the tested bit is also transferred to the carry flag.

6.1.2.8 REGISTER-INDIRECT. In the register-indirect addressing mode, the operand is at the address (in data space) pointed to by the contents of one of the indirect registers (X or Y). The particular X or Y register is selected by bit 4 of the opcode. Bit 4 of the opcode is then decoded into an address which selects the desired X or Y register ($80 or $81). A register-indirect instruction is one byte long.

6.1.2.9 INHERENT. In the inherent addressing mode, all the information necessary to execute the instruction is contained in the opcode. These instructions are one byte long.

6.2 INSTRUCTION SET

The EF6804J2 MCU has a set of 42 basic instructions, which when combined with nine addressing modes produce 242 usable opcodes. They can be divided into five different types: register/memory, read-modify-write, branch, bit manipulation, and control. The following paragraphs briefly explain each type. All the instructions within a given type are presented in individual tables.

6.2.1 Register/Memory Instructions

Most of these instructions use two operands. One operand is the accumulator and the other operand is obtained from memory using one of the addressing modes. The jump unconditional (JMP) and jump to subroutine (JSR) instructions have no register operands. Refer to Table 6-1.

6.2.2 Read-Modify-Write Instructions

These instructions read a memory location or a register, modify or test its contents, and write the modified value back to memory or to the register. There are ten instructions which utilize read-modify-write cycles. All INC and DEC forms along with all bit manipulation instructions use this method. Refer to Table 6-2.

6.2.3 Branch Instructions

The branch instructions cause a branch from the program when a certain condition is met. Refer to Table 6-3.

6.2.4 Bit Manipulation Instructions

These instructions are used on any bit in data space memory. One group either sets or clears. The other group performs the bit test branch operations. Refer to Table 6-4.

Table 6-1. Register/Memory Instructions

		Addressing Modes																			
		Indirect				Immediate			Direct			Inherent			Extended			Short-Direct			
Function	Mnem	Opcode		# Bytes	# Cycles	Opcode	# Bytes	# Cycles	Opcode	# Bytes	# Cycles	Opcode	# Bytes	# Cycles	Opcode	# Bytes	# Cycles	Opcode	# Bytes	# Cycles	Special Notes
		X	Y																		
Load A from Memory	LDA	E0	F0	1	4	E8	2	4	F8	2	4	—	—	—	—	—	—	AC-AF	1	4	1
Load XP from Memory	LDXI	—	—	—	—	B0	3	4	—	—	—	—	—	—	—	—	—	—	—	—	4
Load YP from Memory	LDYI	—	—	—	—	B0	3	4	—	—	—	—	—	—	—	—	—	—	—	—	4
Store A in Memory	STA	E1	F1	1	4	—	—	—	F9	2	4	—	—	—	—	—	—	BC-BF	1	4	2
Add to A	ADD	E2	F2	1	4	EA	2	4	FA	2	4	—	—	—	—	—	—	—	—	—	—
Subtract from A	SUB	E3	F3	1	4	EB	2	4	FB	2	4	—	—	—	—	—	—	—	—	—	—
Arithmetic Compare with Memory	CMP	E4	F4	1	4	EC	2	4	FC	2	4	—	—	—	—	—	—	—	—	—	—
AND Memory to A	AND	E5	F5	1	4	ED	2	4	FD	2	4	—	—	—	—	—	—	—	—	—	—
Jump to Subroutine	JSR	—	—	—	—	—	—	—	—	—	—	—	—	—	8 (TAR)	2	4	—	—	—	3
Jump Unconditional	JMP	—	—	—	—	—	—	—	—	—	—	—	—	—	9 (TAR)	2	4	—	—	—	3
Clear A	CLRA	—	—	—	—	—	—	—	FB	2	4	—	—	—	—	—	—	—	—	—	—
Clear XP	CLRX	—	—	—	—	—	—	—	FB	2	4	—	—	—	—	—	—	—	—	—	—
Clear YP	CLRY	—	—	—	—	—	—	—	FB	2	4	—			—	—	—	—	—	—	—
Complement A	COMA	—	—	—	—	—	—	—	—	—	—	B4	1	4	—	—	—	—	—	—	—
Move Immediate Value to Memory	MVI	—	—	—	—	B0	3	4	B0	3	4	—	—	—	—	—	—	—	—	—	5
Rotate A Left and Carry	ROLA	—	—	—	—	—	—	—	—	—	—	B5	1	4	—	—	—	—	—	—	—
Arithmetic Left Shift of A	ASLA	—	—	—	—	—	—	—	FA	2	4	—	—	—	—	—	—	—	—	—	—

SPECIAL NOTES

1. In Short-Direct addressing, the LDA mnemonic represents opcode AC, AD, AE, and AF. This is equivalent to RAM locations $80 (AC), $81 (AD), $82 (AE), and $83 (AF)
2. In Short-Direct addressing, the STA mnemonic represents opcode BC, BD, BE, and BF. This is equivalent to RAM locations $80 (BC), $81 (BD), $82 (BE), and $83 (BF).
3. In Extended addressing, the four LSBs of the opcode (Mnemonic JSR and JMP) are formed by the four MSBs of the target address. (TAR)
4. In Immediate addressing, the LDXI and LDYI are mnemonics which are recognized as follows:

   LDXI = MVI $80,data
   LDYI = MVI $81,data Where data is a one-byte hexadecimal number.

5. The MVI instruction refers to both Immediate and Direct addressing.

Table 6-2. Read-Modify-Write Instructions

		Addressing Modes										
		Indirect				Direct			Short-Direct			
		Opcode		#	#		#	#		#	#	Special
Function	Mnem	X	Y	Bytes	Cycles	Opcode	Bytes	Cycles	Opcode	Bytes	Cycles	Notes
Increment Memory Location	INC	E6	F6	1	4	FE	2	4	A8-AB	1	4	1, 3
Increment A	INCA	—	—	—	—	FE	2	4	—	—	—	—
Increment XP	INCX	—	—	—	—	—	—	—	A8	1	4	—
Increment YP	INCY	—	—	—	—	—	—	—	A9	1	4	—
Decrement Memory Location	DEC	E7	F7	1	4	FF	2	4	B8-BB	1	4	2, 4
Decrement A	DECA	—	—	—	—	FF	2	4	—	—	—	—
Decrement XP	DECX	—	—	—	—	—	—	—	B8	1	4	—
Decrement YP	DECY	—	—	—	—	—	—	—	B9	1	4	—

SPECIAL NOTES

1. In Short-Direct addressing, the INC mnemonic represents opcode A8, A9, AA, and AB. These are equivalent to RAM locations $80 (A8), $81 (A9), $82 (AA), and $83 (AB).
2. In Short-Direct addressing, the DEC mnemonic represents opcode B8, B9, BA, and BB. These are equivalent to RAM locations $80 (B8), $81 (B9), $82 (BA), and $83 (BB).
3. In Indirect addressing, the INC mnemonic represents opcode E6 or F6, and causes the location pointed to by X (E6 opcode) or Y (F6 opcode) to be incremented.
4. In Indirect addressing, the INC mnemonic represents opcode E7 or F7, and causes the location pointed to by X (E7 opcode) or Y (F7 opcode) to be incremented.

Table 6-3. Branch Instructions

		Relative Addressing Mode			
Function	Mnem	Opcode	# Bytes	# Cycles	Special Notes
Branch if Carry Clear	BCC	40-5F	1	2	1
Branch if Higher or Same	(BHS)	40-5F	1	2	1, 2
Branch if Carry Set	BCS	60-7F	1	2	1
Branch if Lower	(BLO)	60-7F	1	2	1, 3
Branch if Not Equal	BNE	00-1F	1	2	1
Branch if Equal	BEQ	20-3F	1	2	1

SPECIAL NOTES

1. Each mnemonic of the Branch Instructions covers a range of 32 opcodes; e.g., BCC ranges from 40 through 5F. The actual memory location (target address) to which the branch is made is formed by adding the sign extended lower five bits of the opcode to the contents of the program counter.
2. The BHS instruction (shown in parentheses) is identical to the BCC instruction. The C bit is clear if the register was higher or the same as the location in the memory to which it was compared.
3. The BLO instruction (shown in parentheses) is identical to the BCS instruction. The C bit is set if the register was lower than the location in memory to which it was compared.

Table 6-4. Bit Manipulation Instructions

		Addressing Modes						
		Bit Set/Clear			Bit Test and Branch			
Function	Mnem	Opcode	# Bytes	# Cycles	Opcode	# Bytes	# Cycles	Special Note
Branch IFF Bit n is set	BRSET n (n=0 . . . . 7)	–	–	–	C8+n	3	5	1
Branch IFF Bit n is clear	BRCLR n (n=0 . . . . 7)	–	–	–	C0+n	3	5	1
Set Bit n	BSET n (n=0 . . . .7)	D8+n	2	4	–	–	–	1
Clear Bit n	BCLR n (n=0 . . . . 7)	D0+n	2	4	–	–	–	1

SPECIAL NOTE

1. The opcode is formed by adding the bit number (0-7) to the basic opcode. For example: to clear bit six using the BSET6 instruction the opcode becomes DE (D8+6); BCLR5 becomes (C0+5); etc.

6.2.5 Control Instructions

The control instructions control the MCU operations during program execution. Refer to Table 6-5.

6.2.6 Alphabetical Listing

The complete instruction set is given in alphabetical order in Table 6-6. There are certain mnemonics recognized by the assembler and converted to other instructions. The fact that all registers and accumulator are in RAM allows many implied instructions to exist. The implied instructions recognized by the assembler are identified in Table 6-6.

6.2.7 Opcode Map Summary

Table 6-7 contains an opcode map for the instructions used on the MCU.

6.3 IMPLIED INSTRUCTIONS

Since the accumulator and all other registers are located in RAM many implied instructions exist. The assembler-recognized implied instructions are given in Table 6-6. Some examples not recognized by the assembler are shown below.

BCLR,7 $FF	Ensures accumulator is plus
BSET,7 $FF	Ensures accumulator is minus
BRCLR,7 $FF	Branch iff accumulator is plus
BRSET,7 $FF	Branch iff accumulator is minus
BRCLR,7 $80	Branch iff X is plus (BXPL)
BRSET,7 $80	Branch iff X is minus (BXMI)
BRCLR,7 $81	Branch iff Y is plus (BYPL)
BRSET,7 $81	Branch iff Y is minus (BYMI)

Table 6-5. Control Instructions

		Addressing Modes									
		Short-Direct			Inherent			Relative			
Function	Mnem	Opcode	# Bytes	# Cycles	Opcode	# Bytes	# Cycles	Opcode	# Bytes	# Cycles	Special Notes
Transfer A to X	TAX	BC	1	4	–	–	–	–	–	–	–
Transfer A to Y	TAY	BD	1	4	–	–	–	–	–	–	–
Transfer X to A	TXA	AC	1	4	–	–	–	–	–	–	–
Transfer Y to A	TYA	AD	1	4	–	–	–	–	–	–	–
Return from Subroutine	RTS	–	–	–	B3	1	2	–	–	–	–
Return from Interrupt	RTI	–	–	–	B2	1	2	–	–	–	–
No-Operation	NOP	–	–	–	–	–	–	–	–	–	1

SPECIAL NOTE

1 . The NOP instruction is equivalent to a branch if equal (BEQ) to the location designated by PC + 1.

Table 6-6. Instruction Set

	Addressing Modes									Flags	
Mnemonic	Inherent	Immediate	Direct	Short Direct	Bit/Set Clear	Bit-Test-Branch	Register Indirect	Extended	Relative	Z	C
ADD		X	X				X			Λ	Λ
AND		X	X				X			Λ	•
ASLA			Assembler converts this to "ADD $FF"							•	•
BCC									X	Λ	Λ
BCLR					X					•	•
BCS									X	•	•
BEQ									X	•	•
BHS			Assembler converts this to "BCC"							•	•
BLO			Assembler converts this to "BCS"							•	•
BNE									X	•	•
BRCLR						X				•	Λ
BRSET						X				•	Λ
BSET					X					•	•
CLRA			Assembler converts this to "SUB $FF"							Λ	Λ
CLRX			Assembler converts this to "MVI = 0, $80"							•	•
CLRY			Assembler converts this to "MVI = 0, $81"							•	•
CMP		X	X				X			Λ	Λ
COMA	X									Λ	Λ
DEC			X	X			X			Λ	•
DECA			Assembler converts this to "DEC $FF"							Λ	•
DECX			Assembler converts this to "DEC $80"							Λ	•
DECY			Assembler converts this to "DEC $81"							Λ	•
INC			X	X			X			Λ	•
INCA			Assembler converts this to "INC $FF"							Λ	•
INCX			Assembler converts this to "INC $80"							Λ	•
INCY			Assembler converts this to "INC $81"							Λ	•
JMP								X		•	•
JSR								X		•	•
LDA		X	X	X			X			Λ	•
LDXI			Assembler converts this to "MVI DATA, $80"							•	•
LDYI			Assembler converts this to "MVI DATA, $81"							•	•
MVI		X	X							•	•
NOP			Assembler converts this to "BEQ (PC) + 1"							•	•
ROLA	X									Λ	Λ
RTI	X									Λ	Λ
RTS	X									•	•
STA			X	X			X			Λ	•
SUB		X	X				X			Λ	Λ
TAX	Assembler converts this to "STA $80"									Λ	•
TAY	Assembler converts this to "STA $81"									Λ	•
TXA	Assembler converts this to "LDA $80"									Λ	•
TYA	Assembler converts this to "LDA $81"									Λ	•

Flag Symbols: Z = Zero, C = Carry/Borrow, Λ = Test and Set if True, Cleared Otherwise, • = Not Affected

Table 6-7 – EF6804P2 Microcomputer Instruction Set Opcode Map

	Branch Instructions							
Hi Low	0 0000	1 0001	2 0010	3 0011	4 0100	5 0101	6 0110	7 0111
0 0000	2 BNE 1 REL	2 BNE 1 REL	2 BEQ 1 REL	2 BEQ 1 REL	2 BCC 1 REL	2 BCC 1 REL	2 BCS 1 REL	2 BCS 1 REL
1 0001	2 BNE 1 REL	2 BNE 1 REL	2 BEQ 1 REL	2 BEQ 1 REL	2 BCC 1 REL	2 BCC 1 REL	2 BCS 1 REL	2 BCS 1 REL
2 0010	2 BNE 1 REL	2 BNE 1 REL	2 BEQ 1 REL	2 BEQ 1 REL	2 BCC 1 REL	2 BCC 1 REL	2 BCS 1 REL	2 BCS 1 REL
3 0011	2 BNE 1 REL	2 BNE 1 REL	2 BEQ 1 REL	2 BEQ 1 REL	2 BCC 1 REL	2 BCC 1 REL	2 BCS 1 REL	2 BCS 1 REL
4 0100	2 BNE 1 REL	2 BNE 1 REL	2 BEQ 1 REL	2 BEQ 1 REL	2 BCC 1 REL	2 BCC 1 REL	2 BCS 1 REL	2 BCS 1 REL
5 0101	2 BNE 1 REL	2 BNE 1 REL	2 BEQ 1 REL	2 BEQ 1 REL	2 BCC 1 REL	2 BCC 1 REL	2 BCS 1 REL	2 BCS 1 REL
6 0110	2 BNE 1 REL	2 BNE 1 REL	2 BEQ 1 REL	2 BEQ 1 REL	2 BCC 1 REL	2 BCC 1 REL	2 BCS 1 REL	2 BCS 1 REL
7 0111	2 BNE 1 REL	2 BNE 1 REL	2 BEQ 1 REL	2 BEQ 1 REL	2 BCC 1 REL	2 BCC 1 REL	2 BCS 1 REL	2 BCS 1 REL
8 1000	2 BNE 1 REL	2 BNE 1 REL	2 BEQ 1 REL	2 BEQ 1 REL	2 BCC 1 REL	2 BCC 1 REL	2 BCS 1 REL	2 BCS 1 REL
9 1001	2 BNE 1 REL	2 BNE 1 REL	2 BEQ 1 REL	2 BEQ 1 REL	2 BCC 1 REL	2 BCC 1 REL	2 BCS 1 REL	2 BCS 1 REL
A 1010	2 BNE 1 REL	2 BNE 2 REL	2 BEQ 1 REL	2 BEQ 1 REL	2 BCC 1 REL	2 BCC 1 REL	2 BCS 1 REL	2 BCS 1 REL
B 1011	2 BNE 1 REL	2 BNE 1 REL	2 BEQ 1 REL	2 BEQ 1 REL	2 BCC 1 REL	2 BCC 1 REL	2 BCS 1 REL	2 BCS 1 REL
C 1100	2 BNE 1 REL	2 BNE 1 REL	2 BEQ 1 REL	2 BEQ 1 REL	2 BCC 1 REL	2 BCC 1 REL	2 BCS 1 REL	2 BCS 1 REL
D 1101	2 BNE 1 REL	2 BNE 1 REL	2 BEQ 1 REL	2 BEQ 1 REL	2 BCC 1 REL	2 BCC 1 REL	2 BCS 1 REL	2 BCS 1 REL
E 1110	2 BNE 1 REL	2 BNE 1 REL	2 BEQ 1 REL	2 BEQ 1 REL	2 BCC 1 REL	2 BCC 1 REL	2 BCS 1 REL	2 BCS 1 REL
F 1111	2 BNE 1 REL	2 BNE 1 REL	2 BEQ 1 REL	2 BEQ 1 REL	2 BCC 1 REL	2 BCC 1 REL	2 BCS 1 REL	2 BCS 1 REL

Abbreviations for Address Modes

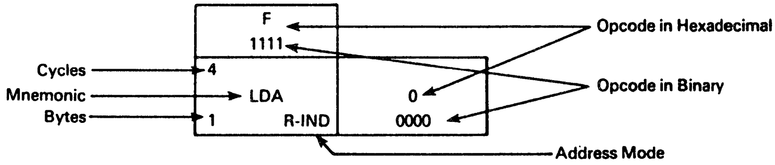
INH	Inherent
S-D	Short Direct
B-T-B	Bit Test and Branch
IMM	Immediate
DIR	Direct
EXT	Extended
REL	Relative
BSC	Bit Set/Clear
R-IND	Register Indirect

*	Indicates Instruction Reserved for Future Use
#	Indicates Illegal Instruction

Instruction Set Opcode Map

Register/Memory, Control, and Read/Modify/Write Instructions				Bit Manipulation Instructions		Register/Memory and Read/Modify/Write		
8 1000	9 1001	A 1010	B 1011	C 1100	D 1101	E 1110	F 1111	Hi Low
4 JSRn 2 EXT	4 JMPn 2 EXT	*	4 MVI 3 IMM	5 BRCLR0 3 B-T-B	4 BCLR0 2 BSC	4 LDA 1 R-IND	4 LDA 1 R-IND	0 0000
4 JSRn 2 EXT	4 JMPn 2 EXT	*	*	5 BRCLR1 3 B-T-B	4 BCLR1 2 BSC	4 STA 1 R-IND	4 STA 1 R-IND	1 0001
4 JSRn 2 EXT	4 JMPn 2 EXT	*	2 RTI 1 INH	5 BRCLR2 3 BT-B	4 BCLR2 2 BSC	4 ADD 1 R-IND	4 ADD 1 R-IND	2 0010
4 JSRn 2 EXT	4 JMPn 2 EXT	*	2 RTS 1 INH	5 BRCLR3 3 B-T-B	4 BCLR3 2 BSC	4 SUB 1 R-IND	4 SUB 1 R-IND	3 0011
4 JSRn 2 EXT	4 JMPn 2 EXT	*	4 COMA 1 INH	5 BRCLR4 3 B-T-B	4 BCLR4 2 BSC	4 CMP 1 R-IND	4 CMP 1 R-IND	4 0100
4 JSRn 2 EXT	4 JMPn 2 EXT	*	4 ROLA 1 INH	5 BRCLR5 3 B-T-B	4 BCLR5 2 BSC	4 AND 1 R-IND	4 AND 1 R-IND	5 0101
4 JSRn 2 EXT	4 JMPn 2 EXT	*	*	5 BRCLR6 3 B-T-B	4 BCLR6 2 BSC	4 INC 1 R-IND	4 INC 1 R-IND	6 0110
4 JSRn 2 EXT	4 JMPn 2 EXT	*	*	5 BRCLR7 3 B-T-B	4 BCLR7 2 BSC	4 DEC 1 R-IND	4 DEC 1 R-IND	7 0111
4 JSRn 2 EXT	4 JMPn 2 EXT	4 INC 1 S-D	4 DEC 1 S-D	5 BRSET0 3 B-T-B	4 BSET0 2 BSC	4 LDA 2 IMM	4 LDA 2 DIR	8 1000
4 JSRn 2 EXT	4 JMPn 2 EXT	4 INC 1 S-D	4 DEC 1 S-D	5 BRSET1 3 B-T-B	4 BSET1 2 BSC	#	4 STA 2 DIR	9 1001
4 JSRn 2 EXT	4 JMPn 2 EXT	4 INC 1 S-D	4 DEC 1 S-D	5 BRSET2 3 B-T-B	4 BSET2 2 BSC	4 ADD 2 IMM	4 ADD 2 DIR	A 1010
4 JSRn 2 EXT	4 JMPn 2 EXT	4 INC 1 S-D	4 DEC 1 S-D	5 BRSET3 3 B-T-B	4 BSET3 2 BSC	4 SUB 2 IMM	4 SUB 2 DIR	B 1011
4 JSRn 2 EXT	4 JMPn 2 EXT	4 LDA 1 S-D	4 STA 1 S-D	5 BRSET4 3 B-T-B	4 BSET4 2 BSC	4 CMP 2 IMM	4 CMP 2 DIR	C 1100
4 JSRn 2 EXT	4 JMPn 2 EXT	4 LDA 1 S-D	4 STA 1 S-D	5 BRSET5 3 B-T-B	4 BSET5 2 BSC	4 AND 2 IMM	4 AND 2 DIR	D 1101
4 JRSn 2 EXT	4 JMPn 2 EXT	4 LDA 1 S-D	4 STA 1 S-D	5 BRSET6 3 B-T-B	4 BSET6 2 BSC	#	4 INC 2 DIR	E 1110
4 JSRn 2 EXT	4 JMPn 2 EXT	4 LDA 1 S-D	4 STA 1 S-D	5 BRSET7 3 B-T-B	4 BSET7 2 BSC	#	4 DEC 2 DIR	F 1111

LEGEND



SECTION 7
ELECTRICAL SPECIFICATIONS

7.1 INTRODUCTION

This section contains the electrical specifications and associated timing for the EF6804J2.

7.2 MAXIMUM RATINGS

Rating	Symbol	Value	Unit
Supply Voltage	V_{CC}	−0.3 to +7.0	V
Input Voltage	V_{in}	−0.3 to (7.0	V
Operating Temperature Range Standard or L suffix V suffix T suffix	T_A	TL to TH 0 to 70 -40 to 85 -40 to 105	°C
Storage Temperature Range	T_{stg}	−55 to 150	°C
Junction Temperature Range Plastic	T_J	150	°C/W

This device contains circuitry to protect the inputs against damage due to high static voltages of electric fields; however, it is advised that normal precautions be taken to avoid application of any voltage higher than maximum rated voltages to this high impedance circuit. For proper operation it is recommended that V_{in} and V_{out} be constrained to the range $V_{SS} \leq (V_{in}$ or $V_{out}) \leq V_{CC}$. Reliability of operation is enhanced if unused inputs except EXTAL are connected to an appropriate logic voltage level (e.g., either V_{SS} or V_{CC}).

7.3 THERMAL CHARACTERISTICS

Characteristic	Symbol	Value	Unit
Thermal Resistance Plastic	θ_{JA}	90	°C/W

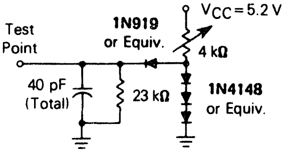


Figure 7-1. LSTTL Equivalent Test Load (Port B)

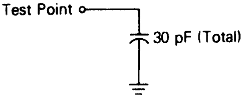


Figure 7-2. CMOS Equivalent Test Load (Ports A and B)

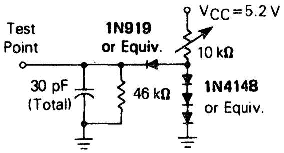


Figure 7-3. LSTTL Equivalent Test Load (Port A and TIMER)

7.4 POWER CONSIDERATIONS

The average chip-junction temperature, T_J, in °C can be obtained from:

$T_J = T_A + (P_D \cdot \theta_{JA})$ (1)

Where:

T_A = Ambient Temperature, °C

θ_{JA} = Package Thermal Resistance, Junction-to-Ambient, °C/W

$P_D = P_{INT} + P_{PORT}$

$P_{INT} = I_{CC} \times V_{CC}$, Watts – Chip Internal Power

P_{PORT} = Port Power Dissipation, Watts – User Determined

For most applications $P_{PORT} \ll P_{INT}$ and can be neglected. P_{PORT} may become significant if the device is configured to drive Darlington bases or sink LED loads.

An approximate relationship between P_D and T_J (if P_{PORT} is neglected) is:

$P_D = K \div (T_J + 273°C)$ (2)

Solving equations 1 and 2 for K gives:

$K = P_D \cdot (T_A + 273°C) + \theta_{JA} \cdot P_D^2$ (3)

Where K is a constant pertaining to the particular part. K can be determined from equation 3 by measuring P_D (at equilibrium) for a known T_A. Using this value of K the values of P_D and T_J can be obtained by solving equations (1) and (2) iteratively for any value of T_A.

7.5 ELECTRICAL CHARACTERISTICS (V_{CC} = +5.0 Vdc ± 0.5 Vdc, V_{SS} = 0 Vdc, $T_A = T_L$ to T_H unless otherwise noted)

Characteristic	Symbol	Min	Typ	Max	Unit
Internal Power Dissipation – No Port Loading $T_A = 0°C$ $T_A = -40°C$	P_{INT} –	– –	135 –	170 210	mV
Input High Voltage	V_{IH}	4.0	–	V_{CC}	V
Input Low Voltage	V_{IL}	V_{SS}	–	0.8	V
Input Capacitance	C_{in}	–	10	–	pF
Input Current ($\overline{IRQ}$, $\overline{RESET}$)	I_{in}	–	2	20	µA

7.6 SWITCHING CHARACTERISTICS (V_{CC} = +5.0 Vdc ± 0.5 Vdc, V_{SS} = GND, $T_A = T_L$ to T_H unless otherwise noted)

Characteristic	Symbol	Min	Typ	Max	Unit
Oscillator Frequency	f_{OSC}	4.0	–	11.0	MHz
Bit Time	t_{bit}	0.364	–	1.0	µs
Byte Cycle Time	t_{byte}	4.36	–	12.0	µs
$\overline{IRQ}$ and TIMER Pulse Width	t_{WL}, t_{WH}	$2 x t_{byte}$	–	–	–
$\overline{RESET}$ Pulse Width	t_{RWL}	$2 x t_{byte}$	–	–	–
$\overline{RESET}$ Delay Time (External Capacitance = 1.0 µF)	t_{RHL}	100	–	–	ms

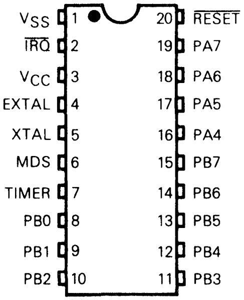
7.7 PORT DC ELECTRICAL CHARACTERISTICS (V_{CC} = +5.0 Vdc ± 0.5 Vdc, V_{SS} = GND, $T_A = T_L$ to T_H unless otherwise noted)

Characteristic	Symbol	Min	Typ	Max	Unit
Timer and Ports A (Standard)					
Output Low Voltage, I_{Load} = 0.4 mA	V_{OL}	–	–	0.5	V
Output High Voltage, I_{Load} = – 50 μA	V_{OH}	2.3	–	–	V
Input High Voltage	V_{IH}	2.0	–	V_{CC}	V
Input Low Voltage	V_{IL}	V_{SS}	–	0.8	V
Hi-Z State Input Current	I_{TSI}	–	4	40	μA
Timer and Ports A (**Open Drain**)					
Output Low Voltage, I_{Load} = 0.4 mA	V_{OL}	–	–	0.5	V
Input High Voltage	V_{IH}	2.0	–	V_{CC}	V
Input Low Voltage	V_{IL}	V_{SS}	–	0.8	V
Hi-Z State Input Current	I_{TSI}	–	4	40	μA
Open Drain Leakage (V_{out} = V_{CC})	I_{LOD}	–	4	40	μA
Timer and Ports A (**CMOS Drive**)					
Output Low Voltage, I_{Load} = 0.4 mA (Sink)	V_{OL}	–	–	0.5	V
Output High Voltage, I_{Load} = – 10 μA	V_{OH}	V_{CC} – 1.0	–	–	V
Output High Voltage, I_{Load} = – 50 μA	V_{OH}	2.3	–	–	V
Input High Voltage, I_{Load} = – 300 μA Max	V_{IH}	2.0	–	V_{CC}	V
Input Low Voltage, I_{Load} = – 300 μA Max	V_{IL}	V_{SS}	–	0.8	V
Hi-Z State Input Current (V_{in} = 0.4 V to V_{CC})	I_{TSI}	–	–	– 300	μA
Port B (Standard)					
Output Low Voltage, I_{Load} = 1.0 mA	V_{OL}	–	–	0.5	V
Output Low Voltage, I_{Load} = 10 mA (Sink)	V_{OL}	–	–	1.5	V
Output High Voltage, I_{Load} = – 100 μA	V_{OH}	2.3	–	–	V
Input High Voltage	V_{IH}	2.0	–	V_{CC}	V
Input Low Voltage	V_{IL}	V_{SS}	–	0.8	V
Hi-Z State Input Current	I_{TSI}	–	8	80	μA
Port B (Open Drain)					
Output Low Voltage, I_{Load} = 1.0 mA	V_{OL}	–	–	0.5	V
Output Low Voltage, I_{Load} = 10 mA (Sink)	V_{OL}	–	–	1.5	V
Input High Voltage	V_{IH}	2.0	–	V_{CC}	V
Input Low Voltage	V_{IL}	V_{SS}	–	0.8	V
Hi-Z State Input Current	I_{TSI}	–	8	80	μA
Open Drain Leakage (V_{out} = V_{CC})	I_{LOD}	–	8	80	μA
Port B (CMOS Drive)					
Output Low Voltage, I_{Load} = 1.0 mA	V_{OL}	–	–	0.5	V
Output High Voltage, I_{Load} = 10 mA (Sink)	V_{OL}	–	–	1.5	V
Output High Voltage, I_{Load} = – 10 μA	V_{OH}	V_{CC} – 1.0	–	–	V
Output High Voltage, I_{Load} = – 100 μA	V_{OH}	2.3	–	–	V
Input High Voltage, I_{Load} = – 300 μA Max	V_{IH}	2.0	–	V_{CC}	V
Input Low Voltage, I_{Load} = – 300 μA Max	V_{IL}	V_{SS}	–	0.8	V
Hi-Z State Input Current (V_{in} = 0.4 V to V_{CC})	I_{TSI}	–	–	– 300	μA

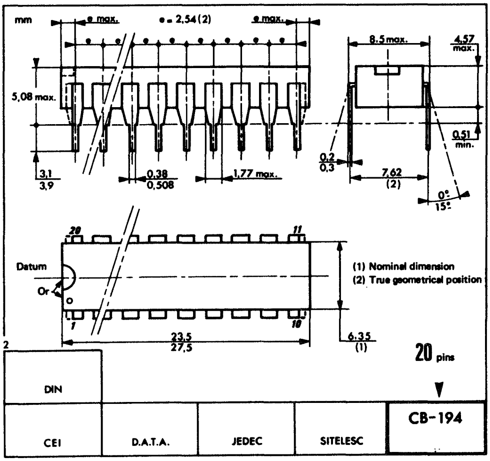
SECTION 8
MECHANICAL DATA

This section contains the pin assignment and package dimension diagrams for the EF6804J2 microcomputer.

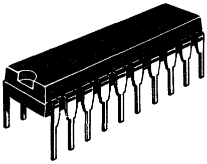
8.1 PIN ASSIGNMENT



8.2 PHYSICAL DIMENSIONS



CB-194



P SUFFIX
PLASTIC PACKAGE

SECTION 9
ORDERING INFORMATION

9.1 INTRODUCTION

The following information is required when ordering a custom MCU. The information may be transmitted to THOMSON SEMICONDUCTEURS in the following media:
EPROM(s), 2716 or 2732
EFDOS/MDOS*, disk file

To initiate a ROM pattern for the MCU, it is necessary to first contact your local field service office, local sales person, or your local THOMSON SEMICONDUCTEURS representative.

9.1.1 EPROMs

One 2716 or one 2732 type EPROM, programmed with the customer program (positive logic sense for address and data), may be submitted for pattern generation. Since all program and data space information will fit on one 2716 or 2732 EPROM, the EPROM must be programmed as follows in order to emulate the EF6804J2 MCU. **For a 2716, start the data space ROM at EPROM address $020 and start program space ROM at EPROM address $410 and continue to memory space $7FF. Memory spaces $7F8 through $7FB are reserved for THOMSON SEMICONDUCTEURS self-test vectors. For a 2732, the memory map shown in Figure 2-1 can be used. All unused bytes, including the user's space, must be set to zero.** For shipment to THOMSON SEMICONDUCTEURS the EPROMs should be placed in a conductive IC carrier and packed securely. Do not use styrofoam.

9.1.2 EFDOS/MDOS* Disk File

An EFDOS/MDOS* disk, programmed with the customer program (positive logic sense for address and data), may be submitted for pattern generation. **When using the EFDOS/MDOS* disk, include the entire memory image of both data and program space. All unused bytes, including the user's space, must be set to zero.**

9.2 VERIFICATION MEDIA

All original pattern media (EPROMs or floppy disk) are filed for contractual purposes and are not returned. A computer listing of the ROM code will be generated and returned along with a listing verification form. The listing should be thoroughly checked and the verification form completed, signed, and returned to THOMSON SEMICONDUCTEURS. The signed verification form constitutes the contractural agreement for creation of the customer mask. If desired, THOMSON SEMICONDUCTEURS will program a blank 2716, 2732, or EFDOS/MDOS* disk **(supplied by the customer)** from the data file used to create the custom mask to aid in the verification process.

* Requires prior factory approval

9.3 ROM VERIFICATION UNITS (RVUs)

Ten MCUs containing the customer's ROM pattern will be sent for program verification. These units will have been made using the custom mask but are for the purpose of ROM verification only. For expediency they are usually tested only at room temperature, five volts and may be unmarked and packaged in ceramic. These RVUs are included in the mask charge and are not production parts.

These RVUs.are not backed nor guaranteed by **THOMSON SEMICONDUCTEURS Quality Assurance.**

9.4 FLEXIBLE DISKS

The disk media submitted must be single-sided single density, 8-inch, EFDOS/MDOS* compatible floppies. The customer must clearly label the disk with the ROM pattern file name. The minimum EFDOS/MDOS* system files as well as the absolute binary object file (file name. LO type of file) from the 6804 cross assembler must be on the disk. An object file made from a memory dump, using the ROLLOUT command is also admissable. Consider submitting a source listing as well as: file name, .LX (DEVICE/EXORciser loadable format). This file will of course be kept confidential and is used 1) to speed up the process in house if any problems arise, and 2) to speed up our customer to factory interface if a user finds any software errors and needs assistance quickly from THOMSON SEMICONDUCTEURS factory representatives.

EFDOS is THOMSON SEMICONDUCTEURS Disk Operating System available on development systems such as DEVICE.
MDOS is MOTOROLA's Disk Operating System available on development systems such as EXORciser . .
*Requires prior factory approval.

Whenever ordering a custom MCU is required, please contact your local THOMSON SEMICONDUCTEURS distributor and/or complete and send the attached "MCU customer ordering sheet SEMICONDUCTEURS representative.

ORDERING INFORMATION

EF6804J2 | P | V |

Device

Package

Screening level

Oper. temp.

The table below horizontally shows all available suffix combinations for package, operating temperature and screening level. Other possibilities on request.

DEVICE	PACKAGE					OPER. TEMP			SCREENING LEVEL			
	C	J	P	E	FN	L*	V	T	Std	D		
EF6804J2			●			●	●		●	●		

Examples : EF6804J2P, EF6804J2PV, EF6804J2PLD, EF6804J2PVD.

Package : **C** : Ceramic DIL, **P** : Plastic DIL, **FN** : PLCC.
Oper. temp. : **L*** : 0°C to +70°C, **V** : -40°C to +85°C, **T** : -40°C to +105°C, * : may be omitted.
Screening level : **Std** : (no-end suffix), **D** : NFC 96883 level D.

EXORciser is a registered trademark of Motorola Inc.

These specifications are subject to change without notice.
Please inquire with our sales offices about the availability of the different products.

Printed in France

Ref.DSEF6804J2

THOMSON
COMPONENTS
SEMICONDUCTEURS

EF6804P2

8 BIT MICROCOMPUTER

ADVANCE INFORMATION

The EF6804P2 Microcomputer Unit (MCU) is a member of the EF6804 Family of very low-cost single-chip microcomputers. This 8-bit microcomputer contains a CPU, on-chip CLOCK, ROM, RAM, I/O, and TIMER. It is designed for the user who needs an economical microcomputer with the proven capabilities of the EF6800-based instruction set. The following are some of the hardware and software highlights of the EF6804P2 MCU.

HARDWARE FEATURES

- 8-Bit Architecture
- Pin Compatible with the EF6805P2 and EF68HC04P3
- 32 Bytes of RAM
- Memory Mapped I/O
- 1020 Bytes of User ROM
- 64 Bytes of ROM for Look-Up Tables
- 20 TTL/CMOS Compatible Bidirectional I/O Lines (Eight Lines are LED Compatible)
- 8-Bit Timer with 7-Bit Software Programmable Prescaler
- On-Chip Clock Generator
- Self-Check Mode and ROM Verify Mode
- Master Reset
- Complete Development System Support on DEVICE®
- 5 Volt Single Supply
- TIMER Pin is Programmable as Input or Output

SOFTWARE FEATURES

- Similar to EF6805 HMOS Family
- Byte Efficient Instruction Set
- Easy to Program
- True Bit Manipulation
- Bit Test and Branch Instruction
- Separate Flags for Interrupt and Normal Processing
- Versatile Indirect Registers
- Conditional Branches
- Single Instruction Memory Examine/Change
- True LIFO Stack Eliminates Stack Pointer
- Eight Powerful Addressing Modes
- Any Bit in Data Space Memory May be Tested
- Any Bit in Data Space Memory Capable of Being Written to May be Set or Cleared.

USER SELECTABLE OPTIONS

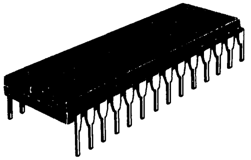
- 20 Bidirectional I/O Lines with LSTTL, LSTTL/CMOS, or Open-Drain Interface
- Crystal or Low-Cost Resistor-Capacitor Oscillator
- Mask Selectable Edge- or Level-Sensitive Interrupt Pin

DEVICE® is THOMSON SEMICONDUCTEURS' development/emulation tool.

HMOS

CASES

CB-132

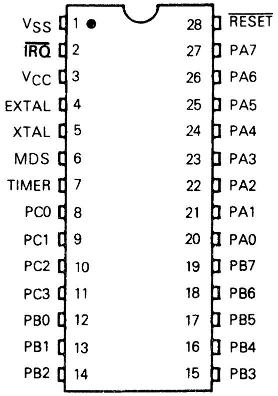


P SUFFIX
PLASTIC PACKAGE

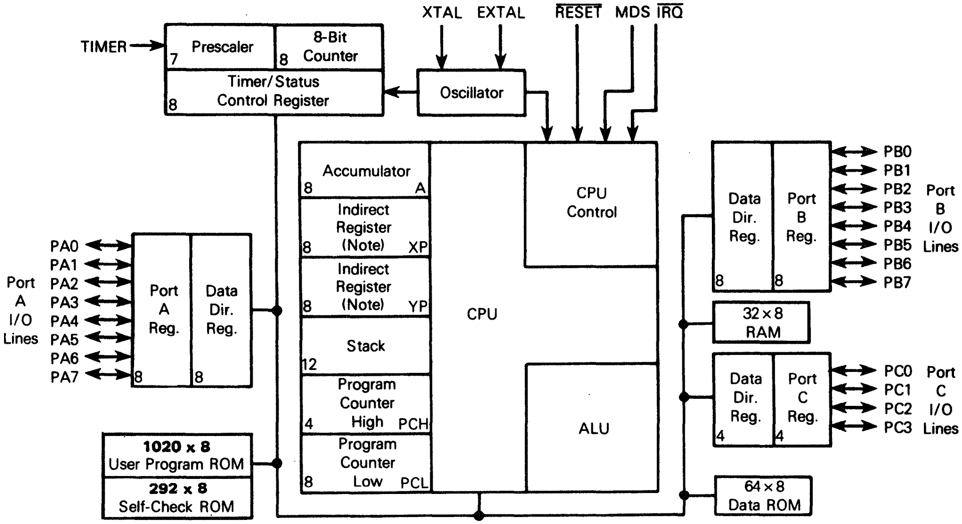
CB-520

FN SUFFIX
PLCC-28

PIN ASSIGNMENT



BLOCK DIAGRAM



NOTE: 8-Bit indirect registers XP and YP, although shown as part of the CPU, are actually located in the 32 x 8 RAM at locations $80 and $81.

SECTION 2
FUNCTIONAL PIN DESCRIPTION, MEMORY, CPU, AND REGISTERS

This section provides a description of the functional pins, memory spaces, the central processing unit (CPU), and the various registers and flags.

2.1 FUNCTIONAL PIN DESCRIPTION

2.1.1 V_{CC} and V_{SS}

Power is supplied to the MCU using these two pins. V_{CC} is power and V_{SS} is the ground connection.

2.1.2 $\overline{IRQ}$

This pin provides the capability for asynchronously applying an external interrupt to the MCU. Refer to **4.1 INTERRUPT** for additional information.

2.1.3 XTAL and EXTAL

These pins provide connections to the on-chip clock oscillator circuit. A crystal, a resistor and capacitor, or an external signal, depending on the user selectable manufacturing mask option, can be connected to these pins to provide a system clock source with various stability/cost tradeoffs. Lead lengths and stray capacitance on these two pins should be minimized. Refer to **4.4 INTERNAL CLOCK GENERATOR OPTIONS** for recommendations concerning these inputs.

2.1.4 TIMER

In the input mode, the timer pin is connected to the prescaler input and serves as the timer clock. In the output mode, the timer pin signals that a time out of the timer has occurred. Refer to **SECTION 3 TIMER** for additional information.

2.1.5 $\overline{RESET}$

The $\overline{RESET}$ pin is used to restart the processor of the EF6804P2 to the beginning of a program. This pin, together with the MDS pin is also used to select the operating mode of the EF6804P2. If the MDS pin is at zero volts, the normal mode is selected and the program counter is loaded with the user restart vector. However, if the MDS pin is at +5 volts, then pins PA6 and PA7 are decoded to allow selection of the operating mode. Refer to **4.3 RESET** for additional information.

2.1.6 MDS

The MDS (mode select) pin is used to place the MCU into special operating modes. If MDS is held at +5 volts at the exit of the reset state, the decoded state of PA6 and PA7 is latched to determine the operating mode (single-chip, self-check, or ROM verify). However, if MDS is held at zero volts at the exit of the reset state, the single-chip operating mode is automatically selected (regardless of PA6 and PA7 state).

For those users familiar with the EF6801 microcomputer, mode selection is similar but much less complex in the EF6804P2. No special external diodes, switches, transistors, etc. are required in the EF6804P2.

2.1.7 Input/Output Lines (PA0-PA7, PB0-PB7, PC0-PC3)

These 20 lines are arranged into two 8-bit ports (A and B) and one 4-bit port (C). All lines are programmable as either inputs or outputs under software control of the data direction registers. Refer to **SECTION 5 INPUT/OUTPUT PORTS** for additional information.

2.2 MEMORY

The MCU operates in three different memory spaces: program space, data space, and stack space. A representation of these memory spaces is shown in Figure 2-1. The program space (Figure 2-1a) contains all of the instructions that are to be executed, as well as the data required for the immediate addressing mode instructions, and the self-check and user vectors. The data space (Figure 2-1b) contains all of the RAM locations, plus I/O locations and some ROM used for storage of tables and constants. The stack space (Figure 2-1c) contains RAM which is used for stacking subroutine and interrupt return addresses.

The MCU is capable of addressing 4096 bytes of program space memory with its program counter and 256 bytes of data space memory with its instructions. The data space memory contains three bytes for port data registers, three bytes for port data direction registers, one byte for timer status/control, 64 bytes ROM, 32 bytes RAM (which includes two bytes for XP and YP indirect registers), two bytes for timer prescaler and count registers, and one byte for the accumulator. The program space section includes 288 bytes of self-check ROM, 1016 bytes program ROM, and eight bytes of vectors for self-check and user programs.

2.3 CENTRAL PROCESSING UNIT

The CPU of the EF6804 Family is implemented independently from the I/O or memory configuration. Consequently, it can be treated as an independent central processor communicating with I/O and memory via internal addresses, data, and control buses.

(a) Program Space Memory Map

Reserved (All Ones)	$000 $ADF
Self-Check ROM	$AE0 $BFF
Program ROM	$C00 $FF7
Self Check $\overline{\text{IRQ}}$ Vector	$FF8-$FF9
Self Check Restart Vector	$FFA-$FFB
User $\overline{\text{IRQ}}$ Vector	$FFC-$FFD
User Restart Vector	$FFE-$FFF

(b) Data Space Memory Map

Port A Data Register	$00
Port B Data Register	$01
1 1 1 1 Port C Data Reg.	$02
Not Used	$03
Port A Data Direction Register	$04
Port B Data Direction Register	$05
1 1 1 1 Port C DDR	$06
Not Used	$07 $08
Timer Status Control Register	$09
Future Expansion	$0A $1F
User Data Space ROM	$20 $5F
Future Expansion	$60 $7F
Indirect Register X	$80
Indirect Register Y	$81
Data Space RAM	$82 $9F
Future Expansion	$A0 $FC
Prescaler Register	$FD
Timer Count Register	$FE
Accumulator	$FF

(c) Stack Space Memory Map

Level 1
Level 2
Level 3
Level 4

Figure 2-1. EF6804P2 MCU Address Map

2.4 REGISTERS

The EF6804 Family CPU has four registers and two flags available to the programmer. They are. shown in Figure 2-2 and are explained in the following paragraphs.

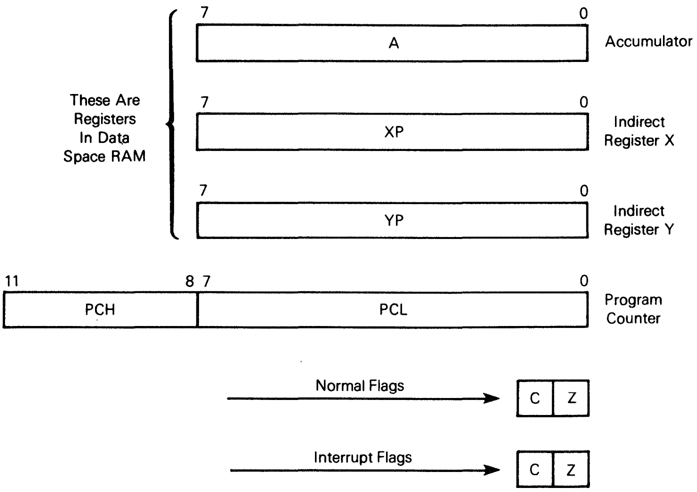


Figure 2-2.- Programming Model

2.4.1 Accumulator (A)

The accumulator is an 8-bit general purpose register used in all arithmetic calculations, logical operations, and data manipulations. The accumulator is implemented as the highest RAM location ($FF) in data space and thus implies that several instructions exist which are not explicitly implemented. Refer to **6.3 IMPLIED INSTRUCTIONS** for additional information.

2.4.2 Indirect Registers (XP, YP)

These two indirect registers are used to maintain pointers to other memory locations in data space. They are used in the register-indirect addressing mode, and can be accessed with the direct, indirect, short direct, or bit set/clear addressing modes. These registers are implemented as two of the 32 RAM locations ($80, $81) and as such generate implied instructions and may be manipulated in a manner similar to any RAM memory location in data space. Refer to **6.3 IMPLIED INSTRUCTIONS** for additional information.

2.4.3 Program Counter (PC)

The program counter is a 12-bit register that contains the address of the next ROM word to be used (may be opcode, operand, or address of operand). The 12-bit program counter is contained in PCL (low byte) and PCH (high nibble).

2.4.4 Flags (C, Z)

The carry (C) bit is set on a carry or a borrow out of the ALU. It is cleared if the result of an arithmetic operation does not result in a carry or a borrow. The (C) bit is also set to the value of the bit tested in a bit test instruction, and participates in the rotate left instruction.

The zero (Z) bit is set if the result of the last arithmetic or logical operation was equal to zero, otherwise it is cleared.

There are two sets of these flags, one set is for interrupt processing, the other for all other routines. When an interrupt occurs, a context switch is made from the program flags to the interrupt flags (interrupt mode). An RTI forces the context switch back to the program flags (program mode). While in either mode, only the flags for that mode are available. Further, the interrupt flags will not be cleared upon entering the interrupt mode. Instead, the flags will be as they were at the exit of the last interrupt mode. Both sets of flags are cleared by reset.

2.4.5 Stack

There is a true LIFO stack incorporated in the EF6804P2 which eliminates the need for a stack pointer. Stack space is implemented in separate RAM (12-bits wide) shown in Figure 2-1c. Whenever a subroutine call (or interrupt) occurs, the contents of the PC are shifted into the top register of the stack. At the same time (same cycle), the top register is shifted to the next level deeper. This happens to all registers with the bottom register falling out the bottom of the stack.

Whenever a subroutine or interrupt return occurs, the top register is shifted into the PC and all lower registers are shifted up one level higher. The stack RAM is four levels deep. If the stack is pulled more than four times without any pushes, the address that was stored in the bottom level will be shifted into the PC.

SECTION 3
TIMER

3.1 INTRODUCTION

A block diagram of the EF6804P2 timer circuitry is shown in Figure 3-1. The timer logic in the MCU is comprised of a simple 8-bit counter (timer count register, TCR) with a 7-bit prescaler, and a timer status/control register (TSCR). The timer count register, which may be loaded under program control, is decremented towards zero by a clock input (prescaler output). The prescaler is used to extend the maximum interval of the overall timer. The prescaler tap is selected by bits 0-2 (PSO-PS2) of the timer status/control register. Bits PSO-PS2 control the actual division of the prescaler within the range of divide-by-1 (2^0) to divide-by-128 (2^7). The timer count register (TCR) and prescaler are decremented on rising clock edges. The coding of the TCSR PSO-PS2 bits produce a division in the prescaler as shown in Table 3-1.

Table 3-1. Prescaler Coding Table

PS2	PS1	PSO	Divide By
0	0	0	1
0	0	1	2
0	1	0	4
0	1	1	8

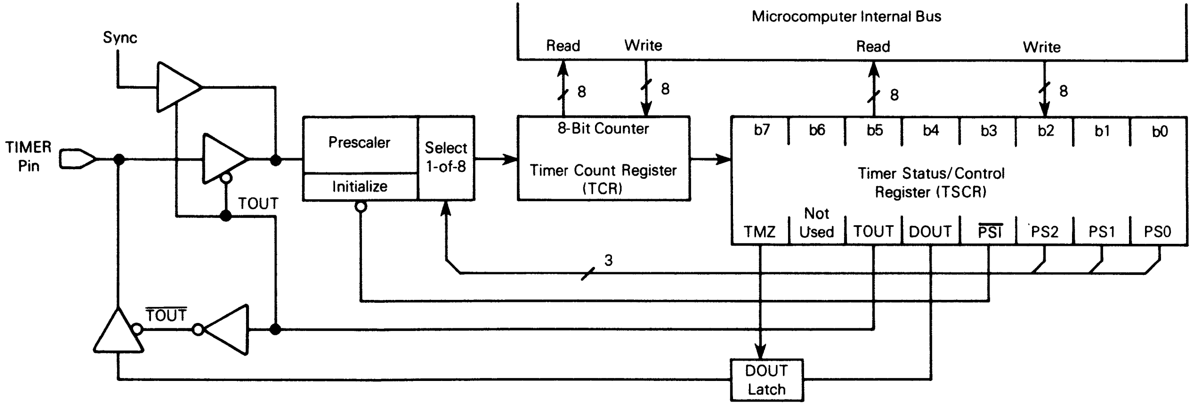
PS2	PS1	PSO	Divide By
1	0	0	16
1	0	1	32
1	1	0	64
1	1	1	128

The TIMER pin may be programmed as either an input or an output depending on the status of TOUT (TSCR bit 5). Refer to Figure 3-1. In the input mode, TOUT is a logic zero and the TIMER pin is connected directly to the prescaler input. Therefore, the timer prescaler is clocked by the signal applied from the TIMER pin. The prescaler then divides its clock input by a value determined by the coding of the TSCR bits PSO-PS2 as shown in Table 3-1. The divided prescaler output then clocks the 8-bit timer count register (TCR). When the TCR is decremented to zero, it sets the TMZ bit in the timer status/control register (TSCR). The TMZ bit can be tested under program control to perform a timer function whenever it goes high. The frequency of the external clock applied to the TIMER pin must be less than t_{byte} (f_{osc}/48).

In the output mode, TOUT is a logic one and the TIMER pin is connected to the DOUT latch. Therefore, the timer prescaler is clocked by the internal sync pulse (divide-by-48 of the internal oscillator). Operation is similar to that described above for the input mode. However, in the output mode, the low-to-high TMZ bit transition is used to latch the DOUT bit of the TSCR and provide it as output for the TIMER pin.

NOTE

TMZ is normally set to logic one when the timer times out (TCR decrements to $00); however, it may be set by a write of $00 to the TCR or by a write to bit 7 of the TSCR.



TIMER Pin Status

TOUT	Prescaler Clock	TIMER Pin
0	TIMER Pin	Input Mode
1	Sync	Output Mode

Figure 3-1. Timer Block Diagram

During reset, the timer count register and prescaler are set to $FF, while the timer status/control register is cleared to $00 and the DOUT LATCH (TIMER pin is in the high-impedance input mode) is forced to a logic high. The prescaler and timer count register are implemented in data space RAM locations ($FD, $FE); therefore, they are both readable and writeable. A write to either will predominate over the TCR decrement-to-$00 function; i.e., if a write and a TCR decrement-to-$00 occur simultaneously, the write will take precedence, and the TMZ bit is not set until the next timer time out.

3.2 TIMER REGISTERS

3.2.1 Timer Count Register (TCR)

7	0
MSB	LSB

TCR Address = $FE

The timer count register indicates the state of the internal 8-bit counter.

3.2.2 Timer Status/Control Register (TSCR)

7	6	5	4	3	2	1	0
TMZ	Not Used	TOUT	DOUT	$\overline{\text{PSI}}$	PS2	PS1	PS0

TSCR Address = $09

b7,TMZ — Low-to-high transition indicates the timer count register has decremented to zero since the timer status/control register was last read. Cleared by a read of TSCR register if TMZ was read as a logic one.

b6 — Not used.

b5, TOUT — When low, this bit selects the input mode for the timer. When high, the output mode is selected.

b4, DOUT — Data sent to the timer output pin when TMZ is set high (output mode only).

b3, $\overline{\text{PSI}}$ — Used to initialize the prescaler and inhibit its counting while $\overline{\text{PSI}} = 0$. The initialized value is set to $FF. The timer count register will also be inhibited (contents unchanged). When $\overline{\text{PSI}} = 1$ the prescaler begins to count downward.

b0, b1, b2 PS0-PS1-PS2 — These bits are used to select the prescaler divide-by ratio; therefore, effecting the clock input frequency to the timer count register.

3.2.3 Timer Prescaler Register

TPR Address= $FD

The timer prescaler register indicates the state of the internal 7-bit prescaler. This 7-bit prescaler divide ratio is normally determined by bits PS0-PS2 of the timer status/control register (see Table 3-1).

SECTION 4
INTERRUPT, SELF-CHECK, RESET, AND INTERNAL CLOCK GENERATOR

4.1 INTERRUPT

The EF6804P2 can be interrupted by applying a logic low signal to the $\overline{\text{IRQ}}$ pin ; however, a mask option selected at the time of manufacture determines whether the negative-going edge or the actual low level is sensed to indicate an interrupt.

4.1.1 Edge-Sensitive Option

When the $\overline{\text{IRQ}}$ pin is pulled low, the internal interrupt request latch is set. Prior to each instruction fetch, the interrupt request latch is tested and, if its output is high, an interrupt sequence is initiated at the end of the current instruction (provided the interrupt mask is cleared). Figure 4-1 contains a flowchart which illustrates both the reset and interrupt sequence. The interrupt sequence consists of one cycle during which: the interrupt request latch is cleared, the interrupt mode flags are selected, the PC is saved on the stack, the interrupt mask is set, and the $\overline{\text{IRQ}}$ vector (single chip mode= $FFC/$FFD, self-check mode= $FF8/$FF9) is loaded into the PC. Internal processing of the interrupt continues until an RTI (return from interrupt) instruction is processed. During the RTI instruction, the interrupt mask is cleared and the program mode flags are selected.The next instruction of the program is then fetched and executed. Once the interrupt was initially detected and the interrupt sequence started, the interrupt request latch is cleared so that the next (second) interrupt may be detected even while the previous (first) one is being serviced. However, even though the second interrupt sets the interrupt request latch during processing of the first interrupt, the second interrupt sequence will not be initiated until completion of the interrupt service routine for the first interrupt. Completion of an interrupt service routine is always accomplished using an RTI instruction to return to the main program. The interrupt mask (which is not directly available to the programmer) is cleared during the last cycle of the RTI instruction.

4.1.2 Level-Sensitive Option

The actual operation of the level-sensitive and edge-sensitive options are similar except that the level-sensitive option does not have an interrupt request latch. With no interrupt request latch, the logic level of the $\overline{\text{IRQ}}$ pin is checked for detection of the interrupt. Also, in the interrupt sequence, there is no need to clear the interrupt request latch. These differences are illustrated in the flowchart of Figure 4-1.

4.1.3 Power Up and Timing

During the power-up sequence the interrupt mask is set to preclude any false or "ghost" interrupts from occurring. To clear the interrupt mask, the programmer should write a JSR (instead of a JMP) instruction to an initialization routine as the first instruction in a program. The initialization routine

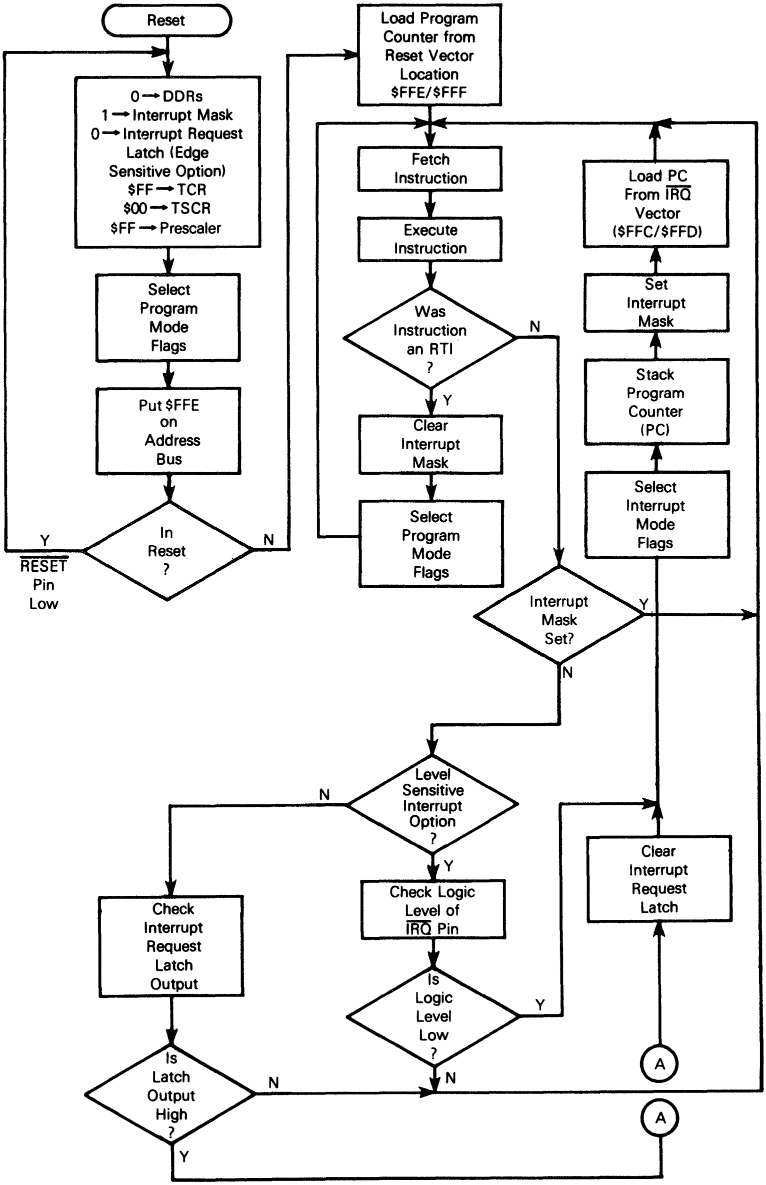

Reset
0 → DDRs
1 → Interrupt Mask
0 → Interrupt Request Latch (Edge Sensitive Option)
$FF → TCR
$00 → TSCR
$FF → Prescaler
Select Program Mode Flags
Put $FFE on Address Bus
In Reset ?
Y
N
RESET Pin Low
Load Program Counter from Reset Vector Location $FFE/$FFF
Fetch Instruction
Execute Instruction
Was Instruction an RTI ?
Y
N
Clear Interrupt Mask
Select Program Mode Flags
Interrupt Mask Set?
Y
N
Load PC From IRQ Vector ($FFC/$FFD)
Set Interrupt Mask
Stack Program Counter (PC)
Select Interrupt Mode Flags
Level Sensitive Interrupt Option ?
N
Y
Check Logic Level of IRQ Pin
Is Logic Level Low ?
Y
N
Check Interrupt Request Latch Output
Is Latch Output High ?
N
Y
Clear Interrupt Request Latch
A
A

Figure 4-1. Reset and Interrupt Processing Flowchart

should end with an RTI (instead of RTS). Maximum interrupt response time is eight machine (t_{byte}) cycles (see **4.4 INTERNAL CLOCK GENERATOR OPTIONS**). This includes five machine cycles for the longest instruction, plus one machine cycle for stacking the PC and switching flags, plus two machine cycles for synchronization of the $\overline{IRQ}$ input with the internal clock. Minimum response time is one machine cycle for stacking PC and switching flags (see **2.4.4 Flags (C, Z)**).

4.2 SELF-CHECK

The self check capability of the EF6804P2 MCU provides an internal check to determine if the part is functional. A schematic diagram of the self-check connections is shown in Figure 4-2. To perform a functional check of the MCU, connect it as shown in Figure 4-2a and monitor the LEDs for a 00100 ($04) pattern on port A. To initiate a ROM self-check of the memory simply connect the circuit as shown in Figure 4-2b and check that the "good" LED turns on to indicate a good memory. The ROM verify uses a cyclical redundancy check (CRC) to conduct a ROM check by means of signature analysis circuit. This circuit consists of two 8-bit shift registers configured to perform the check using the CCITT polynominal.

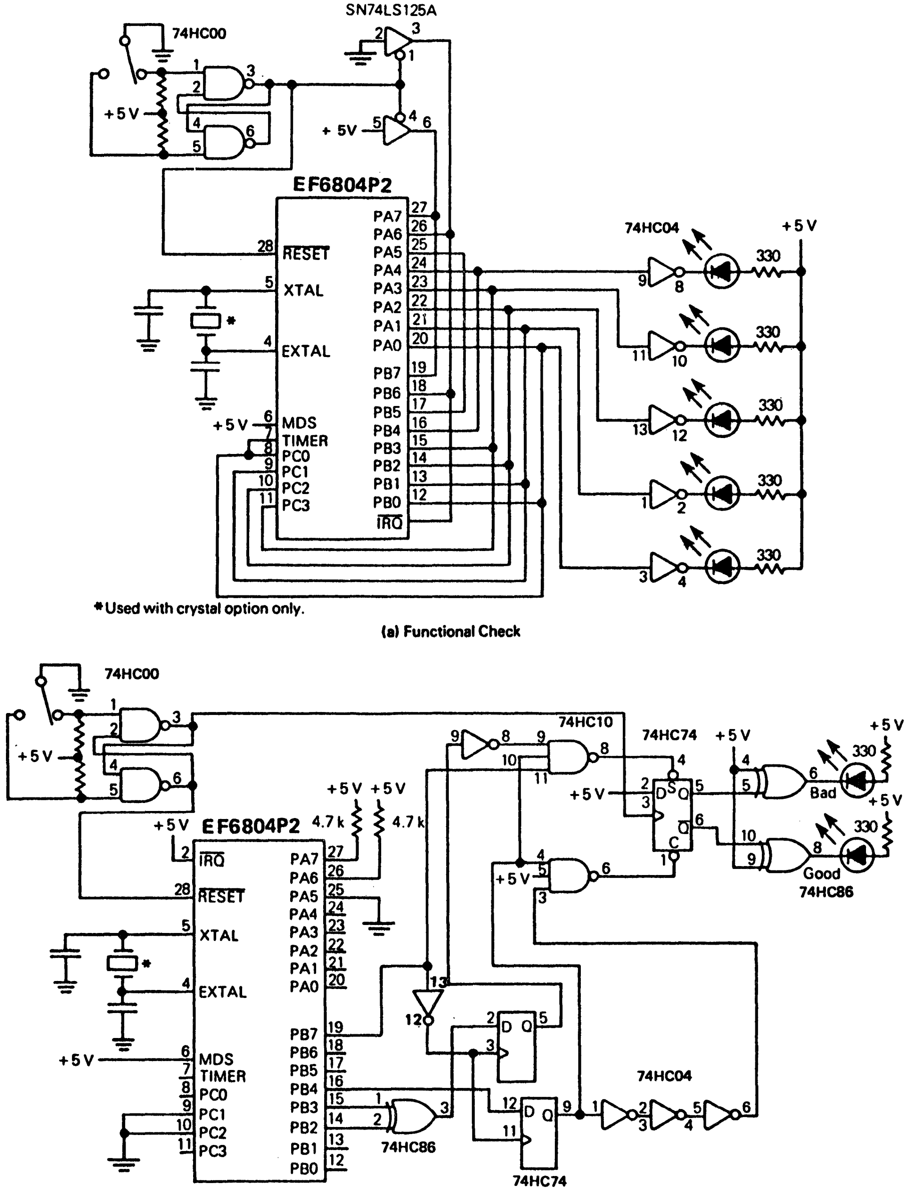
4.3 RESET

The MCU can be reset in two ways: by initial power up (see Figure 4-1) and by the external reset input ($\overline{RESET}$). During power up, a delay of t_{RHL} is needed before allowing the $\overline{RESET}$ input to go high. This time delay allows the internal clock generator to stabilize. Connecting a capacitor and resistor to the $\overline{RESET}$ input, as shown in Figure 4-3, typically provides sufficient delay.

4.4 INTERNAL CLOCK GENERATOR OPTIONS

The internal clock generator circuit is designed to require a minimum of external components. A crystal, a resistor-capacitor, or an external signal may be used to generate a system clock with various stability/cost tradeoffs. A manufacturing mask option is required to select either the crystal oscillator or the RC oscillator circuit. The different clock generator option connection methods are shown in Figure 4-4, crystal specifications and suggested PC board layouts are given in Figure 4-5, resistor-capacitor selection graph is given in Figure 4-6, and a timing diagram is illustrated in Figure 4-7. The crystal oscillator startup time is a function of many variables: crystal parameters (especially R_S), oscillator load capacitance (C_L), IC parameters, ambient temperature, and supply voltage. To ensure rapid oscillator startup, neither the crystal characteristics nor the load capacitance should exceed recommendations.

The oscillator output frequency is internally divided by four to produce the internal $\phi1$ and $\phi2$ clocks. The $\phi1$ clock is divided by twelve to produce a machine byte (cycle) clock. A byte cycle is the smallest unit needed to execute any operation (i.e., increment the program counter). An instruction may need two, four, or five byte cycles to execute.



*Used with crystal option only.

(a) Functional Check

*Used with crystal option only.

(b) Simple ROM Verify Check

Figure 4-2. Self-Check Circuit

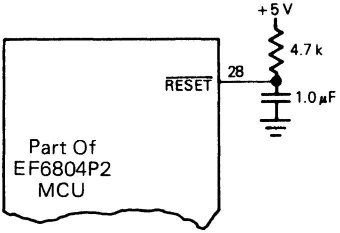
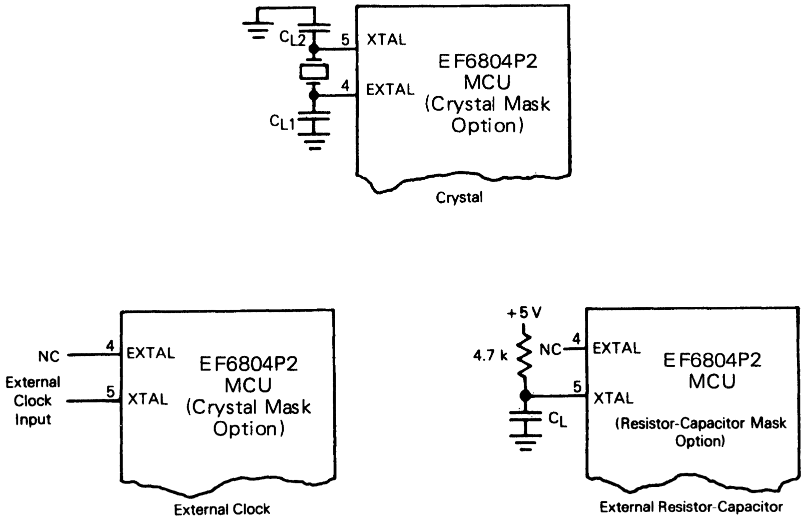


Figure 4-3. Power-Up Reset Delay Circuit

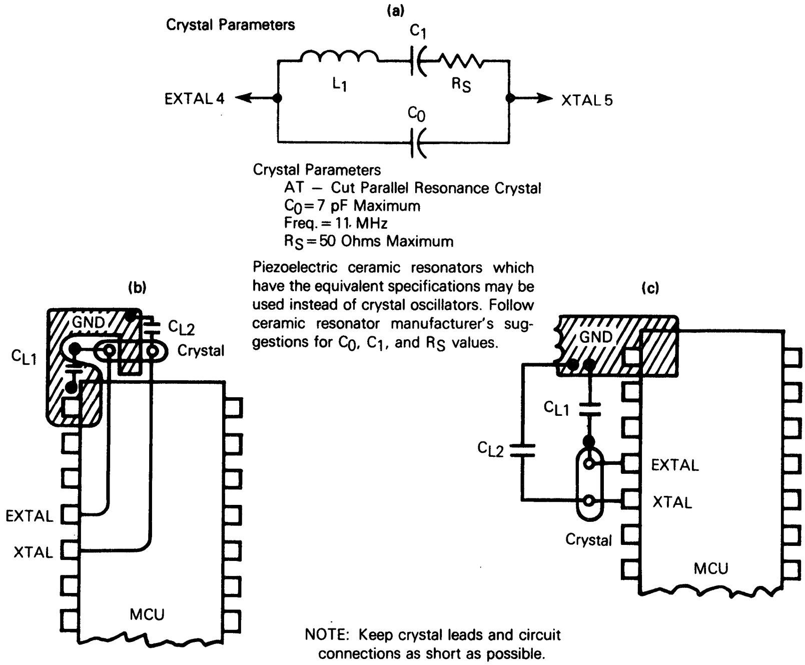


Crystal

External Clock

External Resistor-Capacitor

Figure 4-4. Clock Generator Options



Crystal Parameters
AT – Cut Parallel Resonance Crystal
$C_0 = 7$ pF Maximum
Freq. = 11. MHz
$R_S = 50$ Ohms Maximum

Piezoelectric ceramic resonators which have the equivalent specifications may be used instead of crystal oscillators. Follow ceramic resonator manufacturer's suggestions for C_0, C_1, and R_S values.

NOTE: Keep crystal leads and circuit connections as short as possible.

Figure 4-5. Crystal Motional Arm Parameters and Suggested PC Board Layout

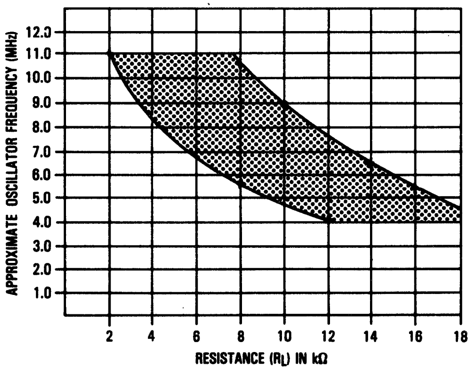
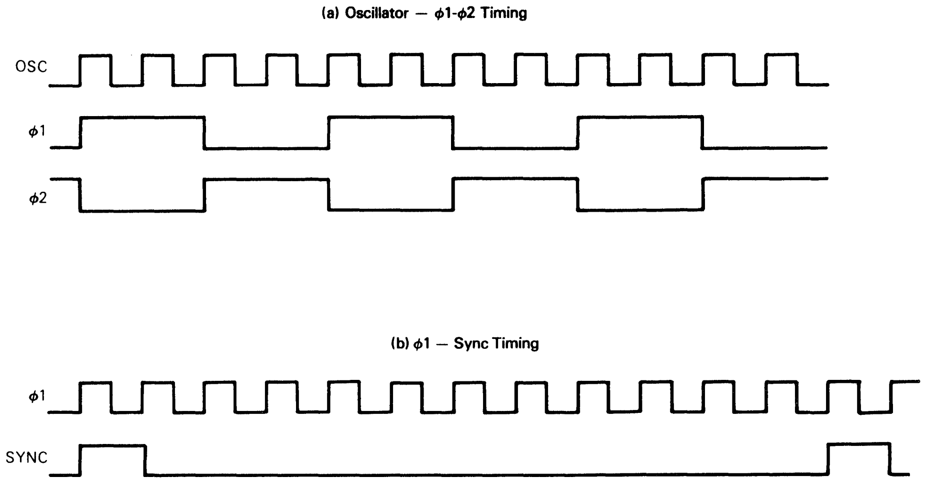


Figure 4-6. Typical Frequency Selection For Resistor-Capacitor Oscillator Option ($C_L = 17$ pF)



(a) Oscillator — $\phi 1$-$\phi 2$ Timing

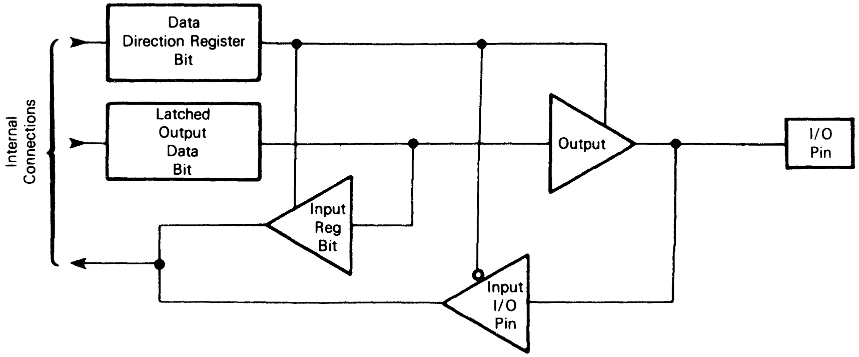
(b) $\phi 1$ — Sync Timing

Figure 4-7. Clock Generator Timing Diagram

SECTION 5
INPUT/OUTPUT PORTS

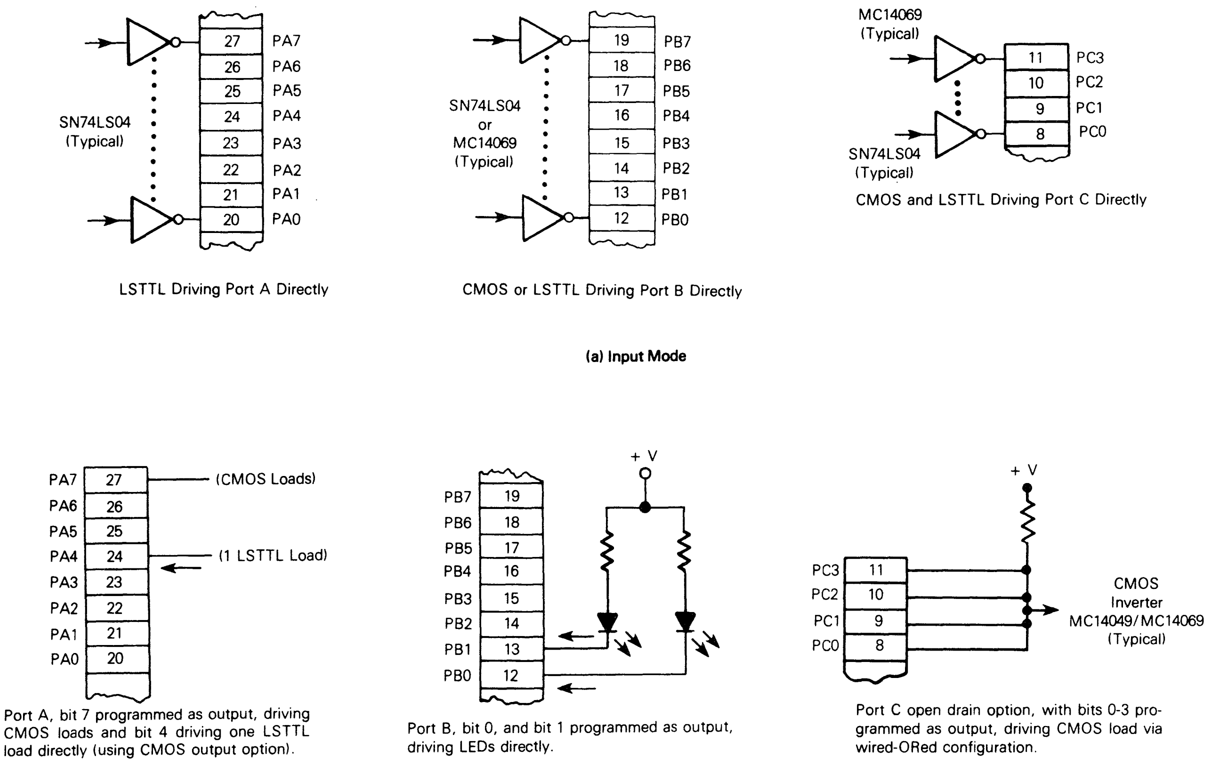
5.1 INPUT/OUTPUT

There are 20 input/output pins. All pins (port A, B, and C) are programmable as either inputs or outputs under software control of the corresponding data direction register (DDR). The port I/O programming is accomplished by writing the corresponding bit in the port DDR to a logic one for output or a logic zero for input. On reset, all the DDRs are initialized to a logic zero state to put the ports in the input mode. The port output registers are not initialized on reset but should be initialized before changing the DDR bits to avoid undefined levels. When programmed as outputs, the latched output data is readable as input data, regardless of the logic levels at the output pin due to output loading; see Figure 5-1. All input/output pins are LSTTL compatible as both inputs and outputs. In addition, all three ports may have one of two mask options: 1) internal pullup resistor for CMOS output compatibility, or 2) open drain output. The address map in Figure 2-1 gives the address of data registers and DDRs. The register configuration is discussed under the registers paragraph below and Figure 5-2 provides some examples of port connections.



Data Direction Register Bit	Output Data Bit	Output State	Input To MCU
1	0	0	0
1	1	1	1
0	X	Hi-Z	Pin

Figure 5-1. Typical I/O Port Circuitry



LSTTL Driving Port A Directly

CMOS or LSTTL Driving Port B Directly

CMOS and LSTTL Driving Port C Directly

(a) Input Mode

Port A, bit 7 programmed as output, driving CMOS loads and bit 4 driving one LSTTL load directly (using CMOS output option).

Port B, bit 0, and bit 1 programmed as output, driving LEDs directly.

Port C open drain option, with bits 0-3 programmed as output, driving CMOS load via wired-ORed configuration.

(b) Output Mode

Figure 5-2. Typical Port Connections

The latched output data bit (see Figure 5-1) may always be written. Therefore, any write to a port writes to all of its data bits even though the port DDR is set to input. This may be used to initialize the data registers and avoid undefined outputs; however, care must be exercised when using read-modify-write instructions since the data read corresponds to the pin level if the DDR is an input (0) and corresponds to the latched output data when the DDR is an output (1). The 20 bidirectional lines may be configured by port to be LSTTL (standard configuration), LSTTL/CMOS (mask option), or open drain (mask option). Port B outputs are LED compatible.

NOTE

The mask option only allows changes by port. For example, if the customer wishes PA7 to be open drain, then PA0-PA7 must all be open drain.

5.2 REGISTERS

The registers described below are implemented as RAM locations and thus may be read or written.

5.2.1 Port Data Register

7	0
MSB	LSB

Port A Address = $00
Port B Address = $01
Port C Address = $02 (Bits 0-3)

The source of data read from the port data register will be the port I/O pin or previously latched output data depending upon the contents of the corresponding data direction register (DDR). The destination of data written to the port data register will be an output data latch. If the corresponding data direction register (DDR) for the port I/O pin is programmed as an output, the data will then appear on the port pin.

5.2.2 Port Data Direction Register

7	0
MSB	LSB

Port A Address = $04
Port B Address = $05
Port C Address = $06 (Bits 0-3)

The port DDRs configure the port pins as either inputs or outputs. Each port pin can be programmed individually to act as an input or an output. A zero in the pins corresponding bit position will program that pin as an input while a one in the pins corresponding bit position will program that pin as an output.

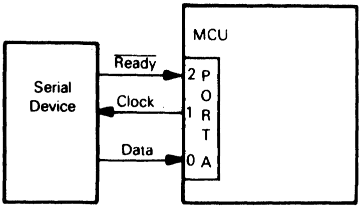
SECTION 6
SOFTWARE AND INSTRUCTION SET

6.1 SOFTWARE

6.1.1 Bit Manipulation

The EF6804P2 MCU has the ability to set or clear any register or single random access memory (RAM) writable bit with a single instruction (BSET, BCLR). Any bit in data space, including ROM, can be tested, using the BRSET and BRCLR instructions, and the program may branch as a result of its state. The carry bit equals the value of the bit referenced by BRSET or BRCLR. A rotate instruction may then be used to accumulate serial input data in a RAM location or register. The capability to work with any bit in RAM, ROM, or I/O allows the user to have individual flags in RAM or to handle I/O bits as control lines.

The coding example in Figure 6-1 illustrates the usefulness of the bit manipulation and test instructions. Assume that the MCU is to communicate with an external serial device. The external device has a data ready signal, a data output line, and a clock line (to clock data one bit at a time, MSB first, out of the device). The MCU waits until the data is ready, clocks the external device, picks up the data in the carry flag (C bit), clears the clock line, and finally accumulates the data bit in the accumulator.



```
                   .
                   .
                   .
SELF  BRSET  2,PORTA,SELF
                   .
                   .
                   .
      BSET   1,PORTA
      BRCLR  0,PORTA,CONT
CONT  BCLR   1,PORTA
      ROLA         .
                   .
                   .
                   .
```

Figure 6-1. Bit Manipulation Example

6.1.2 Addressing Modes

The EF6804P2 MCU has nine addressing modes which are explained briefly in the following paragraphs. The EF6804P2 deals with objects in three different address spaces : program space, data space, and stack space. Program space contains the instructions which are to be executed, plus the data for immediate mode instructions. Data space contains all of the RAM locations, XP and YP registers, accumulator, timer, I/O locations, and some ROM (for storage of tables and constants). Stack space contains RAM for use in stacking the return addresses for subroutines and interrupts.

The term "Effective Address" (EA) is used in describing the address modes. EA is defined as the address from which the argument for an instruction is fetched or stored.

6.1.2.1 IMMEDIATE. In the immediate addressing mode, the operand is located in program ROM and is contained in a byte following the opcode. The immediate addressing mode is used to access constants which do not change during program execution (e.g., a constant used to initialize a loop counter).

6.1.2.2 DIRECT. In the direct addressing mode, the effective address of the argument is contained in a single byte following the opcode byte. Direct addressing allows the user to directly address the 256 bytes in data space memory with a single two-byte instruction.

6.1.2.3 SHORT DIRECT. The MCU also has four locations in data space RAM ($80, $81, $82, $83) which may be used in a short-direct addressing mode. In this mode the opcode determines the data space RAM location, and the instruction is only one byte. Short direct addressing is a subset of the direct addressing mode. (The XP and YP registers are at locations $80 and $81 respectively.)

6.1.2.4 EXTENDED. In the extended addressing mode, the effective address is obtained by concatenating the four least significant bits of the opcode with the byte following the opcode (12-bit address). Instructions using the extended addressing mode (JMP, JSR) are capable of branching anywhere in program space. An extended addressing mode instruction is two bytes long.

6.1.2.5 RELATIVE. The relative addressing mode is only used in conditional branch instructions. In relative addressing, that address is formed by adding the sign extended lower five bits of the opcode (the offset) to the program counter if and only if the condition is true. Otherwise, control proceeds to the next instruction. The span of relative addressing is from − 15 to + 16 from the opcode address. The programmer need not worry about calculating the correct offset when using the assembler since it calculates the proper offset and checks to see if it is within the span of the branch.

6.1.2.6 BIT SET/CLEAR. In the bit set/clear addressing mode, the bit to be set or cleared is part of the opcode, and the byte following the opcode specifies the direct address of the byte in which the specified bit is to be set or cleared. Thus, any bit in the 256 locations of data space memory, which can be written to, can be set or cleared.

6.1.2.7 BIT TEST AND BRANCH. The bit test and branch addressing mode is a combination of direct addressing and relative addressing. The bit and condition (set or clear) which is to be tested is included in the opcode, and the data space address of the byte to be tested is in the single byte immediately following the opcode byte. The third byte is sign extended to twelve bits and becomes the offset added to the program counter if the condition is true. The single three-byte instruction allows the program to branch based on the condition of any bit in data space memory. The span of branching is from – 125 to + 130 from the opcode address. The state of the tested bit is also transferred to the carry flag.

6.1.2.8 REGISTER-INDIRECT. In the register-indirect addressing mode, the operand is at the address (in data space) pointed to by the contents of one of the indirect registers (X or Y). The particular X or Y register is selected by bit 4 of the opcode. Bit 4 of the opcode is then decoded into an address which selects the desired X or Y register ($80 or $81). A register-indirect instruction is one byte long.

6.1.2.9 INHERENT. In the inherent addressing mode, all the information necessary to execute the instruction is contained in the opcode. These instructions are one byte long.

6.2 INSTRUCTION SET

The EF6804P2 MCU has a set of 42 basic instructions which when combined with nine addressing modes produce 242 usable opcodes. They can be divided into five different types: register/memory, read-modify-write, branch, bit manipulation, and control. The following paragraphs briefly explain each type. All the instructions within a given type are presented in individual tables.

6.2.1 Register/Memory Instructions

Most of these instructions use two operands. One operand is the accumulator and the other operand is obtained from memory using one of the addressing modes. The jump unconditional (JMP) and jump to subroutine (JSR) instructions have no register operands. Refer to Table 6-1.

6.2.2 Read-Modify-Write Instructions

These instructions read a memory location or a register, modify or test its contents, and write the modified value back to memory or to the register. There are ten instructions which utilize read-modify-write cycles. All INC and DEC forms along with all bit manipulation instructions use this method. Refer to Table 6-2.

6.2.3 Branch Instructions

The branch instructions cause a branch from the program when a certain condition is met. Refer to Table 6-3.

6.2.4 Bit Manipulation Instructions

These instructions are used on any bit in data space memory. One group either sets or clears. The other group performs the bit test branch operations. Refer to Table 6-4.

Table 6-1. Register/Memory Instructions

		Addressing Modes																			
		Indirect				Immediate			Direct			Inherent			Extended			Short-Direct			
		Opcode																			
Function	Mnem	XP	YP	# Bytes	# Cycles	Opcode	# Bytes	# Cycles	Opcode	# Bytes	# Cycles	Opcode	# Bytes	# Cycles	Opcode	# Bytes	# Cycles	Opcode	# Bytes	# Cycles	Special Notes
Load A from Memory	LDA	E0	F0	1	4	E8	2	4	F8	2	4	—	—	—	—	—	—	AC-AF	1	4	1
Load XP from Memory	LDXI	—	—	—	—	B0	3	4	—	—	—	—	—	—	—	—	—	—	—	—	4
Load YP from Memory	LDYI	—	—	—	—	B0	3	4	—	—	—	—	—	—	—	—	—	—	—	—	4
Store A in Memory	STA	E1	F1	1	4	—	—	—	F9	2	4	—	—	—	—	—	—	BC-BF	1	4	2
Add to A	ADD	E2	F2	1	4	EA	2	4	FA	2	4	—	—	—	—	—	—	—	—	—	—
Subtract from A	SUB	E3	F3	1	4	EB	2	4	FB	2	4	—	—	—	—	—	—	—	—	—	—
Arithmetic Compare with Memory	CMP	E4	F4	1	4	EC	2	4	FC	2	4	—	—	—	—	—	—	—	—	—	—
AND Memory to A	AND	E5	F5	1	4	ED	2	4	FD	2	4	—	—	—	—	—	—	—	—	—	—
Jump to Subroutine	JSR	—	—	—	—	—	—	—	—	—	—	—	—	—	8 (TAR)	2	4	—	—	—	3
Jump Unconditional	JMP	—	—	—	—	—	—	—	—	—	—	—	—	—	9 (TAR)	2	4	—	—	—	3
Clear A	CLRA	—	—	—	—	—	—	—	FB	2	4	—	—	—	—	—	—	—	—	—	—
Clear XP	CLRX	—	—	—	—	—	—	—	FB	2	4	—	—	—	—	—	—	—	—	—	—
Clear YP	CLRY	—	—	—	—	—	—	—	FB	2	4	—			—	—	—	—	—	—	—
Complement A	COMA	—	—	—	—	—	—	—	—	—	—	B4	1	4	—	—	—	—	—	—	—
Move Immediate Value to Memory	MVI	—	—	—	—	B0	3	4	B0	3	4	—	—	—	—	—	—	—	—	—	5
Rotate A Left and Carry	ROLA	—	—	—	—	—	—	—	—	—	—	B5	1	4	—	—	—	—	—	—	—
Arithmetic Left Shift of A	ASLA	—	—	—	—	—	—	—	FA	2	4	—	—	—	—	—	—	—	—	—	—

SPECIAL NOTES

1. In Short-Direct addressing, the LDA mnemonic represents opcode AC, AD, AE, and AF. This is equivalent to RAM locations $80 (AC), $81 (AD), $82 (AE), and $83 (AF)
2. In Short-Direct addressing, the STA mnemonic represents opcode BC, BD, BE, and BF. This is equivalent to RAM locations $80 (BC), $81 (BD), $82 (BE), and $83 (BF).
3. In Extended addressing, the four LSBs of the opcode (Mnemonic JSR and JMP) are formed by the four MSBs of the target address. (TAR)
4. In Immediate addressing, the LDXI and LDYI are mnemonics which are recognized as follows:

   LDXI = MVI $80,data
   LDYI = MVI $81,data Where data is a one-byte hexadecimal number.
5. The MVI instruction refers to both Immediate and Direct addressing.

Table 6-2. Read-Modify-Write Instructions

		Addressing Modes										
		Indirect				Direct			Short-Direct			Special Notes
		Opcode		#	#		#	#		#	#	
Function	Mnem	XP	YP	Bytes	Cycles	Opcode	Bytes	Cycles	Opcode	Bytes	Cycles	
Increment Memory Location	INC	E6	F6	1	4	FE	2	4	A8-AB	1	4	1, 3
Increment A	INCA	—	—	—	—	FE	2	4	—	—	—	—
Increment XP	INCX	—	—	—	—	—	—	—	A8	1	4	—
Increment YP	INCY	—	—	—	—	—	—	—	A9	1	4	—
Decrement Memory Location	DEC	E7	F7	1	4	FF	2	4	B8-BB	1	4	2, 4
Decrement A	DECA	—	—	—	—	FF	2	4	—	—	—	—
Decrement XP	DECX	—	—	—	—	—	—	—	B8	1	4	—
Decrement YP	DECY	—	—	—	—	—	—	—	B9	1	4	—

SPECIAL NOTES

1. In Short-Direct addressing, the INC mnemonic represents opcode A8, A9, AA, and AB. These are equivalent to RAM locations $80 (A8), $81 (A9), $82 (AA), and $83 (AB).
2. In Short-Direct addressing, the DEC mnemonic represents opcode B8, B9, BA, and BB. These are equivalent to RAM locations $80 (B8), $81 (B9), $82 (BA), and $83 (BB).
3. In Indirect addressing, the INC mnemonic represents opcode E6 or F6, and causes the location pointed to by XP (E6 opcode) or YP (F6 opcode) to be incremented.
4. In Indirect addressing, the INC mnemonic represents opcode E7 or F7, and causes the location pointed to by XP (E7 opcode) or YP (F7 opcode) to be incremented.

Table 6-3. Branch Instructions

		Relative Addressing Mode			
Function	Mnem	Opcode	# Bytes	# Cycles	Special Notes
Branch if Carry Clear	BCC	40-5F	1	2	1
Branch if Higher or Same	(BHS)	40-5F	1	2	1, 2
Branch if Carry Set	BCS	60-7F	1	2	1
Branch if Lower	(BLO)	60-7F	1	2	1, 3
Branch if Not Equal	BNE	00-1F	1	2	1
Branch if Equal	BEQ	20-3F	1	2	1

SPECIAL NOTES

1. Each mnemonic of the Branch Instructions covers a range of 32 opcodes; e.g., BCC ranges from 40 through 5F. The actual memory location (target address) to which the branch is made is formed by adding the sign extended lower five bits of the opcode to the contents of the program counter.
2. The BHS instruction (shown in parentheses) is identical to the BCC instruction. The C bit is clear if the register was higher or the same as the location in the memory to which it was compared.
3. The BLO instruction (shown in parentheses) is identical to the BCS instruction. The C bit is set if the register was lower than the location in memory to which it was compared.

Table 6-4. Bit Manipulation Instructions

		Addressing Modes						
		Bit Set/Clear			Bit Test and Branch			
Function	Mnem	Opcode	# Bytes	# Cycles	Opcode	# Bytes	# Cycles	Special Note
Branch IFF Bit n is set	BRSET n (n=0 . . . . 7)	–	–	–	C8+n	3	5	1
Branch IFF Bit n is clear	BRCLR n (n=0 . . . . 7)	–	–	–	C0+n	3	5	1
Set Bit n	BSET n (n=0 . . . .7)	D8+n	2	4	–	–	–	1
Clear Bit n	BCLR n (n=0 . . . . 7)	D0+n	2	4	–	–	–	1

SPECIAL NOTE

1. The opcode is formed by adding the bit number (0-7) to the basic opcode. For example: to clear bit six using the BSET6 instruction the opcode becomes DE (D8+6); BCLR5 becomes (C0+5); etc.

6.2.5 Control Instructions

The control instructions control the MCU operations during program execution. Refer to Table 6-5.

6.2.6 Alphabetical Listing

The complete instruction set is given in alphabetical order in Table 6-6. There are certain mnemonics recognized by the assember and converted to other instructions. The fact that all registers and accumulator are in RAM allows many implied instructions to exist. The implied instructions recognized by the assembler are identified in Table 6-6.

6.2.7 Opcode Map Summary

Table 6-7 contains an opcode map for the instructions used on the MCU.

6.3 IMPLIED INSTRUCTIONS

Since the accumulator and all other registers are located in RAM many implied instructions exist. The assembler-recognized implied instructions are given in Table 6-6. Some examples not recognized by the assembler are shown below.

BCLR,7 $FF	Ensures accumulator is plus
BSET,7 $FF	Ensures accumulator is minus
BRCLR,7 $FF	Branch iff accumulator is plus
BRSET,7 $FF	Branch iff accumulator is minus
BRCLR,7 $80	Branch iff X is plus (BXPL)
BRSET,7 $80	Branch iff X is minus (BXMI)
BRCLR,7 $81	Branch iff Y is plus (BYPL)
BRSET,7 $81	Branch iff Y is minus (BYMI)

Table 6-5. Control Instructions

		Addressing Modes									
		Short-Direct			Inherent			Relative			
Function	Mnem	Opcode	# Bytes	# Cycles	Opcode	# Bytes	# Cycles	Opcode	# Bytes	# Cycles	Special Notes
Transfer A to XP	TAX	BC	1	4	—	—	—	—	—	—	—
Transfer A to YP	TAY	BD	1	4	—	—	—	—	—	—	—
Transfer XP to A	TXA	AC	1	4	—	—	—	—	—	—	—
Transfer YP to A	TYA	AD	1	4	—	—	—	—	—	—	—
Return from Subroutine	RTS	—	—	—	B3	1	2	—	—	—	—
Return from Interrupt	RTI	—	—	—	B2	1	2	—	—	—	—
No-Operation	NOP	—	—	—	—	—	—	—	—	—	1

SPECIAL NOTE

1 . The NOP instruction is equivalent to a branch if equal (BEQ) to the location designated by PC+1.

Table 6-6. Instruction Set

	Addressing Modes									Flags	
Mnemonic	Inherent	Immediate	Direct	Short Direct	Bit/Set Clear	Bit-Test-Branch	Register Indirect	Extended	Relative	Z	C
ADD		X	X				X			∧	∧
AND		X	X				X			∧	•
ASLA			Assembler converts this to "ADD $FF"							•	•
BCC									X	∧	∧
BCLR					X					•	•
BCS									X	•	•
BEQ									X	•	•
BHS			Assembler converts this to "BCC"							•	•
BLO			Assembler converts this to "BCS"							•	•
BNE									X	•	•
BRCLR						X				•	∧
BRSET						X				•	∧
BSET					X					•	•
CLRA			Assembler converts this to "SUB $FF"							∧	∧
CLRX			Assembler converts this to "MVI # 0, $80"							•	•
CLRY			Assembler converts this to "MVI # 0, $81"							•	•
CMP		X	X				X			∧	∧
COMA	X									∧	∧
DEC			X	X			X			∧	•
DECA			Assembler converts this to "DEC $FF"							∧	•
DECX			Assembler converts this to "DEC $80"							∧	•
DECY			Assembler converts this to "DEC $81"							∧	•
INC			X	X			X			∧	•
INCA			Assembler converts this to "INC $FF"							∧	•
INCX			Assembler converts this to "INC $80"							∧	•
INCY			Assembler converts this to "INC $81"							∧	•
JMP								X		•	•
JSR								X		•	•
LDA		X	X	X			X			∧	•
LDXI			Assembler converts this to "MVI DATA, $80"							•	•
LDYI			Assembler converts this to "MVI DATA, $81"							•	•
MVI		X	X							•	•
NOP			Assembler converts this to "BEQ (PC) + 1"							•	•
ROLA	X									∧	∧
RTI	X									∧	∧
RTS	X									•	•
STA			X	X			X			∧	•
SUB		X	X				X			∧	∧
TAX	Assembler converts this to "STA $80"									∧	•
TAY	Assembler converts this to "STA $81"									∧	•
TXA	Assembler converts this to "LDA $80"									∧	•
TYA	Assembler converts this to "LDA $81"									∧	•

Flag Symbols: Z = Zero, C = Carry/Borrow, ∧ = Test and Set if True, Cleared Otherwise, • = Not Affected

Table 6-7 . EF6804P2 Microcomputer

Hi Low	Branch Instructions							
	0 0000	1 0001	2 0010	3 0011	4 0100	5 0101	6 0110	7 0111
0 0000	2 BNE 1 REL	2 BNE 1 REL	2 BEQ 1 REL	2 BEQ 1 REL	2 BCC 1 REL	2 BCC 1 REL	2 BCS 1 REL	2 BCS 1 REL
1 0001	2 BNE 1 REL	2 BNE 1 REL	2 BEQ 1 REL	2 BEQ 1 REL	2 BCC 1 REL	2 BCC 1 REL	2 BCS 1 REL	2 BCS 1 REL
2 0010	2 BNE 1 REL	2 BNE 1 REL	2 BEQ 1 REL	2 BEQ 1 REL	2 BCC 1 REL	2 BCC 1 REL	2 BCS 1 REL	2 BCS 1 REL
3 0011	2 BNE 1 REL	2 BNE 1 REL	2 BEQ 1 REL	2 BEQ 1 REL	2 BCC 1 REL	2 BCC 1 REL	2 BCS 1 REL	2 BCS 1 REL
4 0100	2 BNE 1 REL	2 BNE 1 REL	2 BEQ 1 REL	2 BEQ 1 REL	2 BCC 1 REL	2 BCC 1 REL	2 BCS 1 REL	2 BCS 1 REL
5 0101	2 BNE 1 REL	2 BNE 1 REL	2 BEQ 1 REL	2 BEQ 1 REL	2 BCC 1 REL	2 BCC 1 REL	2 BCS 1 REL	2 BCS 1 REL
6 0110	2 BNE 1 REL	2 BNE 1 REL	2 BEQ 1 REL	2 BEQ 1 REL	2 BCC 1 REL	2 BCC 1 REL	2 BCS 1 REL	2 BCS 1 REL
7 0111	2 BNE 1 REL	2 BNE 1 REL	2 BEQ 1 REL	2 BEQ 1 REL	2 BCC 1 REL	2 BCC 1 REL	2 BCS 1 REL	2 BCS 1 REL
8 1000	2 BNE 1 REL	2 BNE 1 REL	2 BEQ 1 REL	2 BEQ 1 REL	2 BCC 1 REL	2 BCC 1 REL	2 BCS 1 REL	2 BCS 1 REL
9 1001	2 BNE 1 REL	2 BNE 1 REL	2 BEQ 1 REL	2 BEQ 1 REL	2 BCC 1 REL	2 BCC 1 REL	2 BCS 1 REL	2 BCS 1 REL
A 1010	2 BNE 1 REL	2 BNE 2 REL	2 BEQ 1 REL	2 BEQ 1 REL	2 BCC 1 REL	2 BCC 1 REL	2 BCS 1 REL	2 BCS 1 REL
B 1011	2 BNE 1 REL	2 BNE 1 REL	2 BEQ 1 REL	2 BEQ 1 REL	2 BCC 1 REL	2 BCC 1 REL	2 BCS 1 REL	2 BCS 1 REL
C 1100	2 BNE 1 REL	2 BNE 1 REL	2 BEQ 1 REL	2 BEQ 1 REL	2 BCC 1 REL	2 BCC 1 REL	2 BCS 1 REL	2 BCS 1 REL
D 1101	2 BNE 1 REL	2 BNE 1 REL	2 BEQ 1 REL	2 BEQ 1 REL	2 BCC 1 REL	2 BCC 1 REL	2 BCS 1 REL	2 BCS 1 REL
E 1110	2 BNE 1 REL	2 BNE 1 REL	2 BEQ 1 REL	2 BEQ 1 REL	2 BCC 1 REL	2 BCC 1 REL	2 BCS 1 REL	2 BCS 1 REL
F 1111	2 BNE 1 REL	2 BNE 1 REL	2 BEQ 1 REL	2 BEQ 1 REL	2 BCC 1 REL	2 BCC 1 REL	2 BCS 1 REL	2 BCS 1 REL

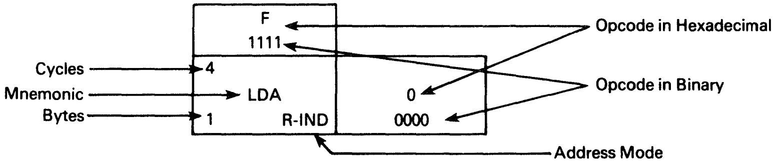
Abbreviations for Address Modes

INH	Inherent	*	Indicates Instruction Reserved for Future Use
S-D	Short Direct	#	Indicates Illegal Instruction
B-T-B	Bit Test and Branch		
IMM	Immediate		
DIR	Direct		
EXT	Extended		
REL	Relative		
BSC	Bit Set/Clear		
R-IND	Register Indirect		

Instruction Set Opcode Map

Register/Memory, Control, and Read/Modify/Write Instructions				Bit Manipulation Instructions		Register/Memory and Read/Modify/Write		
8 1000	9 1001	A 1010	B 1011	C 1100	D 1101	E 1110	F 1111	Hi Low
4 JSRn 2 EXT	4 JMPn 2 EXT	*	4 MVI 3 IMM	5 BRCLR0 3 B-T-B	4 BCLR0 2 BSC	4 LDA 1 R-IND	4 LDA 1 R-IND	0 0000
4 JSRn 2 EXT	4 JMPn 2 EXT	*	*	5 BRCLR1 3 B-T-B	4 BCLR1 2 BSC	4 STA 1 R-IND	4 STA 1 R-IND	1 0001
4 JSRn 2 EXT	4 JMPn 2 EXT	*	2 RTI 1 INH	5 BRCLR2 3 BT-B	4 BCLR2 2 BSC	4 ADD 1 R-IND	4 ADD 1 R-IND	2 0010
4 JSRn 2 EXT	4 JMPn 2 EXT	*	2 RTS 1 INH	5 BRCLR3 3 B-T-B	4 BCLR3 2 BSC	4 SUB 1 R-IND	4 SUB 1 R-IND	3 0011
4 JSRn 2 EXT	4 JMPn 2 EXT	*	4 COMA 1 INH	5 BRCLR4 3 B-T-B	4 BCLR4 2 BSC	4 CMP 1 R-IND	4 CMP 1 R-IND	4 0100
4 JSRn 2 EXT	4 JMPn 2 EXT	*	4 ROLA 1 INH	5 BRCLR5 3 B-T-B	4 BCLR5 2 BSC	4 AND 1 R-IND	4 AND 1 R-IND	5 0101
4 JSRn 2 EXT	4 JMPn 2 EXT	*	*	5 BRCLR6 3 B-T-B	4 BCLR6 2 BSC	4 INC 1 R-IND	4 INC 1 R-IND	6 0110
4 JSRn 2 EXT	4 JMPn 2 EXT	*	*	5 BRCLR7 3 B-T-B	4 BCLR7 2 BSC	4 DEC 1 R-IND	4 DEC 1 R-IND	7 0111
4 JSRn 2 EXT	4 JMPn 2 EXT	4 INC 1 S-D	4 DEC 1 S-D	5 BRSET0 3 B-T-B	4 BSET0 2 BSC	4 LDA 2 IMM	4 LDA 2 DIR	8 1000
4 JSRn 2 EXT	4 JMPn 2 EXT	4 INC 1 S-D	4 DEC 1 S-D	5 BRSET1 3 B-T-B	4 BSET1 2 BSC	#	4 STA 2 DIR	9 1001
4 JSRn 2 EXT	4 JMPn 2 EXT	4 INC 1 S-D	4 DEC 1 S-D	5 BRSET2 3 B-T-B	4 BSET2 2 BSC	4 ADD 2 IMM	4 ADD 2 DIR	A 1010
4 JSRn 2 EXT	4 JMPn 2 EXT	4 INC 1 S-D	4 DEC 1 S-D	5 BRSET3 3 B-T-B	4 BSET3 2 BSC	4 SUB 2 IMM	4 SUB 2 DIR	B 1011
4 JSRn 2 EXT	4 JMPn 2 EXT	4 LDA 1 S-D	4 STA 1 S-D	5 BRSET4 3 B-T-B	4 BSET4 2 BSC	4 CMP 2 IMM	4 CMP 2 DIR	C 1100
4 JSRn 2 EXT	4 JMPn 2 EXT	4 LDA 1 S-D	4 STA 1 S-D	5 BRSET5 3 B-T-B	4 BSET5 2 BSC	4 AND 2 IMM	4 AND 2 DIR	D 1101
4 JRSn 2 EXT	4 JMPn 2 EXT	4 LDA 1 S-D	4 STA 1 S-D	5 BRSET6 3 B-T-B	4 BSET6 2 BSC	#	4 INC 2 DIR	E 1110
4 JSRn 2 EXT	4 JMPn 2 EXT	4 LDA 1 S-D	4 STA 1 S-D	5 BRSET7 3 B-T-B	4 BSET7 2 BSC	#	4 DEC 2 DIR	F 1111

LEGEND



SECTION 7
ELECTRICAL SPECIFICATIONS

7.1 INTRODUCTION

This section contains the electrical specifications and associated timing for the EF6804P2

7.2 MAXIMUM RATINGS

Rating	Symbol	Value	Unit
Supply Voltage	V_{CC}	− 0.3 to + 7.0	V
Input Voltage	V_{in}	− 0.3 to + 7.0	V
Operating Temperature Range Standard or L suffix V suffix T suffix	T_A	TL to TH 0 to 70 -40 to 85 -40 to 105	°C
Storage Temperature Range	T_{stg}	− 55 to 150	°C
Junction Temperature Range Plastic PLCC	Tj	 150 150	°C

This device contains circuitry to protect the inputs against damage due to high static voltages of electric fields; however, it is advised that normal precautions be taken to avoid application of any voltage higher than maximum rated voltages to this high impedance circuit. For proper operation it is recommended that V_{in} and V_{out} be constrained to the range $V_{SS} \leq (V_{in}$ or $V_{out}) \leq V_{CC}$. Reliability of operation is enhanced if unused inputs except EXTAL are connected to an appropriate logic voltage level (e.g., either V_{SS} or V_{CC}).

7.3 THERMAL CHARACTERISTICS

Characteristic	Symbol	Value	Unit
Thermal Resistance Plastic PLCC	θ JA	 70 90	°C/W

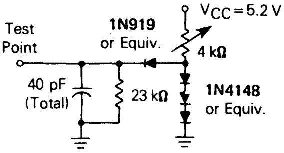


Figure 7-1. LSTTL Equivalent Test Load (Port B)

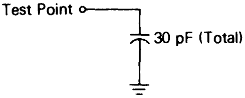


Figure 7-2. CMOS Equivalent Test Load (Ports A, B, C)

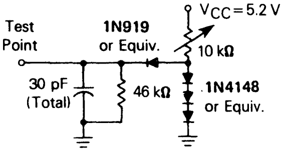


Figure 7-3. LSTTL Equivalent Test Load (Ports A, C, and TIMER)

7.4 POWER CONSIDERATIONS

The average chip-junction temperature, T_J, in °C can be obtained from:

$$T_J = T_A + (P_D \cdot \theta_{JA}) \quad (1)$$

Where:

T_A = Ambient Temperature, °C

θ_{JA} = Package Thermal Resistance, Junction-to-Ambient, °C/W

$P_D = P_{INT} + P_{PORT}$

$P_{INT} = I_{CC} \times V_{CC}$, Watts – Chip Internal Power

P_{PORT} = Port Power Dissipation, Watts – User Determined

For most applications $P_{PORT} \ll P_{INT}$ and can be neglected. P_{PORT} may become significant if the device is configured to drive Darlington bases or sink LED loads.

An approximate relationship between P_D and T_J (if P_{PORT} is neglected) is:

$$P_D = K \div (T_J + 273°C) \quad (2)$$

Solving equations 1 and 2 for K gives:

$$K = P_D \cdot (T_A + 273°C) + \theta_{JA} \cdot P_D^2 \quad (3)$$

Where K is a constant pertaining to the particular part. K can be determined from equation 3 by measuring P_D (at equilibrium) for a known T_A. Using this value of K the values of P_D and T_J can be obtained by solving equations (1) and (2) iteratively for any value of T_A.

7.5 ELECTRICAL CHARACTERISTICS (V_{CC} = +5.0 Vdc ± 0.5 Vdc, V_{SS} = 0 Vdc, $T_A = T_L$ to T_H unless otherwise noted)

Characteristic		Symbol	Min	Typ	Max	Unit
Internal Power Dissipation – No Port Loading	T_A = 0°C T_A = -40°C	P_{INT}	– –	135 –	170 210	mV
Input High Voltage		V_{IH}	4.0	–	V_{CC}	V
Input Low Voltage		V_{IL}	V_{SS}	–	0.8	V
Input Capacitance		C_{in}	–	10	–	pF
Input Current ($\overline{IRQ}$, $\overline{RESET}$)		I_{in}	–	2	20	μA

7.6 SWITCHING CHARACTERISTICS (V_{CC} = +5.0 Vdc ± 0.5 Vdc, V_{SS} = GND, $T_A = T_L$ to T_H unless otherwise noted)

Characteristic	Symbol	Min	Typ	Max	Unit
Oscillator Frequency	f_{OSC}	4.0	–	11.0	MHz
Bit Time	t_{bit}	0.364	–	1.0	μs
Byte Cycle Time	t_{byte}	4.36	–	12.0	μs
$\overline{IRQ}$ and TIMER Pulse Width	t_{WL}, t_{WH}	$2 \times t_{byte}$	–	–	–
$\overline{RESET}$ Pulse Width	t_{RWL}	$2 \times t_{byte}$	–	–	–
$\overline{RESET}$ Delay Time (External Capacitance = 1.0 μF)	t_{RHL}	100	–	–	ms

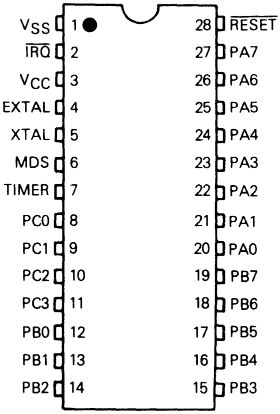
7.7 PORT DC ELECTRICAL CHARACTERISTICS (V_{CC} = +5.0 Vdc ± 0.5 Vdc, V_{SS} = GND, $T_A = T_L$ to T_H unless otherwise noted)

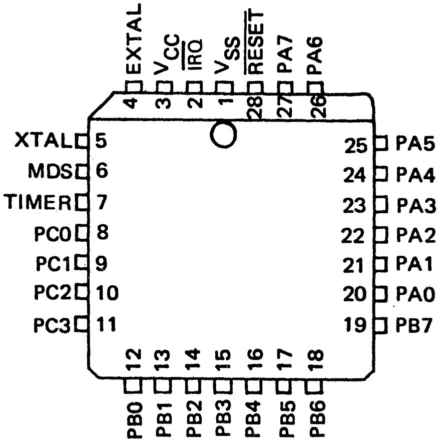
Characteristic	Symbol	Min	Typ	Max	Unit
Timer and Ports A and C (Standard)					
Output Low Voltage, I_{Load} = 0.4 mA	V_{OL}	–	–	0.5	V
Output High Voltage, I_{Load} = – 50 µA	V_{OH}	2.3	–	–	V
Input High Voltage	V_{IH}	2.0	–	V_{CC}	V
Input Low Voltage	V_{IL}	V_{SS}	–	0.8	V
Hi-Z State Input Current	I_{TSI}	–	4	40	µA
Ports A and C (Open Drain)					
Output Low Voltage, I_{Load} = 0.4 mA	V_{OL}	–	–	0.5	V
Input High Voltage	V_{IH}	2.0	–	V_{CC}	V
Input Low Voltage	V_{IL}	V_{SS}	–	0.8	V
Hi-Z State Input Current	I_{TSI}	–	4	40	µA
Open Drain Leakage (V_{out} = V_{CC})	I_{LOD}	–	4	40	µA
Ports A and C (CMOS Drive)					
Output Low Voltage, I_{Load} = 0.4 mA (Sink)	V_{OL}	–	–	0.5	V
Output High Voltage, I_{Load} = – 10 µA	V_{OH}	V_{CC} – 1.0	–	–	V
Output High Voltage, I_{Load} = – 50 µA	V_{OH}	2.3	–	–	V
Input High Voltage, I_{Load} = – 300 µA Max	V_{IH}	2.0	–	V_{CC}	V
Input Low Voltage, I_{Load} = – 300 µA Max	V_{IL}	V_{SS}	–	0.8	V
Hi-Z State Input Current (V_{in} = 0.4 V to V_{CC})	I_{TSI}	–	–	– 300	µA
Port B (Standard)					
Output Low Voltage, I_{Load} = 1.0 mA	V_{OL}	–	–	0.5	V
Output Low Voltage, I_{Load} = 10 mA (Sink)	V_{OL}	–	–	1.5	V
Output High Voltage, I_{Load} = – 100 µA	V_{OH}	2.3	–	–	V
Input High Voltage	V_{IH}	2.0	–	V_{CC}	V
Input Low Voltage	V_{IL}	V_{SS}	–	0.8	V
Hi-Z State Input Current	I_{TSI}	–	8	80	µA
Port B (Open Drain)					
Output Low Voltage, I_{Load} = 1.0 mA	V_{OL}	–	–	0.5	V
Output Low Voltage, I_{Load} = 10 mA (Sink)	V_{OL}	–	–	1.5	V
Input High Voltage	V_{IH}	2.0	–	V_{CC}	V
Input Low Voltage	V_{IL}	V_{SS}	–	0.8	V
Hi-Z State Input Current	I_{TSI}	–	8	80	µA
Open Drain Leakage (V_{out} = V_{CC})	I_{LOD}	–	8	80	µA
Port B (CMOS Drive)					
Output Low Voltage, I_{Load} = 1.0 mA	V_{OL}	–	–	0.5	V
Output High Voltage, I_{Load} = 10 mA (Sink)	V_{OL}	–	–	1.5	V
Output High Voltage, I_{Load} = – 10 µA	V_{OH}	V_{CC} – 1.0	–	–	V
Output High Voltage, I_{Load} = – 100 µA	V_{OH}	2.3	–	–	V
Input High Voltage, I_{Load} = – 300 µA Max	V_{IH}	2.0	–	V_{CC}	V
Input Low Voltage, I_{Load} = – 300 µA Max	V_{IL}	V_{SS}	–	0.8	V
Hi-Z State Input Current (V_{in} = 0.4 V to V_{CC})	I_{TSI}	–	–	– 300	µA

SECTION 8
MECHANICAL DATA

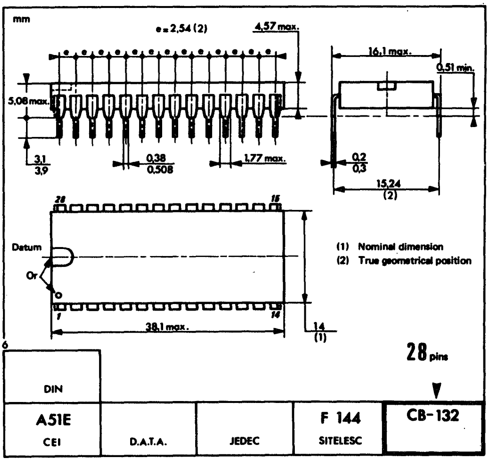
This section contains the pin assignment and package dimension diagrams for the EF6804P2 microcomputer.

8.1 PIN ASSIGNMENTS

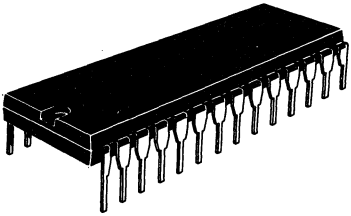




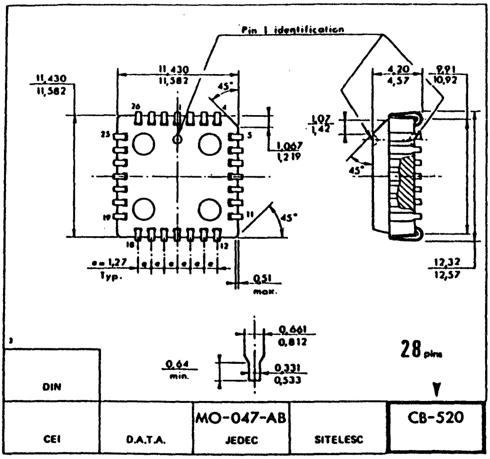
8.2 PHYSICAL DIMENSIONS



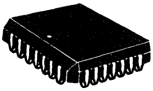
CB-132



P SUFFIX
PLASTIC PACKAGE



CB-520



FN SUFFIX
PLCC 28

SECTION 9
ORDERING INFORMATION

9.1 INTRODUCTION

The following information is required when ordering a custom MCU. The information may be transmitted to THOMSON SEMICONDUCTEURS in the following media:
EPROM(s), 2716 or 2732
EFDOS/MDOS* disk file

To initiate a ROM pattem for the MCU, it is necessary to first contact your local field service office, local sales person, or your local THOMSON SEMICONDUCTEURS representative.

9.1.1 EPROMs

One 2716 or one 2732 type EPROM, programmed with the customer program (positive logic sense for address and data), may be submitted for pattern generation. Since all program and data space information will fit on one 2716 EPROM, the EPROM must be programmed as follows in order to emulate the EF6804P2 MCU: **start the data space ROM at EPROM address $020 and start program space ROM at EPROM address $400 and continue to memory space $7FF. All unused bytes, including the user's space, must be set to zero, memory space $7F8 to $7FB is reserved for self-check vectors.** When using one 2732 EPROM, the memory map shown in Figure 2-1 can be used. For shipment to THOMSON SEMICONDUCTEURS the EPROMs should be placed in a conductive IC carrier and packed securely. Do not use styrofoam.

9.1.2 EFDOS/MDOS* Disk File

An EFDOS/MDOS* disk, programmed with the customer program (positive logic sense for address and data) may be submitted for pattern generation. **When using the EFDOS/MDOS* disk, include the entire memory image of both data and program space. All unused bytes, including the user's space, must be set to zero.**

9.2 VERIFICATION MEDIA

All original pattern media (EPROMs or floppy disk) are filed for contractual purposes and are not returned. A computer listing of the ROM code will be generated and returned along with a listing verification form. The listing should be thoroughly checked and the verification form completed, signed, and returned to THOMSON SEMICONDUCTEURS. The signed verification form constitutes the contracural agreement for creation of the customer mask. If desired, THOMSON SEMICONDUCTEURS will program a blank 2716, 2732, or EFDOS/MDOS* disk **(supplied by the customer)** from the data file used to create the custom mask to aid in the verification process.

* Requires prior factory approval

9.3 ROM VERIFICATION UNITS (RVUs)

Ten MCUs containing the customer's ROM pattern will be sent for program verification. These units will have been made using the custom mask but are for the purpose of ROM verification only. For expediency they are usually unmarked, packaged in ceramic, and tested only at room temperature and five volts. These RVUs are included in the mask charge and are not production parts. These RVUs are not backed nor guaranteed by THOMSON SEMICONDUCTEURS Quality Assurance.

9.4 FLEXIBLE DISKS

The disk media submitted must be single-sided, single density, 8-inch, EFDOS/MDOS* compatible floppies. The customer must clearly label the disk with the ROM pattern file name. The minimum EFDOS/MDOS* system files as well as the absolute binary object file (file name. LO type of file) from the 6804 cross-assembler must be on the disk. An object file made from a memory dump, using the ROLLOUT command is also admissable. Consider submitting a source listing as well as: file name, .LX (DEVICE/EXORciser loadable format). This file will of course be kept confidential and is used 1) to speed up the process in house if any problems arises, and 2) to speed up our customer to factory interface if an user finds any software errors and needs assistance quickly from THOMSON SEMICONDUCTEURS factory representatives.

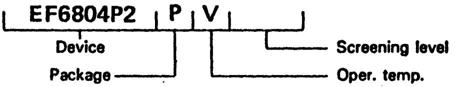
MDOS* (fully compatible with EFDOS) is Motorola's Disk Operating System available on development systems Such as EXORcisers, EXORsets, etc.

EFDOS is THOMSON SEMICONDUCTEURS Disk Operating System available on development systems such as DEVICE/EXOR...

Whenever ordering a custom MCU is required, please contact your local THOMSON-SEMICONDUCTEURS representative or THOMSON-SEMICONDUCTEURS distributor and/or complete and send the attached "MCU customer ordering sheet" to your local THOMSON-SEMICONDUCTEURS representative.

* Requires prior factory approval

ORDERING INFORMATION



The table below horizontally shows all available suffix combinations for package, operating temperature and screening level. Other possibilities on request.

DEVICE	PACKAGE					OPER. TEMP			SCREENING LEVEL			
	C	J	P	E	FN	L*	V	T	Std	D		
EF6804P2			●		●	●	●		●	●		

Examples : EF6804P2P, EF6804P2FN, EF6804P2PV, EF6804P2FNV.

Package : C : Ceramic DIL, **P** : Plastic DIL, **FN** : PLCC.
Oper. temp. : L* : 0°C to +70°C, **V** : -40°C to +85°C, **T** : -40°C to +105°C, * : may be omitted.
Screening level : Std : (no-end suffix), **D** : NFC 96883 level D.

EXORciser is a registered trademark to Motorola Inc.

These specifications are subject to change without notice.
Please inquire with our sales offices about the availability of the different products.

Printed in France

THOMSON COMPONENTS SEMICONDUCTEURS

TS68HC04J3

8-BIT MICROCOMPUTER

PRODUCT PREVIEW

HCMOS

The TS68HC04J3 microcomputer unit (MCU) is a member of the 68HC04 family of very low cost and low power single chip microcomputers. This 8 bit microcomputer contains a CPU, on-chip clock, ROM, RAM, I/O, and timer. It is designed for the user who needs an economical microcomputer with the proven capabilities of the 6800 based instruction set. The following are some of the hardware and software highlights of the TS68HC 04J3 MCU.

HARDWARE FEATURES

- Low power HCMOS
- Power saving stop and wait modes
- Single 2.0 to 6.0 volt power supply
- 8-bit architecture
- Fully static operation
- Pin compatible with 6804J2
- 124 bytes of on-chip RAM with standby mode
- 2 Kbytes of program ROM including 356 bytes for self-check program
- 72 bytes of user data ROM for look-up tables
- 12 TTL/CMOS compatible bidirectional I/O lines
- On-chip clock generator
- Extensive self-check capability allowing complete functional test of the chip (including ROM content)
- Master RESET and power-on-reset
- 8-bit timer with software programmable 7-bit prescaler
- TIMER pin programmable as input or output
- Complete development system support on DEVICE®.

SOFTWARE FEATURES

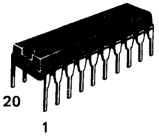
- Similar to 6800 family
- Byte efficient instruction set
- Easy to program
- True bit manipulation
- Stop, Wait and bit manipulation instructions
- Bit test and branch instructions
- Versatile interrupt handling
- Separate flags for normal and interrupt processing
- True LIFO 4-level stack eliminating stack pointer
- Maskable timer interrupt
- Versatile indirect registers
- Conditional branches
- Single instruction memory examine/change
- 9 powerful addressing modes.

USER SELECTABLE OPTIONS

- Crystal or low-cost resistor oscillator option
- Mask selectable internal clock generator options
- Mask selectable edge or level sensitive interrupt pin
- Program ROM protection option
- Optional pull-down devices on I/O lines.

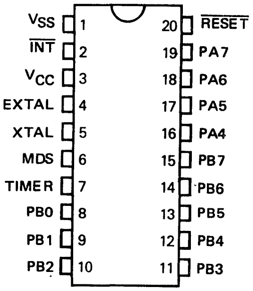
CASE

CB-194



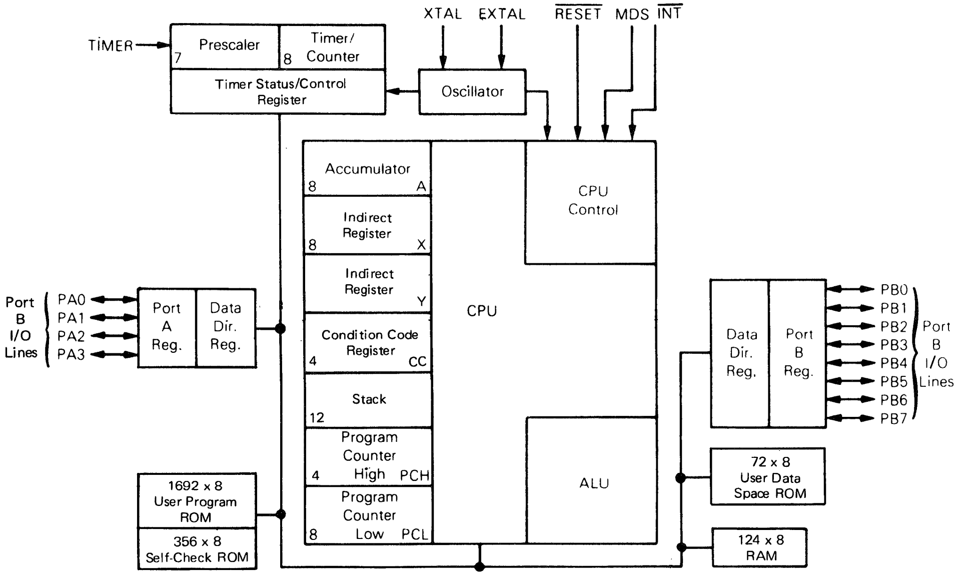
P SUFFIX
PLASTIC PACKAGE

PIN ASSIGNMENT



NOVEMBER 1986 1/2

BLOCK DIAGRAM



NOTE : 8-Bit Indirect Registers X and Y, although shown as part of the CPU are actually located in the 124 x 8 RAM at locations $80 and $81.

These specifications are subject to change without notice.
Please inquire with our sales offices about the availability of the different products.

Printed in France

Ref.DSTS68HC04J3

THOMSON COMPONENTS SEMICONDUCTEURS

EF68HC04P3

8-BIT MICROCOMPUTER

ADVANCE INFORMATION

HCMOS

The EF68HC04P3 microcomputer unit (MCU) is a member of the EF68HC 04 family of very low cost and low power single chip microcomputers. This 8 bit microcomputer contains a CPU, on-chip clock, ROM, RAM, I/O, and timer. It is designed for the user who needs an economical microcomputer with the proven capabilities of the EF6800 based instruction set. The following are some of the hardware and software highlights of the EF68HC04P3 MCU.

HARDWARE FEATURES

- Low power HCMOS
- Power saving stop and wait modes
- Single 2.0 to 6.0 volt power supply
- 8-bit architecture
- Fully static operation
- Pin compatible with the EF6805P2 and EF6804P2
- 124 bytes of on-chip RAM with standby mode
- 2 Kbytes of program ROM including 356 bytes for self-check program
- 72 bytes of user data ROM for look-up tables
- 20 open-collector TTL/CMOS compatible bidirectional I/O lines
- On-chip clock generator
- Extensive self-check capability allowing complete functional test of the chip (including ROM content)
- Master RESET and power-on-reset
- 8-bit timer with 7-bit software programmable prescaler
- TIMER pin programmable as input or output
- Complete development system support on DEVICE®.

SOFTWARE FEATURES

- Similar to EF6800 family
- Byte efficient instruction set
- Easy to program
- True bit manipulation
- Stop, Wait and bit manipulation instructions
- Bit test and branch instructions
- Versatile interrupt handling
- Separate flags for normal and interrupt processing
- True LIFO 4-level stack eliminating stack pointer
- Maskable timer interrupt
- Versatile indirect registers
- Conditional branches
- Single instruction memory examine/change
- 9 powerful addressing modes.

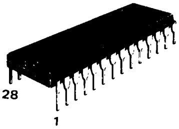
USER SELECTABLE OPTIONS

- Crystal or low-cost resistor oscillator option
- Mask selectable internal clock generator options
- Mask selectable edge or level sensitive interrupt pin
- Program ROM protection option
- Optional pull-down devices on I/O lines.

DEVICE® is THOMSON SEMICONDUCTEURS' development/emulation tool.

CASES

CB-132

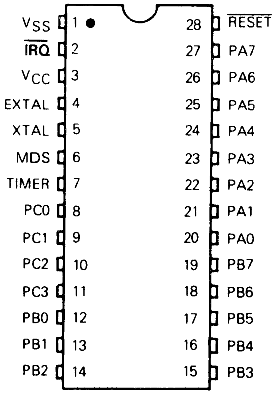


P SUFFIX
PLASTIC PACKAGE

CB-520

FN SUFFIX
PLCC 28

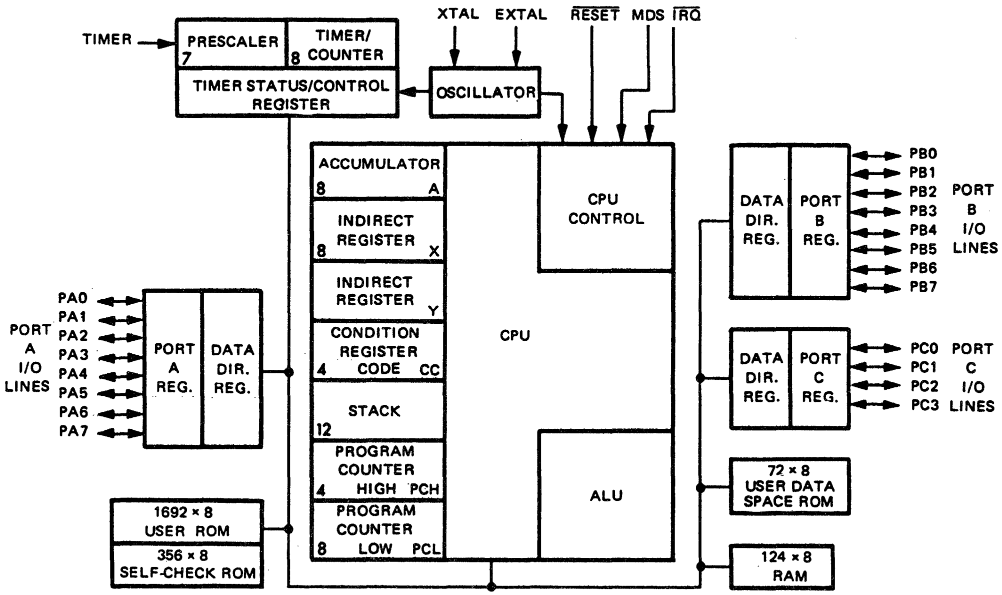
PIN ASSIGNMENT



JANUARY 1987 1/44

43, Avenue de l'Europe - 78140 VÉLIZY-VILLACOUBLAY/FRANCE - Tel. : (1) 39.46.97.19 - Telex TCSF 204780F - Telecopie : (1) 39.46.52.64

BLOCK DIAGRAM



SECTION 2
FUNCTIONAL PIN DESCRIPTION, MEMORY, CPU, AND REGISTERS

This section provides a description of the functional pins, memory spaces, the central processing unit (CPU), and the various registers and flags.

2.1 FUNCTIONAL PIN DESCRIPTION

2.1.1 V_{CC} and V_{SS}

Power is supplied to the MCU using these two pins. V_{CC} is power and V_{SS} is the ground connection.

2.1.2 $\overline{\text{IRQ}}$

This pin provides the capability for asynchronously applying an external interrupt to the MCU. Refer to **4.1 INTERRUPT** for additional information.

2.1.3 XTAL and EXTAL

These pins provide connections to the on-chip clock oscillator circuit. A crystal, a resistor and capacitor, or an external signal, depending on the user selectable manufacturing mask option, can be connected to these pins to provide a system clock source with various stability/cost tradeoffs. Lead lengths and stray capacitance on these two pins should be minimized. Refer to **4.4 INTERNAL CLOCK GENERATOR OPTIONS** for recommendations concerning these inputs.

2.1.4 TIMER

In the input mode, the timer pin is connected to the prescaler input and serves as the timer clock or as an enable input for the internal clock. In the output mode, the timer pin signals that a time out of the timer has occurred. Refer to **SECTION 3 TIMER** for additional information.

2.1.5 $\overline{\text{RESET}}$

The $\overline{\text{RESET}}$ pin is used to restart the processor of the EF68HC04P3 to the beginning of a program. This pin, together with the MDS pin is also used to select the operating mode of the EF68HC04P3. If the MDS pin is at zero volts, the normal mode is selected and the program counter is loaded with the user restart vector. However, if the MDS pin is at + 5 volts, then pins PA6 and PA7 are decoded to allow selection of the operating mode. Refer to **4.3 RESET** for additional information.

2.1.6 MDS

The MDS (mode select) pin is used to place the MCU into special operating modes. If MDS is held at +5 volts at the exit of the reset state, the decoded state of PA6 and PA7 is latched to determine the operating mode (single-chip, self-check, or ROM verify). However, if MDS is held at zero volts at the exit of the reset state, the single-chip operating mode is automatically selected (regardless of PA6 and PA7 state).

For those users familiar with the EF6801 microcomputer, mode selection is similar but much less complex in the EF68HC04P3. No special external diodes, switches, transistors, etc. are required in the EF68HC04P3.

2.1.7 Port Input/Output Lines (PA0-PA7, PB0-PB7, PC0-PC3)

These 20 lines are arranged into two 8-bit ports (A and B) and one 4-bit port (C). All lines are programmable as either inputs or outputs under software control of the data direction registers. Refer to **SECTION 5 INPUT/OUTPUT PORTS** for additional information.

2.2 MEMORY

The MCU operates in three different memory spaces: program space, data space, and stack space. A representation of these memory spaces is shown in Figure 2-1. The program space (Figure 2-1a) contains all of the instructions that are to be executed, as well as the data required for the immediate addressing mode instructions, and the self-check and user vectors. The data space (Figure 2-1b) contains all of the RAM locations, plus I/O locations and some ROM used for storage of tables and constants. The stack space (Figure 2-1c) contains RAM which is used for stacking subroutine and interrupt return addresses.

The MCU is capable of addressing 4096 bytes of program space memory with its program counter and 256 bytes of data space memory with its instructions. The data space memory contains three bytes for port data registers, three bytes for port data direction registers, one byte for timer status/control, 72 bytes ROM, 124 bytes RAM (which includes two bytes for XP and YP indirect registers), two bytes for timer prescaler and count registers, and one byte for the accumulator. The program space section contains 2048 bytes of ROM including 356 bytes of self-check ROM and 8 bytes of vectors for self-check and user programs.

2.3 CENTRAL PROCESSING UNIT

The CPU of the EF68HC04 Family is implemented independently from the I/O or memory configuration. Consequently, it can be treated as an independent central processor communicating with I/O and memory via internal addresses, data, and control buses.

(a) Program Space Memory Map

Reserved (All zeros)	$000 $7FF
Self-Check ROM	$800 $95F
Program ROM	$960 $FF7
Self Check $\overline{\text{IRQ}}$ Vector	$FF8-$FF9
Self Check Restart Vector	$FFA-$FFB
User $\overline{\text{IRQ}}$ Vector	$FFC-$FFD
User Restart Vector	$FFE-$FFF

(b) Data Space Memory Map

Port A Data Register	$00
Port B Data Register	$01
1 1 1 1 Port C Data Reg.	$02
Not Used	$03
Port A Data Direction Register	$04
Port B Data Direction Register	$05
1 1 1 1 Port C DDR	$06
Not Used	$07 $08
Timer Status Control Register	$09
CRC LSB	$0A
CRC MSB	$0B
Not Used	$0C $0F
Reserved	$10 $17
User Data Space ROM	$18 $5F
Future Expansion	$60 $7F
Indirect Register X *	$80
Indirect Register Y *	$81
Data Space RAM	$82 $FB
Reserved	$FC
Prescaler Register	$FD
Timer Count Register	$FE
Accumulator	$FF

(c) Stack Space Memory Map

Level 1
Level 2
Level 3
Level 4

* $80, $81, $82, $83 are used for short direct addressing.

Figure 2-1. EF68HC04P3 MCU Address Map

2.4 REGISTERS

The EF68HC04 Family CPU has four registers and two flags available to the programmer. They are shown in Figure 2-2 and are explained in the following paragraphs.

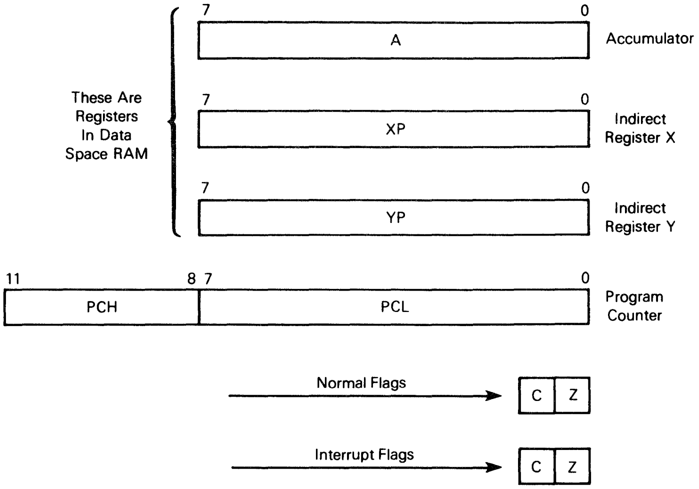


Figure 2-2. Programming Model

2.4.1 Accumulator (A)

The accumulator is an 8-bit general purpose register used in all arithmetic calculations, logical operations, and data manipulations. The accumulator is implemented as the highest RAM location ($FF) in data space and thus implies that several instructions exist which are not explicitly implemented. Refer to **6.3 IMPLIED INSTRUCTIONS** for additional information.

2.4.2 Indirect Registers (XP, YP)

These two indirect registers are used to maintain pointers to other memory locations in data space. They are used in the register-indirect addressing mode, and can be accessed with the direct, indirect, short direct, or bit set/clear addressinq modes. These registers are implemented as two of the 124 RAM locations ($80, $81) and as such generate implied instructions and may be manipulated in a manner similar to any RAM memory location in data space. Refer to **6.3 IMPLIED INSTRUCTIONS** for additional information.

2.4.3 Program Counter (PC)

The program counter is a 12-bit register that contains the address of the next ROM word to be used (may be opcode, operand, or address of operand). The 12-bit program counter is contained in PCL (low byte) and PCH (high nibble).

2.4.4 Flags (C, Z)

The carry (C) bit is set on a carry or a borrow out of the ALU. It is cleared if the result of an arithmetic operation does not result in a carry or a borrow. The (C) bit is also set to the value of the bit tested in a bit test instruction, and participates in the rotate left instruction.

The zero (Z) bit is set if the result of the last arithmetic or logical operation was equal to zero, otherwise it is cleared.

There are two sets of these flags, one set is for interrupt processing, the other for all other routines. When an interrupt occurs, a context switch is made from the program flags to the interrupt flags (interrupt mode). An RTI forces the context switch back to the program flags (program mode). While in either mode, only the flags for that mode are available. Further, the interrupt flags will not be cleared upon entering the interrupt mode. Instead, the flags will be as they were at the exit of the last interrupt mode. Both sets of flags are cleared by reset.

2.4.5 Stack

There is a true LIFO stack incorporated in the EF68HC04P3 which eliminates the need for a stack pointer. Stack space is implemented in separate RAM (12-bits wide) shown in Figure 2-1c. Whenever a subroutine call (or interrupt) occurs, the contents of the PC are shifted into the top register of the stack. At the same time (same cycle), the top register is shifted to the next level deeper. This happens to all registers with the bottom register falling out the bottom of the stack.

Whenever a subroutine or interrupt return occurs, the top register is shifted into the PC and all lower registers are shifted up one level higher. Stack level 4 is loaded with the previous content of stack level 3. The stack RAM is four levels deep.

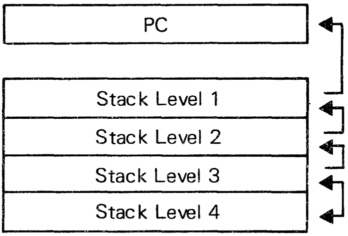


Figure 2-3. Stack Operation when RTS or RTI occurs

SECTION 3
TIMER

3.1 INTRODUCTION

A block diagram of the EF68HC04P3 timer circuitry is shown in Figure 3-1. The timer logic in the MCU is comprised of a simple 8-bit counter (timer count register, TCR) with a 7-bit prescaler, and a timer status/control register (TSCR). The timer count register, which may be loaded under program control is decremented towards zero by a clock input (prescaler output). The prescaler is used to extend the maximum interval of the overall timer. The prescaler tap is selected by bits 0-2 (PSO-PS2) of the timer status/control register. Bits PSO-PS2 control the actual division of the prescaler within the range of divide-by-1 (2^0) to divide-by-128 (2^7). The timer count register (TCR) and prescaler are decremented on rising clock edges. The coding of the TCSR PSO-PS2 bits produces a division in the prescaler as shown in Table 3-1.

Table 3-1. Prescaler Coding Table

PS2	PS1	PS0	Divide By
0	0	0	1
0	0	1	2
0	1	0	4
0	1	1	8

PS2	PS1	PS0	Divide By
1	0	0	16
1	0	1	32
1	1	0	64
1	1	1	128

TIMER pin may be programmed as either an output or an input depending on the status of DOUT and TOUT bits. Three modes are available.

Output mode (TOUT = 1)
The TIMER pin is connected to the DOUT latch. Therefore, the timer prescaler is clocked by the internal SYNC pulse. (Divide-by-12, -24 or -48 of the internal oscillator according to selected mask option, refer to 4.4 INTERNAL CLOCK GENERATOR OPTIONS). The prescaler then divides its clock input by a value determined by the coding of the TSCR bits PS0 - PS2 as shown in table 3-1. The divided prescaler output then clocks the 8-bit timer count register (TCR). When the TCR count reaches zero, it sets the TMZ bit in the TSCR. The TMZ bit can be tested under program control to perform a timer function whenever it goes high. The low-to-high TMZ bit transition is used to latch the DOUT bit of the TSCR and provides it for the TIMER pin.

Controlled mode (TOUT = 0, DOUT = 1)
The TIMER pin is an input which controls the counting by the prescaler-timer. When high, it enables counting Counting is disabled as long as this input remains low. Operation is similar to that described for the output mode.

Clock input mode (TOUT = 0 DOUT = 0)
The TIMER pin is connected directly to the prescaler input. Therefore the timer prescaler is clocked by the signal applied from the TIMER pin. Operation is similar to that described for the output mode. The frequency of the signal applied to the TIMER pin must be less than $1/t_{byte}$ ($f_{osc} \div 12$, $\div 24$ or $\div 48$ according to selected mask option) because of internal synchronization.

NOTE

TMZ is normally set to logic one when the timer times out (TCR count reaches $00) ; however, it may be set by a write of $00 to the TCR or by a write to bit 7 of the TSCR.
TMZ bit is cleared by a read-only of the TSCR even if TMZ bit is not concerned by this read.

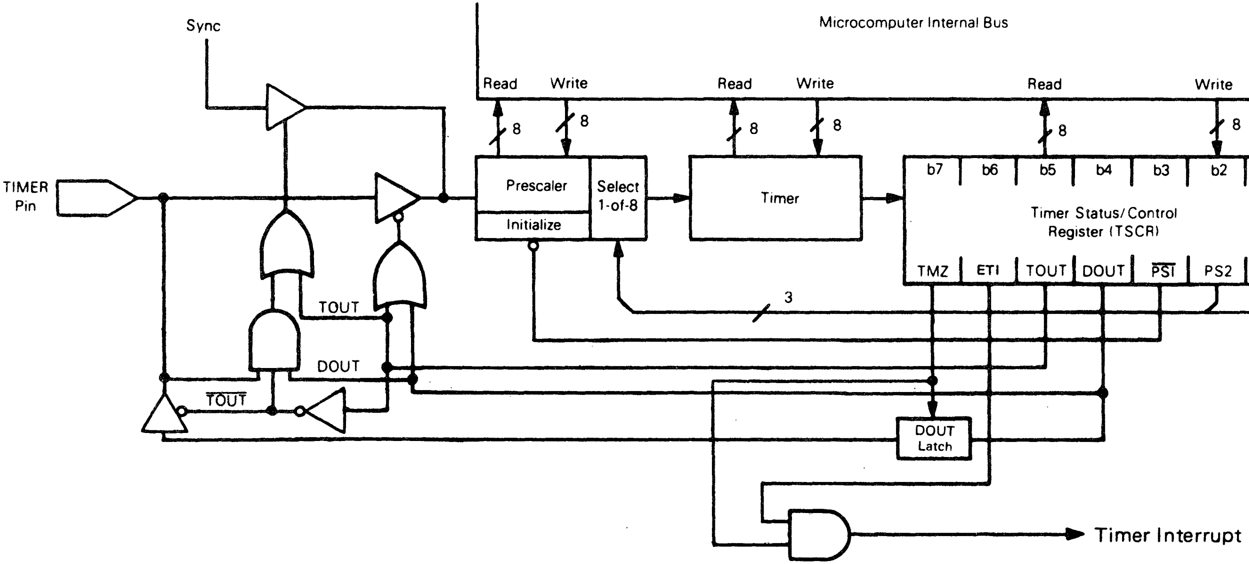


Figure 3-1. Timer Block Diagram

During reset, the timer count register and prescaler are set to $FF, while the timer status/control register is cleared to $00 and the DOUT LATCH (TIMER pin is in the high-impedance input mode) is forced to a logic high. The prescaler and timer count register are implemented in data space RAM locations ($FD, $FE); therefore, they are both readable and writeable. A write to either will predominate over the TCR decrement-to-$00 function; i.e., if a write and a TCR decrement-to-$00 occur simultaneously, the write will take precedence, and the TMZ bit is not set until the next timer time out.

3.2 TIMER REGISTERS

3.2.1 Timer Count Register (TCR)

7	0
MSB	LSB

TCR Address = $FE

The timer count register indicates the state of the internal 8-bit counter.

3.2.2 Timer Status/Control Register (TSCR)

7	6	5	4	3	2	1	0
TMZ	ETI	TOUT	DOUT	$\overline{\text{PSI}}$	PS2	PS1	PS0

TSCR Address = $09

b7,TMZ — Low-to-high transition indicates the timer count register has decremented to zero since the timer status/control register was last read. Cleared by a read of TSCR register if TMZ was read as a logic one.

b6, ETI — This bit, when set, enables the timer interrupt.

b5, TOUT — When low, this bit selects the input modes for the timer. When high, the output mode is selected.

b4, DOUT — Data sent to the timer output pin when TMZ is set high (output mode only). Choice of input mode (input mode only).

b3, $\overline{\text{PSI}}$ — Used to initialize the prescaler and inhibit its counting while $\overline{\text{PSI}} = 0$. The initialized value is set to $FF. The timer count register will also be inhibited (contents unchanged). When $\overline{\text{PSI}} = 1$ the prescaler begins to count downward.

b0, b1, b2 PS0-PS1-PS2 — These bits are used to select the prescaler divide-by ratio; therefore, effecting the clock input frequency to the timer count register.

3.2.3 Timer Prescaler Register

6	0
MSB	LSB

TPR Address = $FD

The timer prescaler register indicates the state of the internal 7-bit prescaler. This 7-bit prescaler divide ratio is normally determined by bits PS0-PS2 of the timer status/control register (see Table 3-1).

SECTION 4
INTERRUPT, SELF-CHECK, RESET, AND INTERNAL CLOCK GENERATOR

4.1 INTERRUPT

The EF68HC04P3 can be interrupted by applying a logic low signal to the $\overline{IRQ}$ pin ; however, a mask option selected at the time of manufacture determines whether the negative-going edge or the actual low level is sensed to indicate an interrupt The EF68HC04P3 can also be interrupted by the timer if ETI bit in TCSR register is set.

4.1.1 Edge-Sensitive Option

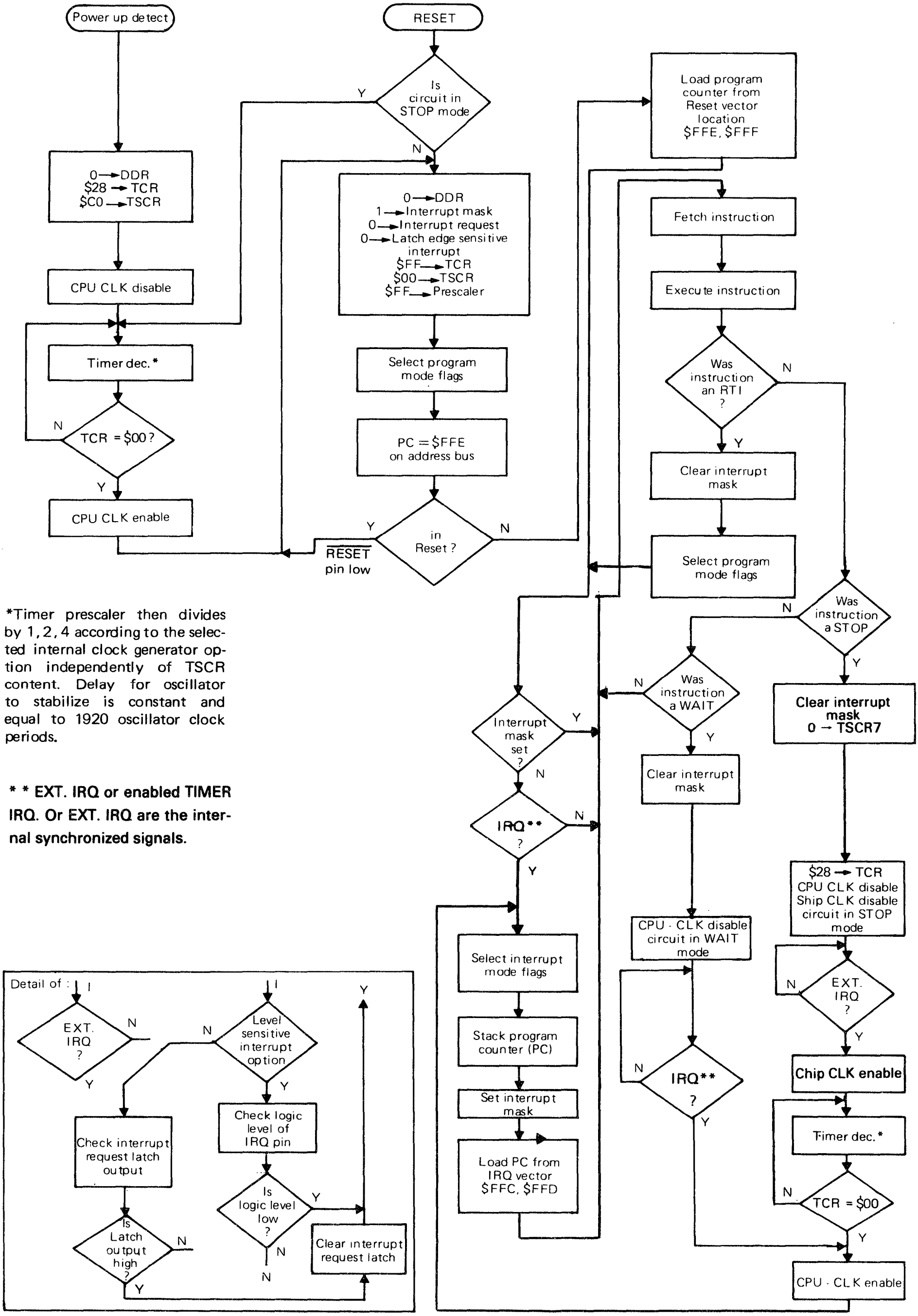
When the $\overline{IRQ}$ pin is pulled low, the internal interrupt request latch is set after a maximum of 3 machine cycles for internal synchronization. Prior to each instruction fetch, the interrupt latch is tested and, if its output is high, an interrupt sequence is initiated (provided the interrupt mask is cleared). Figure 4-1 contains a flowchart which illustrates both the reset and interrupt sequence. The interrupt sequence consists of one cycle during which the interrupt request latch is cleared, the interrupt mode flags are selected, the PC is saved on the stack, the interrupt mask is set, and the $\overline{IRQ}$ vector (single chip mode = $FFC/$FFD, self-check mode = $FF8/$FF9) is loaded into the PC. Internal processing of the interrupt continues until an RTI (return from interrupt) instruction is processed. During the RTI instruction, the interrupt mask is cleared and the program mode flags are selected. The next instruction of the program is then fetched and executed unless the $\overline{IRQ}$ pin has been pulled low at least 3 cycles before the end of the RTI instruction (see above). In this case, another interrupt sequence is initiated. Once the interrupt was initially detected and the interrupt sequence started, the interrupt request latch is cleared so that the next (second) interrupt may be detected even while the previous (first) one is being serviced. However, even though the second interrupt sets the interrupt request latch during processing of the first interrupt, the second interrupt sequence will not be initiated until completion of the interrupt service routine for the first interrupt. Completion of an interrupt service routine is always accomplished using an RTI instruction to return to the main program. The interrupt mask (which is not directly available to the programmer) is cleared during the last cycle of the RTI instruction.

4.1.2 Level-Sensitive Option

The actual operation of the level-sensitive and edge-sensitive options are similar except that the level-sensitive option does not have an interrupt request latch. With no interrupt request latch, the logic level of the $\overline{IRQ}$ pin is checked for detection of the interrupt. Also, in the interrupt sequence, there is no need to clear the interrupt request latch. These differences are illustrated in the flowchart of Figure 4-1.

4.1.3 Power Up and Timing

During the power-up sequence the interrupt mask is set to preclude any false or "ghost" interrupts from occurring. To clear the interrupt mask, the programmer should write a JSR (instead of a JMP) instruction to an initialization routine as the first instruction in a program. The initialization routine



*Timer prescaler then divides by 1, 2, 4 according to the selected internal clock generator option independently of TSCR content. Delay for oscillator to stabilize is constant and equal to 1920 oscillator clock periods.

**** EXT. IRQ or enabled TIMER IRQ. Or EXT. IRQ are the internal synchronized signals.**

Figure 4-1. STOP, WAIT, INTERRUPT and RESET processing flowchart

should end with an RTI (instead of RTS) or with a STOP or WAIT instruction which clears the interrupt mask. Maximum interrupt response time is eight machine (t_{byte}) cycles (see **4.4 INTERNAL CLOCK GENERATOR OPTIONS).** This includes five machine cycles for the longest instruction, plus one machine cycle for stacking the PC and switching flags and plus two machine cycles for synchronization of the $\overline{IRQ}$ input with internal machine cycles. Minimum response time is four machine cycles for internal synchronization (3 cycles) and for stacking PC and switching flags (see **2.4.4 Flags (C, Z)**).

4.2 SELF-CHECK

The self-check capability of the EF68HC04P3 MCU provides an intemal check to determine if the part a functional check of the MCU, connect it as shown in Figure 4-2a and monitor the LEDs for a 00100 ($04) pattem on port A. The MCU is left in the WAIT mode. A logical low signal applied to the $\overline{IRQ}$ pin places the MCU in the STOP mode. A 00101 ($05) pattern appears on port A. Another logical low signal applied on the $\overline{IRQ}$ pin enables exit from the STOP mode. The "final good" pattern (00110 - $06) appears on port A). To Initiate a ROM self-check of the memory simply connect the circuit as shown in Figure 4-2b and check that the "good" LED tums on to indicate a good memory. The ROM verify uses a cyclical redundancy check (CRC) to conduct a ROM check by means of signature analysis circuit. This circuit consists of two 8-bit shift registers configured to perform the check using the CCITT polynominal. A manufacturing mask option inhibits the outputs of the CRC data and the ROM data until the final result is available in order to protect the program ROM when the option is selected.

4.3 RESET

The MCU can be reset in two ways: by the external reset input ($\overline{RESET}$) and by power-up detect (PUD).

4.3.1. $\overline{RESET}$ input

This input can be used to reset the MCU internal state and provides an orderly software start-up procedure.

4.3.2 PUD

It occurs when a positive transition is detected on V_{CC} on initial power-up. No external RC network is needed. PUD is used strictly for power turn-on conditions and should not be used to detect any drops in the power supply voltage. There is no provision in this block for power-down detect. When the MCU is reset by means of PUD, an internal delay of 1920 oscillator clock periods is generated for the oscillator to stabilize. The MCU emerges from the reset condition at the end of this temporization.

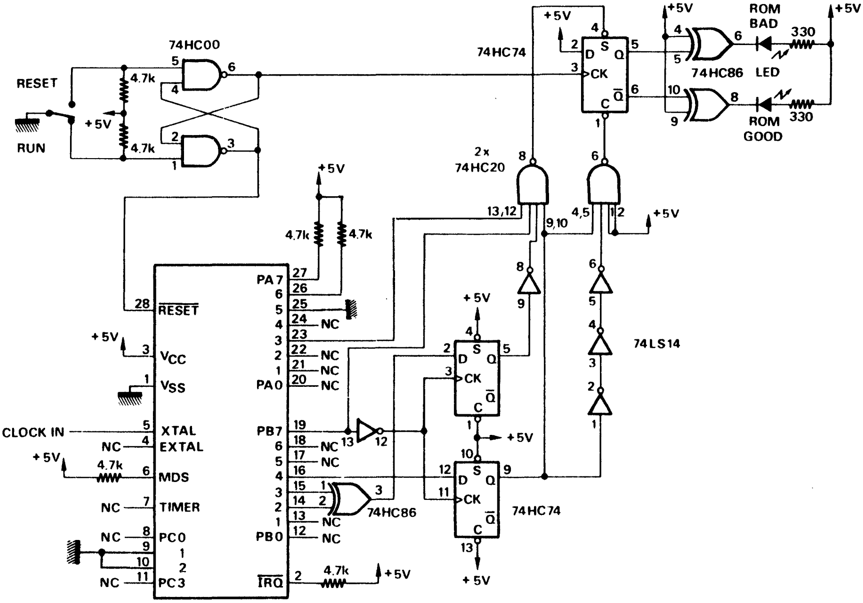
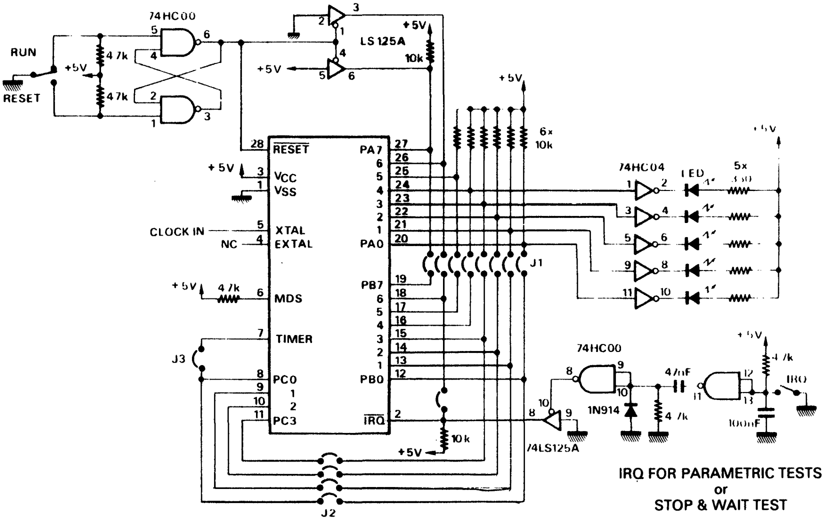


Figure 4-2. ROM Verify Circuit



IRQ FOR PARAMETRIC TESTS
or
STOP & WAIT TEST

(Beware of IRQ̅ pulse width when using level - Sensitive option - 10 machine cycles max. length).

Figure 4-3. Self-Check Circuit

4.4 INTERNAL CLOCK GENERATOR OPTIONS

The internal clock generator circuit is designed to require a minimum of external components. A crystal, a resistor-capacitor, or an external signal may be used to generate a system clock with various stability/cost tradeoffs. A manufacturing mask option is required to select either the crystal oscillator or the RC oscillator circuit. The different clock generator option connection methods are shown in Figure 4-4, crystal specifications and suggested PC board layouts are given in Figure 4-5, resistor-capacitor selection graph is given in Figure 4-6, and a timing diagram is illustrated in Figure 4-7. The crystal oscillator startup time is a function of many variables: crystal parameters (especially R_S), oscillator load capacitance (C_L), IC parameters, ambient temperature, and supply voltage. To ensure rapid oscillator startup, neither the crystal characteristics nor the load capacitance should exceed recommendations.

The oscillator output frequency is internally divided by four, two or one depending upon a manufacturing mask option selection to produce the internal ϕ 1 and ϕ2 clocks. The ϕ1 clock is divided by twelve to produce a machine byte (cycle) clock (internal SYNC pulse). A byte cycle is the smallest unit needed to execute any operation (i.e., increment the program counter). An instruction may need two, four, or five byte cycles to be executed.

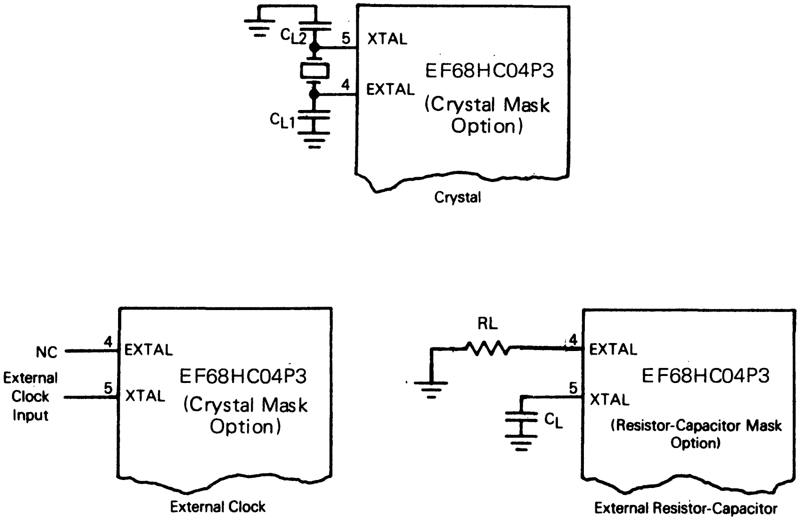
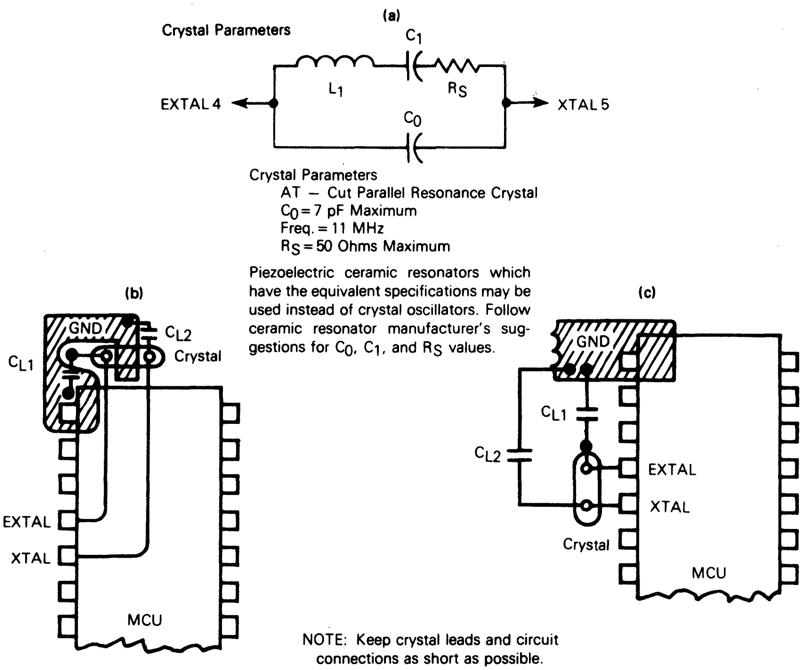


Figure 4-4. Clock Generator Options



Crystal Parameters
AT – Cut Parallel Resonance Crystal
C_0 = 7 pF Maximum
Freq. = 11 MHz
R_S = 50 Ohms Maximum

Piezoelectric ceramic resonators which have the equivalent specifications may be used instead of crystal oscillators. Follow ceramic resonator manufacturer's suggestions for C_0, C_1, and R_S values.

NOTE: Keep crystal leads and circuit connections as short as possible.

Figure 4-5. Crystal Motional Arm Parameters and Suggested PC Board Layout

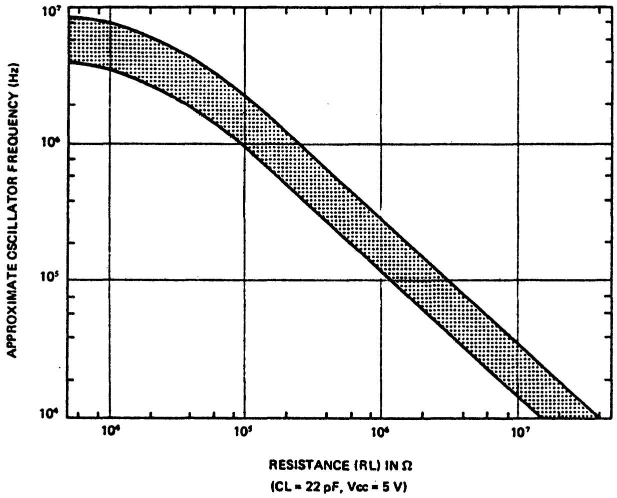
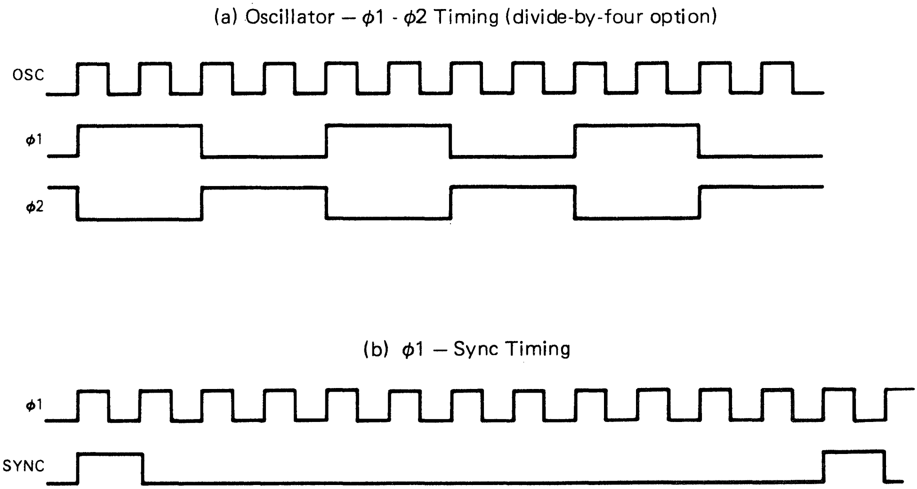


Figure 4-6. Typical Frequency Selection For Resistor-Capacitor Oscillator Option



(a) Oscillator — φ1 - φ2 Timing (divide-by-four option)

(b) φ1 — Sync Timing

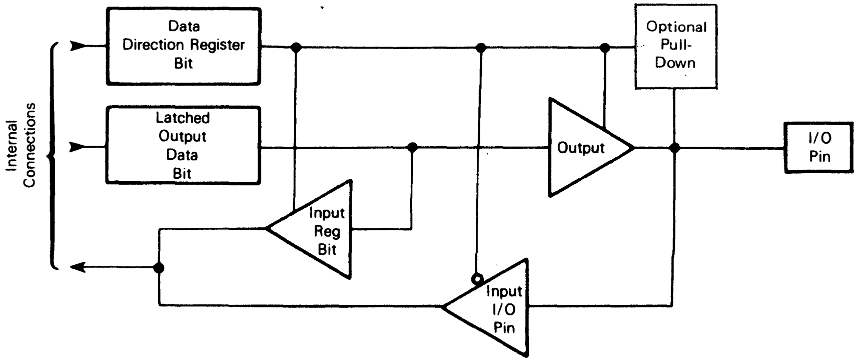
Figure 4-7. Clock Generator Timing Diagram

SECTION 5
INPUT/OUTPUT PORTS

5.1 INPUT/OUTPUT

There are 20 input/output pins. All pins (port A, B, and C) are programmable as either inputs or outputs under software control of the corresponding data direction register (DDR). The port I/O programming is accomplished by writing the corresponding bit in the port DDR to a logic one for output or a logic zero for input. On reset, all the DDRs are initialized to a logic zero state to put the ports in the input mode. The port output registers are not initialized on reset but should be initialized before changing the DDR bits to avoid undefined levels. When programmed as outputs, the latched output data is readable as input data, regardless of the logic levels at the output pin due to output loading ; see Figure 5-1. All input/output pins are open collector LSTTL/CMOS compatible as both inputs and outputs. A manufacturing mask option enables the choice of additional pull-down devices on all I/O pins (Selection in 5 groups: PA7, PA(5:6), PA(1:4), Port B, PA0 + Port C).

The address map in Figure 2-1 gives the address of data registers and DDRs. The register configuration is discussed under the registers paragraph below.



Data Direction Register Bit	Output Data Bit	Output State	Input To MCU
1	0	0	0
1	1	1	1
0	X	Hi-Z	Pin

Figure 5-1. Typical I/O Port Circuitry

The latched output data bit (see Figure 5-1) may always be written. Therefore, any write to a port writes to all of its data bits even though the port DDR is set to input. This may be used to initialize the data registers and avoid undefined outputs; however, care must be exercised when using read-modify-write instructions since the data read corresponds to the pin level if the DDR is an input (0) and corresponds to the latched output data when the DDR is an output (1).

5.2 REGISTERS

The registers described below are implemented as RAM locations and thus may be read or written.

5.2.1 Port Data Register

7	0
MSB	LSB

Port A Address = $00
Port B Address = $01
Port C Address = $02 (Bits 0-3)

The source of data read from the port data register will be the port I/O pin or previously latched output data depending upon the contents of the corresponding data direction register (DDR). The destination of data written to the port data register will be an output data latch. If the corresponding data direction register (DDR) for the port I/O pin is programmed as an output, the data will then appear on the port pin.

5.2.2 Port Data Direction Register

7	0
MSB	LSB

Port A Address = $04
Port B Address = $05
Port C Address = $06 (Bits 0-3)

The port DDRs configure the port pins as either inputs or outputs. Each port pin can be programmed individually to act as an input or an output. A zero in the pins corresponding bit position will program that pin as an input while a one in the pins corresponding bit position will program that pin as an output.

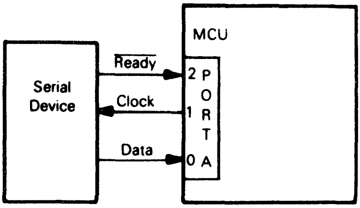
SECTION 6
SOFTWARE AND INSTRUCTION SET

6.1 SOFTWARE

6.1.1 Bit Manipulation

The EF68HC04P3 MCU has the ability to set or clear any register or single random access memory (RAM) writable bit with a single instruction (BSET, BCLR). Any bit in data space, including ROM, can be tested, using the BRSET and BRCLR instructions, and the program may branch as a result of its state. The carry bit equals the value of the bit referenced by BRSET or BRCLR. A rotate instruction may then be used to accumulate serial input data in a RAM location or register. The capability to work with any bit in RAM, ROM, or I/O allows the user to have individual flags in RAM or to handle I/O bits as control lines.

The coding example in Figure 6-1 illustrates the usefulness of the bit manipulation and test instructions. Assume that the MCU is to communicate with an external serial device. The external device has a data ready signal, a data output line, and a clock line (to clock data one bit at a time, MSB first, out of the device). The MCU waits until the data is ready, clocks the external device, picks up the data in the carry flag (C bit), clears the clock line, and finally accumulates the data bit in the accumulator.



```
                  .
                  .
                  .
SELF    BRSET   2,PORTA,SELF
                  .
                  .
                  .
        BSET    1,PORTA
        BRCLR   0,PORTA,CONT
CONT    BCLR    1,PORTA
        ROLA      .
                  .
                  .
                  .
```

Figure 6-1. Bit Manipulation Example

6.1.2 Addressing Modes

The EF68HC04P3 MCU has nine addressing modes which are explained briefly in the following paragraphs. The EF68HC04P3 deals with objects in three different address spaces : program space, data space, and stack space. Program space contains the instructions which are to be executed, plus the data for immediate mode instructions. Data space contains all of the RAM locations, XP and YP registers, accumulator, timer, I/O locations, and some ROM (for storage of tables and constants). Stack space contains RAM for use in stacking the return addresses for subroutines and interrupts.

The term "Effective Address" (EA) is used in describing the address modes. EA is defined as the address from which the argument for an instruction is fetched or stored.

6.1.2.1 IMMEDIATE. In the immediate addressing mode, the operand is located in program ROM and is contained in a byte following the opcode. The immediate addressing mode is used to access constants which do not change during program execution (e.g., a constant used to initialize a loop counter).

6.1.2.2 DIRECT. In the direct addressing mode, the effective address of the argument is contained in a single byte following the opcode byte. Direct addressing allows the user to directly address the 256 bytes in data space memory with a single two-byte instruction.

6.1.2.3 SHORT DIRECT. The MCU also has four locations in data space RAM ($80, $81, $82, $83) which may be used in a short-direct addressing mode. In this mode the opcode determines the data space RAM location, and the instruction is only one byte. Short direct addressing is a subset of the direct addressing mode. (Note : $80 and $81 are the X and Y register locations).

6.1.2.4 EXTENDED. In the extended addressing mode, the effective address is obtained by concatenating the four least significant bits of the opcode with the byte following the opcode (12-bit address). Instructions using the extended addressing mode (JMP, JSR) are capable of branching anywhere in program space. An extended addressing mode instruction is two bytes long.

6.1.2.5 RELATIVE. The relative addressing mode is only used in conditional branch instructions. In relative addressing, that address is formed by adding the sign extended lower five bits of the opcode (the offset) to the program counter if and only if the condition is true. Otherwise, control proceeds to the next instruction. The span of relative addressing is from − 15 to + 16 from the opcode address. The programmer need not worry about calculating the correct offset when using the assembler since it calculates the proper offset and checks to see if it is within the span of the branch.

6.1.2.6 BIT SET/CLEAR. In the bit set/clear addressing mode, the bit to be set or cleared is part of the opcode, and the byte following the opcode specifies the direct address of the byte in which the specified bit is to be set or cleared. Thus, any bit in the 256 locations of data space memory, which can be written to, can be set or cleared.

6.1.2.7 BIT TEST AND BRANCH. The bit test and branch addressing mode is a combination of direct addressing and relative addressing. The bit and condition (set or clear) which is to be tested is included in the opcode, and the data space address of the byte to be tested is in the single byte immediately following the opcode byte. The third byte is sign extended to twelve bits and becomes the offset added to the program counter if the condition is true. The single three-byte instruction allows the program to branch based on the condition of any bit in data space memory. The span of branching is from – 125 to + 130 from the opcode address. The state of the tested bit is also transferred to the carry flag.

6.1.2.8 REGISTER-INDIRECT. In the register-indirect addressing mode, the operand is at the address (in data space) pointed to by the contents of one of the indirect registers (X or Y). The particular X or Y register is selected by bit 4 of the opcode. Bit 4 of the opcode is then decoded into an address which selects the desired X or Y register ($80 or $81). A register-indirect instruction is one byte long.

6.1.2.9 INHERENT. In the inherent addressing mode, all the information necessary to execute the instruction is contained in the opcode. These instructions are one byte long.

6.2 INSTRUCTION SET

The EF68HC04P3 MCU has a set of 44 basic instructions, which when combined with nine addressing modes produce 244 usable opcodes. They can be divided into five different types : register/memory, read-modify-write, branch, bit manipulation, and control. The following paragraphs briefly explain each type. All the instructions within a given type are presented in individual tables.

6.2.1 Register/Memory Instructions

Most of these instructions use two operands. One operand is the accumulator and the other operand is obtained from memory using one of the addressing modes. The jump unconditional (JMP) and jump to subroutine (JSR) instructions have no register operands. Refer to Table 6-1.

6.2.2 Read-Modify-Write Instructions

These instructions read a memory location or a register, modify or test its contents, and write the modified value back to memory or to the register. There are ten instructions which utilize read-modify-write cycles. All INC and DEC forms along with all bit manipulation instructions use this method. Refer to Table 6-2.

6.2.3 Branch Instructions

The branch instructions cause a branch from the program when a certain condition is met. Refer to Table 6-3.

6.2.4 Bit Manipulation Instructions

These instructions are used on any bit in data space memory. One group either sets or clears. The other group performs the bit test branch operations. Refer to Table 6-4.

Table 6-1. Register/Memory Instructions

		Addressing Modes																			
		Indirect				Immediate			Direct			Inherent			Extended			Short-Direct			
Function	Mnem	Opcode		#	#		#	#		#	#		#	#		#	#		#	#	Special
		XP	YP	Bytes	Cycles	Opcode	Bytes	Cycles	Opcode	Bytes	Cycles	Opcode	Bytes	Cycles	Opcode	Bytes	Cycles	Opcode	Bytes	Cycles	Notes
Load A from Memory	LDA	E0	F0	1	4	E8	2	4	F8	2	4	—	—	—	—	—	—	AC-AF	1	4	1
Load XP from Memory	LDXI	—	—	—	—	B0	3	4	—	—	—	—	—	—	—	—	—	—	—	—	4
Load YP from Memory	LDYI	—	—	—	—	B0	3	4	—	—	—	—	—	—	—	—	—	—	—	—	4
Store A in Memory	STA	E1	F1	1	4	—	—	—	F9	2	4	—	—	—	—	—	—	BC-BF	1	4	2
Add to A	ADD	E2	F2	1	4	EA	2	4	FA	2	4	—	—	—	—	—	—	—	—	—	—
Subtract from A	SUB	E3	F3	1	4	EB	2	4	FB	2	4	—	—	—	—	—	—	—	—	—	—
Arithmetic Compare with Memory	CMP	E4	F4	1	4	EC	2	4	FC	2	4	—	—	—	—	—	—	—	—	—	—
AND Memory to A	AND	E5	F5	1	4	ED	2	4	FD	2	4	—	—	—	—	—	—	—	—	—	—
Jump to Subroutine	JSR	—	—	—	—	—	—	—	—	—	—	—	—	—	8 (TAR)	2	4	—	—	—	3
Jump Unconditional	JMP	—	—	—	—	—	—	—	—	—	—	—	—	—	9 (TAR)	2	4	—	—	—	3
Clear A	CLRA	—	—	—	—	—	—	—	FB	2	4	—	—	—	—	—	—	—	—	—	—
Clear XP	CLRX	—	—	—	—	—	—	—	FB	2	4	—	—	—	—	—	—	—	—	—	—
Clear YP	CLRY	—	—	—	—	—	—	—	FB	2	4	—			—	—	—	—	—	—	—
Complement A	COMA	—	—	—	—	—	—	—	—	—	—	B4	1	4	—	—	—	—	—	—	—
Move Immediate Value to Memory	MVI	—	—	—	—	B0	3	4	B0	3	4	—	—	—	—	—	—	—	—	—	5
Rotate A Left and Carry	ROLA	—	—	—	—	—	—	—	—	—	—	B5	1	4	—	—	—	—	—	—	—
Arithmetic Left Shift of A	ASLA	—	—	—	—	—	—	—	FA	2	4	—	—	—	—	—	—	—	—	—	—

SPECIAL NOTES

1. In Short-Direct addressing, the LDA mnemonic represents opcode AC, AD, AE, and AF. This is equivalent to RAM locations $80 (AC), $81 (AD), $82 (AE), and $83 (AF)
2. In Short-Direct addressing, the STA mnemonic represents opcode BC, BD, BE, and BF. This is equivalent to RAM locations $80 (BC), $81 (BD), $82 (BE), and $83 (BF).
3. In Extended addressing, the four LSBs of the opcode (Mnemonic JSR and JMP) are formed by the four MSBs of the target address (TAR).
4. In Immediate addressing, the LDXI and LDYI are mnemonics which are recognized as follows:

   LDXI = MVI $80,data

   LDYI = MVI $81,data Where data is a one-byte hexadecimal number.
5. The MVI instruction refers to both Immediate and Direct addressing.

Table 6-2. Read-Modify-Write Instructions

		Addressing Modes										
		Indirect				Direct			Short-Direct			Special Notes
		Opcode		#	#		#	#		#	#	
Function	Mnem	XP	YP	Bytes	Cycles	Opcode	Bytes.	Cycles	Opcode	Bytes	Cycles	
Increment Memory Location	INC	E6	F6	1	4	FE	2	4	A8-AB	1	4	1, 3
Increment A	INCA	–	–	–	–	FE	2	4	–	–	–	–
Increment XP	INCX	–	–	–	–	–	–	–	A8	1	4	–
Increment YP	INCY	–	–	–	–	–	–	–	A9	1	4	–
Decrement Memory Location	DEC	E7	F7	1	4	FF	2	4	B8-BB	1	4	2, 4
Decrement A	DECA	–	–	–	–	FF	2	4	–	–	–	–
Decrement XP	DECX	–	–	–	–	–	–	–	B8	1	4	–
Decrement YP	DECY	–	–	–	–	–	–	–	B9	1	4	–

SPECIAL NOTES

1. In Short-Direct addressing, the INC mnemonic represents opcode A8, A9, AA, and AB. These are equivalent to RAM locations $80 (A8), $81 (A9), $82 (AA), and $83 (AB).
2. In Short-Direct addressing, the DEC mnemonic represents opcode B8, B9, BA, and BB. These are equivalent to RAM locations $80 (B8), $81 (B9), $82 (BA), and $83 (BB).
3. In Indirect addressing, the INC mnemonic represents opcode E6 or F6, and causes the location pointed to by XP (E6 opcode) or YP (F6 opcode) to be incremented.
4. In Indirect addressing, the INC mnemonic represents opcode E7 or F7, and causes the location pointed to by XP (E7 opcode) or YP (F7 opcode) to be incremented.

Table 6-3. Branch Instructions

		Relative Addressing Mode			
Function	Mnem	Opcode	# Bytes	# Cycles	Special Notes
Branch if Carry Clear	BCC	40-5F	1	2	1
Branch if Higher or Same	(BHS)	40-5F	1	2	1, 2
Branch if Carry Set	BCS	60-7F	1	2	1
Branch if Lower	(BLO)	60-7F	1	2	1, 3
Branch if Not Equal	BNE	00-1F	1	2	1
Branch if Equal	BEQ	20-3F	1	2	1

SPECIAL NOTES

1. Each mnemonic of the Branch Instructions covers a range of 32 opcodes; e.g., BCC ranges from 40 through 5F. The actual memory location (target address) to which the branch is made is formed by adding the sign extended lower five bits of the opcode to the contents of the program counter.
2. The BHS instruction (shown in parentheses) is identical to the BCC instruction. The C bit is clear if the register was higher or the same as the location in the memory to which it was compared.
3. The BLO instruction (shown in parentheses) is identical to the BCS instruction. The C bit is set if the register was lower than the location in memory to which it was compared.

Table 6-4. Bit Manipulation Instructions

		Addressing Modes						
		Bit Set/Clear			Bit Test and Branch			
Function	Mnem	Opcode	# Bytes	# Cycles	Opcode	# Bytes	# Cycles	Special Note
Branch IFF Bit n is set	BRSET n (n=0 . . . . 7)	–	–	–	C8+n	3	5	1
Branch IFF Bit n is clear	BRCLR n (n=0 . . . . 7)	–	–	–	C0+n	3	5	1
Set Bit n	BSET n (n=0 . . . .7)	D8+n	2	4	–	–	–	1
Clear Bit n	BCLR n (n=0 . . . . 7)	D0+n	2	4	–	–	–	1

SPECIAL NOTE

1. The opcode is formed by adding the bit number (0-7) to the basic opcode. For example: to clear bit six using the BSET6 instruction the opcode becomes DE (D8+6); BCLR5 becomes (C0+5); etc.

6.2.5 Control Instructions

The control instructions control the MCU operations during program execution. Refer to Table 6-5.

6.2.5.1 Stop instruction

The STOP instruction places the EF68HC04P3 in its lowest power consumption mode. In STOP mode the internal oscillator is turned off causing all internal processing and the timer to be halted. In STOP mode, timer STATUS/CONTROL register bits 6 (ETI) and 7 (TMZ) are cleared to remove any pending timer interrupt requests and to disable any further timer interrupts. External interrupts are enabled. All I/O lines remain unchanged. The processor can only be brought out of the STOP mode by pulling low $\overline{\text{IRQ}}$ or $\overline{\text{RESET}}$ input pins. During the exit from the STOP mode, the timer is used to provide a delay of 1920 oscillator clock periods for the oscillator to stabilize. If an external clock is used, it should be kept high during all the time the MCU is in STOP mode.

6.2.5.2 Wait instruction

The WAIT instruction places EF68HC04P3 in a low-power consumption mode, but WAIT mode consumes more power than the STOP mode. In WAIT mode the clock is disabled from all internal circuitry except the timer circuit. Thus all internal processing is halted. The timer may, if desired, continue to count down ($\overline{\text{PSI}}$ bit of TCSR).

During the WAIT mode, external interrupts are enabled. All other registers memory, and I/O lines remain in their last state. Timer interrupt (ETI bit) may be enabled by software prior to entering the WAIT mode to allow an exit from the WAIT mode via a Timer Interrupt.

6.2.6 Alphabetical Listing

The complete instruction set is given in alphabetical order in Table 6-6. There are certain mnemonics recognized by the assember and converted to other instructions. The fact that all registers and accumulator are in RAM allows many implied instructions to exist. The implied instructions recognized by the assembler are identified in Table 6-6.

6.2.7 Opcode Map Summary

Table 6-7 contains an opcode map for the instructions used on the MCU.

6.3 IMPLIED INSTRUCTIONS

Since the accumulator and all other registers are located in RAM many implied instructions exist. The assembler-recognized implied instructions are given in Table 6-6. Some examples not recognized by the assembler are shown below.

BCLR,7 $FF	Ensures accumulator is plus
BSET,7 $FF	Ensures accumulator is minus
BRCLR,7 $FF	Branch iff accumulator is plus
BRSET,7 $FF	Branch iff accumulator is minus
BRCLR,7 $80	Branch iff X is plus (BXPL)
BRSET,7 $80	Branch iff X is minus (BXMI)
BRCLR,7 $81	Branch iff Y is plus (BYPL)
BRSET,7 $81	Branch iff Y is minus (BYMI)

Table 6-5. Control Instructions

		Addressing Modes									
		Short-Direct			Inherent			Relative			
Function	Mnem	Opcode	# Bytes	# Cycles	Opcode	# Bytes	# Cycles	Opcode	# Bytes	# Cycles	Special Notes
Transfer A to XP	TAX	BC	1	4	–	–	–	–	–	–	–
Transfer A to YP	TAY	BD	1	4	–	–	–	–	–	–	–
Transfer XP to A	TXA	AC	1	4	–	–	–	–	–	–	–
Transfer YP to A	TYA	AD	1	4	–	–	–	–	–	–	–
Return from Subroutine	RTS	–	–	–	B3	1	2	–	–	–	–
Return from Interrupt	RTI	–	–	–	B2	1	2	–	–	–	–
No-Operation	NOP	–	–	–	–	–	–	–	–	–	1
Stop	STOP	–	–	–	B6	1	2	–	–	–	–
Wait	WAIT	–	–	–	B7	1	2	–	–	–	–

SPECIAL NOTE

1 . The NOP instruction is equivalent to a branch if equal (BEQ) to the location designated by PC+1.

Table 6-6. Instruction Set

	Addressing Modes									Flags	
Mnemonic	Inherent	Immediate	Direct	Short Direct	Bit/Set Clear	Bit-Test-Branch	Register Indirect	Extended	Relative	Z	C
ADD		X	X				X			∧	∧
AND		X	X				X			∧	•
ASLA			Assembler converts this to "ADD $FF"							∧	∧
BCC									X	•	•
BCLR					X					•	•
BCS									X	•	•
BEQ									X	•	•
BHS			Assembler converts this to "BCC"							•	•
BLO			Assembler converts this to "BCS"							•	•
BNE									X	•	•
BRCLR						X				•	∧
BRSET						X				•	∧
BSET					X					•	•
CLRA			Assembler converts this to "SUB $FF"							∧	∧
CLRX			Assembler converts this to "MVI #0, $80"							•	•
CLRY			Assembler converts this to "MVI #0, $81"							•	•
CMP		X	X				X			∧	∧
COMA	X									∧	∧
DEC			X	X			X			∧	•
DECA			Assembler converts this to "DEC $FF"							∧	•
DECX			Assembler converts this to "DEC $80"							∧	•
DECY			Assembler converts this to "DEC $81"							∧	•
INC			X	X			X			∧	•
INCA			Assembler converts this to "INC $FF"							∧	•
INCX			Assembler converts this to "INC $80"							∧	•
INCY			Assembler converts this to "INC $81"							∧	•
JMP								X		•	•
JSR								X		•	•
LDA		X	X	X			X			∧	•
LDXI			Assembler converts this to "MVI DATA, $80"							•	•
LDYI			Assembler converts this to "MVI DATA, $81"							•	•
MVI		X	X							•	•
NOP			Assembler converts this to "BEQ (PC) + 1"							•	•
ROLA	X									∧	∧
RTI	X									∧	∧
RTS	X									•	•
STA			X	X			X			∧	•
STOP	X									•	•
SUB		X	X				X			∧	∧
TAX	Assembler converts this to "STA $80"									∧	•
TAY	Assembler converts this to "STA $81"									∧	•
TXA	Assembler converts this to "LDA $80"									∧	•
TYA	Assembler converts this to "LDA $81"									∧	•
WAIT	X									•	•

Flag Symbols: Z = Zero, C = Carry/Borrow, ∧ = Test and Set if True, Cleared Otherwise • = Not Affected

Table 6-7. EF68HC04P3 Microcomputer

	Branch Instructions							
Hi Low	0 0000	1 0001	2 0010	3 0011	4 0100	5 0101	6 0110	7 0111
0 0000	2 BNE 1 REL	2 BNE 1 REL	2 BEQ 1 REL	2 BEQ 1 REL	2 BCC 1 REL	2 BCC 1 REL	2 BCS 1 REL	2 BCS 1 REL
1 0001	2 BNE 1 REL	2 BNE 1 REL	2 BEQ 1 REL	2 BEQ 1 REL	2 BCC 1 REL	2 BCC 1 REL	2 BCS 1 REL	2 BCS 1 REL
2 0010	2 BNE 1 REL	2 BNE 1 REL	2 BEQ 1 REL	2 BEQ 1 REL	2 BCC 1 REL	2 BCC 1 REL	2 BCS 1 REL	2 BCS 1 REL
3 0011	2 BNE 1 REL	2 BNE 1 REL	2 BEQ 1 REL	2 BEQ 1 REL	2 BCC 1 REL	2 BCC 1 REL	2 BCS 1 REL	2 BCS 1 REL
4 0100	2 BNE 1 REL	2 BNE 1 REL	2 BEQ 1 REL	2 BEQ 1 REL	2 BCC 1 REL	2 BCC 1 REL	2 BCS 1 REL	2 BCS 1 REL
5 0101	2 BNE 1 REL	2 BNE 1 REL	2 BEQ 1 REL	2 BEQ 1 REL	2 BCC 1 REL	2 BCC 1 REL	2 BCS 1 REL	2 BCS 1 REL
6 0110	2 BNE 1 REL	2 BNE 1 REL	2 BEQ 1 REL	2 BEQ 1 REL	2 BCC 1 REL	2 BCC 1 REL	2 BCS 1 REL	2 BCS 1 REL
7 0111	2 BNE 1 REL	2 BNE 1 REL	2 BEQ 1 REL	2 BEQ 1 REL	2 BCC 1 REL	2 BCC 1 REL	2 BCS 1 REL	2 BCS 1 REL
8 1000	2 BNE 1 REL	2 BNE 1 REL	2 BEQ 1 REL	2 BEQ 1 REL	2 BCC 1 REL	2 BCC 1 REL	2 BCS 1 REL	2 BCS 1 REL
9 1001	2 BNE 1 REL	2 BNE 1 REL	2 BEQ 1 REL	2 BEQ 1 REL	2 BCC 1 REL	2 BCC 1 REL	2 BCS 1 REL	2 BCS 1 REL
A 1010	2 BNE 1 REL	2 BNE 2 REL	2 BEQ 1 REL	2 BEQ 1 REL	2 BCC 1 REL	2 BCC 1 REL	2 BCS 1 REL	2 BCS 1 REL
B 1011	2 BNE 1 REL	2 BNE 1 REL	2 BEQ 1 REL	2 BEQ 1 REL	2 BCC 1 REL	2 BCC 1 REL	2 BCS 1 REL	2 BCS 1 REL
C 1100	2 BNE 1 REL	2 BNE 1 REL	2 BEQ 1 REL	2 BEQ 1 REL	2 BCC 1 REL	2 BCC 1 REL	2 BCS 1 REL	2 BCS 1 REL
D 1101	2 BNE 1 REL	2 BNE 1 REL	2 BEQ 1 REL	2 BEQ 1 REL	2 BCC 1 REL	2 BCC 1 REL	2 BCS 1 REL	2 BCS 1 REL
E 1110	2 BNE 1 REL	2 BNE 1 REL	2 BEQ 1 REL	2 BEQ 1 REL	2 BCC 1 REL	2 BCC 1 REL	2 BCS 1 REL	2 BCS 1 REL
F 1111	2 BNE 1 REL	2 BNE 1 REL	2 BEQ 1 REL	2 BEQ 1 REL	2 BCC 1 REL	2 BCC 1 REL	2 BCS 1 REL	2 BCS 1 REL

Abbreviations for Address Modes

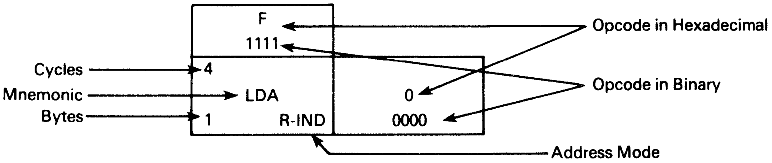
INH	Inherent
S-D	Short Direct
B-T-B	Bit Test and Branch
IMM	Immediate
DIR	Direct
EXT	Extended
REL	Relative
BSC	Bit Set/Clear
R-IND	Register Indirect

*	Indicates Instruction Reserved for Future Use
#	Indicates Illegal Instruction

Instruction Set Opcode Map

Register/Memory, Control, and Read/Modify/Write Instructions				Bit Manipulation Instructions		Register/Memory and Read/Modify/Write		
8 1000	9 1001	A 1010	B 1011	C 1100	D 1101	E 1110	F 1111	Hi / Low
4 JSRn 2 EXT	4 JMPn 2 EXT	*	4 MVI 3 IMM	5 BRCLR0 3 B-T-B	4 BCLR0 2 BSC	4 LDA 1 R-IND	4 LDA 1 R-IND	0 0000
4 JSRn 2 EXT	4 JMPn 2 EXT	*	*	5 BRCLR1 3 B-T-B	4 BCLR1 2 BSC	4 STA 1 R-IND	4 STA 1 R-IND	1 0001
4 JSRn 2 EXT	4 JMPn 2 EXT	*	2 RTI 1 INH	5 BRCLR2 3 BT-B	4 BCLR2 2 BSC	4 ADD 1 R-IND	4 ADD 1 R-IND	2 0010
4 JSRn 2 EXT	4 JMPn 2 EXT	*	2 RTS 1 INH	5 BRCLR3 3 B-T-B	4 BCLR3 2 BSC	4 SUB 1 R-IND	4 SUB 1 R-IND	3 0011
4 JSRn 2 EXT	4 JMPn 2 EXT	*	4 COMA 1 INH	5 BRCLR4 3 B-T-B	4 BCLR4 2 BSC	4 CMP 1 R-IND	4 CMP 1 R-IND	4 0100
4 JSRn 2 EXT	4 JMPn 2 EXT	*	4 ROLA 1 INH	5 BRCLR5 3 B-T-B	4 BCLR5 2 BSC	4 AND 1 R-IND	4 AND 1 R-IND	5 0101
4 JSRn 2 EXT	4 JMPn 2 EXT	*	2 STOP 1 INH	5 BRCLR6 3 B-T-B	4 BCLR6 2 BSC	4 INC 1 R-IND	4 INC 1 R-IND	6 0110
4 JSRn 2 EXT	4 JMPn 2 EXT	*	2 WAIT 1 INH	5 BRCLR7 3 B-T-B	4 BCLR7 2 BSC	4 DEC 1 R-IND	4 DEC 1 R-IND	7 0111
4 JSRn 2 EXT	4 JMPn 2 EXT	4 INC 1 S-D	4 DEC 1 S-D	5 BRSET0 3 B-T-B	4 BSET0 2 BSC	4 LDA 2 IMM	4 LDA 2 DIR	8 1000
4 JSRn 2 EXT	4 JMPn 2 EXT	4 INC 1 S-D	4 DEC 1 S-D	5 BRSET1 3 B-T-B	4 BSET1 2 BSC	#	4 STA 2 DIR	9 1001
4 JSRn 2 EXT	4 JMPn 2 EXT	4 INC 1 S-D	4 DEC 1 S-D	5 BRSET2 3 B-T-B	4 BSET2 2 BSC	4 ADD 2 IMM	4 ADD 2 DIR	A 1010
4 JSRn 2 EXT	4 JMPn 2 EXT	4 INC 1 S-D	4 DEC 1 S-D	5 BRSET3 3 B-T-B	4 BSET3 2 BSC	4 SUB 2 IMM	4 SUB 2 DIR	B 1011
4 JSRn 2 EXT	4 JMPn 2 EXT	4 LDA 1 S-D	4 STA 1 S-D	5 BRSET4 3 B-T-B	4 BSET4 2 BSC	4 CMP 2 IMM	4 CMP 2 DIR	C 1100
4 JSRn 2 EXT	4 JMPn 2 EXT	4 LDA 1 S-D	4 STA 1 S-D	5 BRSET5 3 B-T-B	4 BSET5 2 BSC	4 AND 2 IMM	4 AND 2 DIR	D 1101
4 JRSn 2 EXT	4 JMPn 2 EXT	4 LDA 1 S-D	4 STA 1 S-D	5 BRSET6 3 B-T-B	4 BSET6 2 BSC	#	4 INC 2 DIR	E 1110
4 JSRn 2 EXT	4 JMPn 2 EXT	4 LDA 1 S-D	4 STA 1 S-D	5 BRSET7 3 B-T-B	4 BSET7 2 BSC	#	4 DEC 2 DIR	F 1111

LEGEND



SECTION 7
ELECTRICAL SPECIFICATIONS

7.1 INTRODUCTION

This section contains the electrical specifications and associated timing for the EF68HC04P3.

7.2 MAXIMUM RATINGS

Rating	Symbol	Value	Unit
Supply Voltage	V_{CC}	–0.3 to + 7.0	V
Input Voltage	V_{in}	V_{SS}–0.3 to V_{SS}+ 0.3	V
Current Drain per Pin	I	10	mA
Total Current for Ports A, B, C and XTAL, TIMER Pins Sink Source	 I_I I_O	 30 –15	 mA
Operating Temperature Range L Range D Range V Range	T_A	T_L to T_H 0 to 70 – 25 to 70 – 40 to 85	°C
Storage Temperature Range	T_{stg}	–55 to 150	°C
Thermal Resistance Plastic PLCC	θ_{JA}	 85 85	°C/W

This device contains circuitry to protect the inputs against damage due to high static voltages of electric fields; however, it is advised that normal precautions be taken to avoid application of any voltage higher than maximum rated voltages to this high impedance circuit. For proper operation it is recommended that V_{in} and V_{out} be constrained to the range $V_{SS} \leq (V_{in}$ or $V_{out}) \leq V_{CC}$. Reliability of operation is enhanced if unused inputs except EXTAL are connected to an appropriate logic voltage level (e.g., either V_{SS} or V_{CC}).

7.3 THERMAL CHARACTERISTICS

Characteristic	Symbol	Value	Unit
Junction Temperature Plastic PLCC	T_J	 150 150	°C

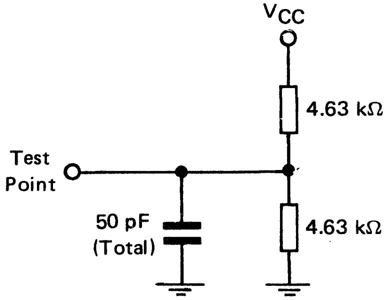


Figure 7-1. Open Collector LSTTL Compatible Equivalent Test Load (Ports A, B, C,)

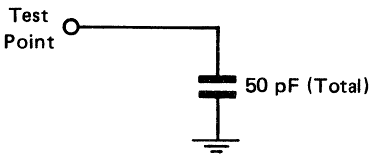


Figure 7-2. CMOS Equivalent Test Load (Ports A, B, C)

7.4 POWER CONSIDERATIONS

The average chip-junction temperature, T_J, in °C can be obtained from:

$T_J = T_A + (P_D \bullet \theta_{JA})$ (1)

Where:

T_A = Ambient Temperature, °C
θ_{JA} = Package Thermal Resistance, Junction-to-Ambient, °C/W
$P_D = P_{INT} + P_{PORT}$
$P_{INT} = I_{CC} \times V_{CC}$, Watts – Chip Internal Power
P_{PORT} = Port Power Dissipation, Watts – User Determined

For most applications $P_{PORT} \ll P_{INT}$ and can be neglected. P_{PORT} may become significant if the device is configured to drive Darlington bases or sink LED loads.

An approximate relationship between P_D and T_J (if P_{PORT} is neglected) is:

$P_D = K \div (T_J + 273°C)$ (2)

Solving equations 1 and 2 for K gives:

$K = P_D \bullet (T_A + 273°C) + \theta_{JA} \bullet P_D^2$ (3)

Where K is a constant pertaining to the particular part. K can be determined from equation 3 by measuring P_D (at equilibrium) for a known T_A. Using this value of K the values of P_D and T_J can be obtained by solving equations (1) and (2) iteratively for any value of T_A.

7.5 ELECTRICAL CHARACTERISTICS @ 5.0 V (V_{CC} = 5.0 Vdc ± 10 % ; V_{SS} = 0 Vdc ; $T_A = T_L$ to T_H unless otherwise noted).

Characteristic	Symbol	Min	Typ	Max	Unit
Output voltage					
(I_{Load} = 10 µA)	V_{OL}	–	–	0.1	V
(I_{Load} = –10 µA)	V_{OH}	V_{CC} – 0.1	–	–	
Output Low Voltage (I_{Load} = 0.8 mA) PA0-PA7, PB0-PB7, PC0-PC3	V_{OL}	–	–	0.4	V
Output High Voltage (I_{Load} = –0.8 mA) PA0-PA7, PB0-PB7, PC0-PC3	V_{OH}	V_{CC} –0.4	–	–	V
Input Low Voltage					
PA0-PA7, PB0-PB7, PC0-PC3, TIMER	V_{IL}	V_{SS}	–	0.3 V_{CC}	V
XTAL, MDS, $\overline{IRQ}$, $\overline{RESET}$	V_{IL}	V_{SS}	–	0.2 V_{CC}	
Input High Voltage					
PA0-PA7, PB0-PB7, PC0-PC3, TIMER	V_{IH}	0.7 V_{CC}	–	V_{CC}	V
XTAL, MDS, $\overline{IRQ}$, $\overline{RESET}$	V_{IH}	0.8 V_{CC}	–	V_{CC}	
Total Supply Current (No dc Loads, (No dc Loads, V_{IL} = 0.2 V ; V_{IH} = V_{CC} – 0.2 V)					
RUN Mode					
÷ 1 Option (f_{osc} = 5.5-MHz)	I_{DD}	–	2	3.3	mA
÷ 2 Option (f_{osc} = 11.0 MHz)	I_{DD}	–	2.4	3.9	mA
÷ 4 Option (f_{osc} = 11.0 MHz)	I_{DD}	–	1.6	2.6	mA
WAIT Mode (See Note)					
÷ 1 Option (f_{osc} = 5.5 MHz)	I_{DD}	–	0.7	1.1	mA
÷ 2 Option (f_{osc} = 11.0 MHz)	I_{DD}	–	1.1	1.8	mA
÷ 4 Option (f_{osc} = 11.0 MHz)	I_{DD}	–	0.9	1.5	mA
STOP Mode (See Note)	I_{DD}	–	1.0	5	µA
I/O Hi-Z Leakage Current (V_{in} = 0.4 V to V_{CC} - 0.4 V) Timer, PA0-PA7, PB0-PB7, PC0-PC3 (with no pull-downs)	I_{TSI}	–10	–	10	µA
Hi-Z State Input Current (V_{in} = V_{CC} – 0.4 V) PA0-PA7, PB0-PB7, PC0-PC3 (with pull-downs)	I_{TSI}	40	–	300	µA
Input Current $\overline{IRQ}$, $\overline{RESET}$, XTAL, MDS	I_{in}	–1	–	+ 1	µA
I/O Output Capacitance Timer, PA0-PA7, PB0-PB7, PC0-PC3	C_{out}	–	–	12	pF
Input Capacitance $\overline{IRQ}$, $\overline{RESET}$, XTAL, MDS	C_{in}	–	–	8	pF

NOTE : Test conditions for I_{DD} as follows :
XTAL input is a square wave from 0.2 V to V_{CC} – 0.2 V.
EXTAL output load = 10 pF
Circuit in self check-mode
In WAIT and STOP Modes, Port A is programmed as output, Ports B and C are programmed as inputs.
In STOP Mode : all inputs are tied to V_{IL} excepted $\overline{IRQ}$, $\overline{RST}$, MDS, XTAL, EXTAL which are tied to V_{IH}.

7.6 CONTROL TIMING @ 5.0 V
(V_{CC} = 5.0 Vdc ± 10 % ; V_{SS} = 0 Vdc ; T_A = T_L to T_H unless otherwise noted).

Characteristic	Symbol	Min	Typ	Max	Unit
Oscillator Frequency					
÷ 2, ÷ 4 options	f_{osc}	0	–	11.0	MHz
÷ 1 option	f_{osc}	0	–	5.5	MHz
Bit Time	t_{bit}	0.182	–	–	µs
Bit Cycle Time	t_{byte}	2.18	–	–	µs
$\overline{\text{IRQ}}$ and TIMER Pulse Width	t_{WL}, t_{WM}	2	–	–	t_{byte}
$\overline{\text{RESET}}$ Pulse Width	t_{RWL}	2	–	–	t_{byte}
Oscillator Pulse Width ÷ 2, ÷ 4 options	t_{OL}, t_{OH}	45	–	–	ns
÷ 1 option		90	–	–	ns

7.7 ELECTRICAL CHARACTERISTICS @ 3.3 V
(V_{CC} = 3.3 Vdc ± 10 % ; V_{SS} = 0 Vdc ; T_A = T_L to T_H unless otherwise noted).

Characteristic	Symbol	Min	Typ	Max	Unit
Output voltage					
(I_{Load} = 10 µA)	V_{OL}	–	–	0.1	V
(I_{Load} = −10 µA)	V_{OH}	V_{CC} – 0.1	–	–	
Output Low Voltage (I_{Load} = 0.4 mA) PA0-PA7, PB0-PB7, PC0-PC3	V_{OL}	–	–	0.4	V
Output High Voltage (I_{Load} = −0.4 mA) PA0-PA7, PB0-PB7, PC0-PC3	V_{OH}	V_{CC} −0.4	–	–	V
Input Low Voltage					
PA0-PA7, PB0-PB7, PC0-PC3, TIMER	V_{IL}	V_{SS}	–	0.3 V_{CC}	V
XTAL, MDS, $\overline{\text{IRQ}}$, $\overline{\text{RESET}}$	V_{IL}	V_{SS}	–	0.2 V_{CC}	
Input High Voltage					
PA0-PA7, PB0-PB7, PC0-PC3, TIMER	V_{IH}	0.7 V_{CC}	–	V_{CC}	V
XTAL, MDS, $\overline{\text{IRQ}}$, $\overline{\text{RESET}}$	V_{IH}	0.8 V_{CC}	–	V_{CC}	
Total Supply Current (No dc Loads, (No dc Loads, V_{IL} = 0.2 V ; V_{IH} = V_{CC} – 0.2 V)					
RUN Mode					
÷ 1 Option (f_{osc} = 3.0 MHz)	I_{DD}	–	0.7	TBD	mA
÷ 2 Option (f_{osc} = 6.0 MHz)	I_{DD}	–	0.9	TBD	mA
÷ 4 Option (f_{osc} = 6.0 MHz)	I_{DD}	–	0.6	TBD	mA
WAIT Mode (See Note)					
÷ 1 Option (f_{osc} = 3.0 MHz)	I_{DD}	–	0.25	TBD	mA
÷ 2 Option (f_{osc} = 6.0 MHz)	I_{DD}	–	0.40	TBD	mA
÷ 4 Option (f_{osc} = 6.0 MHz)	I_{DD}	–	0.35	TBD	mA
STOP Mode (See Note)	I_{DD}	–	0.5	2	µA
I/O Hi-Z Leakage Current (V_{in} = 0.4 V to V_{CC} = 0.4 V) Timer, PA0-PA7, PB0-PB7, PC0-PC3 (with no pull-downs)	I_{TSI}	−10	–	10	µA
Hi-Z State Input Current (V_{in} = V_{CC} – 0.4 V) PA0-PA7, PB0-PB7, PC0-PC3 (with pull-downs)	I_{TSI}	–	125	–	µA
Input Current $\overline{\text{IRQ}}$, $\overline{\text{RESET}}$, XTAL, MDS	I_{in}	−1	–	+ 1	µA
I/O Output Capacitance Timer, PA0-PA7, PB0-PB7, PC0-PC3	C_{out}	–	–	12	pF
Input Capacitance $\overline{\text{IRQ}}$, $\overline{\text{RESET}}$, XTAL, MDS	C_{in}	–	–	8	pF

NOTE : Test conditions for I_{DD} as follows :
XTAL input is a square wave from 0.2 V to V_{CC} – 0.2 V.
EXTAL output load = 10 pF
Circuit in self check-mode
In WAIT and STOP Modes, Port A is programmed as output, Ports B and C are programmed as inputs.
In STOP Mode : all inputs are tied to V_{IL} excepted $\overline{\text{IRQ}}$, $\overline{\text{RST}}$, MDS, XTAL, EXTAL which are tied to V_{IH}.

7.8 CONTROL TIMING @ 3.3 V

(V_{CC} = 3.3 Vdc ± 10 % ; V_{SS} = 0 Vdc ; T_A = T_L to T_H unless otherwise noted).

Characteristic	Symbol	Min	Typ	Max	Unit
Oscillator Frequency ÷ 2, ÷ 4 options ÷ 1 option	 f_{osc} f_{osc}	 0 0	 – –	 6.0 3.0	 MHz MHz
Bit Time	t_{bit}	0.333	–	–	µs
Bit Cycle Time	t_{byte}	4.0	–	–	µs
IRQ and TIMER Pulse Width	t_{WL}, t_{WM}	2	–	–	t_{byte}
RESET Pulse Width	t_{RWL}	2	–	–	t_{byte}
Oscillator Pulse Width ÷ 2, ÷ 4 options ÷ 1 option	t_{OL}, t_{OH}	80 160	– —	– —	ns ns

7.9 ELECTRICAL CHARACTERISTICS @ 2.2 V

(V_{CC} = 2.2 Vdc ± 10 % ; V_{SS} = 0 Vdc ; T_A = T_L to T_H unless otherwise noted).

Characteristic	Symbol	Min	Typ	Max	Unit
Output voltage (I_{Load} = 10 µA) (I_{Load} = −10 µA)	 V_{OL} V_{OH}	 – –	 0.1 V_{CC} - 0.1	 – –	 V
Output Low Voltage (I_{Load} = 0.2 mA) PA0-PA7, PB0-PB7, PC0-PC3	V_{OL}	–	0.4	–	V
Output High Voltage (I_{Load} = 0.2 mA) PA0-PA7, PB0-PB7, PC0-PC3	V_{OH}	–	V_{CC}- 0.4	–	V
Input Low Voltage PA0-PA7, PB0-PB7, PC0-PC3, TIMER XTAL, MDS, IRQ, RESET	 V_{IL} V_{IL}	 – –	 0.3 V_{CC} 0.2 V_{CC}	 – –	 V
Input High Voltage PA0-PA7, PB0-PB7, PC0-PC3, TIMER XTAL, MDS, IRQ, RESET	 V_{IH} V_{IH}	 – –	 0.7 V_{CC} 0.8 V_{CC}	 – –	 V
Total Supply Current (No dc Loads, (No dc Loads, V_{IL} = 0.2 V ; V_{IH} = V_{CC} − 0.2 V) RUN Mode ÷ 1 Option (f_{osc} = 1.0 MHz) ÷ 2 Option (f_{osc} = 2.0 MHz) ÷ 4 Option (f_{osc} = 2.0 MHz) WAIT Mode (See Note) ÷ 1 Option (f_{osc} = 1.0 MHz) ÷ 2 Option (f_{osc} = 2.0 MHz) ÷ 4 Option (f_{osc} = 2.0 MHz) STOP Mode (See Note)	 I_{DD} I_{DD} I_{DD} I_{DD} I_{DD} I_{DD} I_{DD}	 – – – – – – –	 150 175 115 50 80 70 0.5	 TBD TBD TBD TBD TBD TBD 2	 µA µA µA µA µA µA µA
I/O Hi-Z Leakage Current (V_{in} = 0.4 V to V_{CC} - 0.4 V) Timer, PA0-PA7, PB0-PB7, PC0-PC3 (with no pull-downs)	I_{TSI}	−10	–	10	µA
Hi-Z State Input Current (V_{in} = V_{CC} - 0.4 V) PA0-PA7, PB0-PB7, PC0-PC3 (with pull-downs)	I_{TSI}	–	75	–	µA
Input Current IRQ, RESET, XTAL, MDS	I_{in}	−1	–	+ 1	µA
I/O Output CapacitanceTimer, PA0-PA7, PB0-PB7, PC0-PC3	C_{out}	–	–	12	pF
Input Capacitance IRQ, RESET, XTAL, MDS	C_{in}	–	–	8	pF

NOTE : Test conditions for I_{DD} as follows :
XTAL input is a square wave from 0.2 V to V_{CC} − 0.2 V.
EXTAL output load = 10 pF
Device in self check-mode
In WAIT and STOP Modes, Port A is programmed as output, Ports B and C are programmed as inputs.
In STOP Mode : all inputs are tied to V_{IL} excepted IRQ, RST, MDS, XTAL, EXTAL which are tied to V_{IH}.

7.10 CONTROL TIMING @ 2.2 V

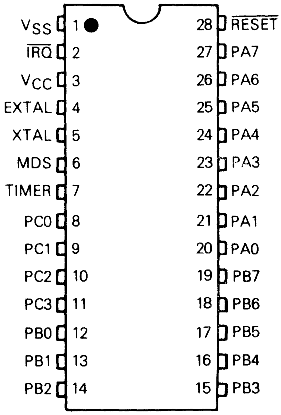
(V_{CC} = 2.2 Vdc ± 10 % ; V_{SS} = 0 Vdc ; T_A = T_L to T_H unless otherwise noted).

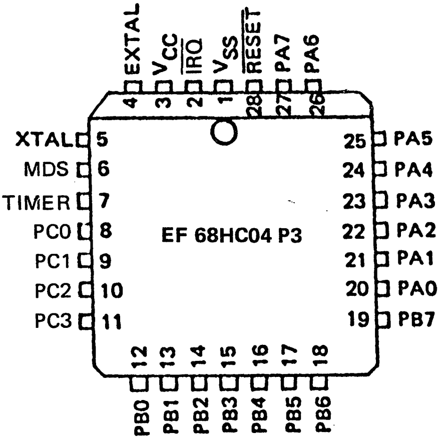
Characteristic	Symbol	Min	Typ	Max	Unit
Oscillator Frequency					
÷ 2, ÷ 4 options	f_{osc}	0	2.0	–	MHz
÷ 1 option	f_{osc}	0	1.0	–	MHz
Bit Time	t_{bit}	–	1.0	–	µs
Bit Cycle Time	t_{byte}	–	12.0	–	µs
$\overline{\text{IRQ}}$ and TIMER Pulse Width	t_{WL}, t_{WM}	2	–	–	t_{byte}
$\overline{\text{RESET}}$ Pulse Width	t_{RWL}	2	–	–	t_{byte}
Oscillator Pulse Width ÷ 2, ÷ 4 options	t_{OL}, t_{OH}	–	250	–	ns
÷ 1 option		–	500	–	ns

SECTION 8
MECHANICAL DATA

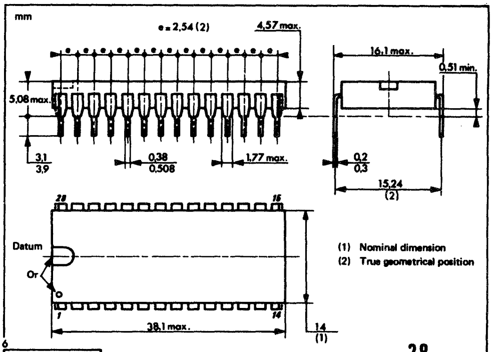
This section contains the pin assignment and package dimension diagrams for the EF68HC04P3 microcomputer.

8.1 PIN ASSIGNMENTS



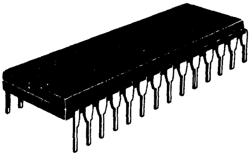


8.2 PHYSICAL DIMENSIONS

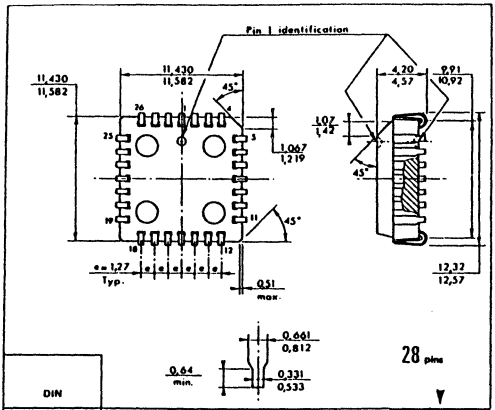


DIN				
A51E CEI	D.A.T.A.	JEDEC	F 144 SITELESC	CB-132

CB-132

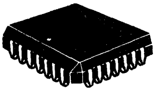


P SUFFIX



DIN				
CEI	D.A.T.A.	MO-047-AB JEDEC	SITELESC	CB-520

CB-520



FN SUFFIX

SECTION 9
ORDERING INFORMATION

9.1 INTRODUCTION

The following information is required when ordering a custom MCU. The information may be transmitted to THOMSON SEMICONDUCTEURS in the following media:

EPROM(s), ET2716 or ET2732
EFDOS/MDOS*, disk file

To initiate a ROM pattern for the MCU, it is necessary to first contact your local field service office, local sales person, or your local THOMSON SEMICONDUCTEURS representative.

9.1.1 EPROMs

A 2716 or 2732 type EPROM, programmed with the customer program (positive logic sense for address and data), may be submitted for pattern generation. Since all program and data space information will fit on one 2716 or 2732 EPROM, the EPROM must be programmed as follows in order to emulate the EF68HC04P3 MCU. **Start the data space ROM at EPROM address $018 and start program space ROM at EPROM address $960 and continue to memory space $FFF. All unused bytes, including the user's space, must be set to zero.** For shipment to THOMSON SEMICONDUCTEURS the EPROMs should be placed in a conductive IC carrier and packed securely. Do not use styrofoam.

9.1.2 EFDOS/MDOS* Disk File

An EFDOS/MDOS* disk, programmed with the customer program (positive logic sense for address and data), may be submitted for pattern generation. **When using the EFDOS/MDOS* disk, include the entire memory image of both data and program space. All unused bytes, including the user's space, must be set to zero.**

9.2 VERIFICATION MEDIA

All original pattern media (EPROMs or floppy disk) are filed for contractual purposes and are not returned. A computer listing of the ROM code will be generated and returned along with a listing verification form. The listing should be thoroughly checked and the verification form completed, signed, and returned to THOMSON SEMICONDUCTEURS. The signed verification form constitutes the contractural agreement for creation of the customer mask. If desired, THOMSON SEMICONDUCTEURS will program a blank 2716, 2732, or EFDOS disk **(supplied by the customer)** from the data file used to create the custom mask to aid in the verification process.

* Requires prior factory approval.

9.3 ROM VERIFICATION UNITS (RVUs)

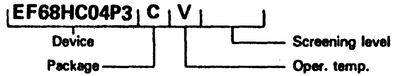
Ten MCUs containing the customer's ROM pattern will be sent for program verification. These units will have been made using the custom mask but are for the purpose of ROM verification only. For expediency they are usually unmarked, packaged in ceramic, and tested only at room temperature and five volts. These RVUs are included in the mask charge and are not production parts. These RVUs are not backed nor guaranteed by THOMSON SEMICONDUCTEURS Quality Assurance.

9.4 FLEXIBLE DISKS

The disk media submitted must be single-sided, EFDOS/MDOS* compatible floppies. The customer must clearly label the disk with the ROM pattern file name. The minimum EFDOS/MDOS* system files as well as the absolute binary object files (filename.LO type of file) and filename. DO if necessary from the EF6804 cross-assembler must be on the disk. An object file made from a memory dump, using the ROLLOUT command is also admissable. Consider submitting a source listing as well as : filename,..LX (DEVICE®/EXORciser® loadable.format) . This file will of course be kept confidential and is used 1) to speed up the process in house if any problems arise, and 2) to speed up our customer to factory interface if a user finds any software errors and needs assistance quickly from the factory representative.

* Requires prior factory approval.
DEVICE® is a registered trademark from THOMSON SEMICONDUCTEURS.
EXORciser® is a registered trademark of MOTOROLA INC.

ORDERING INFORMATION



The table below horizontally shows all available suffix combinations for package, operating temperature and screening level. Other possibilities on request.

DEVICE	PACKAGE					OPER. TEMP			SCREENING LEVEL			
	C		P		FN	L*	D	V**	Std	D	T	
EF68HC04P3			●		●	●	●		●			

Examples : EF68HC04P3P, EF68HC04P3FN.

Package: C: Ceramic DIL, DIL, P: Plastic DIL, FN: PLCC.
Oper. temp.: L*: 0°C to + 70°C, D: — 25°C to + 70°C, V: — 40°C to + 85°C, T: — 40°C to + 105°C.
Screening level: Std: (no-end suffix), D: NFC 96883 level D.
* May be omitted.

A reduced-packaged version of the EF68HC04P3 (28 pins) will be available in a 20-pin package : EF68HC04J3.

These specifications are subject to change without notice.
Please inquire with our sales offices about the availability of the different products.

Ref.DSEF68HC04P3

Printed in France

HOMSON
COMPONENTS
EMICONDUCTEURS

EF6805P2

8-BIT MICROCOMPUTER

ADVANCE INFORMATION

HMOS

The EF6805P2 Microcomputer Unit (MCU) is a member of the EF6805 Family of low-cost single-chip microcomputers. This 8-bit microcomputer contains a CPU, on-chip CLOCK, ROM, RAM, I/O, and TIMER. It is designed for the user who needs an economical microcomputer with the proven capabilities of the 6800 -based instruction set.A comparison of the key features of several members of the 6805 Family is shown at the end of this data sheet. The following are some of the hardware and software highlights of the EF6805P2 MCU.

HARDWARE FEATURES

- 8-Bit Architecture
- 64 Bytes of RAM
- Memory Mapped I/O
- 1100 Bytes of User ROM
- 20 TTL/CMOS Compatible Bidirectional I/O Lines (8 Lines are LED Compatible)
- On-Chip Clock Generator
- Self-Check Mode
- Zero Crossing Detection
- Master Reset
- Complete Development System Support on DEVICE®
- 5 V Single Supply

SOFTWARE FEATURES

- Similar to 6800 Family
- Byte Efficient Instruction Set
- Easy to Program
- True Bit Manipulation
- Bit Test and Branch Instruction
- Versatile Interrupt Handling
- Versatile Index Register
- Powerful Indexed Addressing for Tables
- Full Set of Conditional Branches
- Memory Usable as Register/Flags
- Single Instruction Memory Examine/Change
- 10 Powerful Addressing Modes
- All Addressing Modes Apply to ROM, RAM, and I/O

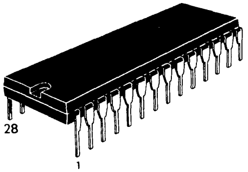
USER SELECTABLE OPTIONS

- Internal 8-Bit Timer with Selectable Clock Source (External Timer Input or Internal Machine Clock)
- Timer Prescaler Option (7 Bits, 2^n)
- 8 Bidirectional I/O Lines with TTL or TTL/CMOS Interface Option
- Crystal or Low-Cost Resistor Oscillator Option
- Low Voltage Inhibit Option
- Vectored Interrupts: Timer, Software, and External

DEVICE® is THOMSON SEMICONDUCTEURS' development/emulation tool.

CASES

CB-132

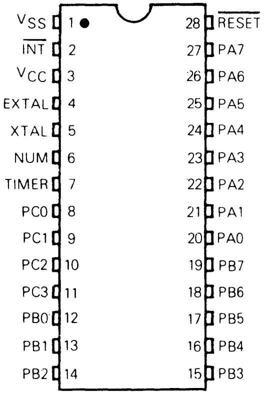


P SUFFIX
PLASTIC PACKAGE

CB-520

FN SUFFIX
PLCC 28

PIN ASSIGNMENT



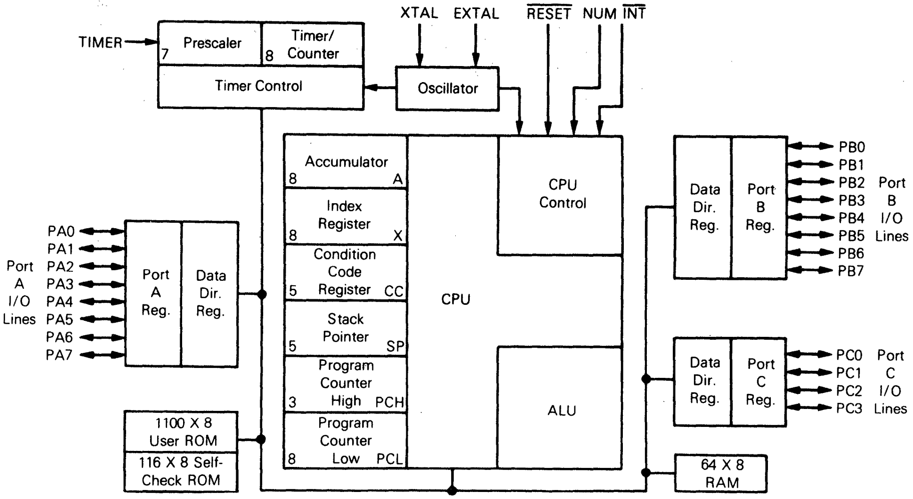


FIGURE 1 – EF6805P2 HMOS MICROCOMPUTER BLOCK DIAGRAM

MAXIMUM RATINGS

Rating	Symbol	Value	Unit
Supply Voltage	V_{CC}	−0.3 to +7.0	V
Input Voltage (Except Pin 6)	V_{in}	−0.3 to +7.0	V
Operating Temperature Range (T_L to T_H) V suffix	T_A	0 to + 70 −40 to + 85	°C
Storage Temperature Range	T_{stg}	−55 to +150	°C
Junction Temperature Plastic PLCC	T_J	 150 150	°C

This device contains circuitry to protect the inputs against damage due to high static voltages or electric fields, however, it is advised that normal precautions be taken to avoid application of any voltage higher than maximum rated voltages to this high-impedance circuit. For proper operation it is recommended that V_{in} and V_{out} be constrained to the range $V_{SS} \leq (V_{in}$ or $V_{out}) \leq V_{CC}$. Reliability of operation is enhanced if unused inputs are tied to an appropriate logic voltage level (e.g., either V_{SS} or V_{CC}).

THERMAL CHARACTERISTICS

Characteristic	Symbol	Value	Unit
Thermal Resistance Plastic PLCC	θ_{JA}	 70 110	°C/W

POWER CONSIDERATIONS

The average chip-junction temperature, T_J, in °C can be obtained from:

$$T_J = T_A + (P_D \bullet \theta_{JA}) \qquad (1)$$

Where:

$T_A \equiv$ Ambient Temperature, °C

$\theta_{JA} \equiv$ Package Thermal Resistance, Junction-to-Ambient, °C/W

$P_D \equiv P_{INT} + P_{PORT}$

$P_{INT} \equiv I_{CC} \times V_{CC}$, Watts – Chip Internal Power

$P_{PORT} \equiv$ Port Power Dissipation, Watts – User Determined

For most applications $P_{PORT} \ll P_{INT}$ and can be neglected. P_{PORT} may become significant if the device is configured to drive Darlington bases or sink LED loads.

An approximate relationship between P_D and T_J (if P_{PORT} is neglected) is:

$$P_D = K \div (T_J + 273°C) \qquad (2)$$

Solving equations 1 and 2 for K gives:

$$K = P_D \bullet (T_A + 273°C) + \theta_{JA} \bullet P_D^2 \qquad (3)$$

Where K is a constant pertaining to the particular part. K can be determined from equation 3 by measuring P_D (at equilibrium) for a known T_A. Using this value of K the values of P_D and T_J can be obtained by solving equations (1) and (2) iteratively for any value of T_A.

ELECTRICAL CHARACTERISTICS (V_{CC} = +5.25 Vdc ±0.5 Vdc, V_{SS} = 0 Vdc, $T_A = T_L$ to T_H unless otherwise noted)

Characteristic	Symbol	Min	Typ	Max	Unit
Input High Voltage	V_{IH}				V
$\overline{\text{RESET}}$ ($4.75 \le V_{CC} \le 5.75$)		4.0	–	V_{CC}	
($V_{CC} < 4.75$)		$V_{CC} - 0.5$	–	V_{CC}	
$\overline{\text{INT}}$ ($4.75 \le V_{CC} \le 5.75$)		4.0	*	V_{CC}	
($V_{CC} < 4.75$)		$V_{CC} - 0.5$	*	V_{CC}	
All Other (Exept TIMER)		2.0	–	V_{CC}	
Input High Voltage Timer	V_{IH}				V
Timer Mode		2.0	–	$V_{CC} + 1$	
Self-Check Mode		9.0	10.0	15 0	
Input Low Voltage	V_{IL}				V
$\overline{\text{INT}}$		V_{SS}	*	1.5	
All Other		V_{SS}	–	0.8	
$\overline{\text{RESET}}$ Hysteresis Voltage (See Figures 10, 11, and 12)					V
"Out of Reset"	V_{IRES+}	2.1	–	4.0	
"Into Reset"	V_{IRES-}	0.8	–	2.0	
$\overline{\text{INT}}$ Zero Crossing Input Voltage, Through a Capacitor	V_{INT}	2.0	–	4.0	$V_{ac\ p-p}$
Internal Power Dissipation – No Port Loading V_{CC} = 5.75 V, T_A = 0°C	P_{INT}	–	400	690	mW
Input Capacitance	C_{in}				pF
XTAL		–	25	–	
All Other		–	10	–	
Low Voltage Recover	V_{LVR}	–	–	4.75	V
Low Voltage Inhibit 0 to +70°C	V_{LVI}	2.75	3.5	–	V
–40 to +85°C		3.1	3.5	–	
Input Current	I_{in}				μA
TIMER (V_{in} = 0.4 V)		–	–	20	
$\overline{\text{INT}}$ (V_{in} = 2.4 V to V_{CC})		–	20	50	
EXTAL (V_{in} = 2.4 V to V_{CC}, Crystal Option)		–	–	10	
(V_{in} = 0.4 V, Crystal Option)		–	–	– 1600	
$\overline{\text{RESET}}$ (V_{in} = 0.8 V) (External Capacitor Charging Current)		–4.0	–	– 40	

* Due to internal biasing, this input (when unused) floats to approximately 2.0 Vdc.

PORT DC ELECTRICAL CHARACTERISTICS (V_{CC} = +5.25 Vdc ±0.5 Vdc, V_{SS} = 0 Vdc, $T_A = T_L$ to T_H unless otherwise noted)

Characteristic	Symbol	Min	Typ	Max	Unit
Port A with CMOS Drive Enabled					
Output Low Voltage, I_{Load} = 1.6 mA	V_{OL}	–	–	0.4	V
Output High Voltage, I_{Load} = – 100 μA	V_{OH}	2.4	–	–	V
Output High Voltage, I_{Load} = – 10 μA	V_{OH}	$V_{CC} - 1.0$	–	–	V
Input High Voltage, I_{Load} = –300 μA (max.)	V_{IH}	2.0	–	V_{CC}	V
Input LowVoltage, I_{Load} = –500 μA (max.)	V_{IL}	V_{SS}	–	0.8	V
Hi-Z State Input Current (V_{in} = 2.0 V to V_{CC})	I_{IH}	–	–	–300	μA
Hi-Z State Input Current (V_{in} = 0.4 V)	I_{IL}	–	–	–500	μA
Port B					
Output Low Voltage, I_{Load} = 3.2 mA	V_{OL}	–	–	0.4	V
Output Low Voltage, I_{Load} = 10 mA (sink)	V_{OL}	–	–	1.0	V
Output High Voltage, I_{Load} = –200 μA	V_{OH}	2.4	–	–	V
Darlington Current Drive (Source), V_O = 1.5 V	I_{OH}	–1.0	–	–10	mA
Input High Voltage	V_{IH}	2.0	–	V_{CC}	V
Input Low Voltage	V_{IL}	V_{SS}	–	0.8	V
Hi-Z State Input Current	I_{TSI}	–	2	10	μA
Port C and Port A with CMOS Drive Disabled					
Output Low Voltage, I_{Load} = 1.6 mA	V_{OL}	–	–	0.4	V
Output High Voltage, I_{Load} = – 100 μA	V_{OH}	2.4	–	–	V
Input High Voltage	V_{IH}	2.0	–	V_{CC}	V
Input Low Voltage	V_{IL}	V_{SS}	–	0.8	V
Hi-Z State Input Current	I_{TSI}	–	2	10	μA

SWITCHING CHARACTERISTICS (V_{CC} = +5.25 Vdc ±0.5 Vdc, V_{SS} = 0 Vdc, $T_A = T_L$ to T_H unless otherwise noted)

Characteristic		Symbol	Min	Typ	Max	Unit
Oscillator Frequency	EF6805P2 EF68A05P2 EF68B05P2	f_{osc}	0.4 0.4 0.4	– – –	4.2 6.0 8.0	MHz
Cycle Time ($4/f_{osc}$)		t_{cyc}	0.95	–	10	μs
$\overline{\text{INT}}$ and TIMER Pulse Width (See Interrupt Section)		t_{WL}, t_{WH}	t_{cyc}+250	–	–	ns
$\overline{\text{RESET}}$ Pulse Width		t_{RWL}	t_{cyc}+250	–	–	ns
$\overline{\text{RESET}}$ Delay Time (External Capacitance = 1.0 μF)		t_{RHL}	–	100	–	ms
$\overline{\text{INT}}$ Zero Crossing Detection Input Frequency (±5° Accuracy)		f_{INT}	0.03	–	1.0	kHz
External Clock Input Duty Cycle (EXTAL)		–	40	50	60	%

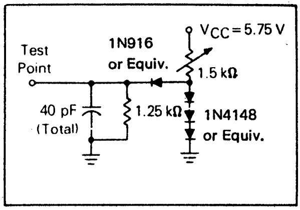


FIGURE 2 – TTL EQUIVALENT TEST LOAD (PORT B)

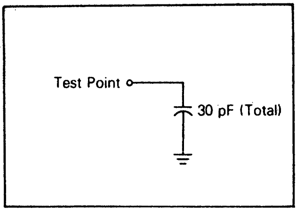


FIGURE 3 – CMOS EQUIVALENT TEST LOAD (PORT A)

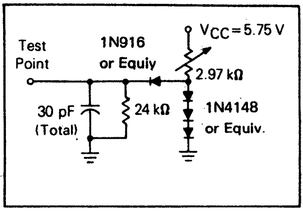


FIGURE 4 – TTL EQUIVALENT TEST LOAD (PORTS A AND C)

SIGNAL DESCRIPTION

The input and output signals for the MCU, shown in Figure 1, are described in the following paragraphs.

V_{CC} AND V_{SS}

Power is supplied to the MCU using these two pins. V_{CC} is power and V_{SS} is the ground connection.

$\overline{\text{INT}}$

This pin provides the capability for asynchronously applying an external interrupt to the MCU. Refer to Interrupts section for additional information.

XTAL AND EXTAL

These pins provide connections to the on-chip clock oscillator circuit. A crystal, a resistor, or an external signal, depending on the user selectable manufacturing mask option, can be connected to these pins to provide a system clock source with various stability/cost tradeoffs. Lead lengths and stray capacitance on these two pins should be minimized. Refer to Internal Clock Generator Options section for recommendations about these inputs.

TIMER

This pin allows an external input to be used to decrement the internal timer circuitry. Refer to Timer section for additional information about the timer circuitry.

$\overline{\text{RESET}}$

This pin allows resetting of the MCU at times other than the automatic resetting capability already in the MCU. Refer to Resets section for additional information.

NUM

This pin is not for user application and must be connected to V_{SS}.

INPUT/OUTPUT LINES (PA0-PA7, PB0-PB7, PC0-PC3)

These 20 lines are arranged into two 8-bit ports (A and B) and one 4-bit port (C). All lines are programmable as either inputs or outputs under software control of the data direction registers. Refer to Inputs/Outputs section for additional information.

MEMORY

As shown in Figure 5, the MCU is capable of addressing 2048 bytes of memory and I/O registers with is program counter. The EF6805P2 MCU has implemented 1288 of these locations. This consists of: 1100 bytes of user ROM, (from $080 to $OFF and from $3CO to $783) 116 bytes of self-check ROM, 64 bytes of user RAM, 6 bytes of port I/O, and 2 timers registers. The ROM division allows 128 bytes of ROM to be addressed with direct instructions.

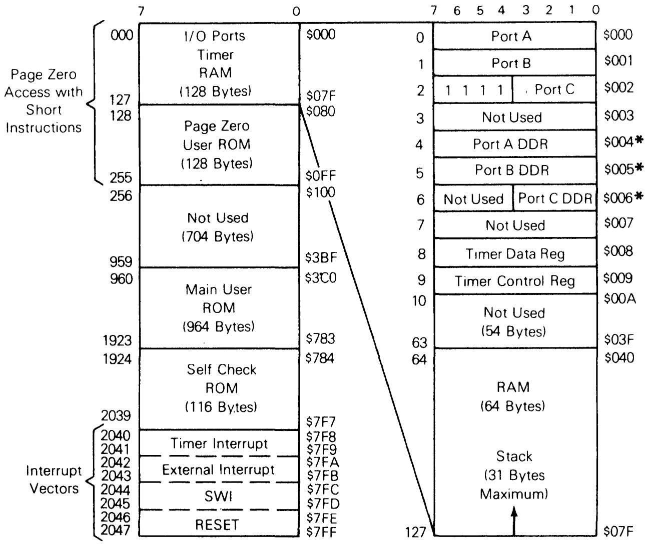
The stack area is used during the processing of interrupt and subroutine calls to save the processor state. The register contents are pushed onto the stack in the order shown in Figure 6. Because the stack pointer decrements during pushes, the low order byte (PCL) of the program counter is stacked first; then the high order three bits (PCH) are stacked. This ensures that the program counter is loaded correctly, during pulls from the stack, since the stack pointer increments during pulls. A subroutine call results in only the program counter (PCL, PCH) contents being pushed onto the stack. The remaining CPU registers are not pushed.

CENTRAL PROCESSING UNIT

The CPU of the EF6805 Family is implemented independently from the I/O or memory configuration. Consequently, it can be treated as an independent central processor communicating with I/O and memory vía internal address, data, and control buses.

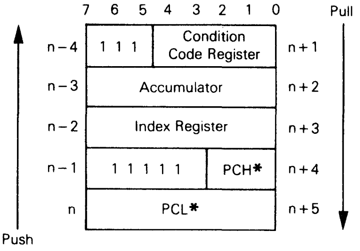
REGISTERS

The 6805 Family CPU has five registers available to the programmer. They are shown in Figure 7 and are explained in the following paragraphs.



*Caution: Data direction registers (DDRs) are write-only; they read as $FF.

FIGURE 5 – MCU ADDRESS MAP



*For subroutine calls, only PCL and PCH are stacked.

FIGURE 6 – INTERRUPT STACKING ORDER

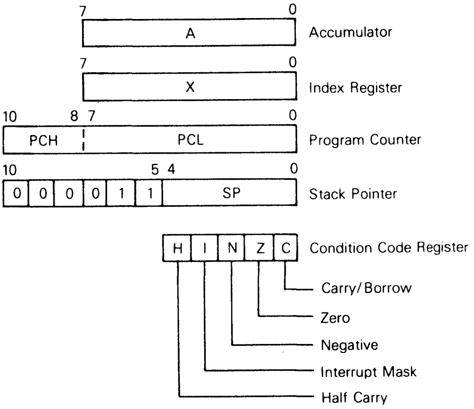


FIGURE 7 – PROGRAMMING MODEL

ACCUMULATOR (A)

The accumulator is a general purpose 8-bit register used to hold operands and results of arithmetic calculations or data manipulations.

INDEX REGISTER (X)

The index register is an 8-bit register used for the indexed addressing mode. It contains an 8-bit value that may be added to an instruction value to create an effective address. The index register can also be used for data manipulations using the read-modify-write instructions. The index register may also be used as a temporary storage area.

PROGRAM COUNTER (PC)

The program counter is an 11-bit register that contains the address of the next instruction to be executed.

STACK POINTER (SP)

The stack pointer is an 11-bit register that contains the address of the next free location on the stack. Initially, the stack pointer is set to location $07F and is decremented as data is pushed onto the stack and incremented as data is pulled from the stack. The six most significant bits of the stack pointer are permanently configured to 000011. During an MCU reset or the reset stack pointer (RSP) instruction, the stack pointer is set to location $07F. Subroutines and interrupts may be nested down to location $061 (31 bytes maximum) which allows the programmer to use up to 15 levels of subroutine calls.

CONDITION CODE REGISTER (CC)

The condition code register is a 5-bit register in which four bits are used to indicate the results of the instruction just executed. These bits can be individually tested by a program and specific action taken as a result of their state. Each individual condition code register bit is explained in the following paragraphs.

HALF CARRY (H) – Set during ADD and ADC instructions to indicate that a carry occurred between bits 3 and 4.

INTERRUPT (I) – This bit is set to mask (disable) the timer and external interrupt ($\overline{\text{INT}}$). If an interrupt occurs while this bit is set, the interrupt is latched and is processed as soon as the interrupt is cleared.

NEGATIVE (N) – Used to indicate that the result of the last arithmetic, logical, or data manipulation was negative (bit 7 in result equal to a logical one).

ZERO (Z) – Used to indicate that the result of the last arithmetic, logical, or data manipulation was zero.

CARRY/BORROW (C) – Used to indicate that a carry or borrow out of the arithmetic logic unit (ALU) occurred during the last arithmetic operation. This bit is also affected during bit test and branch instructions plus shifts and rotates.

TIMER

The EF6805P2 MCU timer circuitry is shown in Figure 8. The 8-bit counter may be loaded under program control and is decremented toward zero by the clock input (prescaler output). When the timer reaches zero, the timer interrupt request bit (bit 7) in the timer control register (TCR) is set. The timer interrupt can be masked (disabled) by setting the timer interrupt mask bit (bit 6) in the TCR. The interrupt bit (I bit) in the condition code register also prevents a timer interrupt from being processed. The MCU responds to this interrupt by saving the present CPU state on the stack, fetching the timer interrupt vector from locations $7F8 and $7F9, and executing the interrupt routine; see the Interrupts section. THE TIMER INTERRUPT REQUEST BIT MUST BE CLEARED BY SOFTWARE.

The clock input to the timer can be from an external source (decrementing of timer counter occurs on a positive transition of the external source) applied to the TIMER input pin or it can be the internal $\phi 2$ signal. Three machine cycles are required for a change in state of the TIMER pin to decrement the timer prescaler. The maximum frequency of a signal that can be recognized by the TIMER or $\overline{\text{INT}}$ pin logic is dependent on the parameter labeled t_{WL}, t_{WH}. The pin logic that recognizes the high (or low) state on the pin must also recognize the low state on the pin in order to "re-arm" the internal logic. Therefore, the period can be calculated as follows: (assumes 50/50 duty cycle for a given period)

$$t_{cyc} \times 2 + 250 \text{ ns} = \text{period} = \frac{1}{\text{freq}}$$

The period is not simply $t_{WL} + t_{WH}$. This computation is allowable, but it does reduce the maximum allowable frequency by defining an unnecessarily longer period (250 ns twice).

When the $\phi 2$ signal is used as the source, it can be gated by an input applied to the TIMER input pin allowing the user to easily perform pulse-width measurements. (NOTE: For

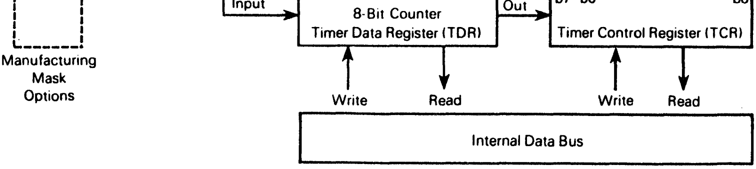


FIGURE 8 – TIMER BLOCK DIAGRAM

ungated $\phi 2$ clock inputs to the timer prescaler, the TIMER pin should be tied to V_{CC}.) The source of the clock input is one of the mask options that is specified before manufacture of the MCU.

A prescaler option can be applied to the clock input that extends the timing interval up to a maximum of 128 counts before decrementing the counter. This prescaling mask option is also specified before manufacture. To avoid truncation errors, the prescaler is cleared when bit 3 of the timer counter register is written to a logic one. (This bit always needs a logic 0).

The timer continues to count past zero, falling from $00 to$FF and then continuing the countdown. Thus, the counter can be read at any time by reading the timer data register (TDR). This allows a program to determine the length of time since a timer interrupt has occurred without disturbing the counting process.

At power-up or reset, the prescaler and counter are initialized with all logical ones, the timer interrupt request bit (bit 7) is cleared and the timer interrupt mask bit (bit 6) is set.

SELF-CHECK

The self-check capability of the EF6805P2 MCU provides an internal check to determine if the part is functional. Connect the MCU as shown in Figure 9 and monitor the output of port C bit 3 for an oscillation of approximately 7 Hz. A 10volt level on the TIMER input, pin 7, energizes the ROM-based self-check feature. The self-check program exercises the RAM, ROM, TIMER, interrupts, and I/O ports.

RESETS

The MCU can be reset three ways: by initial power-up, by the external reset input ($\overline{RESET}$) and by optional, internal, low-voltage detect circuits. The $\overline{RESET}$ input consists mainly of a Schmitt trigger that senses the $\overline{RESET}$ line logic level. A typical reset Schmitt trigger hysteresis curve is shown in Figure 11. The Schmitt trigger provides an internal reset voltage if it senses a logical zero on the $\overline{RESET}$ pin.

POWER-ON RESET (POR)

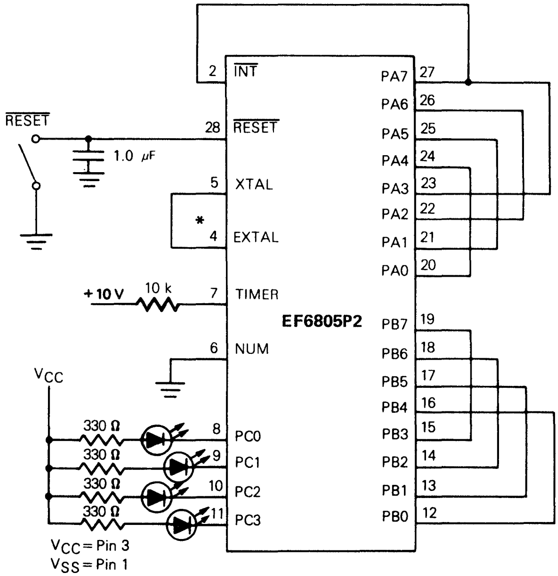
An internal reset is generated upon power-up that allows the internal clock generator to stabilize. A delay of t_{RHL} milliseconds is required before allowings the $\overline{RESET}$ input to go high. See the power and reset timing diagram (see Figure 10). Connecting a capacitor to the $\overline{RESET}$ input (see Figure 12) typically provides sufficient delay. During power-up, the Schmitt trigger switches on (removes reset) when $\overline{RESET}$ rise to $V_{IRES}+$.

EXTERNAL RESET INPUT

The MCU is reset when a logic zero is applied to the $\overline{RESET}$ input for a period longer than one machine cycle (t_{cyc}). Under this type of reset, the Schmitt trigger switches off are $V_{IRES}-$ to provide an internal reset voltage.

LOW VOLTAGE INHIBIT (LVI)

The optional low-voltage detection circuit causes a reset of the MCU if the power supply voltage falls below a certain level (V_{LVI}). The only requirement is that the V_{CC} must remain at or below the V_{LVI} threshold for one t_{cyc} minimum.



*This connection depends on the clock oscillator user selectable mask option. Use crystal if that option is selected.

Self-Check Error Patterns

PC1	PC0	Problem
0	0	Interrupt Failure
0	1	Bad Port A or Port B
1	0	Bad RAM
1	1	Bad RAM
All 4 LEDs Flashing		Good Device

NOTE

When PC1 or PC0 is 0, the LED is on.

FIGURE 9 – SELF-CHECK CONNECTIONS

In typical applications, the V_{CC} bus filter capacitor will eliminate negative-going voltage glitches of less than one t_{cyc}. The output from the low-voltage detector is connected directly to the internal reset circuitry. It also forces the RESET pin low via a strong discharge device through a resistor. The internal reset is removed once the power supply voltage rises above a recovery level (V_{LVR}) at which time a normal power-one reset occurs.

INTERNAL CLOCK GENERATORS OPTIONS

The internal clock generator circuit is designed to require a minimum of external components. A crystal, a resistor, a jumper wire, or an external signal may be used to generate a system clock with various stability/cost tradeoffs.

A manufacturing mask option is used to select crystal or resistor operation.

The different connection methods are shown in Figure 13. Crystal specifications and suggested PC board layouts are given in Figure 14. A resistor selection graph is given in Figure 15.

The crystal oscillator start-up time is a function of many variables: crystal parameters (especially R_S, oscillator load capacitances, IC parameters, ambient temperature, supply voltage and supply voltage turn-on time). To ensure rapid oscillator start-up, neither the crystal characteristics nor the load capacitances should exceed recommendations.

When utilizing the on-board oscillator, the MCU should remain in the reset condition ($\overline{\text{RESET}}$ pin voltage below V_{IRES+}) until the oscillator has stabilized at its operating frequency. Several factors are involved in calculating current specifications.

One V_{CC} minimum is reached, the external $\overline{\text{RESET}}$ capacitor will begin to charge at a rate dependent on the capacitor value. The charging current is supplied from V_{CC} through a large resistor, so it functions almost like a constant current source until the reset voltages rises above V_{IRES+}. Therefore, the $\overline{\text{RESET}}$ pin will charge a approximately

$$(V_{IRES+}) \cdot C_{ext} = I_{RES} \cdot t_{RHL}$$

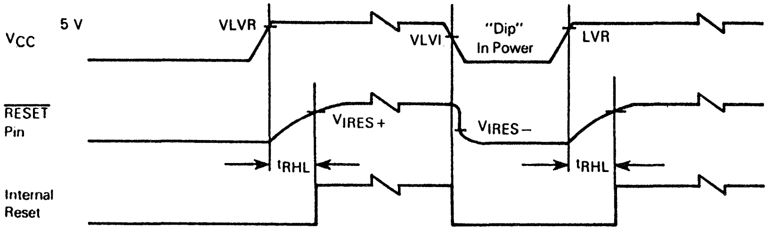


FIGURE 10 – POWER AND RESET TIMING

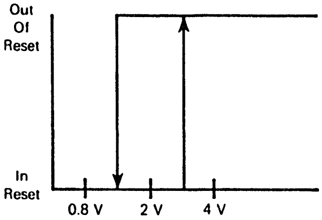


FIGURE 11 – TYPICAL RESET SCHMITT TRIGGER HYSTERESIS

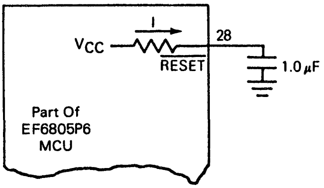
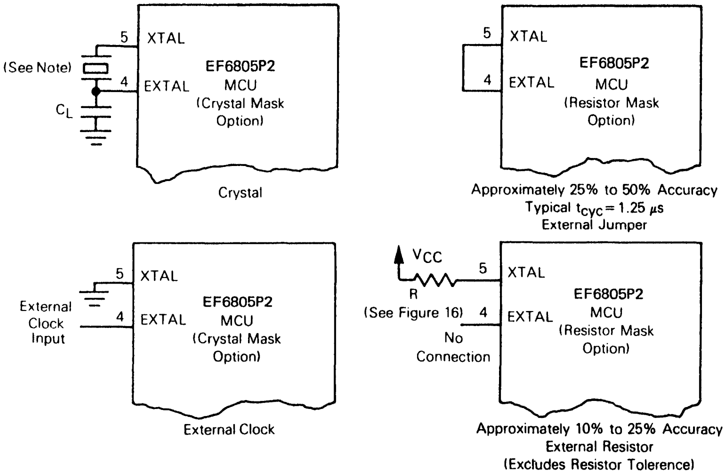


FIGURE 12 – POWER-UP RESET DELAY CIRCUIT



Crystal

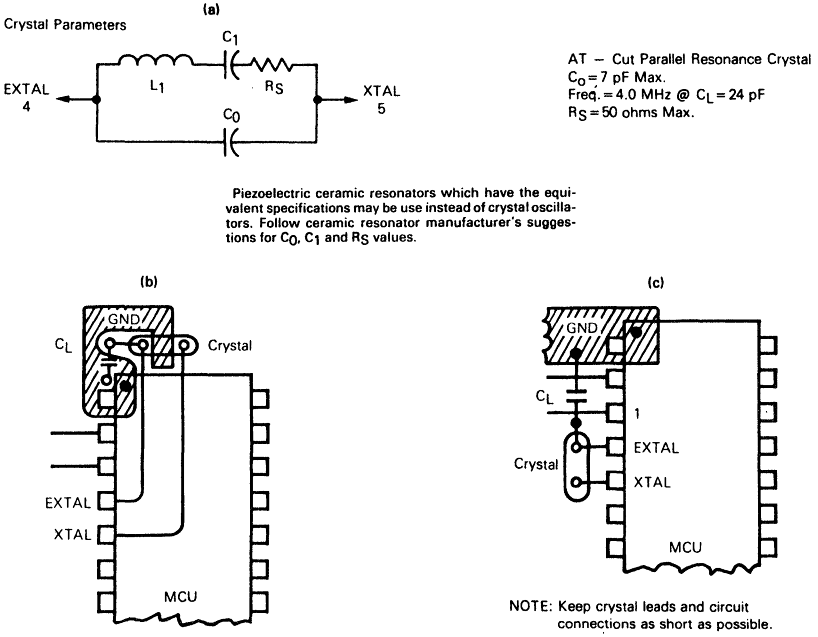
Approximately 25% to 50% Accuracy
Typical $t_{cyc} = 1.25\ \mu s$
External Jumper

External Clock

Approximately 10% to 25% Accuracy
External Resistor
(Excludes Resistor Tolerence)

NOTE: The recommended C_L value with a 4.0 MHz crystal is 27 pF, maximum, including system distributed capacitance. There is an internal capacitance of approximately 25 pF on the XTAL pin. For crystal frequencies other than 4 MHz; the total capacitance on each pin should be scaled as the inverse of the frequency ratio. For example, with a 2 MHz crystal, use approximately 50 pF on EXTAL and approximately 25 pF on XTAL. The exact value depends on the Motional-Arm parameters of the crystal used.

FIGURE 13 – CLOCK GENERATOR OPTIONS



AT – Cut Parallel Resonance Crystal
$C_0 = 7$ pF Max.
Freq. = 4.0 MHz @ $C_L = 24$ pF
$R_S = 50$ ohms Max.

Piezoelectric ceramic resonators which have the equivalent specifications may be use instead of crystal oscillators. Follow ceramic resonator manufacturer's suggestions for C_0, C_1 and R_S values.

NOTE: Keep crystal leads and circuit connections as short as possible.

FIGURE 14 – CRYSTAL MOTIONAL ARM PARAMETERS AND SUGGESTED PC BOARD LAYOUT

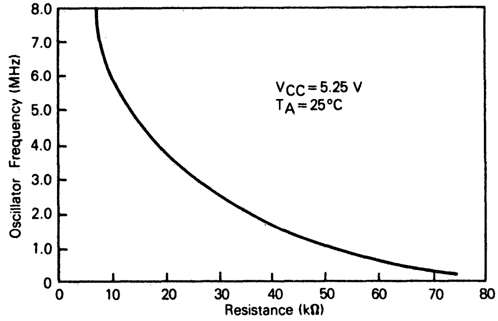


FIGURE 15 – TYPICAL FREQUENCY SELECTION FOR RESISTOR OSCILLATOR OPTION

INTERRUPTS

The EF6805P2 MCU can be interrupted three different ways: through the external interrupt (<span>$\overline{\text{INT}}$</span>) input pin, the internal timer interrupt request, or the software interrupt instruction (SWI). When any interrupt occurs, the current instruction (including SWI) is completed, processing is suspended, the present CPU state is pushed onto the stack, the interrupt bit (I) in the condition code register is set, the address of the interrupt routine is obtained from the appropriate interrupt vector address and the interrupt routine is executed. Stacking the CPU registers, setting the I bit, and vector fetching requires a total of 11 t_{cyc} periods for completion.

A flowchart of the interrupt sequence is shown in Figure 16. The interrupt service routine must end with a return from interrupt (RTI) instruction which allows the MCU to resume processing of the program prior to the interrupt (by unstacking the previous CPU state). Unlike $\overline{\text{RESET}}$, hardware interrupts do not cause the current instruction execution to be halted, but are considered pending until the current instruction execution is complete.

When the current instruction is complete, the processor checks all pending hardware interrupts and if unmasked, proceeds with interrupt processing; otherwise, the next instruction is fetched and executed. Note that masked interrupts are latched for later interrupt service.

If both an external interrupt and a timer interrupt are pending at the end of an instruction execution, the external interrupt is serviced first. The SWI is executed as any other instruction.

The external interrupt is internally synchronized and then latched on the falling edge of $\overline{\text{INT}}$. A sinusoidal input signal (f_{INT} maximum) can be used to generate an external interrupt, as shown in Figure 17(a), for use as a zero-crossing detector (for negative transitions of the ac sinusoid). This allows applications such as servicing time-of-day routines and engaging/disengaging ac power control devices. Off-chip full-wave rectification provides an interrupt at every zero crossing of the ac signal and thereby provides a 2f clock.

For digital applications, the $\overline{\text{INT}}$ pin can be driven by a digital signal. The maximum frequency of a signal that can be recognized by the TIMER or $\overline{\text{INT}}$ pin logic is dependent on the parameter labeled t_{WL}, t_{WH}. The pin logic that recognizes the high (or low) state on the pin must also recognize the low (or high) state on the pin in order to "rearm" the internal logic. Therefore, the period can be calculated as follows: (assumes 50/50 duty cycle for a given period)

$$t_{cyc}x2 + 250 \text{ ns} = \text{period} = \frac{1}{\text{freq}}$$

The period is not simply $t_{WL} + t_{WH}$. This computation is allowable, but it does reduce the maximum allowable frequency by defining an unnecessarily longer period (250 ns twice). See Figure 17(b).

A software interrupt (SWI) is an executable instruction which is executed regardless of the state of the I bit in the condition code register. Note that if the I bit is zero, SWI executes after the other interrupts. SWIs are usually used as break-points for debugging or as system calls.

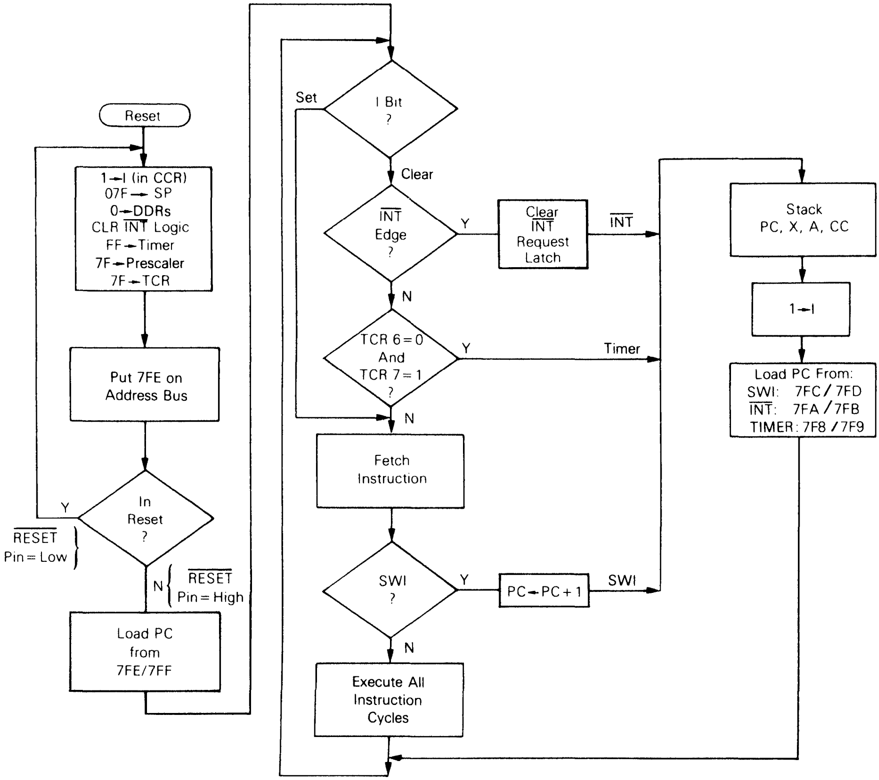

Reset
1→I (in CCR)
07F→SP
0→DDRs
CLR $\overline{\text{INT}}$ Logic
FF→Timer
7F→Prescaler
7F→TCR
Put 7FE on Address Bus
In Reset ?
Y
N
$\overline{\text{RESET}}$ Pin = Low
$\overline{\text{RESET}}$ Pin = High
Load PC from 7FE/7FF
Set
I Bit ?
Clear
$\overline{\text{INT}}$ Edge ?
Y
N
Clear $\overline{\text{INT}}$ Request Latch
$\overline{\text{INT}}$
TCR 6 = 0 And TCR 7 = 1 ?
Y
N
Timer
Fetch Instruction
SWI ?
Y
N
PC←PC + 1
SWI
Execute All Instruction Cycles
Stack PC, X, A, CC
1→I
Load PC From:
SWI: 7FC/7FD
$\overline{\text{INT}}$: 7FA/7FB
TIMER: 7F8/7F9

FIGURE 16 – $\overline{\text{RESET}}$ AND INTERRUPT PROCESSING FLOWCHART

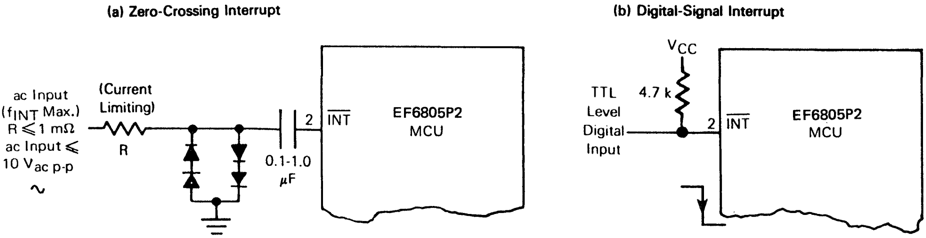


FIGURE 17 – TYPICAL INTERRUPT CIRCUITS

INPUT/OUTPUT

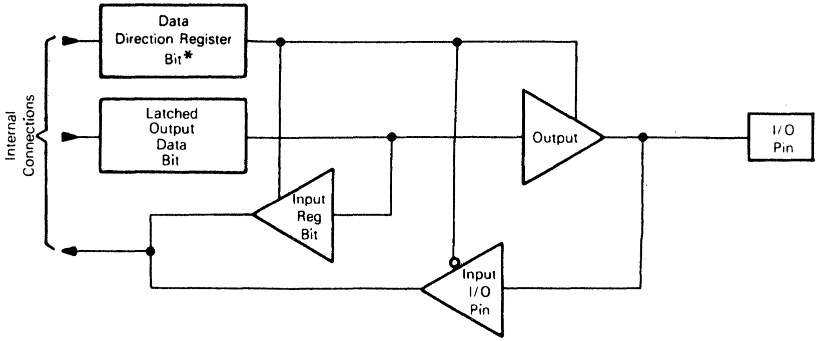
There are 20 input/output pins. The $\overline{INT}$ pin may also be polled with branch instructions to provide an additional input pin. All pins (port A, B, and C) are programmable as either inputs or outputs under software control of the corresponding write-only data direction register (DDR). The port I/O programming is accomplished by writing the corresponding bit in the port DDR to a logic "1" for output or a logic "0" for input. On reset, all the DDRs are initialized to a logic "0" state to put the ports in the input mode. To avoid undefined levels, the port output registers are not initialized on reset, but may be written before setting the DDR bits. When programmed as outputs, the latched output data is readable as input data, regardless of the logic levels at the output pin due to output loading; see Figure 18. When port B is programmed for outputs, it is capable of sinking 10 mA and sourcing 1 mA on each pin.

All input/output lines are TTL compatible as both inputs and outputs. Ports B and C are CMOS compatible as inputs. Port A may be made CMOS compatible as outputs with a mask option. The address map in Figure 5 gives the address of data registers and DDRs. The register configuration is provided in Figure 19 and Figure 20 provides some examples of port connections.

Caution

The corresponding DDRs for ports A, B, and C are write-only registers (registers at $004, $005, and $006). A read operation on these registers is undefined. Since BSET and BCLR are read-modify-write functions, they cannot be used to set or clear a DDR bit (all "unaffected" bits would be set). It is recommended that all DDR bits in a port be written using a single-store instruction.

The latched output data bit (see Figure 18) may always be written. Therefore, any write to a port writes all of its data bits even though the port DDR is set to input. This may be used to initialize the data registers and avoid undefined outputs; however, care must be exercised when using read-modify-write instructions since the data read corresponds to the pin level if the DDR is an input ("0") and corresponds to the latched output data when the DDR is an output ("1").

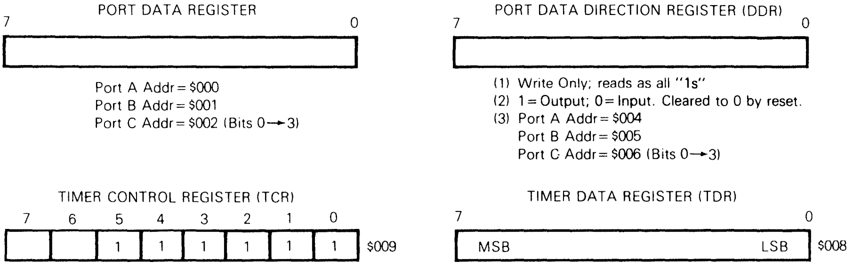


Data Direction Register Bit	Output Data Bit	Latched Output State	Input To MCU
1	0	0	0
1	1	1	1
0	X	High-Z **	Pin

*DDR is a write-only register and reads as all "1s".

**Ports A (with CMOS drive disabled), B, and C are three state ports. Port A has optional internal pullup devices to provide CMOS drive capability. See Electrical Characteristics tables for complete information.

FIGURE 18 – TYPICAL PORT I/O CIRCUITY


PORT DATA REGISTER
7
0
PORT DATA DIRECTION REGISTER (DDR)
7
0
TIMER CONTROL REGISTER (TCR)
7 6 5 4 3 2 1 0
1 1 1 1 1 1
$009
TIMER DATA REGISTER (TDR)
7
0
MSB
LSB
$008

Port A Addr = $000
Port B Addr = $001
Port C Addr = $002 (Bits 0→3)

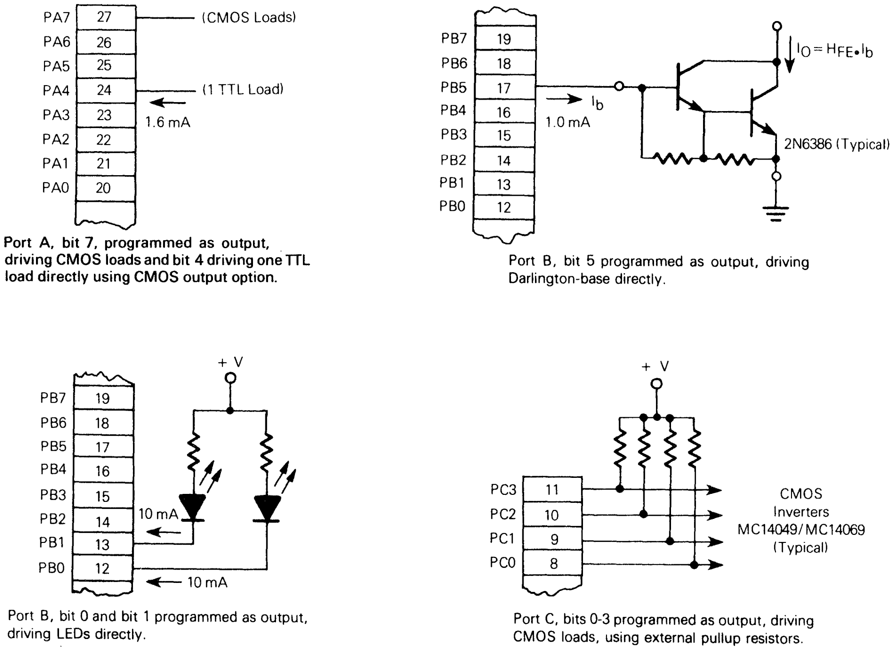
(1) Write Only; reads as all "1s"
(2) 1 = Output; 0 = Input. Cleared to 0 by reset.
(3) Port A Addr = $004
Port B Addr = $005
Port C Addr = $006 (Bits 0→3)

TCR7 – Timer Interrupt Status Bit: Set when TDR goes to zero; must be cleared by software. Cleared to 0 by reset.

TCR6 Bit 6 – Timer Interrupt Mask Bit: 1 = timer interrupt masked (disabled). Set to 1 by reset.

TCR Bits 5, 4, 3, 2, 1, 0 read as "1s" – unused bits.

FIGURE 19 – MCU REGISTER CONFIGURATION



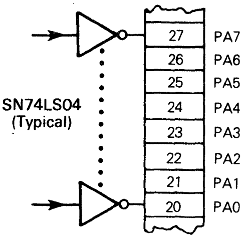
Port A, bit 7, programmed as output, driving CMOS loads and bit 4 driving one TTL load directly using CMOS output option.

Port B, bit 5 programmed as output, driving Darlington-base directly.

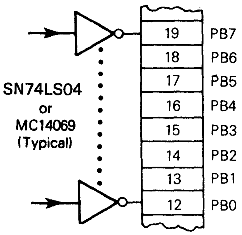
Port B, bit 0 and bit 1 programmed as output, driving LEDs directly.

Port C, bits 0-3 programmed as output, driving CMOS loads, using external pullup resistors.

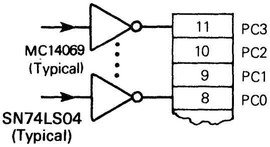
FIGURE 20 (a) – TYPICAL OUTPUT MODE PORT CONNECTIONS



TTL driving port A directly.



CMOS or TTL driving port B directly.



CMOS and TTL driving port C directly.

FIGURE 20 (b) – TYPICAL OUTPUT MODE PORT CONNECTIONS

SOFTWARE

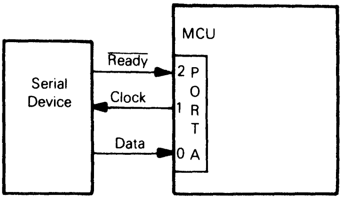
BIT MANIPULATION

The EF6805P2 MCU has the ability to set or clear any single random access memory or input/output bit (except the data direction register, see Caution below), with a single instruction (BSET, BCLR). Any bit in page zero including ROM, except the DDRs, can be tested, using the BRSET and BRCLR instructions, and the program branches as a result of its state. The carry bit equals the value of the bit referenced by BRSET or BRCLR. A rotate instruction may then be used to accumulate serial input data in a RAM location or register. The capability to work with any bit in RAM, ROM, or I/O allows the user to have individual flags in RAM or to handle I/O bits as control lines.

The coding example in Figure 21 illustrates the usefulness of the bit manipulation and test instructions. Assume that the MCU is to communicate with an external serial device. The external device has a data ready signal, a data output line, and a clock line to clock data one bit at a time, LSB first, out of the device. The MCU waits until the data is ready, clocks the external device, picks up the data in the carry flag (C bit), clears the clock line, and finally accumulates the data bit in a RAM location.

Caution

The corresponding DDRs for ports A, B, and C are write-only registers (registers at $004, $005, and $006). A read operation on these registers is undefined. Since BSET and BCLR are read-modify-write functions, they cannot be used to set or clear a DDR bit (all "unaffected" bits would be set). It is recommended that all DDR bits in a port be written using a single-store instruction.



```
                .
                .
                .
SELF  BRSET   2, PORTA, SELF
                .
                .
                .
      BSET    1, PORTA
      BRCLR   0, PORTA, CONT
CONT  BCLR    1, PORTA
      ASR     RAMLOC
                .
                .
                .
```

FIGURE 21 – BIT MANIPULATIONS EXAMPLE

ADDRESSING MODES

The EF6805P2 MCU has 10 addressing modes which are explained briefly in the following paragraphs. For additional details and graphical illustrations, refer to the 6805 Family User's Manual.

The term "effective address" (EA) is used in describing the address modes. EA is defined as the address from which the argument for an instruction is fetched or stored.

IMMEDIATE — In the immediate addressing mode, the operand is contained in the byte immediately following the opcode. The immediate addressing mode is used to access constants which do not change during program execution (e.g., a constant used to initialize a loop counter).

DIRECT — In the direct addressing mode, the effective address of the argument is contained in a single byte following the opcode byte. Direct addressing allows the user to directly address the lowest 256 bytes in memory with a single 2-byte instruction. This includes the on-chip RAM and I/O registers and 128 bytes of ROM. Direct addressing is an effective use of both memory and time.

EXTENDED — In the extended addressing mode, the effective address of the argument is contained in the two bytes following the opcode. Instructions using extended addressing are capable of referencing arguments anywhere in memory with a single 3-byte instruction. When using the Motorola assembler, the programmer need not specify whether an instruction uses direct or extended addressing. The assembler automatically selects the shortest form of the instruction.

RELATIVE — The relative addressing mode is only used in branch instructions. In relative addressing, the contents of the 8-bit signed byte following the opcode (the offset) is added to the PC if and only if the branch condition is true. Otherwise, control proceeds to the next instruction. The span of relative addressing is from − 126 to + 129 from the opcode address. The programmer need not worry about calculating the correct offset when using the Motorola assembler since it calculates the proper offset and checks to see if it is within the span of the branch.

INDEXED, NO OFFSET — In the indexed, no offset addressing mode, the effective address of the argument is contained in the 8-bit index register. Thus, this addressing mode can access the first 256 memory locations. These instructions are only one byte long. This mode is often used to move a pointer through a table or to hold the address of a frequently referenced RAM or I/O location.

INDEXED, 8-BIT OFFSET — In the indexed, 8-bit offset addressing mode, the effective address is the sum of the contents of the unsigned 8-bit index register and the unsigned byte following the opcode. This addressing mode is useful in selecting the kth element in an n element table. With this 2-byte instruction, k would typically be in X with the address of the beginning of the table in the instruction. As such, tables may begin anywhere within the first 256 addressable locations and could extend as far as location 510 ($1FE is the last location at which the instruction may begin).

INDEXED, 16-BIT OFFSET — In the indexed, 16-bit offset addressing mode, the effective address is the sum of the contents of the unsigned 8-bit index register and the two unsigned bytes following the opcode. This addressing mode can be used in a manner similar to indexed, 8-bit offset, except that this 3-byte instruction allows tables to be anywhere in memory. As with direct and extended addressing, the Motorola assembler determines the shortest form of indexed addressing.

BIT SET/CLEAR — In the bit set/clear addressing mode, the bit to be set or cleared is part of the opcode, and the byte following the opcode specifies the direct address of the byte in which the specified bit is to be set or cleared. Thus, any read/write bit in the first 256 locations of memory, including I/O, can be selectively set or cleared with a single 2-byte instruction.

Caution

The corresponding DDRs for ports A, B, and C are write-only registers (registers at $004, $005, and $006). A read operation on these registers is undefined. Since BSET and BCLR are read-modify-write functions, they cannot be used to set or clear a DDR bit (all "unaffected" bits would be set). It is recommended that all DDR bits in a port be written using a single-store instruction.

BIT TEST AND BRANCH — The bit test and branch addressing mode is a combination of direct addressing and relative addressing. The bit and condition (set or clear) which is to be tested is included in the opcode, and the address of the byte to be tested is in the single byte immediately following the opcode byte. The signed relative 8-bit offset is in the third byte and is added to the value of the PC if the branch condition is true. This single 3-byte instruction allows the program to branch based on the condition of any readable bit in the first 256 locations of memory. The span of branching is from − 125 to + 130 from the opcode address. The state of the tested bit is also transferred to the carry bit of the condition code register.

Caution

The corresponding DDRs for ports A, B, and C are write-only registers (registers at $004, $005, and $006). A read operation on these registers is undefined. Since BSET and BCLR are read-modify-write functions, they cannot be used to set or clear a DDR bit (all "unaffected" bits would be set). It is recommended that all DDR bits in a port be written using a single-store instruction.

INHERENT — In the inherent addressing mode, all the information necessary to execute the instruction is contained in the opcode. Operations specifying only the index register or accumulator, as well as control instruction with no other arguments, are included in this mode. These instructions are one byte long.

INSTRUCTION SET

The EF6805P2 MCU has a set of 59 basic instructions, which when combined with the 10 addressing modes produce 207 usable opcodes. They can be divided into five different types: register/memory, read-modify-write, branch, bit manipulation, and control. The following paragraphs briefly explain each type. All the instructions within a given type are presented in individual tables.

REGISTER/MEMORY INSTRUCTIONS — Most of these instructions use two operands. One operand is either the accumulator or the index register. The other operand is obtained from memory using one of the addressing modes. The jump unconditional (JMP) and jump to subroutine (JSR) instructions have no register operands. Refer to Table 1.

READ-MODIFY-WRITE INSTRUCTIONS — These instructions read a memory location or a register, modify or test its contents, and write the modified value back to memory or to the register. The test for negative or zero (TST) instruction is included in read-modify-write instructions through it does not perform the write. Refer to Table 2.

Caution

The corresponding DDRs for ports A, B, and C are write-only registers (registers at $004, $005, and $006). A read operation on these registers is undefined. Since BSET and BCLR are read-modify-write functions, they cannot be used to set or clear a DDR bit (all "unaffected" bits would be set). It is recommended that all DDR bits in a port be written using a single-store instruction.

BRANCH INSTRUCTIONS — The branch instructions cause a branch from the program when a certain condition is met. Refer to Table 3.

BIT MANIPULATION INSTRUCTIONS — These instructions are used on any bit in the first 256 bytes of the memory. One group either sets or clears. The other group performs the bit test branch operations. Refer to Table 4.

Caution

The corresponding DDRs for ports A, B, and C are write-only registers (registers at $004, $005, and $006). A read operation on these registers is undefined. Since BSET and BCLR are read-modify-write functions, they cannot be used to set or clear a DDR bit (all "unaffected" bits would be set). It is recommended that all DDR bits in a port be written using a single-store instruction.

CONTROL INSTRUCTIONS — The control instructions control the MCU operations during program execution. Refer to Table 5.

ALPHABETICAL LISTING — The complete instruction set is given in alphabetical order in Table 6.

OPCODE MAP SUMMARY — Table 7 is an opcode map for the instructions used on the MCU.

TABLE 1 — REGISTER/MEMORY INSTRUCTIONS

		Addressing Modes																	
		Immediate			Direct			Extended			Indexed (No Offset)			Indexed (8-Bit Offset)			Indexed (16-Bit Offset)		
Function	Mnemonic	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles	OP Code	# Bytes	# Cycles
Load A from Memory	LDA	A6	2	2	B6	2	4	C6	3	5	F6	1	4	E6	2	5	D6	3	6
Load X from Memory	LDX	AE	2	2	BE	2	4	CE	3	5	FE	1	4	EE	2	5	DE	3	6
Store A in Memory	STA	—	—	—	B7	2	5	C7	3	6	F7	1	5	E7	2	6	D7	3	7
Store X in Memory	STX	—	—	—	BF	2	5	CF	3	6	FF	1	5	EF	2	6	DF	3	7
Add Memory to A	ADD	AB	2	2	BB	2	4	CB	3	5	FB	1	4	EB	2	5	DB	3	6
Add Memory and Carry to A	ADC	A9	2	2	B9	2	4	C9	3	5	F9	1	4	E9	2	5	D9	3	6
Subtract Memory	SUB	A0	2	2	B0	2	4	C0	3	5	F0	1	4	E0	2	5	D0	3	6
Subtract Memory from A with Borrow	SBC	A2	2	2	B2	2	4	C2	3	5	F2	1	4	E2	2	5	D2	3	6
AND Memory to A	AND	A4	2	2	B4	2	4	C4	3	5	F4	1	4	E4	2	5	D4	3	6
OR Memory with A	ORA	AA	2	2	BA	2	4	CA	3	5	FA	1	4	EA	2	5	DA	3	6
Exclusive OR Memory with A	EOR	A8	2	2	B8	2	4	C8	3	5	F8	1	4	E8	2	5	D8	3	6
Arithmetic Compare A with Memory	CMP	A1	2	2	B1	2	4	C1	3	5	F1	1	4	E1	2	5	D1	3	6
Arithmetic Compare X with Memory	CPX	A3	2	2	B3	2	4	C3	3	5	F3	1	4	E3	2	5	D3	3	6
Bit Test Memory with A (Logical Compare)	BIT	A5	2	2	B5	2	4	C5	3	5	F5	1	4	E5	2	5	D5	3	6
Jump Unconditional	JMP	—	—	—	BC	2	3	CC	3	4	FC	1	3	EC	2	4	DC	3	5
Jump to Subroutine	JSR	—	—	—	BD	2	7	CD	3	8	FD	1	7	ED	2	8	DD	3	9

TABLE 2 — READ-MODIFY-WRITE INSTRUCTIONS

		Addressing Modes														
		Inherent (A)			Inherent (X)			Direct			Indexed (No Offset)			Indexed (8 Bit Offset)		
Function	Mnemonic	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles
Increment	INC	4C	1	4	5C	1	4	3C	2	6	7C	1	6	6C	2	7
Decrement	DEC	4A	1	4	5A	1	4	3A	2	6	7A	1	6	6A	2	7
Clear	CLR	4F	1	4	5F	1	4	3F	2	6	7F	1	6	6F	2	7
Complement	COM	43	1	4	53	1	4	33	2	6	73	1	6	63	2	7
Negate (2's Complement)	NEG	40	1	4	50	1	4	30	2	6	70	1	6	60	2	7
Rotate Left Thru Carry	ROL	49	1	4	59	1	4	39	2	6	79	1	6	69	2	7
Rotate Right Thru Carry	ROR	46	1	4	56	1	4	36	2	6	76	1	6	66	2	7
Logical Shift Left	LSL	48	1	4	58	1	4	38	2	6	78	1	6	68	2	7
Logical Shift Right	LSR	44	1	4	54	1	4	34	2	6	74	1	6	64	2	7
Arithmetic Shift Right	ASR	47	1	4	57	1	4	37	2	6	77	1	6	67	2	7
Test for Negative or Zero	TST	4D	1	4	5D	1	4	3D	2	6	7D	1	6	6D	2	7

TABLE 3 – BRANCH INSTRUCTIONS

		Relative Addressing Mode		
Function	Mnemonic	Op Code	# Bytes	# Cycles
Branch Always	BRA	20	2	4
Branch Never	BRN	21	2	4
Branch IFFHigher	BHI	22	2	4
Branch IFFLower or Same	BLS	23	2	4
Branch IFFCarry Clear	BCC	24	2	4
(BranchIFFHigher or Same)	(BHS)	24	2	4
Branch IFFCarry Set	BCS	25	2	4
(Branch IFF Lower)	(BLO)	25	2	4
BranchIFFNot Equal	BNE	26	2	4
Branch IFFEqual	BEQ	27	2	4
Branch IFF Half Carry Clear	BHCC	28	2	4
BranchIFFHalf Carry Set	BHCS	29	2	4
BranchIFFPlus	BPL	2A	2	4
BranchIFFMinus	BMI	2B	2	4
BranchIFFInterupt Mask Bit is Clear	BMC	2C	2	4
BranchIFFInterrupt Mask Bit is Set	BMS	2D	2	4
BranchIFFInterrupt Line is Low	BIL	2E	2	4
BranchIFFInterrupt Line is High	BIH	2F	2	4
Branch to Subroutine	BSR	AD	2	8

TABLE 4 – BIT MANIPULATION INSTRUCTIONS

		Addressing Modes					
		Bit Set/Clear			Bit Test and Branch		
Function	Mnemonic	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles
Branch IFF Bit n is set	BRSET n (n = 0....7)	—	—	—	2 • n	3	10
Branch IFF Bit n is clear	BRCLR n (n = 0....7)	—	—	—	01 + 2 • n	3	10
Set Bit n	BSET n (n = 0....7)	10 + 2 • n	2	7	—	—	—
Clear bit n	BCLR n (n = 0....7)	11 + 2 • n	2	7	—	—	—

TABLE 5 – CONTROL INSTRUCTIONS

		Inherent		
Function	Mnemonic	Op Code	# Bytes	# Cycles
Transfer A to X	TAX	97	1	2
Transfer X to A	TXA	9F	1	2
Set Carry Bit	SEC	99	1	2
Clear Carry Bit	CLC	98	1	2
Set Interrupt Mask Bit	SEI	9B	1	2
Clear Interrupt Mask Bit	CLI	9A	1	2
Software Interrupt	SWI	83	1	11
Return from Subroutine	RTS	81	1	6
Return from Interrupt	RTI	80	1	9
Reset Stack Pointer	RSP	9C	1	2
No-Operation	NOP	9D	1	2

TABLE 6 — INSTRUCTION SET

	Addressing Modes										Condition Code				
Mnemonic	Inherent	Immediate	Direct	Extended	Relative	Indexed (No Offset)	Indexed (8 Bits)	Indexed (16 Bits)	Bit Set/ Clear	Bit Test & Branch	H	I	N	Z	C
ADC		X	X	X		X	X	X			Λ	●	Λ	Λ	Λ
ADD		X	X	X		X	X	X			Λ	●	Λ	Λ	Λ
AND		X	X	X		X	X	X			●	●	Λ	Λ	●
ASL	X		X			X	X				●	●	Λ	Λ	Λ
ASR	X		X			X	X				●	●	Λ	Λ	Λ
BCC					X						●	●	●	●	●
BCLR									X		●	●	●	●	●
BCS					X						●	●	●	●	●
BEQ					X						●	●	●	●	●
BHCC					X						●	●	●	●	●
BHCS					X						●	●	●	●	●
BHI					X						●	●	●	●	●
BHS					X						●	●	●	●	●
BIH					X						●	●	●	●	●
BIL					X						●	●	●	●	●
BIT		X	X	X		X	X	X			●	●	Λ	Λ	●
BLO					X						●	●	●	●	●
BLS					X						●	●	●	●	●
BMC					X						●	●	●	●	●
BMI					X						●	●	●	●	●
BMS					X						●	●	●	●	●
BNE					X						●	●	●	●	●
BPL					X						●	●	●	●	●
BRA					X						●	●	●	●	●
BRN					X						●	●	●	●	●
BRCLR										X	●	●	●	●	Λ
BRSET										X	●	●	●	●	Λ
BSET									X		●	●	●	●	●
BSR					X						●	●	●	●	●
CLL	X										●	●	●	●	O
CLI	X										●	O	●	●	●
CLR	X		X			X	X				●	●	O	1	●
CMP		X	X	X		X	X	X			●	●	Λ	Λ	Λ
COM	X		X			X	X				●	●	Λ	Λ	1
CPX		X	X	X		X	X	X			●	●	Λ	Λ	Λ
DEC	X		X			X	X				●	●	Λ	Λ	●
EOR		X	X	X		X	X	X			●	●	Λ	Λ	●
INC	X		X			X	X				●	●	Λ	Λ	●
JMP			X	X		X	X	X			●	●	●	●	●
JSR			X	X		X	X	X			●	●	●	●	●
LDA		X	X	X		X	X	X			●	●	Λ	Λ	●
LDX		X	X	X		X	X	X			●	●	Λ	Λ	●
LSL	X		X			X	X				●	●	Λ	Λ	Λ
LSR	X		X			X	X				●	●	O	Λ	Λ
NEQ	X		X			X	X				●	●	Λ	Λ	Λ
NOP	X										●	●	●	●	●
ORA		X	X	X		X	X	X			●	●	Λ	Λ	●
ROL	X		X			X	X				●	●	Λ	Λ	Λ
RSP	X										●	●	●	●	●

Condition Code Symbols:

H Half Carry (From Bit 3)	C Carry/Borrow
I Interrupt Mask	Λ Test and Set if True, Cleared Otherwise
N Negative (Sign Bit)	• Not Affected
Z Zero	

TABLE 6 — INSTRUCTION SET (CONTINUED)

	Addressing Modes										Condition Code				
Mnemonic	Inherent	Immediate	Direct	Extended	Relative	Indexed (No Offset)	Indexed (8 Bits)	Indexed (16 Bits)	Bit Set/ Clear	Bit Test & Branch	H	I	N	Z	C
RTI	X										?	?	?	?	?
RTS	X										●	●	●	●	●
SBC		X	X	X		X	X	X			●	●	∧	∧	∧
SEC	X										●	●	●	●	1
SEI	X										●	1	●	●	●
STA			X	X		X	X	X			●	●	∧	∧	●
STX			X	X		X	X	X			●	●	∧	∧	●
SUB		X	X	X		X	X	X			●	●	∧	∧	∧
SWI	X										●	1	●	●	●
TAX	X										●	●	●	●	●
TST	X		X			X	X				●	●	∧	∧	●
TXA	X										●	●	●	●	●

Condition Code Symbols:

H Half Carry (From Bit 3)
I Interrupt Mask
N Negative (Sign Bit)
Z Zero

C Carry/Borrow
∧ Test and Set if True, Cleared Otherwise
● Not Affected
? Load CC Register From Stack

HMOS 6805 FAMILY

Features	EF6805CT	EF6805P2	EF6805P4	EF6805P6	EF6805R2	EF6805R3	EF6805T2	EF6805U2	EF6805U3	EF6805TV
Technology	HMOS	HMOS	HMOS	HMOS	HMOS	HMOS	HMOS	HMOS	HMOS	HMOS
Number of Pins	40	28	28	28	40	40	28	40	40	40
On-Chip RAM (Bytes)	240	64	112	64	64	112	64	64	112	96
On-Chip User ROM (Bytes)	4096	1100	1100	1796	2048	3776	2508	2048	3776	6144
External Bus	Yes	None	None	None	None	None	None	None	None	None
Bidirectional I/O Lines	29	20	20	20	24	24	19	24	24	32
Unidirectional I/O Lines	None	None	None	None	6 Inputs	6 Inputs	None	8 Inputs	8 Inputs	None
Other I/O Features	Timer,UART	Timer	Timer	Timer	Timer,A/D	Timer,A/D	Timer, PLL	Timer	Timer	Timer,D/A
External Interrupt Inputs	3	1	1	1	2	2	2	2	2	1
STOP and WAIT	No	No	No	No	No	No	No	No	No	No

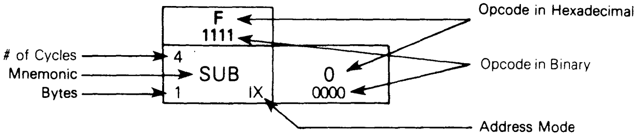
TABLE 7 – 6805 HMOS FAMILY OPCODE MAP

	Bit Manipulation		Branch	Read-Modify-Write					Control		Register/Memory						
	BTB	BSC	REL	DIR	INH	INH	IX1	IX	INH	INH	IMM	DIR	EXT	IX2	IX1	IX	
Low \ Hi	0 0000	1 0001	2 0010	3 0011	4 0100	5 0101	6 0110	7 0111	8 1000	9 1001	A 1010	B 1011	C 1100	D 1101	E 1110	F 1111	Hi \ Low
0 0000	10 BRSET0 3 BTB	7 BSET0 2 BSC	4 BRA 2 REL	6 NEG 2 DIR	4 NEG 1 INH	4 NEG 1 INH	7 NEG 2 IX1	6 NEG 1 IX	9 RTI 1 INH		2 SUB 2 IMM	4 SUB 2 DIR	5 SUB 3 EXT	6 SUB 3 IX2	5 SUB 2 IX1	4 SUB 1 IX	0 0000
1 0001	10 BRCLR0 3 BTB	7 BCLR0 2 BSC	4 BRN 2 REL						6 RTS 1 INH		2 CMP 2 IMM	4 CMP 2 DIR	5 CMP 3 EXT	6 CMP 3 IX2	5 CMP 2 IX1	4 CMP I IX	1 0001
2 0010	10 BRSET1 3 BTB	7 BSET1 2 BSC	4 BHI 2 REL								2 SBC 2 IMM	4 SBC 2 DIR	5 SBC 3 EXT	6 SBC 3 IX2	5 SBC 2 IX1	4 SBC 1 IX	2 0010
3 0011	10 BRCLR1 3 BTB	7 BCLR1 2 BSC	4 BLS 2 REL	6 COM 2 DIR	4 COMA 1 INH	4 COMX 1 INH	7 COM 2 IX1	6 COM 1 IX	11 SWI 1 INH		2 CPX 2 IMM	4 CPX 2 DIR	5 CPX 3 EXT	6 CPX 3 IX2	5 CPX 2 IX1	4 CPX 1 IX	3 0011
4 0100	10 BRSET2 3 BTB	7 BSET2 2 BSC	4 BCC 2 REL	6 LSR 2 DTR	4 LSRA 1 INH	4 LSRX 1 INH	7 LSR 2 IX1	6 LSR 1 IX			2 AND 2 IMM	4 AND 2 DIR	5 AND 3 EXT	6 AND 3 IX2	5 AND 2 IX1	4 AND 1 IX	4 0100
5 0101	10 BRCLR2 3 BTB	7 BCLR2 2 BSC	4 BCS 2 REL								2 BIT 2 IMM	4 BIT 2 DIR	5 BIT 3 EXT	6 BIT 3 IX2	5 BIT 2 IX1	4 BIT 1 IX	5 0101
6 0110	10 BRSET3 3 BTB	7 BSET3 2 BSC	4 BNE 2 REL	6 ROR 2 DIR	4 RORA 1 INH	4 RORX 1 INH	7 ROR 2 IX1	6 ROR 1 IX			2 LDA 2 IMM	4 LDA 2 DIR	5 LDA 3 EXT	6 LDA 3 IX2	5 LDA 2 IX1	4 LDA 1 IX	6 0110
7 0111	10 BRCLR3 3 BTB	7 BCLR3 2 BSC	4 BEQ 2 REL	6 ASR 2 DIR	4 ASRA 1 INH	4 ASRX 1 INH	7 ASR 2 IX1	6 ASR 1 IX		2 TAX 1 INH		5 STA 2 DIR	6 STA 3 EXT	7 STA 3 IX2	6 STA 2 IX1	5 STA 1 IX	7 0111
8 1000	10 BRSET4 3 BTB	7 BSET4 2 BSC	4 BHCC 2 REL	6 LSL 2 DIR	4 LSLA 1 INH	4 LSLX 1 INH	7 LSL 2 IX1	6 LSL 1 IX		2 CLC 1 INH	2 EOR 2 IMM	4 EOR 2 DIR	5 EOR 3 EXT	6 EOR 3 IX2	5 EOR 2 IX1	4 EOR 1 IX	8 1000
9 1001	10 BRCLR4 3 BTB	7 BCLR4 2 BSC	4 BHCS 2 REL	6 ROL 2 DIR	4 ROLA 1 INH	4 ROLX 1 INH	7 ROL 2 IX1	6 ROL 1 IX		2 SEC 1 INH	2 ADC 2 IMM	4 ADC 2 DIR	5 ADC 3 EXT	6 ADC 3 IX2	5 ADC 2 IX1	4 ADC 1 IX	9 1001
A 1010	10 BRSET5 3 BTB	7 BSET5 2 BSC	4 BPL 2 REL	6 DEC 2 DIR	4 DECA 1 INH	4 DECX 1 INH	7 DEC 2 IX1	6 DEC 1 IX		2 CLI 1 INH	2 ORA 2 IMM	4 ORA 2 DIR	5 ORA 3 EXT	6 ORA 3 IX2	5 ORA 2 IX1	4 ORA 1 IX	A 1010
B 1011	10 BRCLR5 3 BTB	7 BCLR5 2 BSC	4 BMI 2 REL							2 SEI 1 INH	2 ADD 2 IMM	4 ADD 2 DIR	5 ADD 3 EXT	6 ADD 3 IX2	5 ADD 2 IX1	4 ADD 1 IX	B 1011
C 1100	10 BRSET6 3 BTB	7 BSET6 2 BSC	4 BMC 2 REL	6 INC 2 DIR	4 INCA 1 INH	4 INCX 1 INH	7 INC 2 IX1	6 INC 1 IX		2 RSP 1 INH		3 JMP 2 DIR	4 JMP 3 EXT	5 JMP 3 IX2	4 JMP 2 IX1	3 JMP 1 IX	C 1100
D 1101	10 BRCLR6 3 BTB	7 BCLR6 2 BSC	4 BMS 2 REL	6 TST 2 DIR	4 TSTA 1 INH	4 TSTX 1 INH	7 TST 2 IX1	6 TST 1 IX		2 NOP 1 INH	8 BSR 2 REL	7 JSR 2 DIR	8 JSR 3 EXT	9 JSR 3 IX2	8 JSR 2 IX1	7 JSR 1 IX	D 1101
E 1110	10 BRSET7 3 BTB	7 BSET7 2 BSC	4 BIL 2 REL								2 LDX 2 IMM	4 LDX 2 DIR	5 LDX 3 EXT	6 LDX 3 IX2	5 LDX 2 IX1	4 LDX 1 IX	E 1110
F 1111	10 BRCLR7 3 BTB	7 BCLR7 2 BSC	4 BIH 2 REL	6 CLR 2 DIR	4 CLRA 1 INH	4 CLRX 1 INH	7 CLR 2 IX1	6 CLR 1 IX		2 TXA 1 INH		5 STX 2 DIR	6 STX 3 EXT	7 STX 3 IX2	6 STX 2 IX1	5 STX 1 IX	F 1111

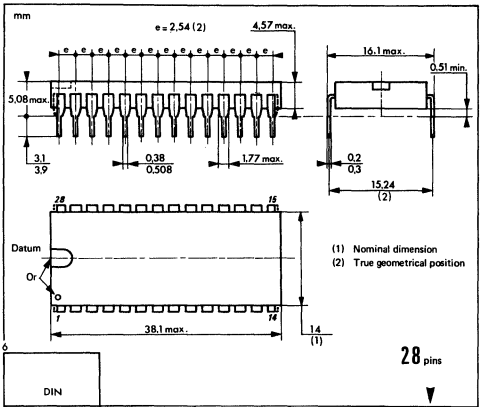
Abbreviations for Address Modes

INH	Inherent
IMM	Immediate
DIR	Direct
EXT	Extended
REL	Relative
BSC	Bit Set/Clear
BTB	Bit Test and Branch
IX	Indexed (No Offset)
IX1	Indexed, 1 Byte (8-Bit) Offset
IX2	Indexed, 2 Byte (16-Bit) Offset

LEGEND

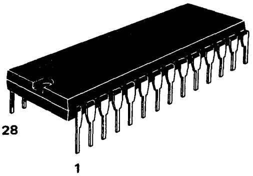


PHYSICAL DIMENSIONS

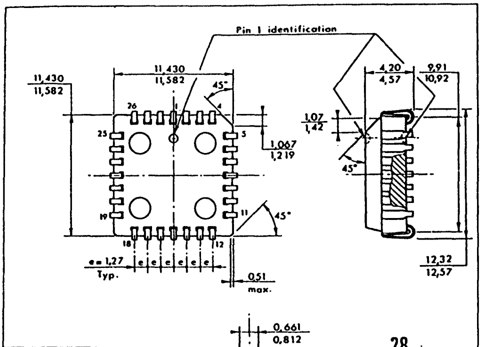


DIN				
A51E CEI	D.A.T.A.	JEDEC	F 144 SITELESC	CB-132

CB-132

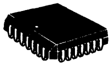


P SUFFIX
PLASTIC PACKAGE

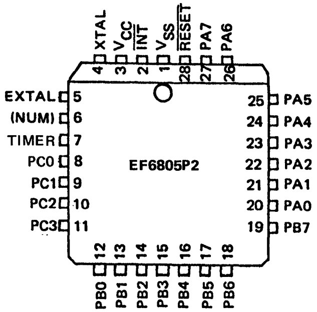


DIN				
CEI	D.A.T.A.	MO-047-AB JEDEC	SITELESC	CB-520

CB-520



FN SUFFIX
PLCC28



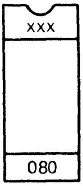
ORDERING INFORMATION

The information required when ordering a custom MCU is listed below. The ROM program may be transmitted to THOMSON SEMICONDUCTEURS on EPROM(s) or an EFDOS/MDOS* disk file.

To initiate a ROM pattern for the MCU, it is necessary to first contact your local THOMSON SEMICONDUCTEURS representative or distributor.

EPROMs

One 2716 or 2732 type EPROM§, programmed with the customer program (positive logic sense for address and data), may be submitted for pattern generation.



XXX = Customer ID

After the EPROM is marked, it should be placed in conductive IC carriers and securely packed. Do not use styrofoam.

VERIFICATION MEDIA

All original pattern media (EPROMs or floppy disk) are filed for contractual purposes and are not returned. A computer listing of the ROM code will be generated and returned along with a listing verification form. The listing should be thoroughly checked and the verification form completed, signed, and returned to THOMSON SEMICONDUCTEURS. The signed verification form constitutes the contractual agreement for creation of the customer mask. If desired, THOMSON SEMICONDUCTEURS will program on blank EPROM from the data file used to create the custom mask and aid in the verification process.

ROM VERIFICATION UNITS (RVUs)

Ten MCUs containing the customer's ROM pattern will be sent for program verification. These units will have been made using the custom mask but are for the purpose of ROM verification only. For expediency they are usually unmarked, packaged in ceramic, and tested only at room temperature and 5 volts. These RVUs are included in the mask charge and are not production parts. The RVUs are thus not guaranteed by THOMSON SEMICONDUCTEURS. Quality Assurance, and should be discarded after verification is completed.

FLEXIBLE DISKS

The disk media submitted must be single-sided, EFDOS/MDOS* compatible floppies.

The customer must write the binary file name and company name on the disk with a felt-tip-pen. The minimum EFDOS/MDOS* system files, as well as the absolute binary object file (Filename .LO type of file) from the 6805 cross assembler, must be on the disk. An object file made from a memory dump using the ROLLOUT command is also acceptable. Consider submitting a source listing as well as the following files : filename .LX (DEVICE/EXORciser loadable format) and filename .SA (ASCII Source Code). These files will of course be kept confidential and are used 1) to speed up the process in-house if any problems arise, and 2) to speed up the user-to-factory interface if the user finds any software errors and needs assistance quickly from THOMSON SEMICONDUCTEURS factory representatives.

EFDOS is THOMSON SEMICONDUCTEURS' Disk Operating System available on development systems such as DEVICE,...

MDOS* is MOTOROLA's Disk Operating System available on development systems such as EXORciser,...

*Requires prior factory approval.

Whenever ordering a custom MCU is required, please contact your local THOMSON SEMICONDUCTEURS representative or THOMSON SEMICONDUCTEURS distributor and/or complete and send the attached "MCU customer ordering sheet" to your local THOMSON SEMICONDUCTEURS representative,

EXORciser is a registered trademark of MOTOROLA Inc.

ORDERING INFORMATION

EF6805P2 | C | V | ___

Device

Package

Oper. temp.

Screening level

The table below horizontally shows all available suffix combinations for package, operating temperature and screening level. Other possibilities on request.

DEVICE	PACKAGE					OPER. TEMP			SCREENING LEVEL			
	C		P		FN	L*	V	T	Std	D		
EF6805P2 (1.0 MHz)			●		●	●	●		●	●		
EF6805P2 (1.5 MHz)**			●		●	●			●	●		
EF6805P2 (2.0 MHz)**			●		●	●			●	●		

Examples : EF6805P2P , EF6805P2FN, EF6805P2 PLD, EF6805P2FNLD.

Package: C: Ceramic DIL, **P:** Plastic DIL, **FN:** PLCC.
Oper. temp.: L*: 0°C to + 70°C, **V:** -40°C to + 85°C, **T:** -40°C to + 105°C, *****: may be omitted.
Screening level: Std: (no-end suffix), **D:** NFC 96883 level D.
******: Requires prior factory approval.

These specifications are subject to change without notice.
Please inquire with our sales offices about the availability of the different products.

Printed in France

THOMSON COMPONENTS

SEMICONDUCTEURS

EF6805P6

8-BIT MICROCOMPUTER

ADVANCE INFORMATION

HMOS

The EF6805P6 Microcomputer Unit (MCU) is a member of the 6805 Family of low-cost single-chip microcomputers. This 8-bit microcomputer contains a CPU, on-chip CLOCK, ROM, RAM, I/O, and TIMER. It is designed for the user who needs an economical microcomputer with the proven capabilities of the 6800-based instruction set. The following are some of the hardware and software highlights of the EF6805P6 MCU.

HARDWARE FEATURES

- 8-Bit Architecture
- 64 Bytes of RAM
- Memory Mapped I/O
- 1796 Bytes of User ROM
- 20 TTL/CMOS Compatible Bidirectional I/O Lines (8 Lines are LED Compatible)
- On-Chip Clock Generator
- Self-Check Mode
- Zero Crossing Detection
- Master Reset
- Complete Development System Support on DEVICE®.
- 5 V Single Supply.

SOFTWARE FEATURES

- Similar to 6800 Family
- Byte Efficient Instruction Set
- Easy to Program
- True Bit Manipulation
- Bit Test and Branch Instruction
- Versatile Interrupt Handling
- Versatile Index Register
- Powerful Indexed Addressing for Tables
- Full Set of Conditional Branches
- Memory Usable as Register/Flags
- Single Instruction Memory Examine/Change
- 10 Powerful Addressing Modes
- All Addressing Modes Apply to ROM, RAM, and I/O

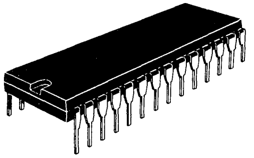
USER SELECTABLE OPTIONS

- Internal 8-Bit Timer with Selectable Clock Source (External Timer Input or Internal Machine Clock)
- Timer Prescaler Option (7 Bits, 2^n)
- 8 Bidirectional I/O Lines with TTL or TTL/CMOS Interface Option
- Crystal or Low-Cost Resistor Oscillator Option
- Low Voltage Inhibit Option
- Vectored Interrupts : Timer, Software, and External.
- Port B Open Drain Drive Option

DEVICE® is THOMSON SEMICONDUCTEURS' development/emulation tool.

CASES

CB-132

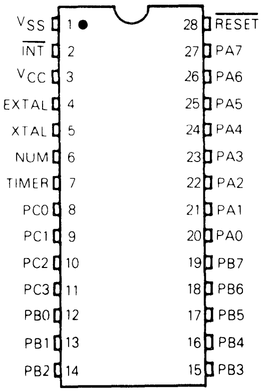


P SUFFIX
PLASTIC PACKAGE

CB-520

FN SUFFIX
PLCC 28

PIN ASSIGNMENT



JUNE 1986 1/24

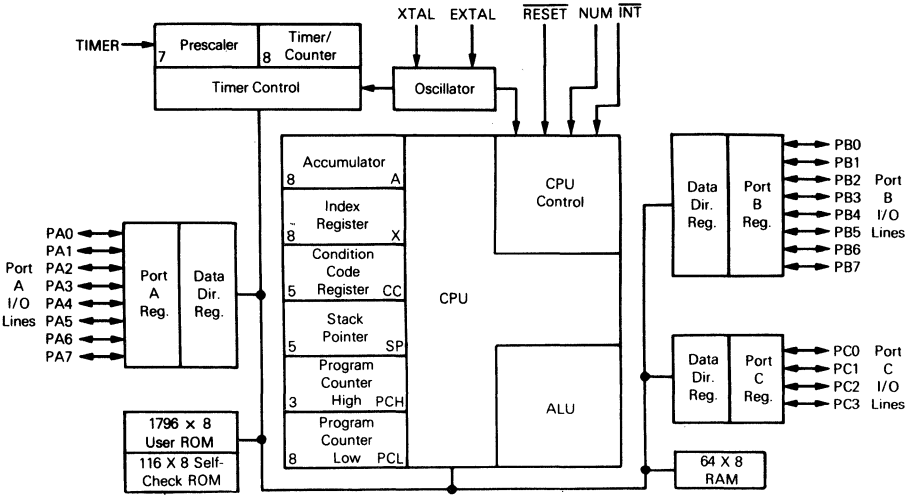


FIGURE 1 — EF6805P6 HMOS MICROCOMPUTER BLOCK DIAGRAM

MAXIMUM RATINGS

Rating	Symbol	Value	Unit
Supply Voltage	V_{CC}	−0.3 to +7.0	V
Input Voltage (Except Pin 6)	V_{in}	−0.3 to +7.0	V
Operating Temperature Range (T_L to T_H) **V suffix**	T_A	**0 to + 70** **−40 to + 85**	°C
Storage Temperature Range	T_{stg}	−55 to + 150	°C
Junction Temperature **Plastic** **PLCC**	T_J	 **150** **150**	°C

This device contains circuitry to protect the inputs against damage due to high static voltages or electric fields, however, it is advised that normal precautions be taken to avoid application of any voltage higher than maximum rated voltages to this high-impedance circuit. For proper operation it is recommended that V_{in} and V_{out} be constrained to the range $V_{SS} \leq (V_{in}$ or $V_{out}) \leq V_{CC}$. Reliability of operation is enhanced if unused inputs are tied to an appropriate logic voltage level (e.g., either V_{SS} or V_{CC}).

THERMAL CHARACTERISTICS

Characteristic	Symbol	Value	Unit
Thermal Resistance Plastic **PLCC**	θ_{JA}	 70 **110**	°C/W

POWER CONSIDERATIONS

The average chip-junction temperature, T_J, in °C can be obtained from:

$$T_J = T_A + (P_D \bullet \theta_{JA}) \qquad (1)$$

Where:

$T_A \equiv$ Ambient Temperature, °C

$\theta_{JA} \equiv$ Package Thermal Resistance, Junction-to-Ambient, °C/W

$P_D \equiv P_{INT} + P_{PORT}$

$P_{INT} \equiv I_{CC} \times V_{CC}$, Watts − Chip Internal Power

$P_{PORT} \equiv$ Port Power Dissipation, Watts − User Determined

For most applications $P_{PORT} \ll P_{INT}$ and can be neglected. P_{PORT} may become significant if the device is configured to drive Darlington bases or sink LED loads.

An approximate relationship between P_D and T_J (if P_{PORT} is neglected) is:

$$P_D = K \div (T_J + 273°C) \qquad (2)$$

Solving equations 1 and 2 for K gives:

$$K = P_D \bullet (T_A + 273°C) + \theta_{JA} \bullet P_D^2 \qquad (3)$$

Where K is a constant pertaining to the particular part. K can be determined from equation 3 by measuring P_D (at equilibrium) for a known T_A. Using this value of K the values of P_D and T_J can be obtained by solving equations (1) and (2) iteratively for any value of T_A.

ELECTRICAL CHARACTERISTICS (V_{CC} = +5.25 Vdc ±0.5 Vdc, V_{SS} = 0 Vdc, $T_A = T_L$ to T_H unless otherwise noted)

Characteristic	Symbol	Min	Typ	Max	Unit
Input High Voltage $\overline{\text{RESET}}$ ($4.75 \le V_{CC} \le 5.75$) ($V_{CC} < 4.75$) $\overline{\text{INT}}$ ($4.75 \le V_{CC} \le 5.75$) ($V_{CC} < 4.75$) **All Other (Exept TIMER)**	V_{IH}	 4.0 $V_{CC} - 0.5$ 4.0 $V_{CC} - 0.5$ 2.0	 – – * * –	 V_{CC} V_{CC} V_{CC} V_{CC} V_{CC}	V
Input High Voltage Timer Timer Mode Self-Check Mode	V_{IH}	 2.0 9.0	 – 10.0	 $V_{CC} + 1$ 15.0	V
Input Low Voltage $\overline{\text{INT}}$ All Other	V_{IL}	 V_{SS} V_{SS}	 * –	 1.5 0.8	V
$\overline{\text{RESET}}$ Hysteresis Voltage (See Figures 10, 11, and 12) "Out of Reset" "Into Reset"	 V_{IRES+} V_{IRES-}	 2.1 0.8	 – –	 4.0 2.0	V
$\overline{\text{INT}}$ Zero Crossing Input Voltage, Through a Capacitor	V_{INT}	2.0	–	4.0	$V_{ac\ p\text{-}p}$
Internal Power Dissipation – No Port Loading V_{CC} = 5.75 V, T_A = 0°C	P_{INT}	–	400	690	mW
Input Capacitance XTAL All Other	C_{in}	 – –	 25 10	 – –	pF
Low Voltage Recover	V_{LVR}	2.75	3.5	4.75	V
Low Voltage Inhibit 0 to +70°C –40 to +85°C	V_{LVI}	2.75 3.1	3.5 3.5	– –	V
Input Current TIMER (V_{in} = 0.4 V) $\overline{\text{INT}}$ (V_{in} = 2.4 V to V_{CC}) EXTAL (V_{in} = 2.4 V to V_{CC}, Crystal Option) (V_{in} = 0.4 V, Crystal Option) $\overline{\text{RESET}}$ (V_{in} = 0.8 V)	I_{in}	 – – – – –4.0	 – 20 – – –	 20 50 10 –1600 –40	µA

* Due to internal biasing, this input (when unused) floats to approximately 2.0 Vdc.

PORT DC ELECTRICAL CHARACTERISTICS (V_{CC} = +5.25 Vdc ±0.5 Vdc, V_{SS} = 0 Vdc, $T_A = T_L$ to T_H unless otherwise noted)

Characteristic	Symbol	Min	Typ	Max	Unit
Port A with CMOS Drive Enabled					
Output Low Voltage, I_{Load} = 1.6 mA	V_{OL}	–	–	0.4	V
Output High Voltage, I_{Load} = –100 µA	V_{OH}	2.4	–	–	V
Output High Voltage, I_{Load} = –10 µA	V_{OH}	$V_{CC} - 1.0$	–	–	V
Input High Voltage, I_{Load} = –300 µA (max.)	V_{IH}	2.0	–	V_{CC}	V
Input LowVoltage, I_{Load} = –500 µA (max.)	V_{IL}	0.3	–	0.8	V
Hi-Z State Input Current (V_{in} = 2.0 V to V_{CC})	I_{IH}	–	–	–300	µA
Hi-Z State Input Current (V_{in} = 0.4 V)	I_{IL}	–	–	–500	µA
Port B					
Output Low Voltage, I_{Load} = 3.2 mA	V_{OL}	–	–	0.4	V
Output Low Voltage, I_{Load} = 10 mA (sink)	V_{OL}	–	–	1.0	V
Output High Voltage, I_{Load} = –200 µA	V_{OH}	2.4	–	–	V
Darlington Current Drive (Source), V_O = 1.5 V	I_{OH}	–1.0	–	–10	mA
Input High Voltage	V_{IH}	2.0	–	V_{CC}	V
Input Low Voltage	V_{IL}	V_{SS}	–	0.8	V
Hi-Z State Input Current	I_{TSI}	–	2	10	µA
Port B with Open Drain Option					
Output High Voltage Hi-Z State Input Current	V_{OH} I_{TSI}	2.4 –	– 2	13.0 20	V µA
Port C and Port A with CMOS Drive Disabled					
Output Low Voltage, I_{Load} = 1.6 mA	V_{OL}	–	–	0.4	V
Output High Voltage, I_{Load} = –100 µA	V_{OH}	2.4	–	–	V
Input High Voltage	V_{IH}	2.0	–	V_{CC}	V
Input Low Voltage	V_{IL}	–0.3	–	0.8	V
Hi-Z State Input Current	I_{TSI}	–	2	10	µA

SWITCHING CHARACTERISTICS (V_{CC} = +5.25 Vdc ±0.5 Vdc, V_{SS} = 0 Vdc, $T_A = T_L$ to T_H unless otherwise noted)

Characteristic		Symbol	Min	Typ	Max	Unit
Oscillator Frequency	EF6805P6 EF68A05P6 EF68B05P6	f_{osc}	0.4 0.4 0.4	– – –	4.2 6.0 8.0	MHz
Cycle Time (4/f_{osc})		t_{cyc}	0.95	–	10	µs
$\overline{INT}$ and TIMER Pulse Width (See Interrupt Section)		t_{WL}, t_{WH}	t_{cyc}+250	–	–	ns
$\overline{RESET}$ Pulse Width		t_{RWL}	t_{cyc}+250	–	–	ns
$\overline{RESET}$ Delay Time (External Capacitance = 1.0 µF)		t_{RHL}	–	100	–	ms
$\overline{INT}$ Zero Crossing Detection Input Frequency		f_{INT}	0.03	–	1.0	kHz
External Clock Input Duty Cycle (EXTAL)		–	40	50	60	%

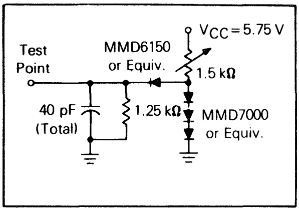


FIGURE 2 – TTL EQUIVALENT TEST LOAD (PORT B)

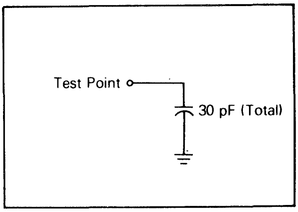


FIGURE 3 – CMOS EQUIVALENT TEST LOAD (PORT A)

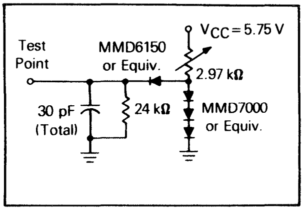


FIGURE 4 – TTL EQUIVALENT TEST LOAD (PORTS A AND C)

SIGNAL DESCRIPTION

The input and output signals for the MCU, shown in Figure 1, are described in the following paragraphs.

V_{CC} AND V_{SS}

Power is supplied to the MCU using these two pins. V_{CC} is power and V_{SS} is the ground connection.

$\overline{INT}$

This pin provides the capability for asynchronously applying an external interrupt to the MCU. Refer to Interrupts section for additional information.

XTAL AND EXTAL

These pins provide connections to the on-chip clock oscillator circuit. A crystal, a resistor, or an external signal, depending on the user selectable manufacturing mask option, can be connected to these pins to provide a system clock source with various stability/cost tradeoffs. Lead lengths and stray capacitance on these two pins should be minimized. Refer to Internal Clock Generator Options section for recommendations about these inputs.

TIMER

This pin allows an external input to be used to decrement the internal timer circuitry. Refer to Timer section for additional information about the timer circuitry.

$\overline{RESET}$

This pin allows resetting of the MCU at times other than the automatic resetting capability already in the MCU. Refer to Resets section for additional information.

NUM

This pin is not for user application and must be connected to V_{SS}.

INPUT/OUTPUT LINES (PA0-PA7, PB0-PB7, PC0-PC3)

These 20 lines are arranged into two 8-bit ports (A and B) and one 4-bit port (C). All lines are programmable as either inputs or outputs under software control of the data direction registers. Refer to Inputs/Outputs section for additional information.

MEMORY

As shown in Figure 6, the MCU is capable of addressing 2048 bytes of memory and I/O registers with its program counter. The EF6805P6 MCU has implemented 1984 of these locations. This consists of: 1796 bytes of user ROM, **(from $080 to $783) 116 bytes of self-check ROM, 64 bytes of user RAM, 6 bytes of port I/O, and 2 timers registers.**

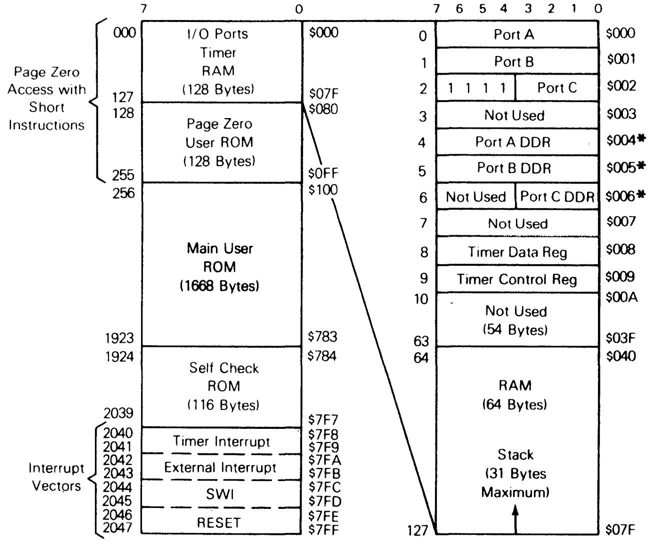
The stack area is used during the processing of interrupt and subroutine calls to save the processor state. The register contents are pushed onto the stack in the order shown in Figure 7. Because the stack pointer decrements during pushes, the low order byte (PCL) of the program counter is stacked first; then the high order three bits (PCH) are stacked. This ensures that the program counter is loaded correctly, during pulls from the stack, since the stack pointer increments during pulls. A subroutine call results in only the program counter (PCL, PCH) contents being pushed onto the stack. The remaining CPU registers are not pushed.

CENTRAL PROCESSING UNIT

The CPU of the EF6805 Family is implemented independently from the I/O or memory configuration. Consequently, it can be treated as an independent central processor communicating with I/O and memory via internal address, data, and control buses.

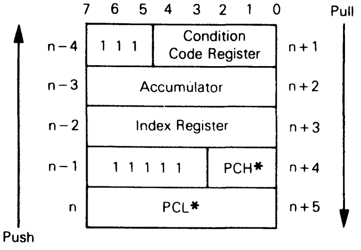
REGISTERS

The 6805 Family CPU has five registers available to the programmer. They are shown in Figure 8 and are explained in the following paragraphs.



*Caution: Data direction registers (DDRs) are write-only; they read as $FF.

FIGURE 5 – MCU ADDRESS MAP



*For subroutine calls, only PCL and PCH are stacked.

FIGURE 6 – INTERRUPT STACKING ORDER

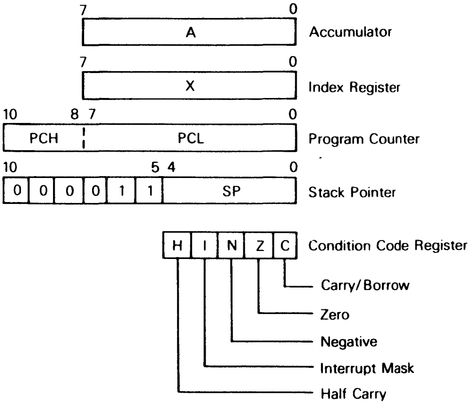


FIGURE 7 – PROGRAMMING MODEL

ACCUMULATOR (A)

The accumulator is a general purpose 8-bit register used to hold operands and results of arithmetic calculations or data manipulations.

INDEX REGISTER (X)

The index register is an 8-bit register used for the indexed addressing mode. It contains an 8-bit value that may be added to an instruction value to create an effective address. The index register can also be used for data manipulations using the read-modify-write instructions. The index register may also be used as a temporary storage area.

PROGRAM COUNTER (PC)

The program counter is an 11-bit register that contains the address of the next instruction to be executed.

STACK POINTER (SP)

The stack pointer is an 11-bit register that contains the address of the next free location on the stack. Initially, the stack pointer is set to location $07F and is decremented as data is pushed onto the stack and incremented as data is pulled from the stack. The six most significant bits of the stack pointer are permanently configured to 000011. During an MCU reset or the reset stack pointer (RSP) instruction, the stack pointer is set to location $07F. Subroutines and interrupts may be nested down to location $061 (31 bytes maximum) which allows the programmer to use up to 15 levels of subroutine calls.

CONDITION CODE REGISTER (CC)

The condition code register is a 5-bit register in which four bits are used to indicate the results of the instruction just executed. These bits can be individually tested by a program and specific action taken as a result of their state. Each individual condition code register bit is explained in the following paragraphs.

HALF CARRY (H) – Set during ADD and ADC instructions to indicate that a carry occurred between bits 3 and 4.

INTERRUPT (I) – This bit is set to mask (disable) the timer and external interrupt ($\overline{\text{INT}}$). If an interrupt occurs while this bit is set, the interrupt is latched and is processed as soon as the interrupt is cleared.

NEGATIVE (N) – Used to indicate that the result of the last arithmetic, logical, or data manipulation was negative (bit 7 in result equal to a logical one).

ZERO (Z) – Used to indicate that the result of the last arithmetic, logical, or data manipulation was zero.

CARRY/BORROW (C) – Used to indicate that a carry or borrow out of the arithmetic logic unit (ALU) occurred during the last arithmetic operation. This bit is also affected during bit test and branch instructions plus shifts and rotates.

TIMER

The EF6805P6 MCU timer circuitry is shown in Figure 8. The 8-bit counter may be loaded under program control and is decremented toward zero by the clock input (prescaler output). When the timer reaches zero, the timer interrupt request bit (bit 7) in the timer control register (TCR) is set. The timer interrupt can be masked (disabled) by setting the timer interrupt mask bit (bit 6) in the TCR. The interrupt bit (I bit) in the condition code register also prevents a timer interrupt from being processed. The MCU responds to this interrupt by saving the present CPU state on the stack, fetching the timer interrupt vector from locations $7F8 and $7F9, and executing the interrupt routine; see the Interrupts section. THE TIMER INTERRUPT REQUEST BIT MUST BE CLEARED BY SOFTWARE.

The clock input to the timer can be from an external source (decrementing of timer counter occurs on a positive transition of the external source) applied to the TIMER input pin or it can be the internal $\phi 2$ signal. Three machine cycles are required for a change in state of the TIMER pin to decrement the timer prescaler. The maximum frequency of a signal that can be recognized by the TIMER or $\overline{\text{INT}}$ pin logic is dependent on the parameter labeled t_{WL}, t_{WH}. The pin logic that recognizes the high (or low) state on the pin must also recognize the low state on the pin in order to "re-arm" the internal logic. Therefore, the period can be calculated as follows: (assumes 50/50 duty cycle for a given period)

$$t_{cyc} \times 2 + 250 \text{ ns} = \text{period} = \frac{1}{\text{freq}}$$

The period is not simply $t_{WL} + t_{WH}$. This computation is allowable, but it does reduce the maximum allowable frequency by defining an unnecessarily longer period (250 ns twice).

When the $\phi 2$ signal is used as the source, it can be gated by an input applied to the TIMER input pin allowing the user to easily perform pulse-width measurements. (NOTE: For

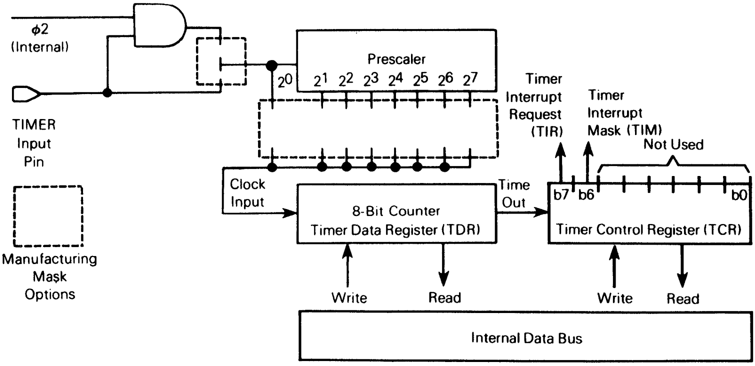


FIGURE 8 – TIMER BLOCK DIAGRAM

ungated $\phi2$ clock inputs to the timer prescaler, the TIMER pin should be tied to V_{CC}.) The source of the clock input is one of the mask options that is specified before manufacture of the MCU.

A prescaler option can be applied to the clock input that extends the timing interval up to a maximum of 128 counts before decrementing the counter. This prescaling mask option is also specified before manufacture. To avoid truncation errors, the prescaler is cleared when bit 3 of the timer counter register is written to a logic one. (This bit always needs a logic 0).

The timer continues to count past zero, falling from $00 to$FF and then continuing the countdown. Thus, the counter can be read at any time by reading the timer data register (TDR). This allows a program to determine the length of time since a timer interrupt has occurred without disturbing the counting process.

At power-up or reset, the prescaler and counter are initialized with all logical ones, the timer interrupt request bit (bit 7) is cleared and the timer interrupt mask bit (bit 6) is set.

SELF-CHECK

The self-check capability of the EF6805P6 MCU provides an internal check to determine if the part is functional. Connect the MCU as shown in Figure 9 and monitor the output of port C bit 3 for an oscillation of approximately 7 Hz. A 10volt level on the TIMER input, pin 7, energizes the ROM-based self-check feature. The self-check program exercises the RAM, ROM, TIMER, interrupts, and I/O ports.

RESETS

The MCU can be reset three ways: by initial power-up, by the external reset input (RESET) and by optional, internal, low-voltage detect circuits. The RESET input consists mainly of a Schmitt trigger that senses the RESET line logic level. A typical reset Schmitt trigger hysteresis curve is shown in Figure 11. The Schmitt trigger provides an internal reset voltage if it senses a logical zero on the RESET pin.

POWER-ON RESET (POR)

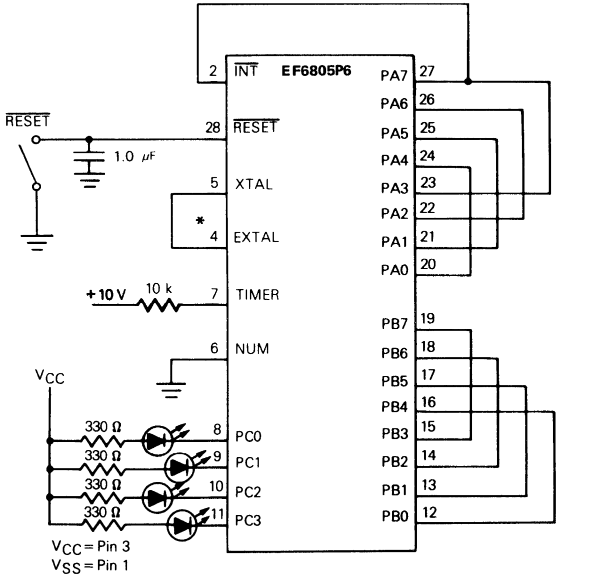
An internal reset is generated upon power-up that allows the internal clock generator to stabilize. A delay of t_{RHL} milliseconds is required before allowings the RESET input to go high. See the power and reset timing diagram (see Figure 10). Connecting a capacitor to the RESET input (see Figure 12) typically provides sufficient delay. During power-up, the Schmitt trigger switches on (removes reset) when RESET rise to $V_{IRES}+$.

EXTERNAL RESET INPUT

The MCU is reset when a logic zero is applied to the RESET input for a period longer than one machine cycle (t_{cyc}). Under this type of reset, the Schmitt trigger switches off are $V_{IRES}-$ to provide an internal reset voltage.

LOW VOLTAGE INHIBIT (LVI)

The optional low-voltage detection circuit causes a reset of the MCU if the power supply voltage falls below a certain level (V_{LVI}). The only requirement is that the V_{CC} must remain at or below the V_{LVI} threshold for one t_{cyc} minimum.



*This connection depends on the clock oscillator user selectable mask option.
Use crystal if that option is selected.

Self-Check Error Patterns

PC1	PC0	Problem
0	0	Interrupt Failure
0	1	Bad Port A or Port B
1	0	Bad RAM
1	1	Bad RAM
All 4 LEDs Flashing		Good Device

NOTE
When PC1 or PC0 is 0, the LED is on.

FIGURE 9 – SELF-CHECK CONNECTIONS

In typical applications, the V_{CC} bus filter capacitor will eliminate negative-going voltage glitches of less than one t_{cyc}. The output from the low-voltage detector is connected directly to the internal reset circuitry. It also forces the RESET pin low via a strong discharge device through a resistor. The internal reset is removed once the power supply voltage rises above a recovery level (V_{LVR}) at which time a normal power-one reset occurs.

INTERNAL CLOCK GENERATORS OPTIONS

The internal clock generator circuit is designed to require a minimum of external components. A crystal, a resistor, a jumper wire, or an external signal may be used to generate a system clock with various stability/cost tradeoffs.

A manufacturing mask option is used to select crystal or resistor operation.

The different connection methods are shown in Figure 13. Crystal specifications and suggested PC board layouts are given in Figure 14. A resistor selection graph is given in Figure 15.

The crystal oscillator start-up time is a function of many variables: crystal parameters (especially R_S, oscillator load capacitances, IC parameters, ambient temperature, supply voltage and supply voltage turn-on time). To ensure rapid oscillator start-up, neither the crystal characteristics nor the load capacitances should exceed recommendations.

When utilizing the on-board oscillator, the MCU should remain in the reset condition ($\overline{\text{RESET}}$ pin voltage below V_{IRES+}) until the oscillator has stabilized at its operating frequency. Several factors are involved in calculating current specifications.

One V_{CC} minimum is reached, the external $\overline{\text{RESET}}$ capacitor will begin to charge at a rate dependent on the capacitor value. The charging current is supplied from V_{CC} through a large resistor, so it functions almost like a constant current source until the reset voltages rises above V_{IRES+}. Therefore, the $\overline{\text{RESET}}$ pin will charge a approximately

$$(V_{IRES+}) \cdot C_{ext} = I_{RES} \cdot t_{RHL}$$

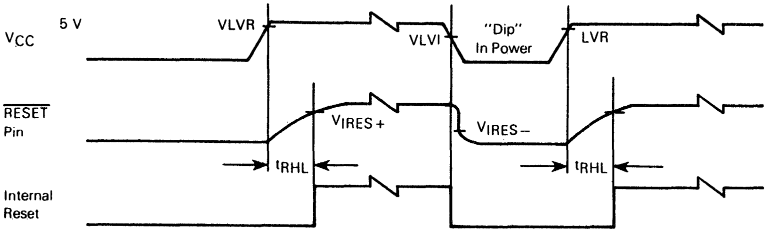


FIGURE 10 – POWER AND RESET TIMING

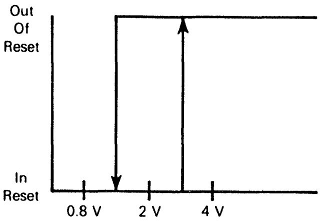


FIGURE 11 – TYPICAL RESET SCHMITT TRIGGER HYSTERESIS

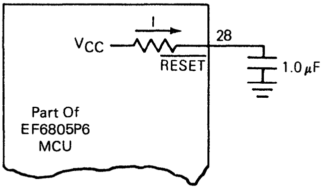
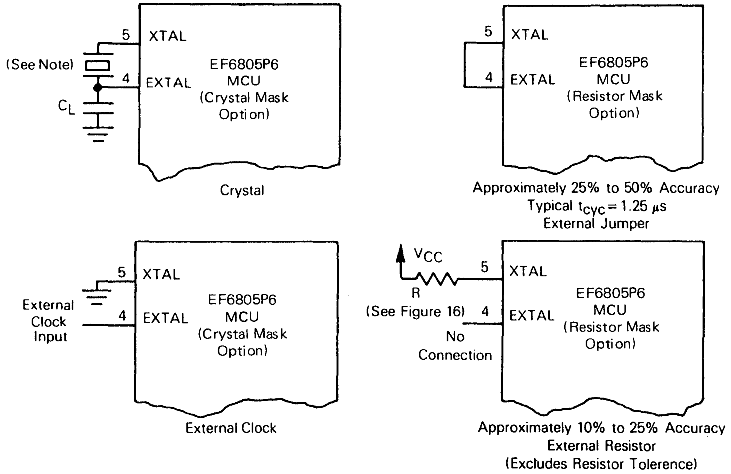


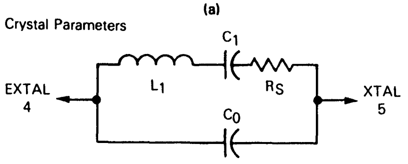
FIGURE 12 – POWER-UP RESET DELAY CIRCUIT



NOTE: The recommended C_L value with a 4.0 MHz crystal is 27 pF, maximum, including system distributed capacitance. There is an internal capacitance of approximately 25 pF on the XTAL pin. For crystal frequencies other than 4 MHz; the total capacitance on each pin should be scaled as the inverse of the frequency ratio. For example, with a 2 MHz crystal, use approximately 50 pF on EXTAL and approximately 25 pF on XTAL. The exact value depends on the Motional-Arm parameters of the crystal used.

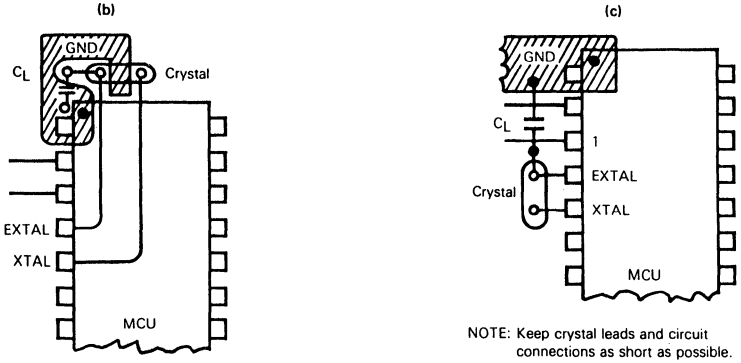
FIGURE 13 – CLOCK GENERATOR OPTIONS

(a)



AT – Cut Parallel Resonance Crystal
C_0 = 7 pF Max.
Freq. = 4.0 MHz @ C_L = 24 pF
R_S = 50 ohms Max.

Piezoelectric ceramic resonators which have the equivalent specifications may be use instead of crystal oscillators. Follow ceramic resonator manufacturer's suggestions for C_0, C_1 and R_S values.



NOTE: Keep crystal leads and circuit connections as short as possible.

FIGURE 14 – CRYSTAL MOTIONAL ARM PARAMETERS AND SUGGESTED PC BOARD LAYOUT

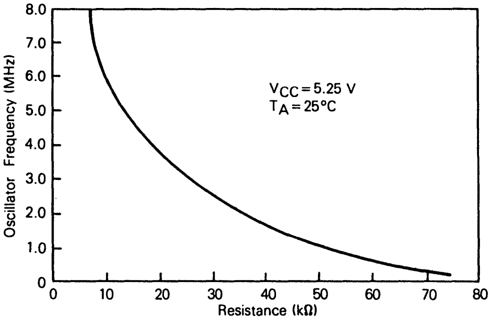


FIGURE 15 – TYPICAL FREQUENCY SELECTION FOR RESISTOR OSCILLATOR OPTION

INTERRUPTS

The EF6805P6 MCU can be interrupted three different ways: through the external interrupt (INT) input pin, the internal timer interrupt request, or the software interrupt instruction (SW1). When any interrupt occurs, the current instruction (including SW1) is completed, processing is suspended, the present CPU state is pushed onto the stack, the interrupt bit (I) in the condition code register is set, the address of the interrupt routine is obtained from the appropriate interrupt vector address and the interrupt routine is executed. Stacking the CPU registers, setting the I bit, and vector fetching requires a total of 11 t_{cyc} periods for completion.

A flowchart of the interrupt sequence is shown in Figure 16. The interrupt service routine must end with a return from interrupt (RTI) instruction which allows the MCU to resume processing of the program prior to the interrupt (by unstacking the previous CPU state). Unlike $\overline{\text{RESET}}$, hardware interrupts do not cause the current instruction execution to be halted, but are considered pending until the current instruction execution is complete.

When the current instruction is complete, the processor checks all pending hardware interrupts and if unmasked, proceeds with interrupt processing; otherwise, the next instruction is fetched and executed. Note that masked interrupts are latched for later interrupt service.

If both an external interrupt and a timer interrupt are pending at the end of an instruction execution, the external interrupt is serviced first. The SWI is executed as any other instruction.

The external interrupt is internally synchronized and then latched on the falling edge of $\overline{\text{INT}}$. A sinusoidal input signal (f_{INT} maximum) can be used to generate an external interrupt, as shown in Figure 17(a), for use as a zero-crossing detector (for negative transitions of the ac sinusoid). This allows applications such as servicing time-of-day routines and engaging/disengaging ac power control devices. Off-chip full-wave rectification provides an interrupt at every zero-crossing of the ac signal and thereby provides a 2f clock.

For digital applications, the $\overline{\text{INT}}$ pin can be driven by a digital signal. The maximum frequency of a signal that can be recognized by the TIMER or $\overline{\text{INT}}$ pin logic is dependent on the parameter labeled t_{WL}, t_{WH}. The pin logic that recognizes the high (or low) state on the pin must also recognize the low (or high) state on the pin in order to "rearm" the internal logic. Therefore, the period can be calculated as follows: (assumes 50/50 duty cycle for a given period)

$$t_{cyc} x2 + 250 \text{ ns} = \text{period} = \frac{1}{\text{freq}}$$

The period is not simply $t_{WL} + t_{WH}$. This computation is allowable, but it does reduce the maximum allowable frequency by defining an unnecessarily longer period (250 ns twice). See Figure 17(b).

A software interrupt (SWI) is an executable instruction which is executed regardless of the state of the I bit in the condition code register. Note that if the I bit is zero, SWI executes after the other interrupts. SWIs are usually used as break-points for debugging or as system calls.

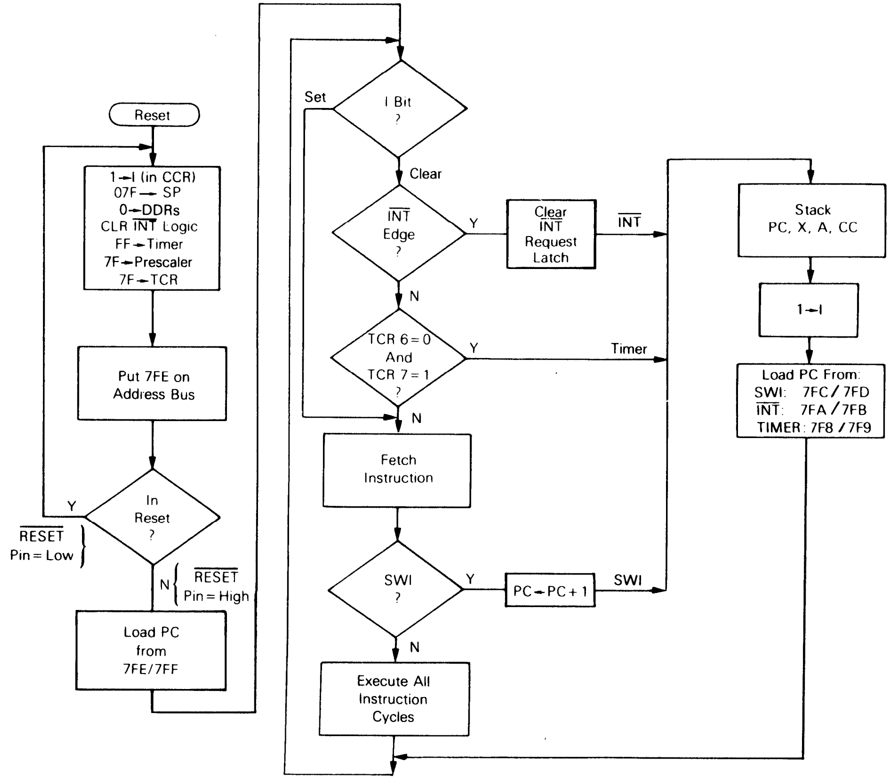


FIGURE 16 – RESET AND INTERRUPT PROCESSING FLOWCHART

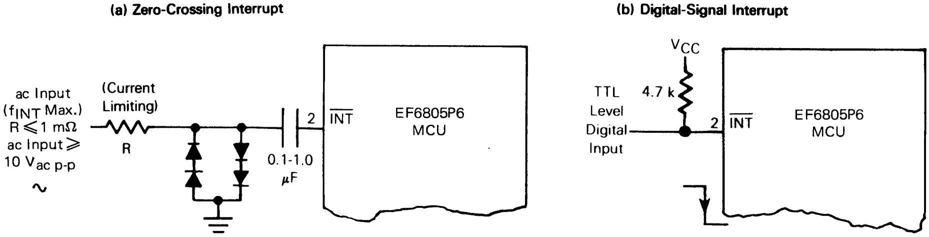


FIGURE 17 – TYPICAL INTERRUPT CIRCUITS

INPUT/OUTPUT

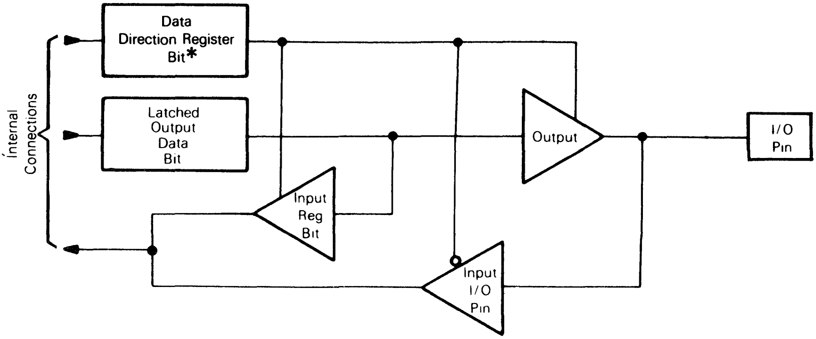
There are 20 input/output pins. The $\overline{INT}$ pin may also be polled with branch instructions to provide an additional input pin. All pins (port A, B, and C) are programmable as either inputs or outputs under software control of the corresponding write-only data direction register (DDR). The port I/O programming is accomplished by writing the corresponding bit in the port DDR to a logic "1" for output or a logic "0" for input. On reset, all the DDRs are initialized to a logic "0" state to put the ports in the input mode. To avoid undefined levels, the port output registers are not initialized on reset, but may be written before setting the DDR bits. When programmed as outputs, the latched output data is readable as input data.regardless of the logic levels at the output pin due to output loading; see Figure 18.When port B is programmed for outputs, it is capable of sinking 10 mA and sourcing 1 mA on each pin.

All input/output lines are TTL compatible as both inputs and outputs. Ports B and C are CMOS compatible as inputs. Port A may be made CMOS compatible as outputs with a mask option. The address map in Figure 5 gives the address of data registers and DDRs. The register configuration is provided in Figure 19 and Figure 20 provides some examples of port connections.

Caution

The corresponding DDRs for ports A, B, and C are write-only registers (registers at $004, $005, and $006). A read operation on these registers is undefined. Since BSET and BCLR are read-modify-write functions, they cannot be used to set or clear a DDR bit (all "unaffected" bits would be set). It is recommended that all DDR bits in a port be written using a single-store instruction.

The latched output data bit (see Figure 18) may always be written. Therefore, any write to a port writes all of its data bits even though the port DDR is set to input. This may be used to initialize the data registers and avoid undefined outputs; however, care must be exercised when using read-modify-write instructions since the data read corresponds to the pin level if the DDR is an input ("0") and corresponds to the latched output data when the DDR is an output ("1").

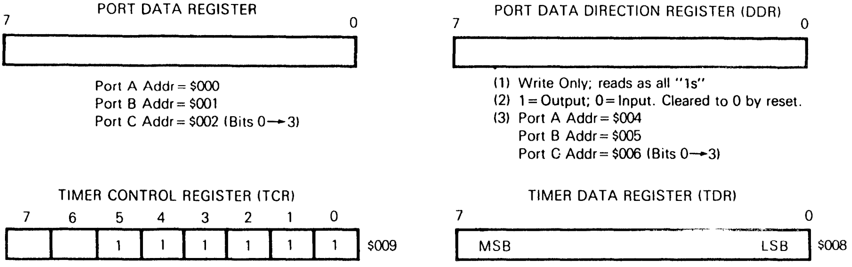


Data Direction Register Bit	Output Data Bit	Latched Output State	Input To MCU
1	0	0	0
1	1	1	1
0	X	High-Z **	Pin

* DDR is a write-only register and reads as all "1s".

** Ports A (with CMOS drive disabled), B, and C are three state ports. Port A has optional internal pullup devices to provide CMOS drive capability. See Electrical Characteristics tables for complete information.

FIGURE 18 – TYPICAL PORT I/O CIRCUITRY


PORT DATA REGISTER
7
0
PORT DATA DIRECTION REGISTER (DDR)
7
0
TIMER CONTROL REGISTER (TCR)
7 6 5 4 3 2 1 0
1 1 1 1 1 1
$009
TIMER DATA REGISTER (TDR)
7
0
MSB
LSB
$008

Port A Addr = $000
Port B Addr = $001
Port C Addr = $002 (Bits 0→3)

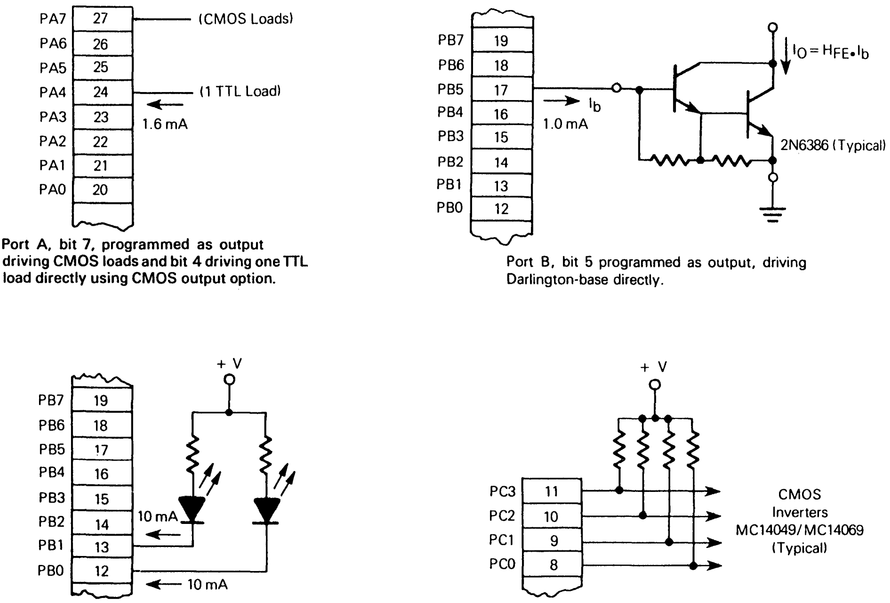
(1) Write Only; reads as all "1s"
(2) 1 = Output; 0 = Input. Cleared to 0 by reset.
(3) Port A Addr = $004
Port B Addr = $005
Port C Addr = $006 (Bits 0→3)

TCR7 – Timer Interrupt Status Bit: Set when TDR goes to zero; must be cleared by software. Cleared to 0 by reset.

TCR6 Bit 6 – Timer Interrupt Mask Bit: 1 = timer interrupt masked (disabled). Set to 1 by reset.

TCR Bits 5, 4, 3, 2, 1, 0 read as "1s" – unused bits.

FIGURE 19 – MCU REGISTER CONFIGURATION



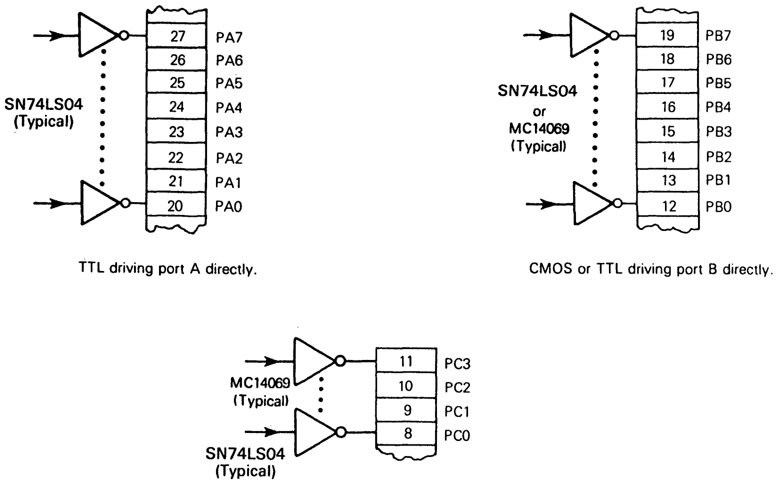
Port A, bit 7, programmed as output driving CMOS loads and bit 4 driving one TTL load directly using CMOS output option.

Port B, bit 5 programmed as output, driving Darlington-base directly.

Port B, bit 0 and bit 1 programmed as output, driving LEDs directly.

Port C, bits 0-3 programmed as output, driving CMOS loads, using external pullup resistors.

FIGURE 20 (a) – TYPICAL OUTPUT MODE PORT CONNECTIONS



TTL driving port A directly.

CMOS or TTL driving port B directly.

CMOS and TTL driving port C directly.

FIGURE 20 (b) – TYPICAL INPUT MODE PORT CONNECTIONS

SOFTWARE

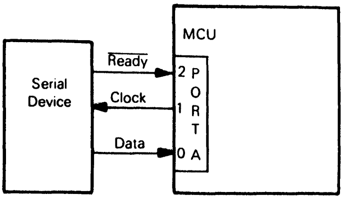
BIT MANIPULATION

The EF6805P6 MCU has the ability to set or clear any single random access memory or input/output bit (except the data direction register, see Caution below), with a single instruction (BSET, BCLR). Any bit in page zero including ROM, except the DDRs, can be tested, using the BRSET and BRCLR instructions, and the program branches as a result of its state. The carry bit equals the value of the bit referenced by BRSET or BRCLR. A rotate instruction may then be used to accumulate serial input data in a RAM location or register. The capability to work with any bit in RAM, ROM, or I/O allows the user to have individual flags in RAM or to handle I/O bits as control lines.

The coding example in Figure 21 illustrates the usefulness of the bit manipulation and test instructions. Assume that the MCU is to communicate with an external serial device. The external device has a data ready signal, a data output line, and a clock line to clock data one bit at a time, LSB first, out of the device. The MCU waits until the data is ready, clocks the external device, picks up the data in the carry flag (C bit), clears the clock line, and finally accumulates the data bit in a RAM location.

Caution

The corresponding DDRs for ports A, B, and C are write-only registers (registers at $004, $005, and $006). A read operation on these registers is undefined. Since BSET and BCLR are read-modify-write functions, they cannot be used to set or clear a DDR bit (all "unaffected" bits would be set). It is recommended that all DDR bits in a port be written using a single-store instruction.



```
                     •
                     •
                     •
SELF    BRSET    2, PORTA, SELF
                     •
                     •
                     •
        BSET     1, PORTA
        BRCLR    0, PORTA, CONT
CONT    BCLR     1, PORTA
        ASR      RAMLOC
                     •
                     •
                     •
```

FIGURE 21 – BIT MANIPULATION EXAMPLE

ADDRESSING MODES

The EF6805P6 MCU has 10 addressing modes which are explained briefly in the following paragraphs. For additional details and graphical illustrations, refer to the 6805 Family User's Manual.

The term "effective address" (EA) is used in describing the address modes. EA is defined as the address from which the argument for an instruction is fetched or stored.

IMMEDIATE — In the immediate addressing mode, the operand is contained in the byte immediately following the opcode. The immediate addressing mode is used to access constants which do not change during program execution (e.g., a constant used to initialize a loop counter).

DIRECT — In the direct addressing mode, the effective address of the argument is contained in a single byte following the opcode byte. Direct addressing allows the user to directly address the lowest 256 bytes in memory with a single 2-byte instruction. This includes the on-chip RAM and I/O registers and 128 bytes of ROM. Direct addressing is an effective use of both memory and time.

EXTENDED — In the extended addressing mode, the effective address of the argument is contained in the two bytes following the opcode. Instructions using extended addressing are capable of referencing arguments anywhere in memory with a single 3-byte instruction. When using the Motorola assembler, the programmer need not specify whether an instruction uses direct or extended addressing. The assembler automatically selects the shortest form of the instruction.

RELATIVE — The relative addressing mode is only used in branch instructions. In relative addressing, the contents of the 8-bit signed byte following the opcode (the offset) is added to the PC if and only if the branch condition is true. Otherwise, control proceeds to the next instruction. The span of relative addressing is from − 126 to + 129 from the opcode address. The programmer need not worry about calculating the correct offset when using the Motorola assembler since it calculates the proper offset and checks to see if it is within the span of the branch.

INDEXED, NO OFFSET — In the indexed, no offset addressing mode, the effective address of the argument is contained in the 8-bit index register. Thus, this addressing mode can access the first 256 memory locations. These instructions are only one byte long. This mode is often used to move a pointer through a table or to hold the address of a frequently referenced RAM or I/O location.

INDEXED, 8-BIT OFFSET — In the indexed, 8-bit offset addressing mode, the effective address is the sum of the contents of the unsigned 8-bit index register and the unsigned byte following the opcode. This addressing mode is useful in selecting the kth element in an n element table. With this 2-byte instruction, k would typically be in X with the address of the beginning of the table in the instruction. As such, tables may begin anywhere within the first 256 addressable locations and could extend as far as location 510 ($1FE is the last location at which the instruction may begin).

INDEXED, 16-BIT OFFSET — In the indexed, 16-bit offset addressing mode, the effective address is the sum of the contents of the unsigned 8-bit index register and the two unsigned bytes following the opcode. This addressing mode can be used in a manner similar to indexed, 8-bit offset, except that this 3-byte instruction allows tables to be anywhere in memory. As with direct and extended addressing, the Motorola assembler determines the shortest form of indexed addressing.

BIT SET/CLEAR — In the bit set/clear addressing mode, the bit to be set or cleared is part of the opcode, and the byte following the opcode specifies the direct address of the byte in which the specified bit is to be set or cleared. Thus, any read/write bit in the first 256 locations of memory, including I/O, can be selectively set or cleared with a single 2-byte instruction.

Caution

The corresponding DDRs for ports A, B, and C are write-only registers (registers at $004, $005, and $006). A read operation on these registers is undefined. Since BSET and BCLR are read-modify-write functions, they cannot be used to set or clear a DDR bit (all "unaffected" bits would be set). It is recommended that all DDR bits in a port be written using a single-store instruction.

BIT TEST AND BRANCH — The bit test and branch addressing mode is a combination of direct addressing and relative addressing. The bit and condition (set or clear) which is to be tested is included in the opcode, and the address of the byte to be tested is in the single byte immediately following the opcode byte. The signed relative 8-bit offset is in the third byte and is added to the value of the PC if the branch condition is true. This single 3-byte instruction allows the program to branch based on the condition of any readable bit in the first 256 locations of memory. The span of branching is from − 125 to + 130 from the opcode address. The state of the tested bit is also transferred to the carry bit of the condition code register.

Caution

The corresponding DDRs for ports A, B, and C are write-only registers (registers at $004, $005, and $006). A read operation on these registers is undefined. Since BSET and BCLR are read-modify-write functions, they cannot be used to set or clear a DDR bit (all "unaffected" bits would be set). It is recommended that all DDR bits in a port be written using a single-store instruction.

INHERENT — In the inherent addressing mode, all the information necessary to execute the instruction is contained in the opcode. Operations specifying only the index register or accumulator, as well as control instruction with no other arguments, are included in this mode. These instructions are one byte long.

INSTRUCTION SET

The EF6805P6 MCU has a set of 59 basic instructions, which when combined with the 10 addressing modes produce 207 usable opcodes. They can be divided into five different types: register/memory, read-modify-write, branch, bit manipulation, and control. The following paragraphs briefly explain each type. All the instructions within a given type are presented in individual tables.

REGISTER/MEMORY INSTRUCTIONS – Most of these instructions use two operands. One operand is either the accumulator or the index register. The other operand is obtained from memory using one of the addressing modes. The jump unconditional (JMP) and jump to subroutine (JSR) instructions have no register operands. Refer to Table 1.

READ-MODIFY-WRITE INSTRUCTIONS — These instructions read a memory location or a register, modify or test its contents, and write the modified value back to memory or to the register. The test for negative or zero (TST) instruction is included in read-modify-write instructions through it does not perform the write. Refer to Table 2.

Caution

The corresponding DDRs for ports A, B, and C are write-only registers (registers at $004, $005, and $006). A read operation on these registers is undefined. Since BSET and BCLR are read-modify-write functions, they cannot be used to set or clear a DDR bit (all "unaffected" bits would be set). It is recommended that all DDR bits in a port be written using a single-store instruction.

BRANCH INSTRUCTIONS – The branch instructions cause a branch from the program when a certain condition is met. Refer to Table 3.

BIT MANIPULATION INSTRUCTIONS — These instructions are used on any bit in the first 256 bytes of the memory. One group either sets or clears. The other group performs the bit test branch operations. Refer to Table 4.

Caution

The corresponding DDRs for ports A, B, and C are write-only registers (registers at $004, $005, and $006). A read operation on these registers is undefined. Since BSET and BCLR are read-modify-write functions, they cannot be used to set or clear a DDR bit (all "unaffected" bits would be set). It is recommended that all DDR bits in a port be written using a single-store instruction.

CONTROL INSTRUCTIONS – The control instructions control the MCU operations during program execution. Refer to Table 5.

ALPHABETICAL LISTING – The complete instruction set is given in alphabetical order in Table 6.

OPCODE MAP SUMMARY – Table 7 is an opcode map for the instructions used on the MCU.

TABLE 1 — REGISTER/MEMORY INSTRUCTIONS

		Addressing Modes																	
		Immediate			Direct			Extended			Indexed (No Offset)			Indexed (8-Bit Offset)			Indexed (16-Bit Offset)		
Function	Mnemonic	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles	OP Code	# Bytes	# Cycles
Load A from Memory	LDA	A6	2	2	B6	2	4	C6	3	5	F6	1	4	E6	2	5	D6	3	6
Load X from Memory	LDX	AE	2	2	BE	2	4	CE	3	5	FE	1	4	EE	2	5	DE	3	6
Store A in Memory	STA	—	—	—	B7	2	5	C7	3	6	F7	1	5	E7	2	6	D7	3	7
Store X in Memory	STX	—	—	—	BF	2	5	CF	3	6	FF	1	5	EF	2	6	DF	3	7
Add Memory to A	ADD	AB	2	2	BB	2	4	CB	3	5	FB	1	4	EB	2	5	DB	3	6
Add Memory and Carry to A	ADC	A9	2	2	B9	2	4	C9	3	5	F9	1	4	E9	2	5	D9	3	6
Subtract Memory	SUB	A0	2	2	B0	2	4	C0	3	5	F0	1	4	E0	2	5	D0	3	6
Subtract Memory from A with Borrow	SBC	A2	2	2	B2	2	4	C2	3	5	F2	1	4	E2	2	5	D2	3	6
AND Memory to A	AND	A4	2	2	B4	2	4	C4	3	5	F4	1	4	E4	2	5	D4	3	6
OR Memory with A	ORA	AA	2	2	BA	2	4	CA	3	5	FA	1	4	EA	2	5	DA	3	6
Exclusive OR Memory with A	EOR	A8	2	2	B8	2	4	C8	3	5	F8	1	4	E8	2	5	D8	3	6
Arithmetic Compare A with Memory	CMP	A1	2	2	B1	2	4	C1	3	5	F1	1	4	E1	2	5	D1	3	6
Arithmetic Compare X with Memory	CPX	A3	2	2	B3	2	4	C3	3	5	F3	1	4	E3	2	5	D3	3	6
Bit Test Memory with A (Logical Compare)	BIT	A5	2	2	B5	2	4	C5	3	5	F5	1	4	E5	2	5	D5	3	6
Jump Unconditional	JMP	—	—	—	BC	2	3	CC	3	4	FC	1	3	EC	2	4	DC	3	5
Jump to Subroutine	JSR	—	—	—	BD	2	7	CD	3	8	FD	1	7	ED	2	8	DD	3	9

TABLE 2 — READ-MODIFY-WRITE INSTRUCTIONS

		Addressing Modes														
		Inherent (A)			Inherent (X)			Direct			Indexed (No Offset)			Indexed (8 Bit Offset)		
Function	Mnemonic	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles
Increment	INC	4C	1	4	5C	1	4	3C	2	6	7C	1	6	6C	2	7
Decrement	DEC	4A	1	4	5A	1	4	3A	2	6	7A	1	6	6A	2	7
Clear	CLR	4F	1	4	5F	1	4	3F	2	6	7F	1	6	6F	2	7
Complement	COM	43	1	4	53	1	4	33	2	6	73	1	6	63	2	7
Negate (2's Complement)	NEG	40	1	4	50	1	4	30	2	6	70	1	6	60	2	7
Rotate Left Thru Carry	ROL	49	1	4	59	1	4	39	2	6	79	1	6	69	2	7
Rotate Right Thru Carry	ROR	46	1	4	56	1	4	36	2	6	76	1	6	66	2	7
Logical Shift Left	LSL	48	1	4	58	1	4	38	2	6	78	1	6	68	2	7
Logical Shift Right	LSR	44	1	4	54	1	4	34	2	6	74	1	6	64	2	7
Arithmetic Shift Right	ASR	47	1	4	57	1	4	37	2	6	77	1	6	67	2	7
Test for Negative or Zero	TST	4D	1	4	5D	1	4	3D	2	6	7D	1	6	6D	2	7

TABLE 3 — BRANCH INSTRUCTIONS

		Relative Addressing Mode		
Function	Mnemonic	Op Code	# Bytes	# Cycles
Branch Always	BRA	20	2	4
Branch Never	BRN	21	2	4
Branch IFF Higher	BHI	22	2	4
Branch IFF Lower or Same	BLS	23	2	4
Branch IFF Carry Clear	BCC	24	2	4
(Branch IFF Higher or Same)	(BHS)	24	2	4
Branch IFF Carry Set	BCS	25	2	4
(Branch IFF Lower)	(BLO)	25	2	4
Branch IFF Not Equal	BNE	26	2	4
Branch IFF Equal	BEQ	27	2	4
Branch IFF Half Carry Clear	BHCC	28	2	4
Branch IFF Half Carry Set	BHCS	29	2	4
Branch IFF Plus	BPL	2A	2	4
Branch IFF Minus	BMI	2B	2	4
Branch IFF Interupt Mask Bit is Clear	BMC	2C	2	4
Branch IFF Interrupt Mask Bit is Set	BMS	2D	2	4
Branch IFF Interrupt Line is Low	BIL	2E	2	4
Branch IFF Interrupt Line is High	BIH	2F	2	4
Branch to Subroutine	BSR	AD	2	8

TABLE 4 — BIT MANIPULATION INSTRUCTIONS

		Addressing Modes					
		Bit Set/Clear			Bit Test and Branch		
Function	Mnemonic	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles
Branch IFF Bit n is set	BRSET n (n = 0....7)	—	—	—	2 • n	3	10
Branch IFF Bit n is clear	BRCLR n (n = 0....7)	—	—	—	01 + 2 • n	3	10
Set Bit n	BSET n (n = 0.. 7)	10 + 2 • n	2	7	—	—	—
Clear bit n	BCLR n (n = 0....7)	11 + 2 • n	2	7	—	—	—

TABLE 5 — CONTROL INSTRUCTIONS

		Inherent		
Function	Mnemonic	Op Code	# Bytes	# Cycles
Transfer A to X	TAX	97	1	2
Transfer X to A	TXA	9F	1	2
Set Carry Bit	SEC	99	1	2
Clear Carry Bit	CLC	98	1	2
Set Interrupt Mask Bit	SEI	9B	1	2
Clear Interrupt Mask Bit	CLI	9A	1	2
Software Interrupt	SWI	83	1	11
Return from Subroutine	RTS	81	1	6
Return from Interrupt	RTI	80	1	9
Reset Stack Pointer	RSP	9C	1	2
No-Operation	NOP	9D	1	2

TABLE 6 — INSTRUCTION SET

	Addressing Modes										Condition Code				
Mnemonic	Inherent	Immediate	Direct	Extended	Relative	Indexed (No Offset)	Indexed (8 Bits)	Indexed (16 Bits)	Bit Set/ Clear	Bit Test & Branch	H	I	N	Z	C
ADC		X	X	X		X	X	X			∧	●	∧	∧	∧
ADD		X	X	X		X	X	X			∧	●	∧	∧	∧
AND		X	X	X		X	X	X			●	●	∧	∧	●
ASL	X		X			X	X				●	●	∧	∧	∧
ASR	X		X			X	X				●	●	∧	∧	∧
BCC					X						●	●	●	●	●
BCLR									X		●	●	●	●	●
BCS					X						●	●	●	●	●
BEQ					X						●	●	●	●	●
BHCC					X						●	●	●	●	●
BHCS					X						●	●	●	●	●
BHI					X						●	●	●	●	●
BHS					X						●	●	●	●	●
BIH					X						●	●	●	●	●
BIL					X						●	●	●	●	●
BIT		X	X	X		X	X	X			●	●	∧	∧	●
BLO					X						●	●	●	●	●
BLS					X						●	●	●	●	●
BMC					X						●	●	●	●	●
BMI					X						●	●	●	●	●
BMS					X						●	●	●	●	●
BNE					X						●	●	●	●	●
BPL					X						●	●	●	●	●
BRA					X						●	●	●	●	●
BRN					X						●	●	●	●	●
BRCLR										X	●	●	●	●	∧
BRSET										X	●	●	●	●	∧
BSET									X		●	●	●	●	●
BSR					X						●	●	●	●	●
CLL	X										●	●	●	●	O
CLI	X										●	O	●	●	●
CLR	X		X			X	X				●	●	O	1	●
CMP		X	X	X		X	X	X			●	●	∧	∧	∧
COM	X		X			X	X				●	●	∧	∧	1
CPX		X	X	X		X	X	X			●	●	∧	∧	∧
DEC	X		X			X	X				●	●	∧	∧	●
EOR		X	X	X		X	X	X			●	●	∧	∧	●
INC	X		X			X	X				●	●	∧	∧	●
JMP			X	X		X	X	X			●	●	●	●	●
JSR			X	X		X	X	X			●	●	●	●	●
LDA		X	X	X		X	X	X			●	●	∧	∧	●
LDX		X	X	X		X	X	X			●	●	∧	∧	●
LSL	X		X			X	X				●	●	∧	∧	∧
LSR	X		X			X	X				●	●	O	∧	∧
NEQ	X		X			X	X				●	●	∧	∧	∧
NOP	X										●	●	●	●	●
ORA		X	X	X		X	X	X			●	●	∧	∧	●
ROL	X		X			X	X				●	●	∧	∧	∧
RSP	X										●	●	●	●	●

Condition Code Symbols:

H Half Carry (From Bit 3)	C Carry/Borrow
I Interrupt Mask	∧ Test and Set if True, Cleared Otherwise
N Negative (Sign Bit)	• Not Affected
Z Zero	

TABLE 6 – INSTRUCTION SET (CONTINUED)

	Addressing Modes										Condition Code				
Mnemonic	Inherent	Immediate	Direct	Extended	Relative	Indexed (No Offset)	Indexed (8 Bits)	Indexed (16 Bits)	Bit Set/ Clear	Bit Test & Branch	H	I	N	Z	C
RTI	X										?	?	?	?	?
RTS	X										●	●	●	●	●
SBC		X	X	X		X	X	X			●	●	Λ	Λ	Λ
SEC	X										●	●	●	●	1
SEI	X										●	1	●	●	●
STA			X	X		X	X	X			●	●	Λ	Λ	●
STX			X	X		X	X	X			●	●	Λ	Λ	●
SUB		X	X	X		X	X	X			●	●	Λ	Λ	Λ
SWI	X										●	1	●	●	●
TAX	X										●	●	●	●	●
TST	X		X			X	X				●	●	Λ	Λ	●
TXA	X										●	●	●	●	●

Condition Code Symbols:

H	Half Carry (From Bit 3)
I	Interrupt Mask
N	Negative (Sign Bit)
Z	Zero
C	Carry/Borrow
Λ	Test and Set if True, Cleared Otherwise
●	Not Affected
?	Load CC Register From Stack

EF6805 HMOS FAMILY

Features	EF6805CT	EF6805P2	EF6805P4	EF6805P6	EF6805R2	EF6805R3	EF6805T2	EF6805U2	EF6805U3	EF6805TV
Technology	HMOS	HMOS	HMOS	HMOS	HMOS	HMOS	HMOS	HMOS	HMOS	HMOS
Number of Pins	40	28	28	28	40	40	28	40	40	40
On-Chip RAM (Bytes)	240	64	112	64	64	112	64	64	112	96
On-Chip User ROM (Bytes)	4096	1100	1100	1796	2048	3776	2508	2048	3776	6144
External Bus	Yes	None	None	None	None	None	None	None	None	None
Bidirectional I/O Lines	29	20	20	20	24	24	19	24	24	32
Unidirectional I/O Lines	None	None	None	None	6 Inputs	6 Inputs	None	8 Inputs	8 Inputs	None
Other I/O Features	Timer,UART	Timer	Timer	Timer	Timer,A/D	Timer,A/D	Timer, PLL	Timer	Timer	Timer,D/A
External Interrupt Inputs	3	1	1	1	2	2	2	2	2	1
STOP and WAIT	No	No	No	No	No	No	No	No	No	No

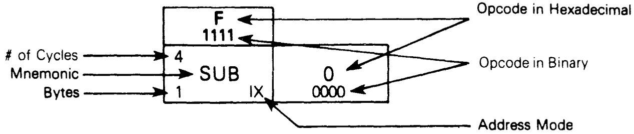
TABLE 7 – EF6805 HMOS FAMILY OPCODE MAP

	Bit Manipulation		Branch	Read-Modify-Write					Control		Register/Memory						
	BTB	BSC	REL	DIR	INH	INH	IX1	IX	INH	INH	IMM	DIR	EXT	IX2	IX1	IX	
Low \ Hi	0 0000	1 0001	2 0010	3 0011	4 0100	5 0101	6 0110	7 0111	8 1000	9 1001	A 1010	B 1011	C 1100	D 1101	E 1110	F 1111	Hi / Low
0 0000	10 BRSET0 3 BTB	7 BSET0 2 BSC	4 BRA 2 REL	6 NEG 2 DIR	4 NEG 1 INH	4 NEG 1 INH	7 NEG 2 IX1	6 NEG 1 IX	9 RTI 1 INH		2 SUB 2 IMM	4 SUB 2 DIR	5 SUB 3 EXT	6 SUB 3 IX2	5 SUB 2 IX1	4 SUB 1 IX	0 0000
1 0001	10 BRCLR0 3 BTB	7 BCLR0 2 BSC	4 BRN 2 REL						6 RTS 1 INH		2 CMP 2 IMM	4 CMP 2 DIR	5 CMP 3 EXT	6 CMP 3 IX2	5 CMP 2 IX1	4 CMP 1 IX	1 0001
2 0010	10 BRSET1 3 BTB	7 BSET1 2 BSC	4 BHI 2 REL								2 SBC 2 IMM	4 SBC 2 DIR	5 SBC 3 EXT	6 SBC 3 IX2	5 SBC 2 IX1	4 SBC 1 IX	2 0010
3 0011	10 BRCLR1 3 BTB	7 BCLR1 2 BSC	4 BLS 2 REL	6 COM 2 DIR	4 COMA 1 INH	4 COMX 1 INH	7 COM 2 IX1	6 COM 1 IX	11 SWI 1 INH		2 CPX 2 IMM	4 CPX 2 DIR	5 CPX 3 EXT	6 CPX 3 IX2	5 CPX 2 IX1	4 CPX 1 IX	3 0011
4 0100	10 BRSET2 3 BTB	7 BSET2 2 BSC	4 BCC 2 REL	6 LSR 2 DTR	4 LSRA 1 INH	4 LSRX 1 INH	7 LSR 2 IX1	6 LSR 1 IX			2 AND 2 IMM	4 AND 2 DIR	5 AND 3 EXT	6 AND 3 IX2	5 AND 2 IX1	4 AND 1 IX	4 0100
5 0101	10 BRCLR2 3 BTB	7 BCLR2 2 BSC	4 BCS 2 REL								2 BIT 2 IMM	4 BIT 2 DIR	5 BIT 3 EXT	6 BIT 3 IX2	5 BIT 2 IX1	4 BIT 1 IX	5 0101
6 0110	10 BRSET3 3 BTB	7 BSET3 2 BSC	4 BNE 2 REL	6 ROR 2 DIR	4 RORA 1 INH	4 RORX 1 INH	7 ROR 2 IX1	6 ROR 1 IX			2 LDA 2 IMM	4 LDA 2 DIR	5 LDA 3 EXT	6 LDA 3 IX2	5 LDA 2 IX1	4 LDA 1 IX	6 0110
7 0111	10 BRCLR3 3 BTB	7 BCLR3 2 BSC	4 BEQ 2 REL	6 ASR 2 DIR	4 ASRA 1 INH	4 ASRX 1 INH	7 ASR 2 IX1	6 ASR 1 IX		2 TAX 1 INH		5 STA 2 DIR	6 STA 3 EXT	7 STA 3 IX2	6 STA 2 IX1	5 STA 1 IX	7 0111
8 1000	10 BRSET4 3 BTB	7 BSET4 2 BSC	4 BHCC 2 REL	6 LSL 2 DIR	4 LSLA 1 INH	4 LSLX 1 INH	7 LSL 2 IX1	6 LSL 1 IX		2 CLC 1 INH	2 EOR 2 IMM	4 EOR 2 DIR	5 EOR 3 EXT	6 EOR 3 IX2	5 EOR 2 IX1	4 EOR 1 IX	8 1000
9 1001	10 BRCLR4 3 BTB	7 BCLR4 2 BSC	4 BHCS 2 REL	6 ROL 2 DIR	4 ROLA 1 INH	4 ROLX 1 INH	7 ROL 2 IX1	6 ROL 1 IX		2 SEC 1 INH	2 ADC 2 IMM	4 ADC 2 DIR	5 ADC 3 EXT	6 ADC 3 IX2	5 ADC 2 IX1	4 ADC 1 IX	9 1001
A 1010	10 BRSET5 3 BTB	7 BSET5 2 BSC	4 BPL 2 REL	6 DEC 2 DIR	4 DECA 1 INH	4 DECX 1 INH	7 DEC 2 IX1	6 DEC 1 IX		2 CLI 1 INH	2 ORA 2 IMM	4 ORA 2 DIR	5 ORA 3 EXT	6 ORA 3 IX2	5 ORA 2 IX1	4 ORA 1 IX	A 1010
B 1011	10 BRCLR5 3 BTB	7 BCLR5 2 BSC	4 BMI 2 REL							2 SEI 1 INH	2 ADD 2 IMM	4 ADD 2 DIR	5 ADD 3 EXT	6 ADD 3 IX2	5 ADD 2 IX1	4 ADD 1 IX	B 1011
C 1100	10 BRSET6 3 BTB	7 BSET6 2 BSC	4 BMC 2 REL	6 INC 2 DIR	4 INCA 1 INH	4 INCX 1 INH	7 INC 2 IX1	6 INC 1 IX		2 RSP 1 INH		3 JMP 2 DIR	4 JMP 3 EXT	5 JMP 3 IX2	4 JMP 2 IX1	3 JMP 1 IX	C 1100
D 1101	10 BRCLR6 3 BTB	7 BCLR6 2 BSC	4 BMS 2 REL	6 TST 2 DIR	4 TSTA 1 INH	4 TSTX 1 INH	7 TST 2 IX1	6 TST 1 IX		2 NOP 1 INH	8 BSR 2 REL	7 JSR 2 DIR	8 JSR 3 EXT	9 JSR 3 IX2	8 JSR 2 IX1	7 JSR 1 IX	D 1101
E 1110	10 BRSET7 3 BTB	7 BSET7 2 BSC	4 BIL 2 REL								2 LDX 2 IMM	4 LDX 2 DIR	5 LDX 3 EXT	6 LDX 3 IX2	5 LDX 2 IX1	4 LDX 1 IX	E 1110
F 1111	10 BRCLR7 3 BTB	7 BCLR7 2 BSC	4 BIH 2 REL	6 CLR 2 DIR	4 CLRA 1 INH	4 CLRX 1 INH	7 CLR 2 IX1	6 CLR 1 IX		2 TXA 1 INH		5 STX 2 DIR	6 STX 3 EXT	7 STX 3 IX2	6 STX 2 IX1	5 STX 1 IX	F 1111

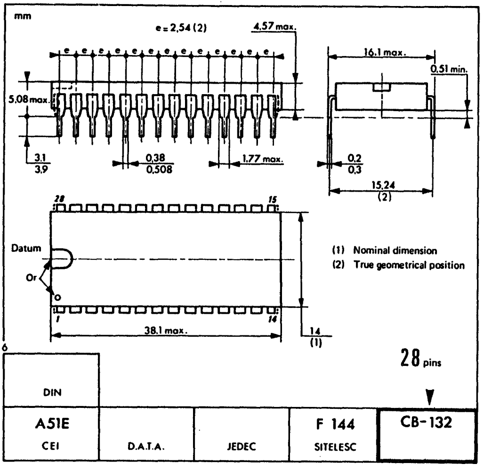
Abbreviations for Address Modes

INH	Inherent
IMM	Immediate
DIR	Direct
EXT	Extended
REL	Relative
BSC	Bit Set/Clear
BTB	Bit Test and Branch
IX	Indexed (No Offset)
IX1	Indexed, 1 Byte (8-Bit) Offset
IX2	Indexed, 2 Byte (16-Bit) Offset

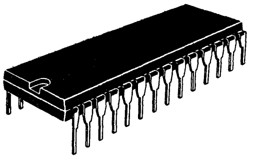
LEGEND



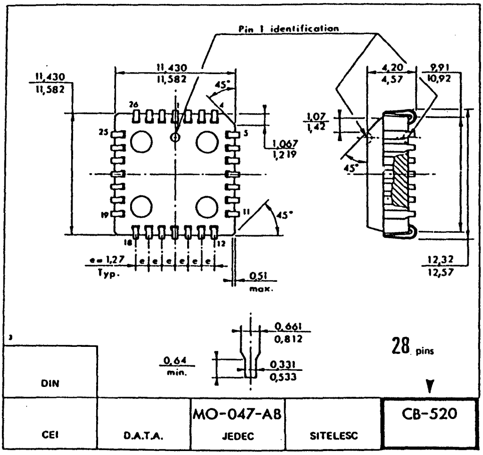
PHYSICAL DIMENSIONS



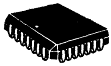
CB-132



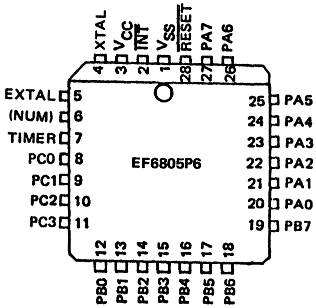
P SUFFIX
PLASTIC PACKAGE



CB-520



FN SUFFIX
PLCC28



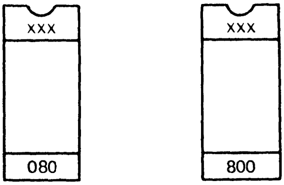
ORDERING INFORMATION

The information required when ordering a custom MCU is listed below. The ROM program may be transmitted to THOMSON SEMICONDUCTEURS on EPROM(s) or an EFDOS/MDOS* disk file.

To initiate a ROM pattern for the MCU, it is necessary to first contact your local THOMSON SEMICONDUCTEURS representative or distributor.

EPROMs

Two 2716 or one 2732 type EPROMs, programmed with the customer program (positive logic sense for address and data), may be submitted for pattern generation. The EPROM must be clearly marked to indicate which EPROM corresponds to which address space. The recommended marking procedure is illustrated below :



XXX = Customer ID

After the EPROM(s) are marked, they should be placed in conductive IC carriers and securely packed. Do not use styrofoam.

VERIFICATION MEDIA

All original pattern media (EPROMs or floppy disk) are filed for contractual purposes and are not returned. A computer listing of the ROM code will be generated and returned along with a listing verification form. The listing should be thoroughly checked and the verification form completed, signed, and returned to THOMSON SEMICONDUCTEURS. The signed verification form constitutes the contractual agreement for creation of the customer mask. If desired, THOMSON SEMICONDUCTEURS will program on blank EPROM from the data file used to create the custom mask and aid in the verification process.

ROM VERIFICATION UNITS (RVUs)

Ten MCUs containing the customer's ROM pattern will be sent for program verification. These units will have been made using the custom mask but are for the purpose of ROM verification only. For expediency they are usually unmarked, packaged in ceramic, and tested only at room temperature and 5 volts. These RVUs are included in the mask charge and are not production parts. The RVUs are thus not guaranteed by THOMSON SEMICONDUCTEURS. Quality Assurance, and should be discarded after verification is completed.

FLEXIBLE DISKS

The disk media submitted must be single-sided, EFDOS/MDOS* compatible floppies.

The customer must write the binary file name and company name on the disk with a felt-tip-pen. The minimum EFDOS/MDOS* system files, as well as the absolute binary object file (Filename .LO type of file) from the 6805 cross assembler, must be on the disk. An object file made from a memory dump using the ROLLOUT command is also acceptable. Consider submitting a source listing as well as the following files : filename .LX (DEVICE/EXORciser loadable format) and filename .SA (ASCII Source Code). These files will of course be kept confidential and are used 1) to speed up the process in-house if any problems arise, and 2) to speed up the user-to-factory interface if the user finds any software errors and needs assistance quickly from THOMSON SEMICONDUCTEURS factory representatives.

EFDOS is THOMSON SEMICONDUCTEURS' Disk Operating System available on development systems such as DEVICE,...

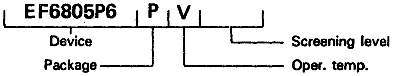
MDOS* is MOTOROLA's Disk Operating System available on development systems such as EXORciser,...

*Requires prior factory approval.

Whenever ordering a custom MCU is required, please contact your local THOMSON SEMICONDUCTEURS representative or THOMSON SEMICONDUCTEURS distributor and/or complete and send the attached "MCU customer ordering sheet" to your local THOMSON SEMICONDUCTEURS representative.

EXORciser is a registered trademark of MOTOROLA Inc.

ORDERING INFORMATION



The table below horizontally shows all available suffix combinations for package, operating temperature and screening level. Other possibilities on request.

DEVICE	PACKAGE					OPER. TEMP			SCREENING LEVEL			
	C		P		FN	L*	V	T	Std	D		
EF6805P6 (1.0 MHz)			●		●	●	●		●	●		
EF6805P6 (1.5 MHz)**			●		●	●			●	●		
EF6805P6 (2.0 MHz)**			●		●	●			●	●		

Examples : EF6805P6P, EF6805P6FN, EF6805P6PLD, EF6805P6FNLD

Package: C: Ceramic DIL, P: Plastic DIL, FN: PLCC.
Oper. temp.: L*: 0°C to + 70°C, V: -40°C to + 85°C, T: -40°C to + 105°C, *: may be omitted.
Screening level: Std: (no-end suffix), D: NFC 96883 level D.
**: Requires prior factory approval.

These specifications are subject to change without notice.
Please inquire with our sales offices about the availability of the different products.

Printed in France

THOMSON COMPONENTS SEMICONDUCTEURS

EF6805R2

8-BIT MICROCOMPUTER WITH A/D

ADVANCE INFORMATION

The EF6805R2 Microcomputer Unit (MCU) is a member of the 6805 Family of low-cost single-chip Microcomputers. The 8-bit microcomputer contains a CPU, on-chip CLOCK, ROM, RAM, I/O, 4-channel 8-bit A/D, and TIMER. It is designed for the user who needs an economical microcomputer with the proven capabilities of the 6800-based instruction set. A comparison of the key features of several members of the 6805 Family of Microcomputers is shown at this end of this data sheet. The following are some of the hardware and software highlights of the EF6805R2 MCU.

HARDWARE FEATURES

- A/D Converter
  8-Bit Conversion, Monotonic
  4 Multiplexed Analog Inputs
  Ratiometric Conversion
- 32 TTL/CMOS Compatible I/O Lines
  24 Bidirectional (8 Lines are LED Compatible)
  8 Input-Only
- 2048 Bytes of User ROM
- 64 Bytes of RAM
- Self-Check Mode.
- Zero-Crossing Detect/Interrupt
- Internal 8-Bit Timer with 7-Bit Mask Programmable Prescaler and Clock Source
- 5 V Single Supply

SOFTWARE FEATURES

- 10 Powerful Addressing Modes
- Byte Efficient Instruction Set with True Bit Manipulation, Bit Test, and Branch Instructions
- Single Instruction Memory Examine/Change
- Powerful Indexed Addressing for Tables
- Full Set of Conditional Branches
- Memory Usable as Register/Flags
- Complete Development System Support on DEVICE®

USER SELECTABLE OPTIONS

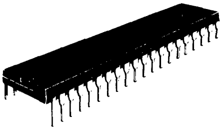
- Internal 8-Bit Timer with Selectable Clock Source (External Timer Input or Internal Machine Clock)
- Timer Prescaler Option (7 Bits, 2^n)
- 8 Bidirectional I/O Lines with TTL or TTL/CMOS Interface Option
- 8 Bidirectional I/O Lines with TTL or Open-Drain Interface Option
- Crystal or Low-Cost Resistor Oscillator Option
- Low Voltage Inhibit Option
- Vectored Interrupts: Timer, Software, and External
- User Callable Self-Check Subroutines

DEVICE® is THOMSON SEMICONDUCTEURS' development/emulation tool.

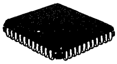
HMOS

CASES

CB-182

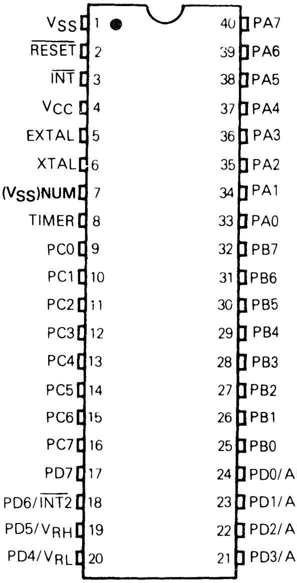


P SUFFIX
PLASTIC PACKAGE
CB-521



FN SUFFIX
PLCC44

PIN ASSIGNMENT



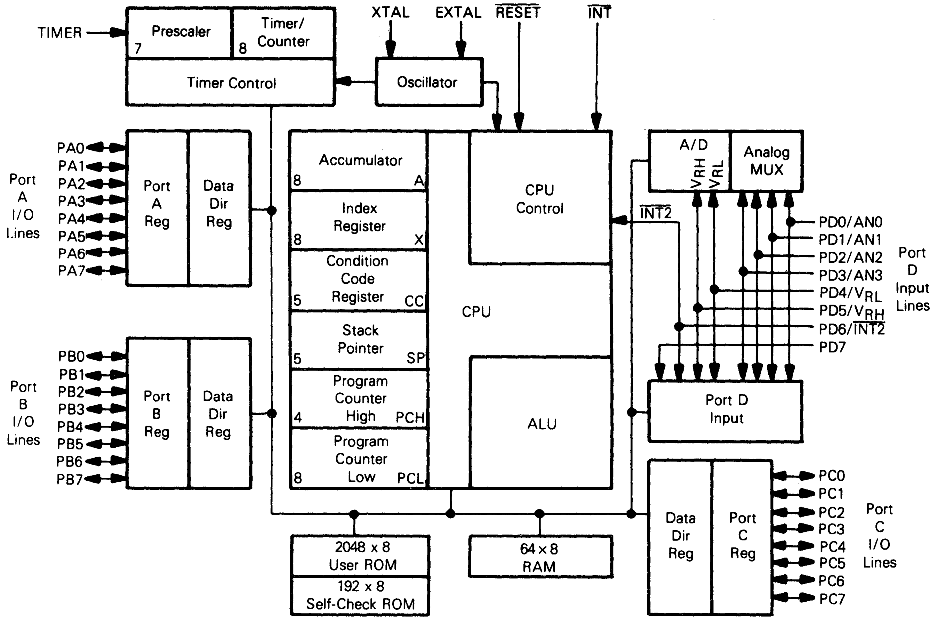


FIGURE 1 – EF6805R2 HMOS MICROCOMPUTER BLOCK DIAGRAM

MAXIMUM RATING

Rating	Symbol	Value	Unit
Supply Voltage	V_{CC}	–0.3 to + 7.0	V
Input Voltage (Except TIMER in Self-Check Mode and Open-Drain Inputs)	V_{in}	–0.3 to + 7.0	V
Input Voltage (Open-Drain Pins, TIMER Pin in Self-Check Mode)	V_{in}	–0.3 to + 15.0	V
Operating Temperature Range V suffix (T_L to T_H) T suffix	T_A	0 to + 70 –40 to + 85 –40 to + 105	°C
Storage Temperature Range	T_{stg}	–55 to + 150	°C
Junction Temperature Plastic Package PLCC	T_J	 150 150	°C

This device contains circuitry to protect the inputs against damage due to high static voltages or electrical fields; however, it is advised that normal precautions be taken to avoid application of any voltage higher than maximum rated voltages to this high-impedance circuit. For proper operation it is recommended that V_{in} and V_{out} be constrained to the range $V_{SS} \leq (V_{in}$ or $V_{out}) \leq V_{CC}$. Reliability of operation is enhanced if unused inputs except EXTAL are tied to an appropriate logic voltage level (e.g., either V_{SS} or V_{CC}).

THERMAL CHARACTERISTICS

Characteristic	Symbol	Value	Unit
Thermal Resistance Plastic PLCC	θ_{JA}	 50 80	°C/W

POWER CONSIDERATIONS

The average chip-junction temperature, T_J, in °C can be obtained from:

$$T_J = T_A + (P_D \bullet \theta_{JA}) \qquad (1)$$

Where:

T_A ■ Ambient Temperature, °C

θ_{JA} ■ Package Thermal Resistance, Junction-to-Ambient, °C/W

P_D ■ $P_{INT} + P_{PORT}$

P_{INT} ■ $I_{CC} \times V_{CC}$, Watts – Chip Internal Power

P_{PORT} ■ Port Power Dissipation, Watts – User Determined

For most applications $P_{PORT} \ll P_{INT}$ and can be neglected. P_{PORT} may become significant if the device is configured to drive Darlington bases or sink LED loads.

An approximate relationship between P_D and T_J (if P_{PORT} is neglected) is:

$$P_D = K \div (T_J + 273°C) \qquad (2)$$

Solving equations 1 and 2 for K gives:

$$K = P_D \bullet (T_A + 273°C) + \theta_{JA} \bullet P_D^2 \qquad (3)$$

Where K is a constant pertaining to the particular part. K can be determined from equation 3 by measuring P_D (at equilibrium) for a known T_A. Using this value of K the values of P_D and T_J can be obtained by solving equations (1) and (2) iteratively for any value of T_A.

ELECTRICAL CHARACTERISTICS (V_{CC} =+ 5.25 Vdc ± 0.5 Vdc, V_{SS} = 0 Vdc, $T_A = T_L$ to T_H Unless Otherwise Noted)

Characteristic	Symbol	Min	Typ	Max	Unit
Input High Voltage $\overline{RESET}$ ($4.75 \leq V_{CC} \leq 5.75$) ($V_{CC} < 4.75$) $\overline{INT}$ ($4.75 \leq V_{CC} \leq 5.75$) ($V_{CC} < 4.75$) All Other (Except Timer)	V_{IH}	 4.0 V_{CC} – 0.5 4.0 V_{CC} – 0.5 2.0	 – – * * –	 V_{CC} V_{CC} V_{CC} V_{CC} V_{CC}	V
Input High Voltage Timer Timer Mode Self-Check Mode	V_{IH}	 2.0 9.0	 – 10.0	 V_{CC} + 1.0 15.0	V
Input Low Voltage $\overline{RESET}$ $\overline{INT}$ All Other (Except A/D Inputs)	V_{IL}	 V_{SS} V_{SS} V_{SS}	 – * –	 0.8 1.5 0.8	V
$\overline{RESET}$ Hysteresis Voltages (See Figures 10,11 and 12) "Out of Reset" "Into Reset"	 V_{IRES+} V_{IRES-}	 2.1 0.8	 – –	 4.0 2.0	V
$\overline{INT}$ Zero Crossing Input Voltage, Through a Capacitor	V_{INT}	2	–	4	$V_{ac\ p\text{-}p}$
Power Dissipation – (No Port Loading, V_{CC} = 5.75 V) T_A = 0°C T_A = –40°C	P_D	 – –	 520 580	 740 800	mW
Input Capacitance EXTAL All Other Except Analog Inputs (See Note)	C_{in}	 – –	 25 10	 – –	pF
Low Voltage Recover	V_{LVR}	–	–	4.75	V
Low Voltage Inhibit	V_{LVI}	2.75	3.75	4.70	V
Input Current TIMER (V_{in} = 0.4 V) $\overline{INT}$ (V_{in} = 2.4 V to V_{CC}) EXTAL (V_{in} = 2.4 V to V_{CC} - Crystal Option) (V_{in} = 0.4 V - Crystal Option) $\overline{RESET}$ (V_{in} = 0.8 V) - External Capacitor Charging Current	I_{in}	 – – – – –4.0	 – 20 – – –	 20 50 10 –1600 –40	µA

NOTE : Port D Analog Inputs, when selected, C_{in} = 25 pF for the first 5 out of 30 cycles.
*Due to internal biasing this input (when unused) floats to approximately 2.2 V.

SWITCHING CHARACTERISTICS (V_{CC} =+ 5.25 Vdc ± 0.5 Vdc, V_{SS} = 0 Vdc, T_A = T_L to T_H Unless Otherwise Noted)

Characteristic	Symbol	Min	Typ	Max	Unit
Oscillator Frequency	f_{osc}	0.4	–	4.2	MHz
Cycle Time (4/f_{osc})	t_{cyc}	0.95	–	10	μs
$\overline{INT}$, $\overline{INT2}$, and TIMER Pulse Width (See Interrupt Section)	t_{WL}, t_{WH}	t_{cyc}+ 250	–	–	ns
$\overline{RESET}$ Pulse Width	t_{RWL}	t_{cyc}+ 250	–	–	ns
$\overline{INT}$ Zero-Crossing Detection Input Frequency	f_{INT}	0.03	–	1	kHz
External Clock Input Duty Cycle (EXTAL)	–	40	50	60	%
Crystal Oscillator Start-Up Time *	–	–	–	100	ms

* See Figure 16 for typical crystal parameters

A/D CONVERTER CHARACTERISTICS (V_{CC} =+ 5.25 Vdc ± 0.5 Vdc, V_{SS} = 0 Vdc, T_A = T_L to T_H Unless Otherwise Noted)

Characteristic	Min	Typ	Max	Unit	Comments
Resolution	8	8	8	Bits	
Non-Linearity	–	–	± 1/2	LSB	For V_{RH} = 4.0 to 5.0 V and V_{RL} = 0 V
Quantizing Error	–	–	± 1/2	LSB	
Conversion Range	V_{RL}	–	V_{RH}	V	
V_{RH}	–	–	V_{CC}	V	A/D accuracy may decrease proportionately as V_{RH} is reduced below 4.0 V. The sum of V_{RH} and V_{RL} must not exceed V_{CC}
V_{RL}	V_{SS}	–	0.2	V	
Conversion Time	30	30	30	t_{cyc}	Includes sampling time
Monotonicity	Inherent (within total error)				
Zero Input Reading	00	00	01	hexadecimal	V_{in} = 0
Ratiometric Reading	FE	FF	FF	hexadecimal	V_{in} = V_{RH}
Sample Time	5	5	5	t_{cyc}	
Sample/Hold Capacitance, Input	–	–	25	pF	
Analog Input Voltage	V_{RL}	–	V_{RH}	V	Negative transients on any analog lines (Pins 19-24) are not allowed at any time during conversion

PORT ELECTRICAL CHARACTERISTICS (V_{CC} =+ 5.25 Vdc ± 0.5 Vdc, V_{SS} = 0 Vdc, T_A = T_L to T_H Unless Otherwise Noted)

Characteristic	Symbol	Min	Typ	Max	Unit
Port A with CMOS Drive Enabled					
Output Low Voltage (I_{Load} = 1.6 mA)	V_{OL}	–	–	0.4	V
Output High Voltage I_{Load} = −100 μA I_{Load} = −10 μA	V_{OH}	 2.4 V_{CC} − 1.0	 – –	 – –	V
Input High Voltage (I_{Load} = −300 μA max.)	V_{IH}	2.0	– –	V_{CC}	V
Input Low Voltage (I_{Load} = −500 μs max.)	V_{IL}	V_{SS}	–	0.8	V
High Z State Input Current (V_{in} = 2.0 V to V_{CC})	I_{IH}	–	–	−300	μA
High Z State Input Current (V_{in} = 0.4 V)	I_{IL}	–	–	−500	μA
Port B					
Output Low Voltage I_{Load} = 3.2 mA I_{Load} = 10 mA (Sink)	V_{OL}	 – –	 – –	 0.4 1.0	V
Output High Voltage I_{Load} = −200 μA	V_{OH}	2.4	–	–	V
Darlington Current Drive (Source) V_O = 1.5 V	I_{OH}	−1.0	–	−10	mA
Input High Voltage	V_{IH}	2.0	–	V_{CC}	V
Input Low Voltage	V_{IL}	V_{SS}	–	0.8	V
High Z State Input Current	I_{TSI}	–	<2	10	μA

PORT ELECTRICAL CHARACTERISTICS (Cont'd)

Port C and Port A with CMOS Drive Disabled					
Output Low Voltage I_{Load}= 1.6 mA	V_{OL}	–	–	0.4	V
Output High Voltage I_{Load}= −100 μA	V_{OH}	2.4	–	–	V
Input High Voltage	V_{IH}	2.0	–	V_{CC}	V
Input Low Voltage	V_{IL}	V_{SS}	–	0.8	V
High Z State Input Current	I_{TSI}	–	<2	10	μs
Port C (Open-Drain Option)					
Input High Voltage	V_{IH}	2.0	–	13.0	V
Input Low Voltage	V_{IL}	V_{SS}	–	0.8	V
Input Leakage Current	I_{LOD}	–	<3	15	μA
Output Low Voltage I_{Load}= 1.6 mA	V_{OL}	–	–	0.4	V
Port D (Digital Inputs Only)					
Input High Voltage	V_{IH}	2.0	–	V_{CC}	V
Input Low Voltage	V_{IL}	V_{SS}	–	0.8	V
Input Current *	I_{in}	–	<1	5	μA

*PD4/V_{RL} –PD5/V_{RH} : The A/D conversion resistor (15 kΩ typical) is connected internally between these two lines, impacting their use as digital inputs in some applications.

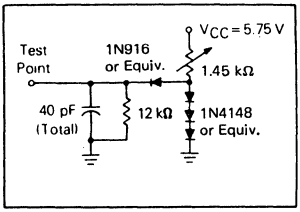


FIGURE 2 – TTL EQUIVALENT TEST LOAD (PORT B)

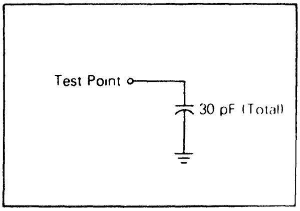


FIGURE 3 – CMOS EQUIVALENT TEST LOAD (PORT A)

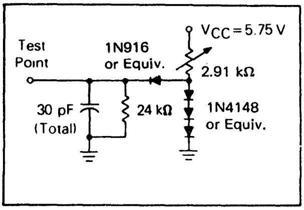


FIGURE 4 – TTL EQUIVALENT TEST LOAD (PORTS A AND C)

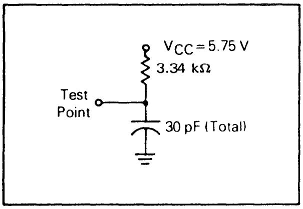


FIGURE 5 – OPEN - DRAIN EQUIVALENT TEST LOAD (PORT C)

SIGNAL DESCRIPTION

The input and output signals for the MCU, shown in Figure 1, are described in the following paragraphs.

V_{CC} AND V_{SS} – Power is supplied to the MCU using these two pins. V_{CC} is power and V_{SS} is the ground connection.

$\overline{INT}$ – This pin provides the capability for asynchronously applying an external interrupt to the MCU. Refer to Interrupts Section for additional information.

NUM (NOM USER MODE) - This pin is not for user application and must be connected to V_{SS}.

XTAL AND EXTAL – These pins provide control input for the on-chip clock oscillator circuit. A crystal, a resistor, or an external signal, depending on user selectable manufacturing mask option, can be connected to these pins to provide a system clock with various degrees of stability/cost tradeoffs. Lead length and stray capacitance on these two pins should be minimized. Refer to Internal Clock Generator Options Section for recommendations about these inputs.

TIMER – The pin allows an external input to be used to control the internal timer circuitry and also to initiate the self test program. Refer to Timer Section for additional information about the timer circuitry.

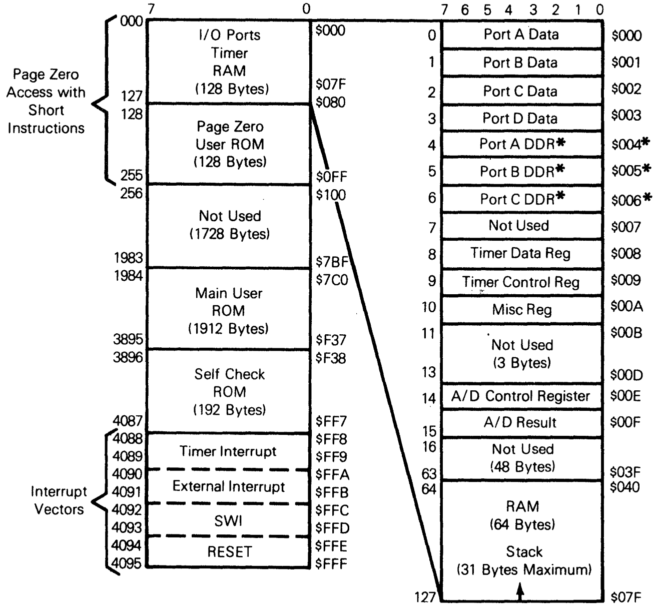
$\overline{\text{RESET}}$ – This pin allows resetting of the MCU at times other than the automatic resetting capability already in the MCU. The MCU can be reset by pulling $\overline{\text{RESET}}$ low. Refer to Resets Section for additional information.

INPUT/OUTPUT LINES (PA0-PA7, PB0-PB7, PC0-PC7, PD0-PD7) – These 32 lines are arranged into four 8-bit ports (A, B, C, and D). Ports A, B, and C are programmable as either inputs or outputs under software control of the data direction registers (DDRs). Port D has up to four analog inputs, plus two voltage reference inputs when the A/D converter is used (PD5/V_{RH}, PD4/V_{RL}), and an $\overline{\text{INT2}}$ input, and from one to eight digital inputs. If any analog input is used, then the voltage reference pins (PD5/V_{RH}, PD4/V_{RL}) must be used in the analog mode. Refer to Input/Output Section, A/D Converter Section, and Interrupts Section for additional information.

MEMORY – The MCU is capable of addressing 4096 bytes of memory and I/O registers with its program counter. The EF6805R2 MCU has implemented 2316 of these bytes. This consists of: 2048 user ROM bytes, 192 self-check ROM bytes, 64 user RAM bytes, 7 port I/O bytes, 2 timer registers. 2 A/D registers, and a miscellaneous register; see Figure 6 for the Address map. The user ROM has been split into three areas. The main user ROM area is from $080 to $OFF and from $7C0 to $F37. The last 8 user ROM locations at the bottom of memory are for the interrupt vectors.

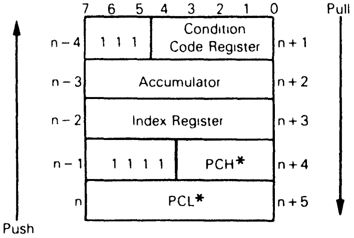
The MCU reserves the first 16 memory locations for I/O features, of which 12 have been implemented. These locations are used for the ports, the port DDRs, the timer, the $\overline{\text{INT2}}$ miscellaneous register, and the A/D. Of the 64 RAM bytes, 31 bytes are shared with the stack area. The stack must be used with care when data shares the stack area.

The shared stack area is used during the processing of an interrupt or subroutine calls to save the contents of the CPU state. The register contents are pushed onto the stack in the order shown in Figure 7. Since the stack pointer decrements during pushes, the low order byte (PCL) of the program counter is stacked first, then the high order four bits (PCH) are stacked. This ensures that the program counter is loaded correctly during pulls from the stack since the stack pointer increments when it pulls data from the stack. A subroutine call results in only the program counter (PCL, PCH) contents being pushed onto the stack; the remaining CPU registers are not pushed.



*Caution: Data direction registers (DDRs) are write-only, they read as $FF.

FIGURE 6 – EF6805R2 MCU ADDRESS MAP



*For subroutine calls, only PCH and PCL are stacked

FIGURE 7 – INTERRUPT STACKING ORDER

CENTRAL PROCESSING UNIT

The CPU of the EF6805 Family is implemented independently from the I/O or memory configuration. Consequently, it can be treated as an independent central processor communicating with I/O and memory via internal address, data, and control buses.

REGISTERS

The 6805 Family CPU has five registers available to the programmer. They are shown in Figure 8 and are explained in the following paragraphs.

ACCUMULATOR (A) – The accumulator is a general purpose 8-bit register used to hold operands and results of arithmetic calculations or data manipulations.

INDEX REGISTER (X) – The index register is an 8-bit register used for the indexed addressing mode. It contains an 8-bit value that may be added to an instruction value to create an effective address. The index register can also be used for data manipulations using the read-modify-write instructions. The Index Register may also be used as a temporary storage area.

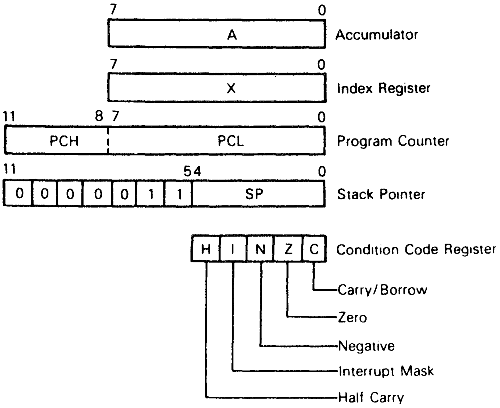


FIGURE 8 – PROGRAMMING MODEL

PROGRAM COUNTER (PC) – The program counter is a 12-bit register that contains the address of the next instruction to be executed.

STACK POINTER (SP) – The stack pointer is a 12-bit register that contains the address of the next free location on the stack. During an MCU reset or the reset stack pointer (RSP) instruction, the stack pointer is set to location $07F. The stack pointer is then decremented as data is pushed onto the stack and incremented as data is then pulled from the stack. The seven most significant bits of the stack pointer are permanently set to 0000011. Subroutines and interrupts may be nested down to location $061 (31 bytes maximum) which allows the programmer to use up to 15 levels of subroutine calls (less if interrupts are allowed).

CONDITION CODE REGISTER (CC) – The condition code register is a 5-bit register in which four bits are used to indicate the results of the instruction just executed. These bits can be individually tested by a program and specific action taken as a result of their state. Each bit is explained in the following paragraphs.

Half Carry (H) – Set during ADD and ADC operations to indicate that a carry occurred between bits 3 and 4.

Interrupt (I) – When this bit is set, the timer and external interrupts ($\overline{INT}$ and $\overline{INT2}$) are masked (disabled). If an interrupt occurs while this bit is set, the interrupt is latched and is processed as soon as the interrupt bit is cleared.

Negative (N) – When set, this bit indicates that the result of the last arithmetic, logical, or data manipulation was negative (bit 7 in the result is a logical "1").

Zero (Z) – When set, this bit indicates that the result of the last arithmetic, logical, or data manipulation was zero.

Carry/Borrow (C) – When set, this bit indicates that a carry or borrow out of the Arithmetic Logic Unit (ALU) occurred during the last arithmetic operation. This bit is also affected during bit test and branch instructions plus shifts and rotates.

TIMER

The timer circuitry for the MC6805R2 is shown in Figure 10. The 8-bit counter may be loaded under program control and is decremented toward zero by the clock input (or prescaler output). When the timer reaches zero, the timer interrupt request bit (bit 7) in the timer control register (TCR) is set. The timer interrupt can be masked (disabled) by setting the timer interrupt mask bit (bit 6) in the TCR. The interrupt bit (I bit) in the condition code register also prevents a timer interrupt from being processed. The MCU responds to this interrupt by saving the present CPU state on the stack, fetching the timer interrupt vector from locations $FF8 and $FF9 and executing the interrupt routine (see **RESET, CLOCK, AND INTERRUPT STRUCTURE SECTIONS**). *The timer interrupt request bit must be cleared by software.* The TIMER and INT2 share the same interrupt vector. *The interrupt routine must check the request bits to determine the source of the interrupt.*

The clock input to the timer can be from an external source (decrementing of timer counter occurs on a positive transition of the external source) applied to the TIMER input pin, or it can be the internal phase two signal. Three machine cycles are required for a change in state of the TIMER pin to decrement the timer prescaler. The maximum frequency of a signal that can be recognized by the TIMER pin logic is dependent on the parameter labeled t_{WH}. The pin logic that recognizes the high state on the pin must also recognize the low state on the pin in order to "re-arm" the internal logic. Therefore, the period can be calculated as follows (assumes 50/50 duty cycle for a given period):

$$t_{cyc} \times 2 + 250 \text{ ns} = \text{period} = \frac{1}{freq}$$

The period is not simply $t_{WL} + t_{WH}$. This computation is allowable, but it does reduce the maximum allowable frequency by defining an unnecessarily longer period (250 nanoseconds times two).

When the phase two signal is used as the source, it can be gated by an input applied to the TIMER input pin allowing the user to easily perform pulse-width measurements. The source of the clock input is one of the mask options that is specified before manufacture of the MCU.

NOTE

For ungated phase two clock input to the timer prescaler, the TIMER pin should be tied to V_{CC}.

A prescaler option, divide by 2^n, can be applied to the clock input that extends the timing interval up to a maximum of 128 counts before decrementing the counter. This prescaling mask option is also specified before manufacture. To avoid truncation errors, the prescaler is cleared when bit 3 of the timer control register is written to a logic one (this bit always reads a logic zero). See Figure 9.

7	6	5	4	3	2	1	0	
		1	1		1	1	1	$009

TCR7 – Timer Interrupt Request Status Bit: Set when TDR goes to zero; must be cleared by software. Cleared to 0 by Reset.

TCR6 – Timer Interrupt Mask Bit: 1 = timer interrupt masked (disabled). Set to 1 by Reset.

TCR3 – Clear prescaler always reads as a 0; clears prescaler when written to a logic "1".

TCR Bits 5, 4, 2, 1, 0 reads "1s" – unused bits.

FIGURE 9 – TIMER CONTROL REGISTER (TCR)

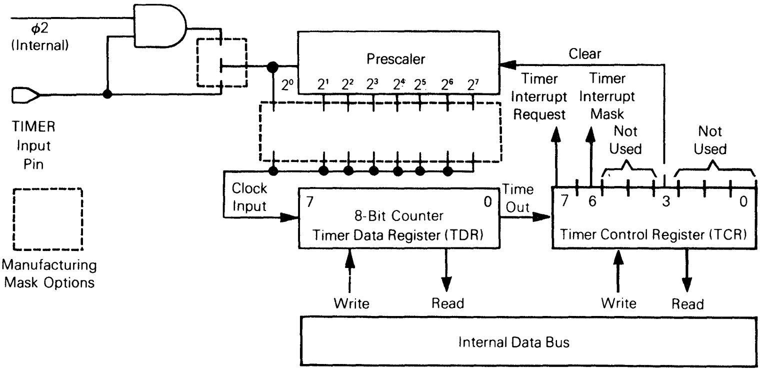


FIGURE 10 – TIMER BLOCK DIAGRAM

The timer continues to count past zero, falling through to $FF from $00 and then continuing the countdown. Thus, the counter can be read at any time by reading the timer data register (TDR). This allows a program to determine the length of time since a timer interrupt has occurred, and not disturb the counting process.

At power up or reset, the prescaler and counter are initialized with all logic ones; the timer interrupt request bit (bit 7) is cleared and the timer interrupt mask bit (bit 6) is set.

SELF-CHECK

The self-check capability of the EF6805R2MCU provides an internal check to determine if the part is functional. Connect the MCU as shown in Figure 11 and monitor the output of Port C bit 3 for an oscillation of approximately 7 Hz. A 10-volt level (through a 10 k resistor) on the timer input, pin 8 and pressing then releasing the RESET button, energizes the ROM-based self-check feature. The self-check program exercises the RAM, ROM, TIMER, A/D, interrupts, and I/O ports.

Several of the self-check subroutines can be called by a user program with a JSR or BSR instruction. They are the RAM, ROM, and 4-channel A/D tests. The timer routine may also be called if the timer input is the internal $\phi2$ clock.

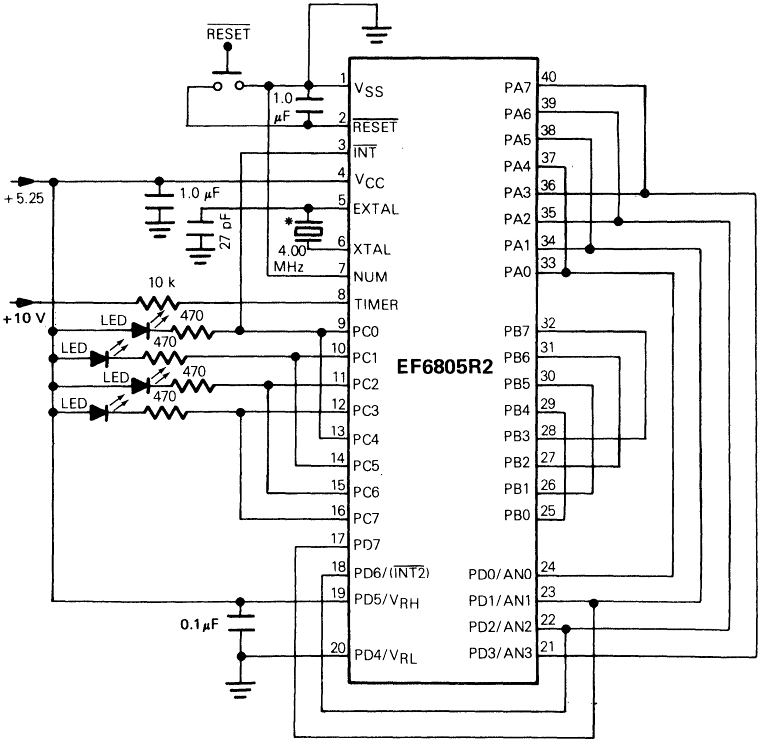
To call those subroutines in customer applications, please contact your local THOMSON SEMICONDUCTEURS sales office in order to obtain the complete description of the self-check program and the entrance/exit conditions.

RAM SELF-CHECK SUBROUTINE – The RAM self-check is called at location $F6F and returns with the Z bit clear if any error is detected ; otherwise the Z bit is set. The walking diagnostic pattern method is used on the EF6805R2.

The RAM test must be called with the stack pointer at $07F. When run, the test checks every RAM cell except for $07F and $07E which are assumed to contain the return address.

The A and X registers and all RAM locations except $07F and $07E are modified.

ROM CHECKSUM SUBROUTINE — The ROM self-check is called at location $F8A. If any error is detected, it returns with the Z bit cleared; otherwise Z = 1, X = 0 on return, and A is zero if the test passes. RAM location $040 to $043 is overwritten. The checksum is the complement of the execution OR of the contents of the user ROM.



*This connection depends on clock oscillator user selectable mask option. Use jumper if the RC mask option is selected.

FIGURE 11 – SELF-CHECK CONNECTIONS

LED Meanings

PC0	PC1	PC2	PC3	Remarks [1:LED ON; 0:LED OFF]
1	0	1	0	Bad I/O
0	0	1	0	Bad Timer
1	1	0	0	Bad RAM
0	1	0	0	Bad ROM
1	0	0	0	Bad A/D
0	0	0	0	Bad Interrupts or Request Flag
Any Flashing				Good Device

Anything else bad Device, Bad Port C, etc.

ANALOG-TO-DIGITAL CONVERTER SELF-CHECK – The A/D self-check is called at location $FA4 and returns with the Z bit cleared if any error was found; otherwise Z = 1.

The A and X register contents are lost. The X register must be set to 4 before the call. On return, X = 8 and A/D channel 7 is selected. The A/D test uses the internal voltage references and confirms port connections.

TIMER SELF-CHECK SUBROUTINE – The timer self-check is called at location $FCF and returns with the Z bit cleared if any error was found; otherwise Z = 1.

In order to work correctly as a user subroutine, the internal $\phi 2$ clock must be the clocking source and interrupts must be disabled. Also, on exit, the clock is running and the interrupt mask is not set so the caller must protect from interrupts if necessary.

The A and X register contents are lost. The timer self-check routine counts how many times the clock counts in 128 cycles. The number of counts should be a power of 2 since the prescaler is a power of 2. If not, the timer is probably not counting correctly. The routine also detects a timer which is not running.

RESET

The MCU can be reset three ways : by initial powerup, by the external reset input ($\overline{\text{RESET}}$) and by an optional internal low-voltage detect circuit. The $\overline{\text{RESET}}$ input consists mainly of a Schmitt trigger which senses the $\overline{\text{RESET}}$ line logic level. A typical reset Schmitt trigger hysteresis curve is shown in Figure 12. The Schmitt trigger provides an internal reset voltage if it senses a logical zero on the $\overline{\text{RESET}}$ pin.

Power-On Reset (POR). – An internal reset is generated upon powerup that allows the internal clock generator to stabilize. A delay of t_{RHL} milliseconds is required before allowing the $\overline{\text{RESET}}$ input to go high. Refer to the power and reset timing diagram of Figure 13. Connecting a capacitor to the $\overline{\text{RESET}}$ input (as illustrated in Figure 14) typically provides sufficient delay. During powerup, the Schmitt trigger switches on (removes reset) when $\overline{\text{RESET}}$ rises to V_{IRES+}.

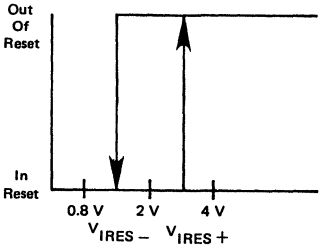


FIGURE 12 – TYPICAL RESET SCHMITT TRIGGER HYSTERESIS

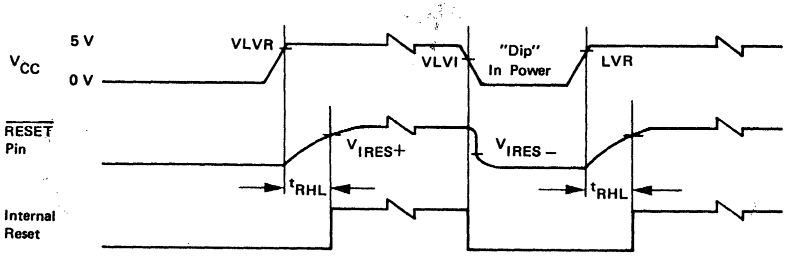


FIGURE 13 – POWER AND RESET TIMING

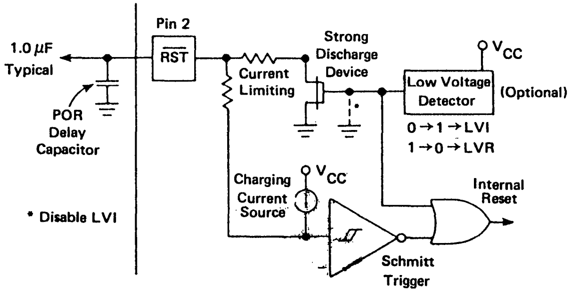


FIGURE 14 – $\overline{\text{RESET}}$ CONFIGURATION

External Reset Input – The MCU will be reset if a logical zero is applied to the $\overline{\text{RESET}}$ input for a period longer than one machine cycle (t_{cyc}). Under this type of reset, the Schmitt trigger switches off at V_{IRES-} to provide an internal reset voltage.

Low-Voltage Inhibit (LVI) – The optional low-voltage detection circuit causes a reset of the MCU if the power supply voltage falls below a certain level (V_{LVI}). The only requirement is that V_{CC} remains at or below the V_{LVI} threshold for one t_{cyc} minimum. In typical applications, the V_{CC} bus filter capacitor will eliminate negative-going voltage glitches of less than one t_{cyc}. The output from the low-voltage detector is connected directly to the internal reset circuitry. It also forces the $\overline{\text{RESET}}$ pin low via a strong discharge device through a resistor. The internal reset will be removed once the power supply voltage rises above a recovery level (V_{LVR}), at which time a normal power-on-reset occurs.

INTERNAL CLOCK GENERATOR OPTIONS

The internal clock generator circuit is designed to require a minimum of external components. A crystal, a resistor, a jumper wire, or an external signal may be used to generate a system clock with various stability/cost tradeoffs. **The oscillator frequency is internally divided by four to produce the internal system clocks. A manufacturing mask option is used to select crystal or resistor operation.**

The different connection methods are shown in Figure 15. Crystal specifications and suggested PC board layouts are given in Figure 16. A resistor selection graph is given in Figure 17.

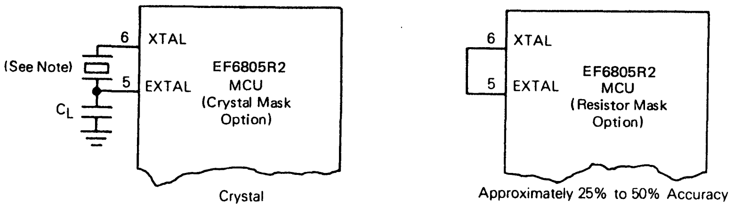
The crystal oscillator start-up time is a function of many variables : crystal parameters (especially R_S), oscillator load capacitances, IC parameters, ambient temperature, and supply voltage. To ensure rapid oscillator start up, neither the crystal characteristics nor the load capacitances should exceed recommendations.

When utilizing the on-board oscillator, the MCU should remain in a reset condition (reset pin voltage below V_{IRES+}) until the oscillator has stabilized at its operating frequency. Several factors are involved in calculating the external reset capacitor required to satisfy this condition: the oscillator start-up voltage, the oscillator stabilization time, the minimum V_{IRES+}, and the reset charging current specification.

Once V_{CC} minimum is reached, the external $\overline{\text{RESET}}$ capacitor will begin to charge at a rate dependent on the capacitor value. The charging current is supplied from V_{CC} through a large resistor, so it appears almost like a constant current source until the reset voltage rises above V_{IRES+}. Therefore, the $\overline{\text{RESET}}$ pin will charge at approximately :

$$(V_{IRES+})\bullet C_{ext} = I_{RES}\bullet t_{RHL}$$

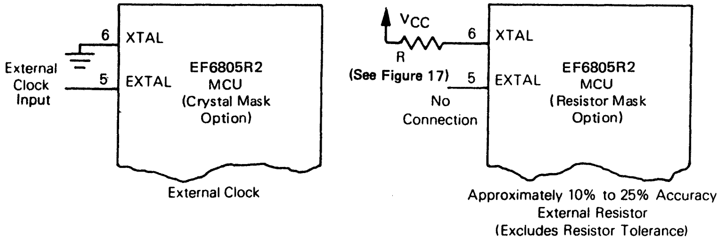
Assuming the external capacitor is initially discharged.



Crystal

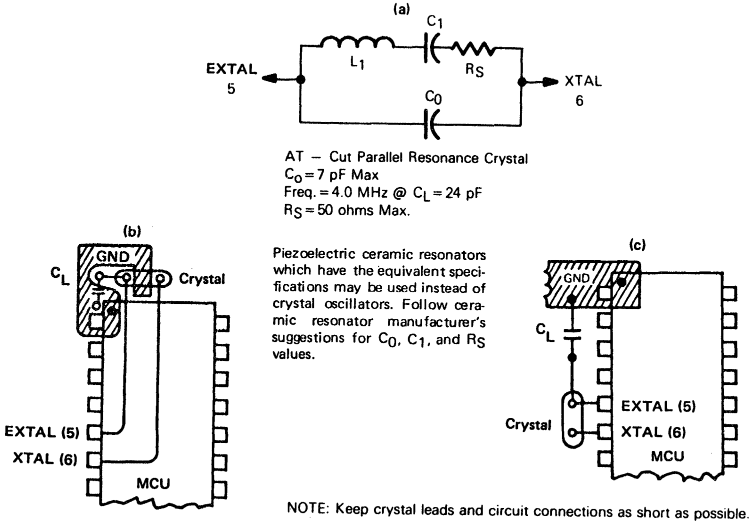
Approximately 25% to 50% Accuracy
Typical t_{cyc} = 1.25 μs
External Jumper

FIGURE 15 – CLOCK GENERATOR OPTIONS



NOTE: The recommended C_L value with a 4.0 MHz crystal is 27 pF, maximum, including system distributed capacitance. There is an internal capacitance of approximately 25 pF on the XTAL pin. For crystal frequencies other than 4 MHz, the total capacitance on each pin should be scaled as the inverse of the frequency ratio. For example, with a 2 MHz crystal, use approximately 50 pF on EXTAL and approximately 25 pF on XTAL. The exact value depends on the Motional-Arm parameters of the crystal used.

FIGURE 15 – CLOCK GENERATOR OPTIONS (Cont'd)



AT – Cut Parallel Resonance Crystal
$C_0 = 7$ pF Max
Freq. = 4.0 MHz @ $C_L = 24$ pF
$R_S = 50$ ohms Max.

Piezoelectric ceramic resonators which have the equivalent specifications may be used instead of crystal oscillators. Follow ceramic resonator manufacturer's suggestions for C_0, C_1, and R_S values.

NOTE: Keep crystal leads and circuit connections as short as possible.

FIGURE 16 – CRYSTAL MOTIONAL ARM PARAMETERS AND SUGGESTED PC BOARD LAYOUT

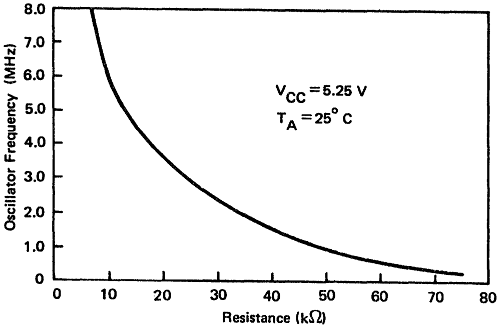


FIGURE 17 – TYPICAL FREQUENCY SELECTION FOR RESISTOR (OSCILLATOR OPTION)

INTERRUPTS

The microcomputers can be interrupted four different ways : through the external interrupt ($\overline{INT}$) input pin, the internal timer interrupt request, the external port D bit 6 ($\overline{INT2}$) input pin, or the software interrupt instruction (SWI). When any interrupt occurs : the current instruction (including SWI) is completed, processing is suspended, the present CPU state is pushed onto the stack, the interrupt bit (I) in the condition code register is set, the address of the interrupt routine is obtained from the appropriate interrupt vector address, and the interrupt routine is executed. Stacking the CPU register, setting the I bit, and vector fetching require a total of 11 t_{cyc} periods for completion. A flowchart of the interrupt sequence is shown in Figure 18. The interrupt service routine must end with a return from interrupt (RTI) instruction which allows the MCU to resume processing of the program prior to the interrupt (by unstacking the previous CPU state). Unlike $\overline{RESET}$, hardware interrupts do not cause the current instruction execution to be halted, but are considered pending until the current instruction execution is complete.

When the current instruction is complete, the processor checks all pending hardware interrupts and if unmasked, proceeds with interrupt processing ; otherwise the next instruction is fetched and executed. Note that masked interrupts are latched for later interrupt service.

If both an external interrupt and a timer interrupt are pending at the end of an instruction execution, the external interrupt is serviced first. The SWI is executed as any other instruction.

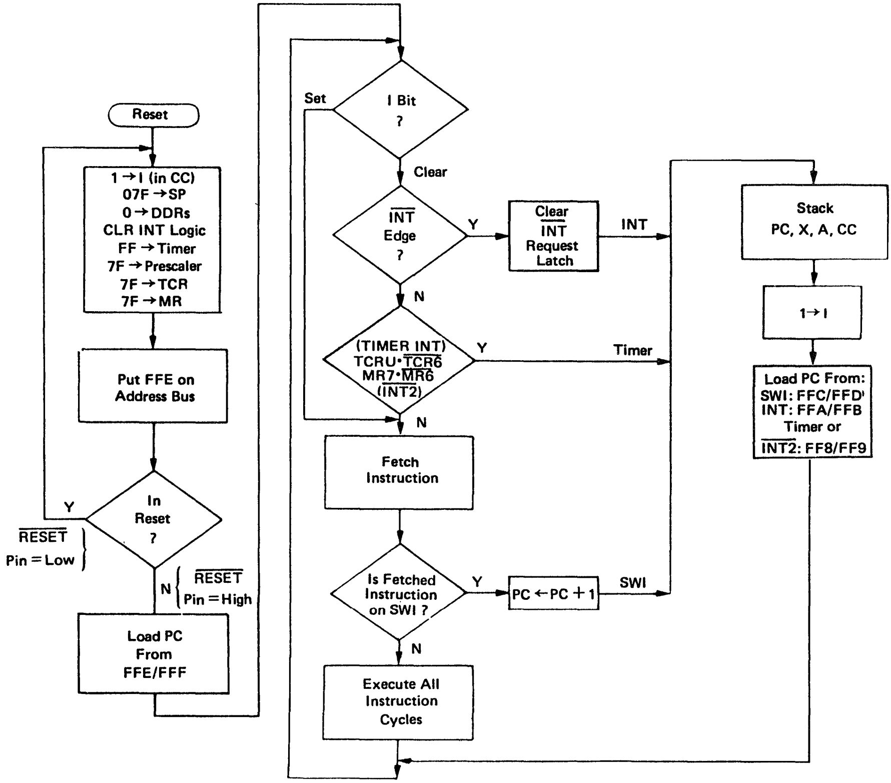


FIGURE 18 – RESET AND INTERRUPT PROCCESSING FLOWCHARD

NOTE

The timer and $\overline{\text{INT2}}$ interrupts share the same vector address. The interrupt routine must determine the source by examining the interrupt request bits (TCR b7 and MR b7). Both TCR b7 and MR b7 can only be written to zero by software.

The external interrupt, $\overline{\text{INT}}$ and $\overline{\text{INT2}}$, are synchronized and then latched on the falling edge of the input signal. The $\overline{\text{INT2}}$ interrupt has an interrupt request bit (bit 7) and a mask bit (bit 6) located in the miscellaneous register (MR). The $\overline{\text{INT2}}$ interrupt is inhibited when the mask bit is set. The $\overline{\text{INT2}}$ is always read as a digital input on port D. The $\overline{\text{INT2}}$ and timer interrupt request bits, if set, cause the MCU to process an interrupt when the condition code I bit is clear.

A sinusoidal input signal (f_{INT} maximum) can be used to generate an external interrupt for use as a zero-crossing detector. This allows applications such as servicing time-of-day routines and engaging/disengaging ac power control devices. Off-chip full wave rectification provides an interrupt at every zero crossing of the ac signal and thereby provides a 2f clock. See Figure 19.

NOTE

The $\overline{\text{INT}}$ (pin 3) is internally biased at approximately 2.2 V due to the internal zero-crossing detection.

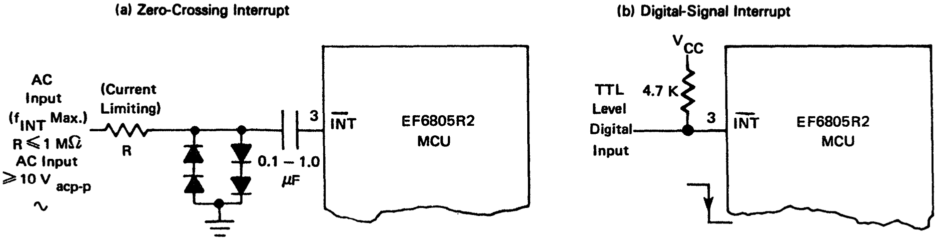
A software interrupt (SWI) is an executable instruction which is executed regardless of the state of the I bit in the condition code register. SWIs are usually used as breakpoints for debugging or as system calls.

A/D CONTROL REGISTER (ACR)

7	6	5	4	3	2	1	0	
	1	1	1	1				$00E

Bit 7 – Conversion Complete Status Flag: Set when conversion is complete; cleared only on a write to ACR.
Readable, not writable.

Bits 2, 1, 0 – A/D input Mux Selection (See Table 2).
Bits 6, 5, 4, 3 read as "1s" – unused bits.


(a) Zero-Crossing Interrupt

(b) Digital-Signal Interrupt

FIGURE 19 – TYPICAL INTERRUPT CIRCUITS

INPUT/OUTPUT CIRCUITRY

There are 32 input/output pins. The $\overline{\text{INT}}$ pin may be polled with branch instructions to provide an additional input pin. All pins on ports A, B, and C are programmable as either inputs or outputs under software control of the corresponding data direction register (DDR). See below I/O port control registers configuration. The port I/O programming is accomplished by writing the corresponding bit in the port DDR to a logic one for output or a logic zero for input. On reset all the DDRs are initialized to a logic zero state, placing the ports in the input mode. The port output registers are not initialized on reset and should be initialized by software before changing the DDRs from input to output. A read operation on a port programmed as an output will read the contents of the output latch regardless of the logic levels at the output pin, due to output loading. Refer to Figure 20.

PORT DATA REGISTER

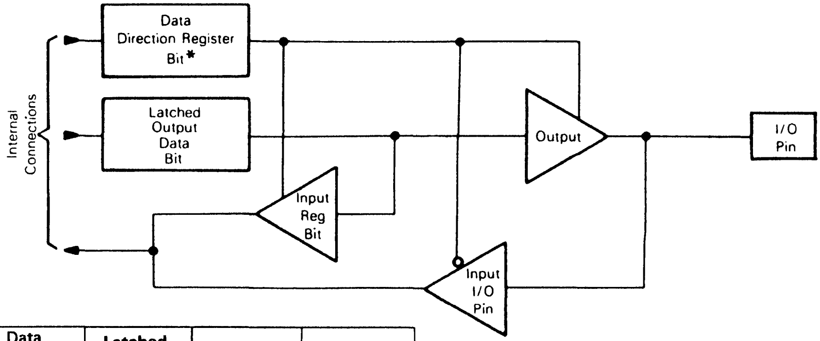
7   0

Port A Addr = $000
Port B Addr = $001
Port C Addr = $002
Port D Addr = $003

PORT DATA DIRECTION REGISTER (DDR)

7   0

(1) Write Only; reads as all "1s"
(2) 1 = Output, 0 = Input Cleared to 0 by Reset
(3) Port A Addr = $004
Port B Addr = $005
Port C Addr = $006



Data Direction Register Bit	Latched Output Data Bit	Output State	Input To MCU
1	0	0	0
1	1	1	1
0	X	High-Z **	Pin

*DDR is a write-only register and reads as all "1s".
** Ports B, and C are three-state ports.
Port A has optional internal pullup devices to provide CMOS data drive capability. See Electrical Characteristics tables for complete information.

FIGURE 20 – TYPICAL PORT I/O CIRCUITRY

All input/output lines are TTL compatible as both inputs and outputs. Port A lines are CMOS compatible as outputs (mask option) while port B, C, and D lines are CMOS compatible as inputs. Port D lines are input only; thus, there is no corresponding DDR. When programmed as outputs, port B is capable of sinking 10 milliamperes and sourcing 1 milliampere on each pin.

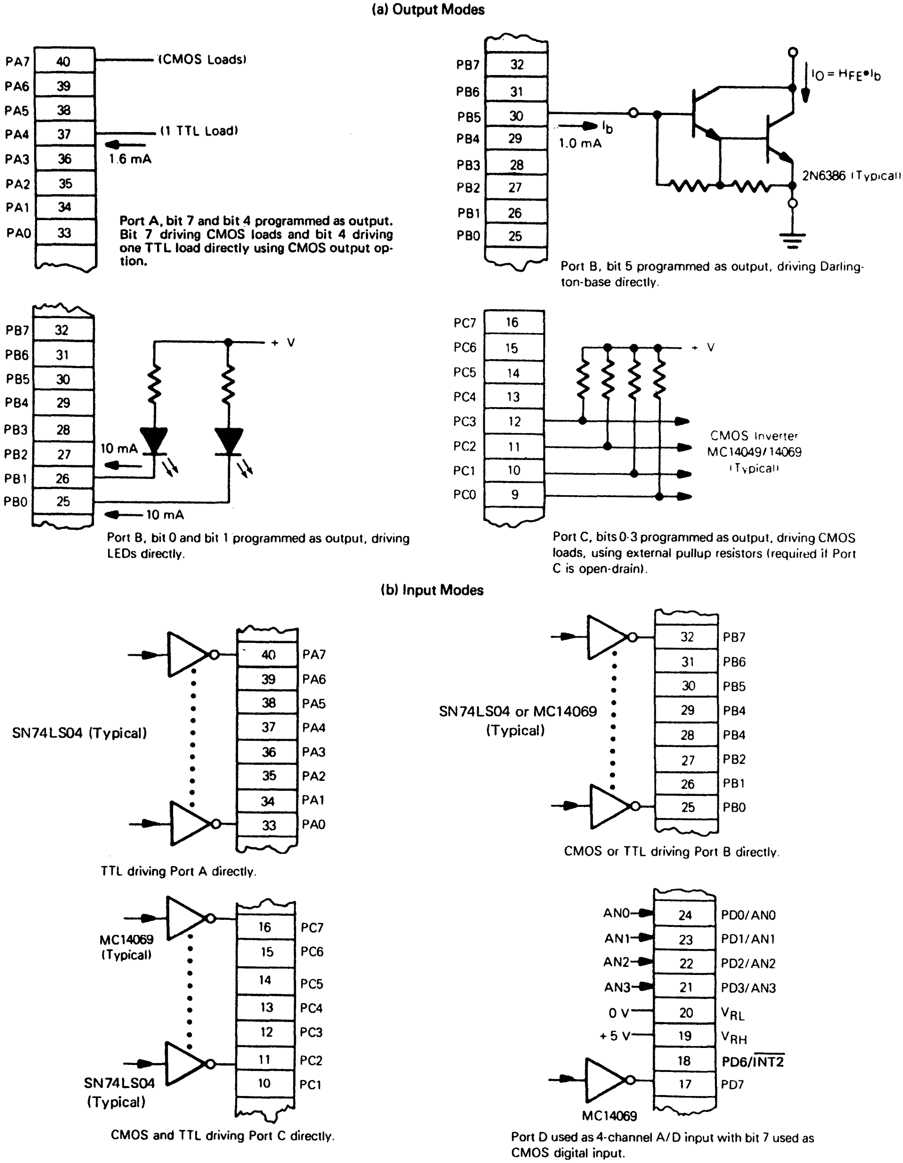
Port D provides the reference voltage, $\overline{\text{INT2}}$, and multiplexed analog inputs. All of these lines are shared with the port D digital inputs. Port D may always be used as digital inputs and may also be used as analog inputs providing V_{RH} and V_{RL} are connected to the appropriate reference voltages. The V_{RL} and V_{RH} lines (PD4 and PD5) are internally connected to the A/D resistor. Analog inputs may be prescaled to attain the V_{RL} and V_{RH} recommended input voltage range.

The address map (Figure 6) gives the addresses of data registers and data direction registers. Figure 21 provides some examples of port connections.

CAUTION

The corresponding DDRs for ports A, B, and C are write-only registers (registers at $004, $005, $006). A read operation on these registers is undefined. Since BSET and BCLR are read-modify-write in function, they cannot be used to set or clear a single DDR bit (all "unaffected" bits would be set). It is recommended that all DDR bits in a port be written using a single-store instruction.

The latched output data bit (see Figure 20) must always be written. Therefore, any write to a port writes all of its data bits even though the port DDR is set to input. This may be used to initialize the data register and avoid undefined outputs ; however, care must be exercised when using read-modify-write instructions, since the data read corresponds to the pin level if the DDR is an input (zero) and corresponds to the latched output data when the DDR is an output (one).


(a) Output Modes

Port A, bit 7 and bit 4 programmed as output. Bit 7 driving CMOS loads and bit 4 driving one TTL load directly using CMOS output option.

Port B, bit 5 programmed as output, driving Darlington-base directly.

Port B, bit 0 and bit 1 programmed as output, driving LEDs directly.

Port C, bits 0-3 programmed as output, driving CMOS loads, using external pullup resistors (required if Port C is open-drain).

(b) Input Modes

TTL driving Port A directly.

CMOS or TTL driving Port B directly.

CMOS and TTL driving Port C directly.

Port D used as 4-channel A/D input with bit 7 used as CMOS digital input.

FIGURE 21 – TYPICAL PORT CONNECTIONS

ANALOG-TO-DIGITAL COUVERTER

The EF6805R2 has an 8-bit analog-to-digital (A/D) converter implemented on the chip using a successive approximation technique, as shown in Figure 22. Up to four external analog inputs, via port D, are connected to the A/D through a multiplexer. Four internal analog channels may be selected for calibration .purposes (V_{RH}-V_{RL}, V_{RH}-V_{RL}/2, V_{RH}-V_{RL}/4, and V_{RL}). The accuracy of these internal channels will not necessarily meet the accuracy specifications of the external channels.

The multiplexer selection is controlled by the A/D control register (ACR) bits 0, 1, and 2 ; see Table 1. This register is cleared during any reset condition.

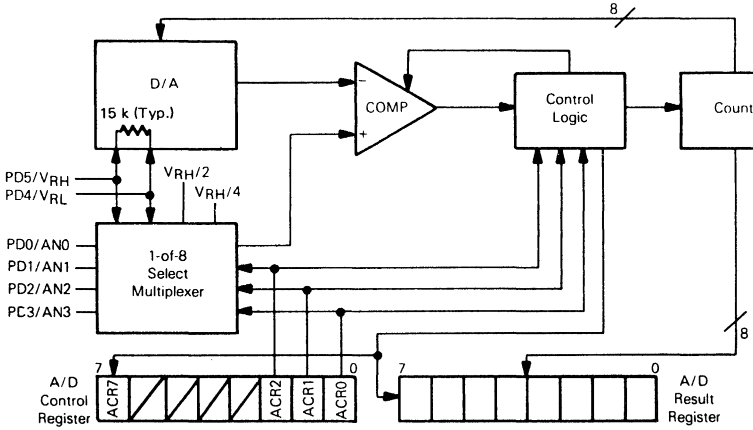


FIGURE 22 – A/D BLOCK DIAGRAM

A/D Control Register			Input Selected	A/D Output (Hex)		
ACR2	ACR1	ACR0		Min	Typ	Max
0	0	0	AN0			
0	0	1	AN1			
0	1	0	AN2			
0	1	1	AN3			
1	0	0	V_{RH}*	FE	FF	FF
1	0	1	V_{RL}*	00	00	01
1	1	0	V_{RH}/4*	3F	40	41
1	1	1	V_{RH}/2*	7F	80	81

*Internal (Calibration) levels

TABLE 1 – A/D INPUT MUX SELECTION

MISCELLANEOUS REGISTER (MR)

7	6						0	
		1	1	1	1	1	1	$00A

MR7 Bit 7 – $\overline{\text{INT2}}$ Interrupt Request Bit: Set when falling edge detected on $\overline{\text{INT2}}$ pin, must be cleared by software. Cleared to 0 by Reset

MR6 Bit 6 – $\overline{\text{INT2}}$ Interrupt Mask Bit: 1 = $\overline{\text{INT2}}$ Interrupt masked (disabled). Set to 1 by Reset.

MR Bits 5, 4, 3, 2, 1, 0 – Read as "1s" – unused bits.

A/D RESULT REGISTER (ARR)

7	0	
MSB	LSB	$00F

Whenever the ACR is written, the conversion in progress is aborted, the conversion complete flag (ACR bit 7) is cleared, and the selected input is sampled for five machine cycles and held internally. During these five cycles, the analog input will appear approximately like a 25 picofarad (maximum) capacitor (plus approximately 10 pF for packaging) charging through a 2.6 kiloohm resistor (typical). Refer to Figure 23.

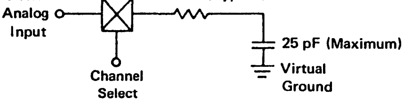


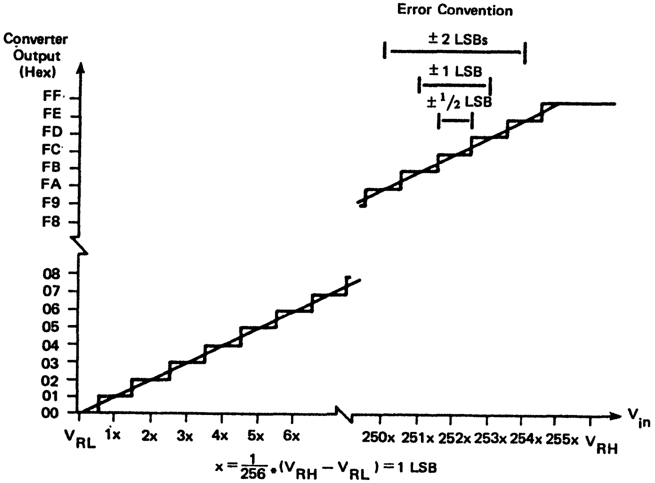
FIGURE 23 – EFFECTIVE ANALOG INPUT IMPEDANCE (DURING SAMPLING ONLY)

The converter operates continuously using 30 machine cycles to complete a conversion of the sampled analog input. When the conversion is complete , the digitized sample of digital value is placed in the A/D result register (ARR), the conversion complete flag is set, the selected input is sampled again, and a new conversion is started.

The A/D is ratiometric. Two reference voltages (V_{RH} and V_{RL}) are supplied to the converter via port D pins. An input voltage equal to V_{RH} converts to $FF (full scale) and an input voltage equal to V_{RL} converts to $00. An input voltage greater than V_{RH} converts to SFF and no overflow indication is provided. Similarly, an input voltage less than V_{RL}, but greater than V_{SS} converts to $00. Maximum and minimum ratings must not be exceeded. For ratiometric conversion, the source of each analog input should use V_{RH} as the supply voltage and be referenced to V_{RL}. To maintain the full accuracy on the A/D, V_{RH} should be equal to or less than V_{DD}, V_{RL} should be equal to or greater than V_{SS} but less than the maximum specification and (V_{RH}-V_{RL}) should be equal to or greater than 4 volts.

The A/D has a built-in 1/2 LSB offset intended to reduce the magnitude of the quantizing error to ± 1/2 LSB, rather than +0, — 1 LSB with no offset. This implies that, ignoring errors, the transition point from $00 to $01 occurs at 1/2 LSB above V_{RL}. Similarly, the transition from $FE to $FF occurs 1 1/2 LSB below V_{RH}, ideally. Refer to Figure 24 and 25.

On release of reset, the A/D control register (ACR) is cleared therefore after reset, channel zero will be selected and the conversion complete flag will be clear.



$x = \frac{1}{256} \cdot (V_{RH} - V_{RL}) = 1$ LSB

FIGURE 24 – IDEAL CONVERTER TRANSFER CHARACTERISTIC

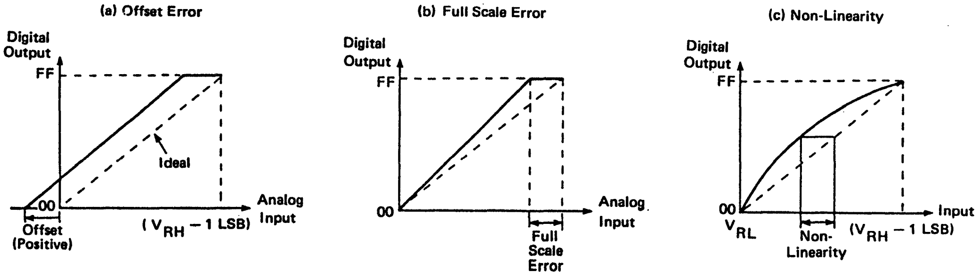


FIGURE 25 – TYPES OF CONVERSION ERRORS

BIT MANIPULATION

The EF6805R2 as the ability to set or clear any single RAM or I/O bit (except the data direction registers) with a single instruction (BSET, BCLR) (see Caution below). Any bit in page zero can be tested using the BRSET and BRCLR instructions and the program branches as a result of its state. The carry bit equals the value of the bit references by BRSET or BRCLR. The capability to working with any bit in RAM, ROM, or I/O allows the user to have individual flags in RAM or to handle single I/O bits as control lines.

CAUTION

The corresponding data direction registers for ports A, B, and C are write-only registers (locations $004, $005, and $006). A read operation on these registers is undefined. Since BSET and BCLR are read-modify-write functions, they cannot be used to set a data direction register bit (all "unaffected" bits would be set). It is recommended that all data direction register bits in a port be written using a single-store instruction.

The coding examples shown in Figure 26 illustrate the usefulness of the bit manipulation and test instruction. Assume that the microcomputer is to communicate with an external serial device. The external device has a data ready signal, a data output line, and a clock line to clock data one bit at a time, least significant bit first out,of the device. The microcomputer waits until the data is ready, clocks the external device, picks up the data in the carry flag, clears the clock line, and finally accumulates the data bit in a random-access memory location.

```
*       BRSET   2,PORTA, *          WAIT FOR READY

        BSET    1, PORTA            CLOCK NEXT BIT IN
        BRCLR   0, PORTA, NEXT      PICKUP BIT IN C-BIT
NEXT    BCLR    1, PORTA            RETURN CLOCK LINE HIGH
        ASR     RAMLOC              MOVE C-BIT INTO RAM
```

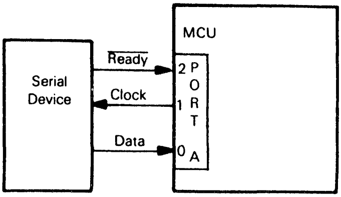


FIGURE 26 - BIT MANIPULATION EXAMPLE

ADDRESSING MODES

The EF6805R2 MCU has ten addressing modes available for use by the programmer. They are explained briefly in the following paragraphs. For additional details and graphical illustrations, refer to the EF6805 Family Users Manual.

The term "effective address" (EA) is used in describing the addressing modes. EA is defined as the address from which the argument for an instruction is fetched or stored.

IMMEDIATE – In the immediate addressing mode, the operand is contained in the byte immediately following the opcode. The immediate addressing mode is used to access constants which do not change during program execution (e.g., a constant used to initialize a loop counter).

DIRECT – In the direct addressing mode, the effective address of the argument is contained in a single byte following the opcode byte. Direct addressing allows the user to directly address the lowest 256 bytes in memory with a single 2-byte instruction. This address area includes all on-chip RAM and I/O registers and 128 bytes of ROM. Direct addressing is an effective use of both memory and time.

EXTENDED – In the extended addressing mode, the effective address of the argument is contained in the two bytes following the opcode. Instructions with extended addressing mode are capable of referencing arguments anywhere in memory with a single 3-byte instruction. When using the assembler, the user need not specify whether an instruction uses direct or extended addressing. The assembler automatically selects the shortest form of the instruction.

RELATIVE – The relative addressing mode is only used in branch instructions. In relative addressing, the contents of the 8-bit signed byte following the opcode (the offset) is added to the PC if, and only if, the branch condition is true. Otherwise, control proceeds to the next instruction. The span of relative addressing is from – 126 to + 129 from the opcode address. The programmer need not worry about calculating the correct offset if he uses the assembler, since it calculates the proper offset and checks to see if it is within the span of the branch.

INDEXED, NO OFFSET – In the indexed, no offset addressing mode, the effective address of the argument is contained in the 8-bit index register. Thus, this addressing mode can access the first 256 memory locations. These instructions are only one byte long. This mode is often used to move a pointer through a table or to hold the address of a frequently referenced RAM or I/O location.

INDEXED, 8-BIT OFFSET – In the indexed, 8-bit offset addressing mode, the effective address is the sum of the contents of the unsigned 8-bit index register and the unsigned byte following the opcode. This addressing mode is useful in selecting the kth element in an n element table. With this 2-byte instruction, k would typically be in X with the address of the beginning of the table in the instruction. As such, tables may begin anywhere within the first 256 addressable locations and could extend as far as location 510 ($1FE is the last location at which the instruction may begin).

INDEXED, 16-BIT OFFSET – In the indexed, 16-bit offset addressing mode, the effective address is the sum of the contents of the unsigned 8-bit index register and the two unsigned bytes following the opcode. This addressing mode can be used in a manner similar to indexed, 8-bit offset except that this 3-byte instruction allows tables to be anywhere in memory. As with direct and extended, the assembler determines the shortest form of indexed addressing.

BIT SET/CLEAR – In the bit set/clear addressing mode, the bit to be set or cleared is part of the opcode, and the byte following the opcode specifies the direct address of the byte in which the specified bit is to be set or cleared. Thus, any read/write bit in the first 256 locations of memory, including I/O, can be selectively set or cleared with a single 2-byte instruction.

CAUTION

The corresponding DDRs for ports A, B, and C are write-only registers (registers at $004, $005, $006). A read operation on these registers is undefined. Since

BSET and BCLR are read-modify-write in function, they cannot be used to set or clear a single DDR bit (all "unaffected" bits would be set). **It is recommended that all DDR bits in a port be must written using a single-store instruction.**

BIT TEST AND BRANCH – The bit test and branch addressing mode is a combination of direct addressing and relative addressing. The bit which is to be tested and condition (set or clear) is included in the opcode, and the address of the byte to be tested is in the single byte immediately following the opcode byte. The signed relative 8-bit offset in the third byte is added to the PC if the specified bit is set or cleared in the specified memory location. This single 3-byte instruction allows the program to branch based on the condition of any readable bit in the first 256 locations of memory. The span of branching is from – 125 to + 130 from the opcode address. The state of the tested bit is also transferred to the carry bit of the condition code registers.

CAUTION

The corresponding DDRs for ports A, B, and C are write-only registers (registers at $004, $005, $006). A read operation on these registers is undefined. Since BSET and BCLR are read-modify-write in function, they cannot be used to set or clear a single DDR bit (all "unaffected" bits would be set). **It is recommended that all DDR bits in a port be must written using a single-store instruction.**

INHERENT – In the inherent addressing mode, all the information necessary to execute the instruction is contained in the opcode. Operations specifying only the index register or accumulator, as well as control instructions with no other arguments, are included in this mode. These instructions are one byte long.

INSTRUCTION SET

The EF6805R2 MCU has a set of 59 basic instructions, which when combined with the 10 addressing modes produce 207 usable opcodes. They can be divided into five different types: register/memory, read-modify-write, branch, bit manipulation, and control. The following paragraphs briefly explain each type. All the instructions within a given type are presented in individual tables.

REGISTER/MEMORY INSTRUCTIONS – Most of these instructions use two operands. One operand is either the accumulator or the index register. The other operand is obtained from memory using one of the addressing modes. The jump unconditional (JMP) and jump to subroutine (JSR) instructions have no register operand. Refer to Table 1.

READ-MODIFY-WRITE INSTRUCTIONS – These instructions read a memory location or a register, modify or test its contents, and write the modified value back to memory or to the register; see Caution under Input/Output section. The test for negative or zero (TST) instruction is included in the read-modify-write instruction though it does not perform the write. Refer to Table 2.

BRANCH INSTRUCTIONS – The branch instructions cause a branch from the program when a certain condition is met. Refer to Table 3.

BIT MANIPULATION INSTRUCTIONS – The instructions are used on any bit in the first 256 bytes of the memory; One group either sets or clears. The other group performs the bit test and branch operations. Refer to Table 4.

CAUTION

The corresponding DDRs for ports A, B, and C are write-only registers (registers at $004, $005, $006). A read operation on these registers is undefined. Since BSET and BCLR are read-modify-write in function, they cannot be used to set or clear a single DDR bit (all "unaffected" bits would be set). **It is recommended that all DDR bits in a port be must written using a single-store instruction.**

CONTROL INSTRUCTION – The control instructions control the MCU operations during program execution. Refer to Table 5.

ALPHABETICAL LISTING – The complete instruction set is given in alphabetical order in Table 6.

OPCODE MAP – Table 8 is an opcode map for the instruction used on the MCU.

TABLE 1 – REGISTER/MEMORY INSTRUCTIONS

		Addressing Modes																	
		Immediate			Direct			Extended			Indexed (No Offset)			Indexed (8-Bit Offset)			Indexed (16-Bit Offset)		
Function	Mnemonic	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles	OP Code	# Bytes	# Cycles
Load A from Memory	LDA	A6	2	2	B6	2	4	C6	3	5	F6	1	4	E6	2	5	D6	3	6
Load X from Memory	LDX	AE	2	2	BE	2	4	CE	3	5	FE	1	4	EE	2	5	DE	3	6
Store A in Memory	STA	—	—	—	B7	2	5	C7	3	6	F7	1	5	E7	2	6	D7	3	7
Store X in Memory	STX	—	—	—	BF	2	5	CF	3	6	FF	1	5	EF	2	6	DF	3	7
Add Memory to A	ADD	AB	2	2	BB	2	4	CB	3	5	FB	1	4	EB	2	5	DB	3	6
Add Memory and Carry to A	ADC	A9	2	2	B9	2	4	C9	3	5	F9	1	4	E9	2	5	D9	3	6
Subtract Memory	SUB	A0	2	2	B0	2	4	C0	3	5	F0	1	4	E0	2	5	D0	3	6
Subtract Memory from A with Borrow	SBC	A2	2	2	B2	2	4	C2	3	5	F2	1	4	E2	2	5	D2	3	6
AND Memory to A	AND	A4	2	2	B4	2	4	C4	3	5	F4	1	4	E4	2	5	D4	3	6
OR Memory with A	ORA	AA	2	2	BA	2	4	CA	3	5	FA	1	4	EA	2	5	DA	3	6
Exclusive OR Memory with A	EOR	A8	2	2	B8	2	4	C8	3	5	F8	1	4	E8	2	5	D8	3	6
Arithmetic Compare A with Memory	CMP	A1	2	2	B1	2	4	C1	3	5	F1	1	4	E1	2	5	D1	3	6
Arithmetic Compare X with Memory	CPX	A3	2	2	B3	2	4	C3	3	5	F3	1	4	E3	2	5	D3	3	6
Bit Test Memory with A (Logical Compare)	BIT	A5	2	2	B5	2	4	C5	3	5	F5	1	4	E5	2	5	D5	3	6
Jump Unconditional	JMP	—	—	—	BC	2	3	CC	3	4	FC	1	3	EC	2	4	DC	3	5
Jump to Subroutine	JSR	—	—	—	BD	2	7	CD	3	8	FD	1	7	ED	2	8	DD	3	9

TABLE 2 – READ-MODIFY-WRITE INSTRUCTIONS

		Addressing Modes														
		Inherent (A)			Inherent (X)			Direct			Indexed (No Offset)			Indexed (8 Bit Offset)		
Function	Mnemonic	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles
Increment	INC	4C	1	4	5C	1	4	3C	2	6	7C	1	6	6C	2	7
Decrement	DEC	4A	1	4	5A	1	4	3A	2	6	7A	1	6	6A	2	7
Clear	CLR	4F	1	4	5F	1	4	3F	2	6	7F	1	6	6F	2	7
Complement	COM	43	1	4	53	1	4	33	2	6	73	1	6	63	2	7
Negate (2's Complement)	NEG	40	1	4	50	1	4	30	2	6	70	1	6	60	2	7
Rotate Left Thru Carry	ROL	49	1	4	59	1	4	39	2	6	79	1	6	69	2	7
Rotate Right Thru Carry	ROR	46	1	4	56	1	4	36	2	6	76	1	6	66	2	7
Logical Shift Left	LSL	48	1	4	58	1	4	38	2	6	78	1	6	68	2	7
Logical Shift Right	LSR	44	1	4	54	1	4	34	2	6	74	1	6	64	2	7
Arithmetic Shift Right	ASR	47	1	4	57	1	4	37	2	6	77	1	6	67	2	7
Test for Negative or Zero	TST	4D	1	4	5D	1	4	3D	2	6	7D	1	6	6D	2	7

TABLE 3 – BRANCH INSTRUCTIONS

		Relative Addressing Mode		
Function	Mnemonic	Op Code	# Bytes	# Cycles
Branch Always	BRA	20	2	4
Branch Never	BRN	21	2	4
Branch IFF Higher	BHI	22	2	4
Branch IFF Lower or Same	BLS	23	2	4
Branch IFF Carry Clear	BCC	24	2	4
(Branch IFF Higher or Same)	(BHS)	24	2	4
Branch IFF Carry Set	BCS	25	2	4
(Branch IFF Lower)	(BLO)	25	2	4
Branch IFF Not Equal	BNE	26	2	4
Branch IFF Equal	BEQ	27	2	4
Branch IFF Half Carry Clear	BHCC	28	2	4
Branch IFF Half Carry Set	BHCS	29	2	4
Branch IFF Plus	BPL	2A	2	4
Branch IFF Minus	BMI	2B	2	4
Branch IFF Interupt Mask Bit is Clear	BMC	2C	2	4
Branch IFF Interrupt Mask Bit is Set	BMS	2D	2	4
Branch IFF Interrupt Line is Low	BIL	2E	2	4
Branch IFF Interrupt Line is High	BIH	2F	2	4
Branch to Subroutine	BSR	AD	2	8

TABLE 4 – BIT MANIPULATION INSTRUCTIONS

		Addressing Modes					
		Bit Set/Clear			Bit Test and Branch		
Function	Mnemonic	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles
Branch IFF Bit n is set	BRSET n (n = 0....7)	—	—	—	2 • n	3	10
Branch IFF Bit n is clear	BRCLR n (n = 0 . .7)	—	—	—	01 + 2 • n	3	10
Set Bit n	BSET n (n = 0. . 7)	10 + 2 • n	2	7	—	—	—
Clear bit n	BCLR n (n = 0....7)	11 + 2 • n	2	7	—	—	—

TABLE 5 – CONTROL INSTRUCTIONS

		Inherent		
Function	Mnemonic	Op Code	# Bytes	# Cycles
Transfer A to X	TAX	97	1	2
Transfer X to A	TXA	9F	1	2
Set Carry Bit	SEC	99	1	2
Clear Carry Bit	CLC	98	1	2
Set Interrupt Mask Bit	SEI	9B	1	2
Clear Interrupt Mask Bit	CLI	9A	1	2
Software Interrupt	SWI	83	1	11
Return from Subroutine	RTS	81	1	6
Return from Interrupt	RTI	80	1	9
Reset Stack Pointer	RSP	9C	1	2
No-Operation	NOP	9D	1	2

TABLE 6 – INSTRUCTION SET

	Addressing Modes										Condition Code				
Mnemonic	Inherent	Immediate	Direct	Extended	Relative	Indexed (No Offset)	Indexed (8 Bits)	Indexed (16 Bits)	Bit Set/ Clear	Bit Test & Branch	H	I	N	Z	C
ADC		X	X	X		X	X	X			Λ	●	Λ	Λ	Λ
ADD		X	X	X		X	X	X			Λ	●	Λ	Λ	Λ
AND		X	X	X		X	X	X			●	●	Λ	Λ	●
ASL	X		X			X	X				●	●	Λ	Λ	Λ
ASR	X		X			X	X				●	●	Λ	Λ	Λ
BCC					X						●	●	●	●	●
BCLR									X		●	●	●	●	●
BCS					X						●	●	●	●	●
BEQ					X						●	●	●	●	●
BHCC					X						●	●	●	●	●
BHCS					X						●	●	●	●	●
BHI					X						●	●	●	●	●
BHS					X						●	●	●	●	●
BIH					X						●	●	●	●	●
BIL					X						●	●	●	●	●
BIT		X	X	X		X	X	X			●	●	Λ	Λ	●
BLO					X						●	●	●	●	●
BLS					X						●	●	●	●	●
BMC					X						●	●	●	●	●
BMI					X						●	●	●	●	●
BMS					X						●	●	●	●	●
BNE					X						●	●	●	●	●
BPL					X						●	●	●	●	●
BRA					X						●	●	●	●	●
BRN					X						●	●	●	●	●
BRCLR										X	●	●	●	●	Λ
BRSET										X	●	●	●	●	Λ
BSET									X		●	●	●	●	●
BSR					X						●	●	●	●	●
CLL	X										●	●	●	●	O
CLI	X										●	O	●	●	●
CLR	X		X			X	X				●	●	O	1	●
CMP		X	X	X		X	X	X			●	●	Λ	Λ	Λ
COM	X		X			X	X				●	●	Λ	Λ	1
CPX		X	X	X		X	X	X			●	●	Λ	Λ	Λ
DEC	X		X			X	X				●	●	Λ	Λ	●
EOR		X	X	X		X	X	X			●	●	Λ	Λ	●
INC	X		X			X	X				●	●	Λ	Λ	●
JMP			X	X		X	X	X			●	●	●	●	●
JSR			X	X		X	X	X			●	●	●	●	●
LDA		X	X	X		X	X	X			●	●	Λ	Λ	●
LDX		X	X	X		X	X	X			●	●	Λ	Λ	●
LSL	X		X			X	X				●	●	Λ	Λ	Λ
LSR	X		X			X	X				●	●	O	Λ	Λ
NEQ	X		X			X	X				●	●	Λ	Λ	Λ
NOP	X										●	●	●	●	●
ORA		X	X	X		X	X	X			●	●	Λ	Λ	●
ROL	X		X			X	X				●	●	Λ	Λ	Λ
RSP	X										●	●	●	●	●

Condition Code Symbols:

H	Half Carry (From Bit 3)	C	Carry/Borrow
I	Interrupt Mask	Λ	Test and Set if True, Cleared Otherwise
N	Negative (Sign Bit)	•	Not Affected
Z	Zero		

TABLE 6 – INSTRUCTION SET (CONTINUED)

	Addressing Modes										Condition Code				
Mnemonic	Inherent	Immediate	Direct	Extended	Relative	Indexed (No Offset)	Indexed (8 Bits)	Indexed (16 Bits)	Bit Set/ Clear	Bit Test & Branch	H	I	N	Z	C
RTI	X										?	?	?	?	?
RTS	X										●	●	●	●	●
SBC		X	X	X		X	X	X			●	●	∧	∧	∧
SEC	X										●	●	●	●	1
SEI	X										●	1	●	●	●
STA			X	X		X	X	X			●	●	∧	∧	●
STX			X	X		X	X	X			●	●	∧	∧	●
SUB		X	X	X		X	X	X			●	●	∧	∧	∧
SWI	X										●	1	●	●	●
TAX	X										●	●	●	●	●
TST	X		X			X	X				●	●	∧	∧	●
TXA	X										●	●	●	●	●

Condition Code Symbols:

H	Half Carry (From Bit 3)	C	Carry/Borrow
I	Interrupt Mask	∧	Test and Set if True, Cleared Otherwise
N	Negative (Sign Bit)	•	Not Affected
Z	Zero	?	Load CC Register From Stack

EF6805 HMOS FAMILY

Features	EF6805CT	EF6805P2	EF6805P4	EF6805P6	EF6805R2	EF6805R3	EF6805T2	EF6805U2	EF6805U3	EF6805TV
Technology	HMOS	HMOS	HMOS	HMOS	HMOS	HMOS	HMOS	HMOS	HMOS	HMOS
Number of Pins	40	28	28	28	40	40	28	40	40	40
On-Chip RAM (Bytes)	240	64	112	64	64	112	64	64	112	96
On-Chip User ROM (Bytes)	4096	1100	1100	1796	2048	3776	2508	2048	3776	6144
External Bus	Yes	None	None	None	None	None	None	None	None	None
Bidirectional I/O Lines	29	20	20	20	24	24	19	24	24	32
Unidirectional I/O Lines	None	None	None	None	6 Inputs	6 Inputs	None	8 Inputs	8 Inputs	None
Other I/O Features	Timer,UART	Timer	Timer	Timer	Timer,A/D	Timer,A/D	Timer, PLL	Timer	Timer	Timer,D/A
External Interrupt Inputs	3	1	1	1	2	2	2	2	2	1
STOP and WAIT	No	No	No	No	No	No	No	No	No	No

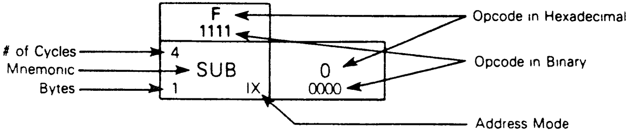
TABLE 7 – 6805 HMOS FAMILY INSTRUCTION SET OPCODE MAP

	Bit Manipulation		Branch	Read-Modify-Write					Control		Register/Memory						
	BTB	BSC	REL	DIR	INH	INH	IX1	IX	INH	INH	IMM	DIR	EXT	IX2	IX1	IX	
Hi / Low	0 0000	1 0001	2 0010	3 0011	4 0100	5 0101	6 0110	7 0111	8 1000	9 1001	A 1010	B 1011	C 1100	D 1101	E 1110	F 1111	Hi / Low
0 0000	10 BRSET0 3 BTB	7 BSET0 2 BSC	4 BRA 2 REL	6 NEG 2 DIR	4 NEG 1 INH	4 NEG 1 INH	7 NEG 2 IX1	6 NEG 1 IX	9 RTI 1 INH		2 SUB 2 IMM	4 SUB 2 DIR	5 SUB 3 EXT	6 SUB 3 IX2	5 SUB 2 IX1	4 SUB 1 IX	0 0000
1 0001	10 BRCLR0 3 BTB	7 BCLR0 2 BSC	4 BRN 2 REL						6 RTS 1 INH		2 CMP 2 IMM	4 CMP 2 DIR	5 CMP 3 EXT	6 CMP 3 IX2	5 CMP 2 IX1	4 CMP 1 IX	1 0001
2 0010	10 BRSET1 3 BTB	7 BSET1 2 BSC	4 BHI 2 REL								2 SBC 2 IMM	4 SBC 2 DIR	5 SBC 3 EXT	6 SBC 3 IX2	5 SBC 2 IX1	4 SBC 1 IX	2 0010
3 0011	10 BRCLR1 3 BTB	7 BCLR1 2 BSC	4 BLS 2 REL	6 COM 2 DIR	4 COMA 1 INH	4 COMX 1 INH	7 COM 2 IX1	6 COM 1 IX	11 SWI 1 INH		2 CPX 2 IMM	4 CPX 2 DIR	5 CPX 3 EXT	6 CPX 3 IX2	5 CPX 2 IX1	4 CPX 1 IX	3 0011
4 0100	10 BRSET2 3 BTB	7 BSET2 2 BSC	4 BCC 2 REL	6 LSR 2 DTR	4 LSRA 1 INH	4 LSRX 1 INH	7 LSR 2 IX1	6 LSR 1 IX			2 AND 2 IMM	4 AND 2 DIR	5 AND 3 EXT	6 AND 3 IX2	5 AND 2 IX1	4 AND 1 IX	4 0100
5 0101	10 BRCLR2 3 BTB	7 BCLR2 2 BSC	4 BCS 2 REL								2 BIT 2 IMM	4 BIT 2 DIR	5 BIT 3 EXT	6 BIT 3 IX2	5 BIT 2 IX1	4 BIT 1 IX	5 0101
6 0110	10 BRSET3 3 BTB	7 BSET3 2 BSC	4 BNE 2 REL	6 ROR 2 DIR	4 RORA 1 INH	4 RORX 1 INH	7 ROR 2 IX1	6 ROR 1 IX			2 LDA 2 IMM	4 LDA 2 DIR	5 LDA 3 EXT	6 LDA 3 IX2	5 LDA 2 IX1	4 LDA 1 IX	6 0110
7 0111	10 BRCLR3 3 BTB	7 BCLR3 2 BSC	4 BEQ 2 REL	6 ASR 2 DIR	4 ASRA 1 INH	4 ASRX 1 INH	7 ASR 2 IX1	6 ASR 1 IX		2 TAX 1 INH		5 STA 2 DIR	6 STA 3 EXT	7 STA 3 IX2	6 STA 2 IX1	5 STA 1 IX	7 0111
8 1000	10 BRSET4 3 BTB	7 BSET4 2 BSC	4 BHCC 2 REL	6 LSL 2 DIR	4 LSLA 1 INH	4 LSLX 1 INH	7 LSL 2 IX1	6 LSL 1 IX		2 CLC 1 INH	2 EOR 2 IMM	4 EOR 2 DIR	5 EOR 3 EXT	6 EOR 3 IX2	5 EOR 2 IX1	4 EOR 1 IX	8 1000
9 1001	10 BRCLR4 3 BTB	7 BCLR4 2 BSC	4 BHCS 2 REL	6 ROL 2 DIR	4 ROLA 1 INH	4 ROLX 1 INH	7 ROL 2 IX1	6 ROL 1 IX		2 SEC 1 INH	2 ADC 2 IMM	4 ADC 2 DIR	5 ADC 3 EXT	6 ADC 3 IX2	5 ADC 2 IX1	4 ADC 1 IX	9 1001
A 1010	10 BRSET5 3 BTB	7 BSET5 2 BSC	4 BPL 2 REL	6 DEC 2 DIR	4 DECA 1 INH	4 DECX 1 INH	7 DEC 2 IX1	6 DEC 1 IX		2 CLI 1 INH	2 ORA 2 IMM	4 ORA 2 DIR	5 ORA 3 EXT	6 ORA 3 IX2	5 ORA 2 IX1	4 ORA 1 IX	A 1010
B 1011	10 BRCLR5 3 BTB	7 BCLR5 2 BSC	4 BMI 2 REL							2 SEI 1 INH	2 ADD 2 IMM	4 ADD 2 DIR	5 ADD 3 EXT	6 ADD 3 IX2	5 ADD 2 IX1	4 ADD 1 IX	B 1011
C 1100	10 BRSET6 3 BT6	7 BSET6 2 BSC	4 BMC 2 REL	6 INC 2 DIR	4 INCA 1 INH	4 INCX 1 INH	7 INC 2 IX1	6 INC 1 IX		2 RSP 1 INH		3 JMP 2 DIR	4 JMP 3 EXT	5 JMP 3 IX2	4 JMP 2 IX1	3 JMP 1 IX	C 1100
D 1101	10 BRCLR6 3 BTB	7 BCLR6 2 BSC	4 BMS 2 REL	6 TST 2 DIR	4 TSTA 1 INH	4 TSTX 1 INH	7 TST 2 IX1	6 TST 1 IX		2 NOP 1 INH	8 BSR 2 REL	7 JSR 2 DIP	8 JSR 3 EXT	9 JSR 3 IX2	8 JSR 2 IX1	7 JSR 1 IX	D 1101
E 1110	10 BRSET7 3 BTB	7 BSET7 2 BSC	4 BIL 2 REL								2 LDX 2 IMM	4 LDX 2 DIR	5 LDX 3 EXT	6 LDX 3 IX2	5 LDX 2 IX1	4 LDX 1 IX	E 1110
F 1111	10 BRCLR7 3 BTB	7 BCLR7 2 BSC	4 BIH 2 REL	6 CLR 2 DIR	4 CLRA 1 INH	4 CLRX 1 INH	7 CLR 2 IX1	6 CLR 1 IX		2 TXA 1 INH		5 STX 2 DIR	6 STX 3 EXT	7 STX 3 IX2	6 STX 2 IX1	5 STX 1 IX	F 1111

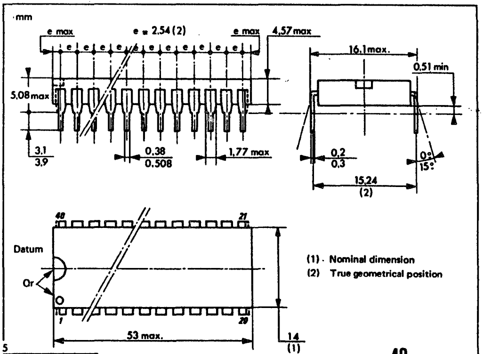
Abbreviations for Address Modes

INH	Inherent
IMM	Immediate
DIR	Direct
EXT	Extended
REL	Relative
BSC	Bit Set/Clear
BTB	Bit Test and Branch
IX	Indexed (No Offset)
IX1	Indexed, 1 Byte (8-Bit) Offset
IX2	Indexed, 2 Byte (16-Bit) Offset

LEGEND

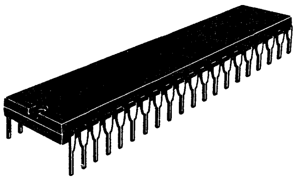


PHYSICAL DIMENSIONS



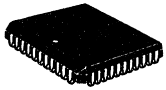
DIN				
A51F CEI	D.A.T.A.	JEDEC	F-119 SITELESC	CB-182

CB-182

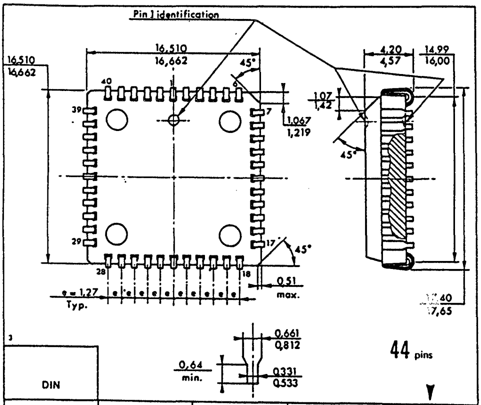


P SUFFIX
PLASTIC PACKAGE

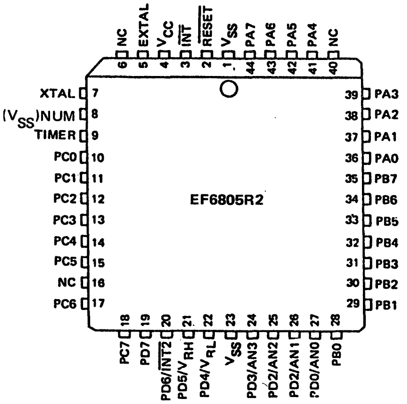
CB-521



FN SUFFIX
PLCC 44



DIN				
CEI	D.A.T.A.	MO-047-AC JEDEC	SITELESC	CB-521



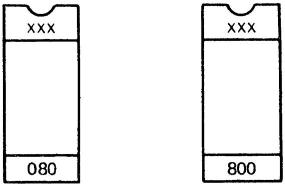
ORDERING INFORMATION

The information required when ordering a custom MCU is listed below. The ROM program may be transmitted to THOMSON SEMICONDUCTEURS on EPROM(s) or an EFDOS/MDOS* disk file.

To initiate a ROM pattern for the MCU, it is necessary to first contact your local THOMSON SEMICONDUCTEURS representative or distributor.

EPROMs

Two 2716 or one 2732 type EPROMs, programmed with the customer program (positive logic sense for address and data), may be submitted for pattern generation. The EPROM must be clearly marked to indicate which EPROM corresponds to which address space. The recommended marking procedure is illustrated below :



XXX = Customer ID

After the EPROM(s) are marked, they should be placed in conductive IC carriers and securely packed. Do not use styrofoam.

VERIFICATION MEDIA

All original pattern media (EPROMs or floppy disk) are filed for contractual purposes and are not returned. A computer listing of the ROM code will be generated and returned along with a listing verification form. The listing should be thoroughly checked and the verification form completed, signed, and returned to THOMSON SEMICONDUCTEURS. The signed verification form constitutes the contractual agreement for creation of the customer mask. If desired, THOMSON SEMICONDUCTEURS will program on blank EPROM from the data file used to create the custom mask and aid in the verification process.

ROM VERIFICATION UNITS (RVUs)

Ten MCUs containing the customer's ROM pattern will be sent for program verification. These units will have been made using the custom mask but are for the purpose of ROM verification only. For expediency they are usually unmarked, packaged in ceramic, and tested only at room temperature and 5 volts. These RVUs are included in the mask charge and are not production parts. The RVUs are thus not guaranteed by THOMSON SEMICONDUCTEURS. Quality Assurance, and should be discarded after verification is completed.

FLEXIBLE DISKS

The disk media submitted must be single-sided, EFDOS/MDOS* compatible floppies.

The customer must write the binary file name and company name on the disk with a felt-tip-pen. The minimum EFDOS/MDOS* system files, as well as the absolute binary object file (Filename .LO type of file) from the 6805 cross assembler, must be on the disk. An object file made from a memory dump using the ROLLOUT command is also acceptable. Consider submitting a source listing as well as the following files : filename .LX (DEVICE/EXORciser loadable format) and filename .SA (ASCII Source Code). These files will of course be kept confidential and are used 1) to speed up the process in-house if any problems arise, and 2) to speed up the user-to-factory interface if the user finds any software errors and needs assistance quickly from THOMSON SEMICONDUCTEURS factory representatives.

EFDOS is THOMSON SEMICONDUCTEURS' Disk Operating System available on development systems such as DEVICE,...

MDOS* is MOTOROLA's Disk Operating System available on development systems such as EXORciser,...

*Requires prior factory approval.

Whenever ordering a custom MCU is required, please contact your local THOMSON SEMICONDUCTEURS representative or THOMSON SEMICONDUCTEURS distributor and/or complete and send the attached "MCU customer ordering sheet" to your local THOMSON SEMICONDUCTEURS representative.

EXORciser is a registered trademark of MOTOROLA Inc.

ORDERING INFORMATION

EF6805R2 P V

Device

Package

Screening level

Oper. temp.

The table below horizontally shows all available suffix combinations for package, operating temperature and screening level. Other possibilities on request.

DEVICE	PACKAGE					OPER. TEMP			SCREENING LEVEL			
	C	J	P	E	FN	L*	V	T	Std	D	G/B	B/B
EF6805R2			●		●	●	●	●	●	●		

Examples : EF6805R2P, EF6805R2FN, EF6805R2PV, EF6805R2FNV

Package : C : Ceramic DIL, **J :** Cerdip DIL, **P :** Plastic DIL, **E :** LCCC, **FN :** PLCC.
Oper. temp. : L* : 0°C to +70°C, **V :** −40°C to +85°C, **T :** −40°C to +105°C, *** :** may be omitted.
Screening level : Std : (no-end suffix), **D :** NFC 96883 level D,
G/B : NFC 96883 level G, **B/B :** NFC 96883 level B and MIL-STD-883C level B.

These specifications are subject to change without notice.
Please inquire with our sales offices about the availability of the different products.

Ref.DSEF6805R2

Printed in France

THOMSON COMPONENTS
SEMICONDUCTEURS

EF6805R3

8 BIT MICROCOMPUTER WITH A/D

ADVANCE INFORMATION

The EF6805R3 Microcomputer Unit (MCU) is a member of the 6805 Family of low-cost single-chip Microcomputers. The 8-bit microcomputer contains a CPU, on-chip CLOCK, ROM, RAM, I/O, 4-channel 8-bit A/D, and TIMER. It is designed for the user who needs an economical microcomputer with the proven capabilities of the 6800-based instruction set. A comparison of the key features of several members of the 6805 Family of Microcomputers is shown at the end of this data sheet. The following are some of the hardware and software highlights of the EF6805R3 MCU.

HARDWARE FEATURES

- A/D Converter
  8-Bit Conversion, Monotonic
  4 Multiplexed Analog Inputs
  Ratiometric Conversion
- 32 TTL/CMOS Compatible I/O Lines
  24 Bidirectional (8 Lines are LED Compatible)
  8 Input-Only
- 3776 Bytes of User ROM
- 112 Bytes of RAM
- Self-Check Mode
- Zero-Crossing Detect/Interrupt
- Internal 8-Bit Timer with 7-Bit Software Programmable Prescaler and Clock Source
- 5 V Single Supply

SOFTWARE FEATURES

- 10 Powerful Addressing Modes
- Byte Efficient Instruction Set with True Bit Manipulation, Bit Test, and Branch Instructions
- Single Instruction Memory Examine/Change
- Powerful Indexed Addressing for Tables
- Full Set of Conditional Branches
- Memory Usable as Register/Flags
- Complete Development System Support on DEVICE®

USER SELECTABLE OPTIONS

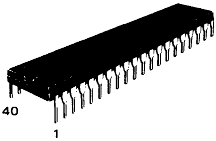
- 8 Bidirectional I/O Lines with TTL or TTL/CMOS Interface Option
- 8 Bidirectional I/O Lines with TTL or Open-Drain Interface Option
- Crystal or Low-Cost Resistor Oscillator Option
- Low Voltage Inhibit Option
- Vectored Interrupts: Timer, Software, and External
- User Callable Self-Check Subroutines

DEVICE® is THOMSON SEMICONDUCTEURS' development/emulation tool.

HMOS

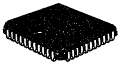
CASES

CB-182



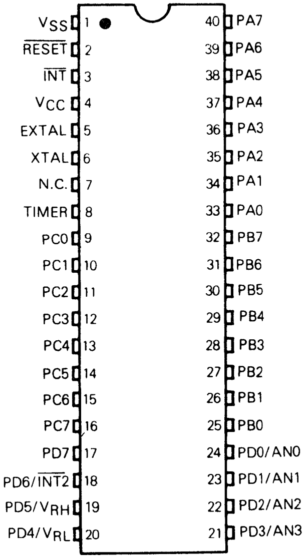
P SUFFIX
PLASTIC PACKAGE

CB-521



FN SUFFIX
PLCC44

PIN ASSIGNMENT



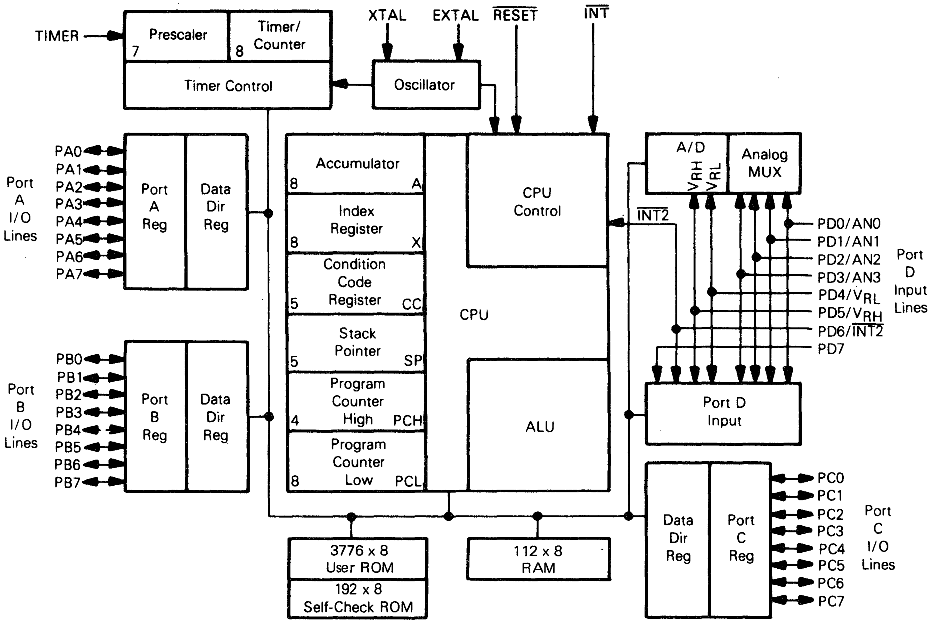


FIGURE 1 – EF6805R3 HMOS MICROCOMPUTER BLOCK DIAGRAM

MAXIMUM RATING

Rating	Symbol	Value	Unit
Supply Voltage	V_{CC}	–0.3 to + 7.0	V
Input Voltage (Except TIMER in Self-Check Mode and Open-Drain Inputs)	V_{in}	–0.3 to + 7.0	V
Input Voltage (Open-Drain Pins, TIMER Pin in Self-Check Mode)	V_{in}	–0.3 to + 15.0	V
Operating Temperature Range (T_L to T_H) V suffix T suffix	T_A	0 to + 70 –40 to + 85 –40 to + 105	°C
Storage Temperature Range	T_{stg}	–55 to + 150	°C
Junction Temperature Plastic Package PLCC	T_J	 150 150	°C

This device contains circuitry to protect the inputs against damage due to high static voltages or electrical fields; however, it is advised that normal precautions be taken to avoid application of any voltage higher than maximum rated voltages to this high-impedance circuit. For proper operation it is recommended that V_{in} and V_{out} be constrained to the range $V_{SS} \leq (V_{in}$ or $V_{out}) \leq V_{CC}$. Reliability of operation is enhanced if unused inputs except EXTAL are tied to an appropriate logic voltage level (e.g., either V_{SS} or V_{CC}).

THERMAL CHARACTERISTICS

Characteristic	Symbol	Value	Unit
Thermal Resistance Plastic PLCC	θ_{JA}	 50 80	°C/W

POWER CONSIDERATIONS

The average chip-junction temperature, T_J, in °C can be obtained from:

$$T_J = T_A + (P_D \bullet \theta_{JA}) \qquad (1)$$

Where:

$T_A \equiv$ Ambient Temperature, °C

$\theta_{JA} \equiv$ Package Thermal Resistance, Junction-to-Ambient, °C/W

$P_D \equiv P_{INT} + P_{PORT}$

$P_{INT} \equiv I_{CC} \times V_{CC}$, Watts – Chip Internal Power

$P_{PORT} \equiv$ Port Power Dissipation, Watts – User Determined

For most applications $P_{PORT} \ll P_{INT}$ and can be neglected. P_{PORT} may become significant if the device is configured to drive Darlington bases or sink LED loads.

An approximate relationship between P_D and T_J (if P_{PORT} is neglected) is:

$$P_D = K \div (T_J + 273°C) \qquad (2)$$

Solving equations 1 and 2 for K gives:

$$K = P_D \bullet (T_A + 273°C) + \theta_{JA} \bullet P_D^2 \qquad (3)$$

Where K is a constant pertaining to the particular part. K can be determined from equation 3 by measuring P_D (at equilibrium) for a known T_A. Using this value of K the values of P_D and T_J can be obtained by solving equations (1) and (2) iteratively for any value of T_A.

ELECTRICAL CHARACTERISTICS (V_{CC} =+ 5.25 Vdc ± 0.5 Vdc, V_{SS} = 0 Vdc, $T_A = T_L$ to T_H Unless Otherwise Noted)

Characteristic	Symbol	Min	Typ	Max	Unit
Input High Voltage	V_{IH}				V
RESET (4.75 ⩽ V_{CC} ⩽ 5.75)		4.0	–	V_{CC}	
(V_{CC} < 4.75)		V_{CC} – 0.5	–	V_{CC}	
INT (4.75 ⩽ V_{CC} ⩽ 5.75)		4.0	*	V_{CC}	
(V_{CC} < 4.75)		V_{CC} – 0.5	*	V_{CC}	
All Other (Except Timer)		2.0	–	V_{CC}	
Input High Voltage Timer	V_{IH}				V
Timer Mode		2.0	–	V_{CC} + 1.0	
Self-Check Mode		9.0	10.0	15.0	
Input Low Voltage	V_{IL}				V
RESET		V_{SS}	–	0.8	
INT		V_{SS}	*	1.5	
All Other (Except A/D Inputs)		V_{SS}	–	0.8	
RESET Hysteresis Voltages (See Figures 10,11 and 12)					V
"Out of Reset"	V_{IRES+}	2.1	–	4.0	
"Into Reset"	V_{IRES-}	0.8	–	2.0	
INT Zero Crossing Input Voltage, Through a Capacitor	V_{INT}	2	–	4	$V_{ac\ p\text{-}p}$
Power Dissipation – (No Port Loading, V_{CC} = 5.75 V)	P_D				mW
T_A = 0°C		–	520	740	
T_A = –40°C		–	580	800	
Input Capacitance	C_{in}				pF
EXTAL		–	25	–	
All Other Except Analog Inputs (See Note)		–	10	–	
Low Voltage Recover	V_{LVR}	–	–	4.75	V
Low Voltage Inhibit	V_{LVI}	2.75	3.75	4.70	V
Input Current	I_{in}				µA
TIMER (V_{in} = 0.4 V)		–	–	20	
INT (V_{in} = 2.4 V to V_{CC})		–	20	50	
EXTAL (V_{in} = 2.4 V to V_{CC} - Crystal Option)		–	–	10	
(V_{in} = 0.4 V - Crystal Option)		–	–	–1600	
RESET (V_{in} = 0.8 V) - External Capacitor Charging Current		– –4.0	– – –	–40	

NOTE : Port D Analog Inputs, when selected, C_{in} = 25 pF for the first 5 out of 30 cycles.

*Due to internal biasing this input (when unused) floats to approximately 2.2 V.

SWITCHING CHARACTERISTICS (V_{CC} =+ 5.25 Vdc ± 0.5 Vdc, V_{SS} = 0 Vdc, $T_A = T_L$ to T_H Unless Otherwise Noted)

Characteristic	Symbol	Min	Typ	Max	Unit
Oscillator Frequency	f_{osc}	0.4	–	4.2	MHz
Cycle Time ($4/f_{osc}$)	t_{cyc}	0.95	–	10	μs
$\overline{INT}$, $\overline{INT2}$, and TIMER Pulse Width (See Interrupt Section)	t_{WL}, t_{WH}	t_{cyc}+ 250	–	–	ns
$\overline{RESET}$ Pulse Width	t_{RWL}	t_{cyc}+ 250	–	–	ns
$\overline{INT}$ Zero-Crossing Detection Input Frequency	f_{INT}	0.03	–	1	kHz
External Clock Input Duty Cycle (EXTAL)	–	40	50	60	%
Crystal Oscillator Start-Up Time *	–	–	–	100	ms

* See Figure 16 for typical crystal parameters

A/D CONVERTER CHARACTERISTICS (V_{CC} =+ 5.25 Vdc ± 0.5 Vdc, V_{SS} = 0 Vdc, $T_A = T_L$ to T_H Unless Otherwise Noted)

Characteristic	Min	Typ	Max	Unit	Comments
Resolution	8	8	8	Bits	
Non-Linearity	–	–	± 1/2	LSB	For V_{RH} = 4.0 to 5.0 V and V_{RL} = 0 V
Quantizing Error	–	–	± 1/2	LSB	
Conversion Range	V_{RL}	–	V_{RH}	V	
V_{RH}	–	–	V_{CC}	V	A/D accuracy may decrease proportionately as V_{RH} is reduced below 4.0 V. The sum of V_{RH} and V_{RL} must not exceed V_{CC}
V_{RL}	V_{SS}	–	0.2	V	
Conversion Time	30	30	30	t_{cyc}	Includes sampling time
Monotonicity	Inherent (within total error)				
Zero Input Reading	00	00	01	hexadecimal	V_{in} = 0
Ratiometric Reading	FE	FF	FF	hexadecimal	V_{in} = V_{RH}
Sample Time	5	5	5	t_{cyc}	
Sample/Hold Capacitance, Input	–	–	25	pF	
Analog Input Voltage	V_{RL}	–	V_{RH}	V	Negative transients on any analog lines (Pins 19-24) are not allowed at any time during conversion

PORT ELECTRICAL CHARACTERISTICS (V_{CC} =+ 5.25 Vdc ± 0.5 Vdc, V_{SS} = 0 Vdc, $T_A = T_L$ to T_H Unless Otherwise Noted)

Characteristic	Symbol	Min	Typ	Max	Unit
Port A with CMOS Drive Enabled					
Output Low Voltage (I_{Load} = 1.6 mA)	V_{OL}	–	–	0.4	V
Output High Voltage I_{Load} = −100 μA I_{Load} = −10 μA	V_{OH}	 2.4 V_{CC} − 1.0	 – –	 – –	V
Input High Voltage (I_{Load} = −300 μA max.)	V_{IH}	2.0	–	V_{CC}	V
Input Low Voltage (I_{Load} = −500 μs max.)	V_{IL}	V_{SS}	–	0.8	V
High Z State Input Current (V_{in} = 2.0 V to V_{CC})	I_{IH}	–	–	−300	μA
High Z State Input Current (V_{in} = 0.4 V)	I_{IL}	–	–	−500	μA
Port B					
Output Low Voltage I_{Load} = 3.2 mA I_{Load} = 10 mA (Sink)	V_{OL}	 – –	 – –	 0.4 1.0	V
Output High Voltage I_{Load} = −200 μA	V_{OH}	2.4	–	–	V
Darlington Current Drive (Source) V_O = 1.5 V	I_{OH}	−1.0	–	−10	mA
Input High Voltage	V_{IH}	2.0	–	V_{CC}	V
Input Low Voltage	V_{IL}	V_{SS}	–	0.8	V
High Z State Input Current	I_{TSI}	–	<2	10	μA

PORT ELECTRICAL CHARACTERISTICS (Cont'd)

Port C and Port A with CMOS Drive Disabled					
Output Low Voltage I_{Load}= 1.6 mA	V_{OL}	–	–	0.4	V
Output High Voltage I_{Load}= –100 µA	V_{OH}	2.4	–	–	V
Input High Voltage	V_{IH}	2.0	–	V_{CC}	V
Input Low Voltage	V_{IL}	V_{SS}	–	0.8	V
High Z State Input Current	I_{TSI}	–	<2	10	µs
Port C (Open-Drain Option)					
Input High Voltage	V_{IH}	2.0	–	13.0	V
Input Low Voltage	V_{IL}	V_{SS}	–	0.8	V
Input Leakage Current	I_{LOD}	–	<3	15	µA
Output Low Voltage I_{Load}= 1.6 mA	V_{OL}	–	–	0.4	V
Port D (Digital Inputs Only)					
Input High Voltage	V_{IH}	2.0	–	V_{CC}	V
Input Low Voltage	V_{IL}	V_{SS}	–	0.8	V
Input Current *	I_{in}	–	<1	5	µA

*PD4/V_{RL} –PD5/V_{RH} : The A/D conversion resistor (15 kΩ typical) is connected internally between these two lines, impacting their use as digital inputs in some applications.

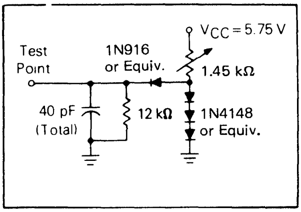


FIGURE 2 – TTL EQUIVALENT TEST LOAD (PORT B)

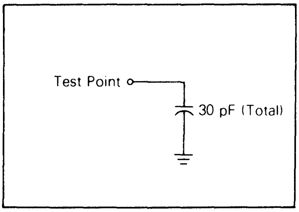


FIGURE 3 – CMOS EQUIVALENT TEST LOAD (PORT A)

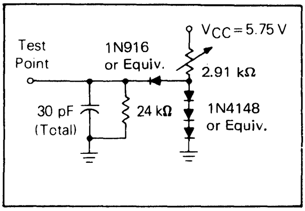


FIGURE 4 – TTL EQUIVALENT TEST LOAD (PORTS A AND C)

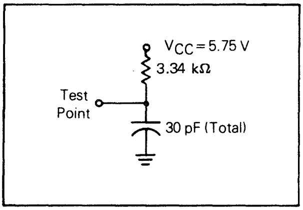


FIGURE 5 – OPEN - DRAIN EQUIVALENT TEST LOAD (PORT C)

SIGNAL DESCRIPTION

The input and output signals for the MCU, shown in Figure 1, are described in the following paragraphs.

V_{CC} AND V_{SS} – Power is supplied to the MCU using these two pins. V_{CC} is power and V_{SS} is the ground connection.

$\overline{INT}$ – This pin provides the capability for asynchronously applying an external interrupt to the MCU. Refer to Interrupts Section for additional information.

XTAL AND EXTAL – These pins provide control input for the on-chip clock oscillator circuit. A crystal, a resistor, or an external signal, depending on user selectable manufacturing mask option, can be connected to these pins to provide a system clock with various degrees of stability/cost tradeoffs. Lead length and stray capacitance on these two pins should be minimized. Refer to Internal Clock Generator Options Section for recommendations about these inputs.

NOTE: Pin 7 in DIL package/Pin 8 in PLCC package is connected to internal protection.

TIMER – The pin allows an external input to be used to control the internal timer circuitry and also to initiate the self test program. Refer to Timer Section for additional information about the timer circuitry.

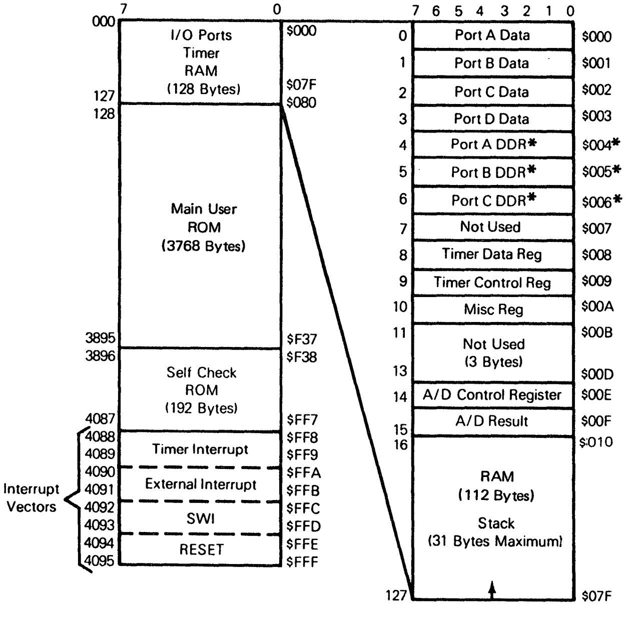
$\overline{\text{RESET}}$ – This pin allows resetting of the MCU at times other than the automatic resetting capability already in the MCU. The MCU can be reset by pulling $\overline{\text{RESET}}$ low. Refer to Resets Section for additional information.

INPUT/OUTPUT LINES (PA0-PA7, PB0-PB7, PC0-PC7, PD0-PD7) – These 32 lines are arranged into four 8-bit ports (A, B, C, and D). Ports A, B, and C are programmable as either inputs or outputs under software control of the data direction registers (DDRs). Port D has up to four analog inputs, plus two voltage reference inputs when the A/D converter is used (PD5/V_{RH}, PD4/V_{RL}), and an $\overline{\text{INT2}}$ input, and from one to eight digital inputs. If any analog input is used, then the voltage reference pins (PD5/V_{RH}, PD4/V_{RL}) must be used in the analog mode. The two analog reference inputs are tied together internally with a resistor, therefore, if they are both used as digital inputs problems may occur. Refer to Input/Output Section, A/D Converter Section, and Interrupts Section for additional information.

MEMORY – The MCU is capable of addressing 4096 bytes of memory and I/O registers with its program counter. The EF6805R3 MCU has implemented 4092 of these bytes. This consists of : 3776 user ROM bytes, 192 self-check ROM bytes, 112 user RAM bytes, 7 port I/O bytes, 2 timer registers, 2 A/D registers, and a miscellaneous register ; see Figure 6 for the Address map. The user ROM has been split into two areas. The main user ROM area is from $080 to $F37. The last 8 user ROM locations at the bottom of memory are for the interrupt vectors.

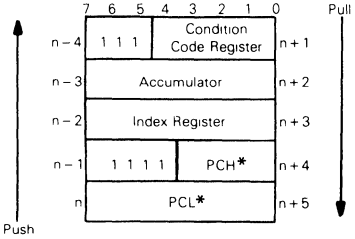
The MCU reserves the first 16 memory locations for I/O' features, of which 12 have been implemented. These locations are used for the ports, the port DDRs, the timer, the $\overline{\text{INT2}}$ miscellaneous register, and the A/D. Of the 112 RAM bytes, 31 bytes are shared with the stack area. The stack must be used with care when data shares the stack area.

The shared stack area is used during the processing of an interrupt or subroutine calls to save the contents of the CPU state. The register contents are pushed onto the stack in the order shown in Figure 7. Since the stack pointer decrements during pushes, the low order byte (PCL) of the program counter is stacked first, then the high order four bits (PCH) are stacked. This ensures that the program counter is loaded correctly during pulls from the stack since the stack pointer increments when it pulls data from the stack. A subroutine call results in only the program counter (PCL, PCH) contents being pushed onto the stack; the remaining CPU registers are not pushed.



*Caution: Data direction registers (DDRs) are write-only; they read as $FF.

FIGURE 6 – EF6805R3 MCU ADDRESS MAP



*For subroutine calls, only PCH and PCL are stacked

FIGURE 7 – INTERRUPT STACKING ORDER

CENTRAL PROCESSING UNIT

The CPU of the EF6805 Family is implemented independently from the I/O or memory configuration. Consequently, it can be treated as an independent central processor communicating with I/O and memory via internal address, data, and control buses.

REGISTERS

The 6805 Family CPU has five registers available to the programmer. They are shown in Figure 8and are explained in the following paragraphs.

ACCUMULATOR (A) – The accumulator is a general purpose 8-bit register used to hold operands and results of arithmetic calculations or data manipulations.

INDEX REGISTER (X) – The index register is an 8-bit register used for the indexed addressing mode. It contains an 8-bit value that may be added to an instruction value to create an effective address. The index register can also be used for data manipulations using the read-modify-write instructions. The Index Register may also be used as a temporary storage area.

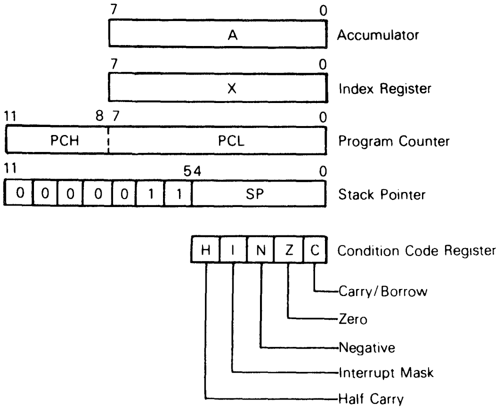


FIGURE 8 – PROGRAMMING MODEL

PROGRAM COUNTER (PC) – The program counter is a 12-bit register that contains the address of the next instruction to be executed.

STACK POINTER (SP) – The stack pointer is a 12-bit register that contains the address of the next free location on the stack. During an MCU reset or the reset stack pointer (RSP) instruction, the stack pointer is set to location $07F. The stack pointer is then decremented as data is pushed onto the stack and incremented as data is then pulled from the stack. The seven most significant bits of the stack pointer are permanently set to 0000011. Subroutines and interrupts may be nested down to location $061 (31 bytes maximum) which allows the programmer to use up to 15 levels of subroutine calls (less if interrupts are allowed).

CONDITION CODE REGISTER . (CC) – The condition code register is a 5-bit register in which four bits are used to indicate the results of the instruction just executed. These bits can be individually tested by a program and specific action taken as a result of their state. Each bit is explained in the following paragraphs.

Half Carry (H) – Set during ADD and ADC operations to indicate that a carry occurred between bits 3 and 4.

Interrupt (I) – When this bit is set, the timer and external interrupts ($\overline{INT}$ and $\overline{INT2}$) are masked (disabled). If an interrupt occurs while this bit is set, the interrupt is latched and is processed as soon as the interrupt bit is cleared.

Negative (N) – When set, this bit indicates that the result of the last arithmetic, logical, or data manipulation was negative (bit 7 in the result is a logical "1").

Zero (Z) – When set, this bit indicates that the result of the last arithmetic, logical, or data manipulation was zero.

Carry/Borrow (C) – When set, this bit indicates that a carry or borrow out of the Arithmetic Logic Unit (ALU) occurred during the last arithmetic operation. This bit is also affected during bit test and branch instructions plus shifts and rotates.

TIMER

The timer circuitry for the EF6805R3 is shown in Figure 10. The timer contains a single 8-bit software programmable counter with a 7-bit software selectable prescaler. The counter may be preset under program control and decrements toward zero. When the counter decrements to zero, the timer interrupt request bit, i.e., bit 7 of the timer control register (TCR), is set. Then if the timer interrupt is not masked, i.e., bit 6 of the TCR and the I bit in the condition code register are both cleared, the processor receives an interrupt. After completion of the current instruction, the processor proceeds to store the appropriate registers on the stack, and then fetches the timer interrupt vector from locations $FF8 and $FF9 in order to begin servicing the interrupt.

The counter continues to count after it reaches zero, allowing the software to determine the number of internal or external input clocks since the timer interrupt request bit was set. The counter may be read at any time by the processor without disturbing the count. The contents of the counter become stable prior to the read portion of a cycle and do not change during the read. The timer interrupt request bit remains set until cleared by the software. If a write occurs before the timer interrupt is serviced, the interrupt is lost. TCR7 may also be used as a scanned status bit in a non-interrupt mode of operation **(TCR6 = 1)**.

The prescaler is a 7-bit divider which is used to extend the maximum length of the timer. Bit 0, bit 1, and bit 2 of the TCR are programmed to choose the appropriate prescaler output which is used as the counter input. The processor cannot write into or read from the prescaler ; however, its contents are cleared to all zeros by the write operation into TCR when bit 3 of the written data equals one, which allows for truncation-free counting.

The timer input can be configured for three different operating modes, plus a disable mode, depending on the value written to the TCR4 and TCR5 control bits. For further information see Figure 9.

Timer Input Mode 1 – If TCR5 and TCR4 are both programmed to a zero, the input to the timer is from an internal clock and the external TIMER input is disabled. The internal clock mode can be used for periodic interrupt generation, as well as a reference in frequency and event measurement. The internal clock is the instruction cycle clock.

Timer Input Mode 2 – With TCR5 = 0 and TCR4 = 1, the internal clock and the TIMER input pin are ANDed to form the timer input signal. This mode can be used to measure external pulse widths. The external timer input pulse simply turns on the internal clock for the duration of the pulse widths.

Timer Input Mode 3 – If TCR5 = 1 and TCR4 = 0, then all inputs to the timer are disabled.

Timer Input Mode 4 – If TCR5 = 1 and TCR4 = 1, the internal clock input to the timer is disabled and the TIMER input pin becomes the input to the timer. The external TIMER pin can, in this mode, be used to count external events as well as external frequencies for generating periodic interrupts.

7	6	5	4	3	2	1	0	
TCR7	TCR6	TCR5	TCR4	TCR3*	TCR2	TCR1	TCR0	$009

*Write only (read as zero).

TCR7 – Timer Interrupt Request Bit :
1 - Set when TDR goes to zero, or under program control
0 - Cleared on external Reset, Power-On-Reset, or under Program Control.

TCR6 – Timer Interrupt Mask Bit :
1 - Timer Interrupt masked (disabled) Set on external Reset, Power-On-Reset, or under Program Control
0 - Cleared under Program Control.

TCR5 – External or Internal Clock Source Bit :
1 - External Clock Source. Set on external Reset, Power On-Reset, or under Program Control
0 - Cleared under Program Control.

TCR4 – External Enable Bit :
1 - Enable external TIMER pin. Set on external Reset, Power-On-Reset, or under Program Control.
0 - Cleared under Program Control.

TCR5	TCR4	Result
0	0	Internal clock to timer
0	1	AND of internal clock and TIMER pin to timer
1	0	Input to timer disabled
1	1	TIMER pin to timer

TCR3 – Timer prescaler reset bit : A read of TCR3 always indicates a zero.
1 - Set on external Reset, Power-On-Reset or under Program Control.
0 - Cleared under Program Control.

TCR2, TCR1, and TCR0 – Prescaler address bits :
1 - All set on external Reset, Power-On-Reset or under Program Control.
0 - Cleared under Program Control.

TCR2	TCR1	TCR0	Result
0	0	0	÷ 1
0	0	1	÷ 2
0	1	0	÷ 4
0	1	1	÷ 8

TCR2	TCR1	TCR0	Result
1	0	0	÷ 16
1	0	1	÷ 32
1	1	0	÷ 64
1	1	1	÷ 128

FIGURE 9 – TIMER CONTROL REGISTER (TCR)

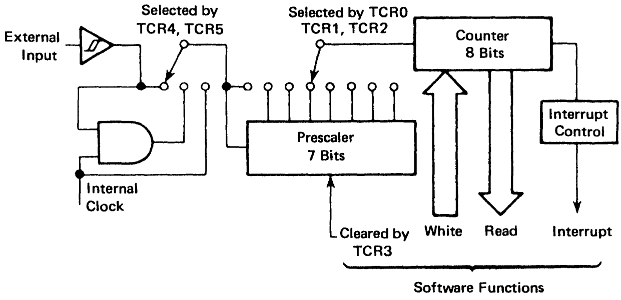


FIGURE 10 – TIMER BLOCK DIAGRAM

NOTES:
1. Prescaler and 8-bit counter are clocked on the failing edge of the internal clock (AS) or external input
2. Counter is written to during data strobe (DS) and counts down continuously

SELF-CHECK

The self-check capability of the EF6805R3 MCU provides an internal check to determine if the part is functional. Connect the MCU as shown in Figure 11 and monitor the output of Port C bit 3 for an oscillation of approximately 7 Hz. A 10-volt level (through a 10 k resistor) on the timer input, pin 8 and pressing then releasing the RESET button, energizes the ROM-based self-check feature. The self-check program exercises the RAM, ROM, TIMER, A/D, interrupts, and I/O ports.

Several of the self-check subroutines can be called by a user program with a JSR or BSR instruction. They are the RAM, ROM, and 4-channel A/D tests. The timer routine may also be called if the timer input is the internal $\phi 2$ clock.

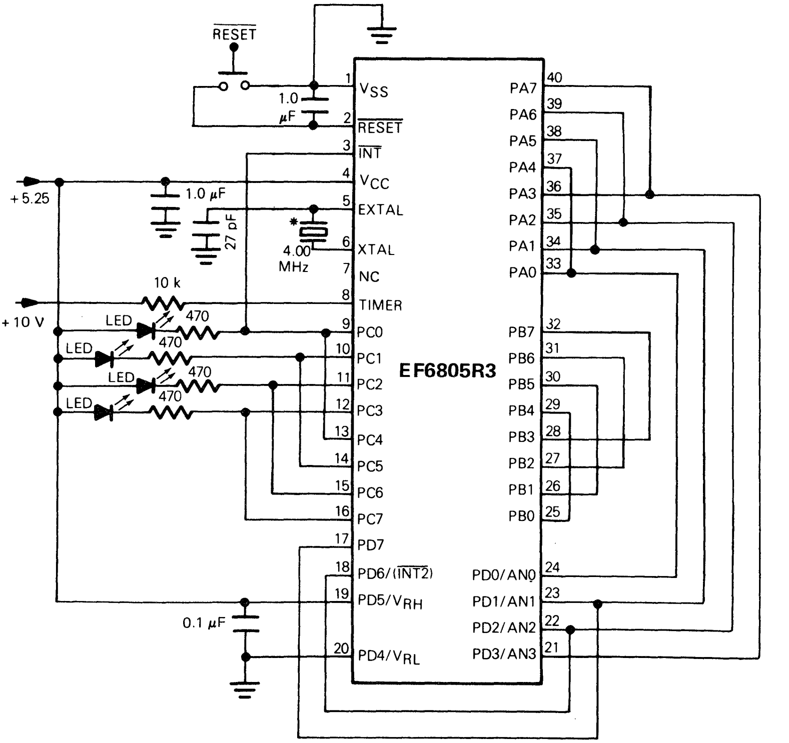
To call those subroutines in customer applications, please contact your local THOMSON SEMICONDUCTEURS sales office in order to obtain the complete description of the self-check program and the entrance/exit conditions.

RAM SELF-CHECK SUBROUTINE – The RAM self-check is called at location $F84 and returns with the Z bit clear if any error is detected ; otherwise the Z bit is set. The RAM test causes each byte to count from 0 up to 0 again with a check after each count.

The RAM test must be called with the stack pointer at $07F and A = 0. When run, the test checks every RAM cell except for $07F and $07E which are assumed to contain the return address.

The A and X registers and all RAM locations except $07F and $07E are modified.

ROM CHECKSUM SUBROUTINE – The ROM self-check is called at location $F95. The A register should be cleared before calling the routine. If any error is detected, it returns with the Z bit cleared ; otherwise Z = 1, X = 0 on return, and A is zero if the test passes. RAM location $40 to $043 is overwritten. The cheksum is the complement of the execution OR of the contents of the user ROM.



*This connection depends on clock oscillator user selectable mask option. Use jumper if the RC mask option is selected.

LED Meanings

PC0	PC1	PC2	PC3	Remarks [1:LED ON; 0:LED OFF]
1	0	1	0	Bad I/O
0	0	1	0	Bad Timer
1	1	0	0	Bad RAM
0	1	0	0	Bad ROM
1	0	0	0	Bad A/D
0	0	0	0	Bad Interrupts or Request Flag
All Flashing				Good Device

Anything else bad Device, Bad Port C, etc.

FIGURE 11 – SELF-CHECK CONNECTIONS

ANALOG-TO-DIGITAL CONVERTER SELF-CHECK– The A/D self-check is called at location $FAE. It returns with the Z bit cleared if any error was found ; otherwise the Z bit is set. The A and X register contents are lost. The X register must be set to four before the call. On return, X = 8 and A/D channel 7 is selected. The A/D test uses the internal voltage references and confirms port connections.

TIMER SELF-CHECK SUBROUTINE – The timer self-check is called at location $F6D and returns with the Z bit cleared if any error was found; otherwise Z = 1.

In order to work correctly as a user subroutine, the internal $\phi 2$ clock must be the clocking source and interrupts must be disabled. Also, on exit, the clock is running and the interrupt mask is not set so the caller must protect from interrupts if necessary.

The A and X register contents are lost. This routine sets the prescaler for divide-by-128 and the timer data register is cleared. The X register is configured to count down the same as the timer data register. The two registers are then compared every 128 cycles until they both count down to zero. Any mismatch during the count down is considered as an error. The A and X registers are cleared on exit from the routine.

RESET

The MCU can be reset three ways : by initial powerup, by the external reset input ($\overline{\text{RESET}}$) and by an optional internal low-voltage detect circuit. The $\overline{\text{RESET}}$ input consists mainly of a Schmitt trigger which senses the $\overline{\text{RESET}}$ line logic level. A typical reset Schmitt trigger hysteresis curve is shown in Figure 12. The Schmitt trigger provides an internal reset voltage if it senses a logical zero on the $\overline{\text{RESET}}$ pin.

Power-On Reset (POR) – An internal reset is generated upon powerup that allows the internal clock generator to stabilize. A delay of t_{RHL} milliseconds is required before allowing the $\overline{\text{RESET}}$ input to go high. Refer to the power and reset timing diagram of Figure 13. Connecting a capacitor to the $\overline{\text{RESET}}$ input (as illustrated in Figure 14) typically provides sufficient delay. During powerup, the Schmitt trigger switches on (removes reset) when $\overline{\text{RESET}}$ rises to V_{IRES+}.

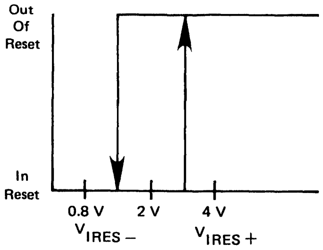


FIGURE 12 – TYPICAL RESET SCHMITT TRIGGER HYSTERESIS

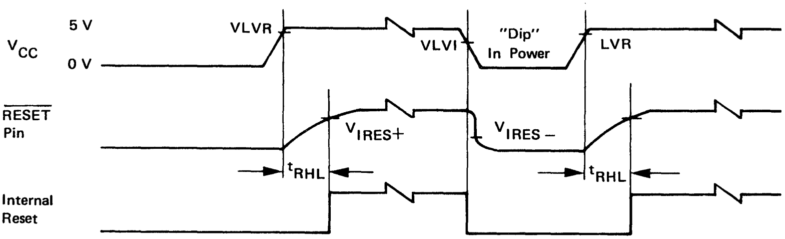


FIGURE 13 – POWER AND RESET TIMING

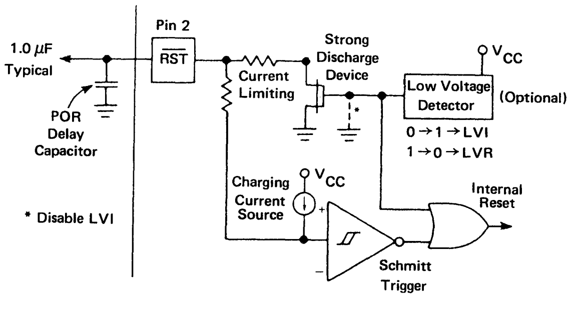


FIGURE 14 – $\overline{\text{RESET}}$ CONFIGURATION

External Reset Input – The MCU will be reset if a logical zero is applied to the RESET input for a period longer than one machine cycle (t_{cyc}). Under this type of reset, the Schmitt trigger switches off at V_{IRES-} to provide an internal reset voltage.

Low-Voltage Inhibit (LVI) – The optional low-voltage detection circuit causes a reset of the MCU if the power supply voltage falls below a certain level (V_{LVI}). The only requirement is that V_{CC} remains at or below the V_{LVI} threshold for one t_{cyc} minimum. In typical applications, the V_{CC} bus filter capacitor will eliminate negative-going voltage glitches of less than one t_{cyc}. The output from the low-voltage detector is connected directly to the internal reset circuitry. It also forces the RESET pin low via a strong discharge device through a resistor. The internal reset will be removed once the power supply voltage rises above a recovery level (V_{LVR}), at which time a normal power-on-reset occurs.

INTERNAL CLOCK GENERATOR OPTIONS

The internal clock generator circuit is designed to require a minimum of external components. A crystal, a resistor, a jumper wire, or an external signal may be used to generate a system clock with various stability/cost tradeoffs. The oscillator frequency is internally divided by four to produce the internal system clocks. A manufacturing mask option is used to select crystal or resistor operation.

The different connection methods are shown in Figure 15. Crystal specifications and suggested PC board layouts are given in Figure 16. A resistor selection graph is given in Figure 17.

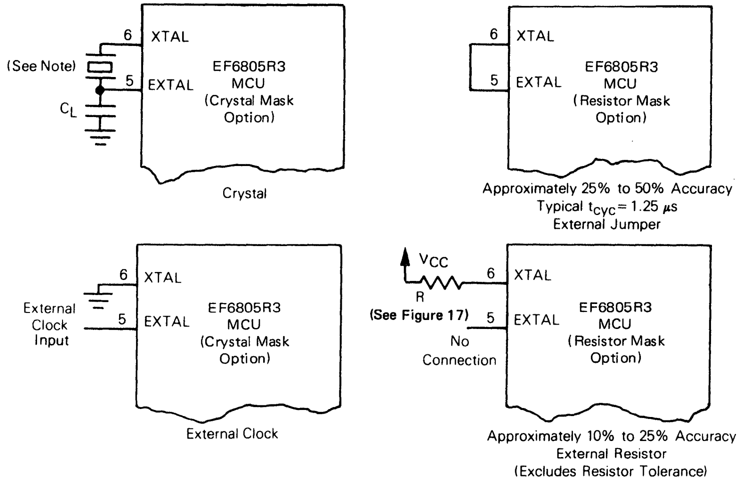
The crystal oscillator start-up time is a function of many variables : crystal parameters (especially R_S), oscillator load capacitances, IC parameters, ambient temperature, and supply voltage. To ensure rapid oscillator start up, neither the crystal characteristics nor the load capacitances should exceed recommendations.

When utilizing the on-board oscillator, the MCU should remain in a reset condition (reset pin voltage below V_{IRES+}) until the oscillator has stabilized at its operating frequency. Several factors are involved in calculating the external reset capacitor required to satisfy this condition: the oscillator start-up voltage, the oscillator stabilization time, the minimum V_{IRES+}, and the reset charging current specification.

Once V_{CC} minimum is reached, the external RESET capacitor will begin to charge at a rate dependent on the capacitor value. The charging current is supplied from V_{CC} through a large resistor, so it appears almost like a constant current source until the reset voltage rises above V_{IRES+}. Therefore, the RESET pin will charge at approximately :

$$(V_{IRES+}) \bullet C_{ext} = I_{RES} \bullet t_{RHL}$$

Assuming the external capacitor is initially discharged.



Crystal

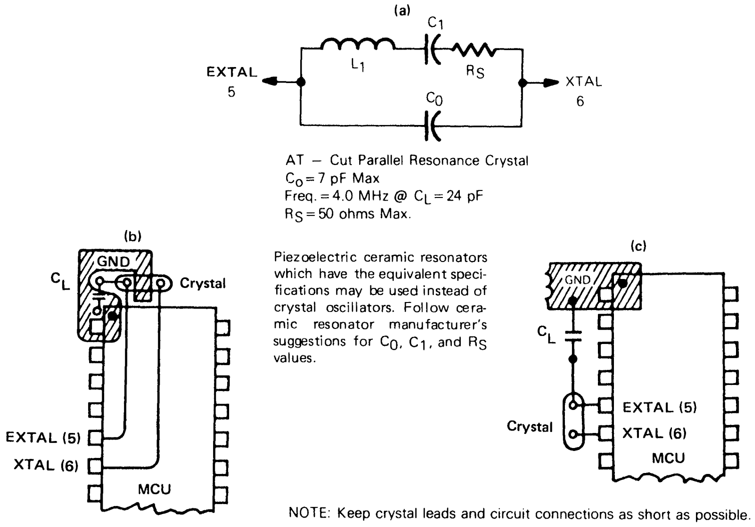
Approximately 25% to 50% Accuracy
Typical t_{cyc} = 1.25 µs
External Jumper

External Clock

Approximately 10% to 25% Accuracy
External Resistor
(Excludes Resistor Tolerance)

NOTE: The recommended C_L value with a 4.0 MHz crystal is 27 pF, maximum, including system distributed capacitance. There is an internal capacitance of approximately 25 pF on the XTAL pin. For crystal frequencies other than 4 MHz, the total capacitance on each pin should be scaled as the inverse of the frequency ratio. For example, with a 2 MHz crystal, use approximately 50 pF on EXTAL and approximately 25 pF on XTAL. The exact value depends on the Motional-Arm parameters of the crystal used.

FIGURE 15 – CLOCK GENERATOR OPTIONS



AT – Cut Parallel Resonance Crystal
$C_0 = 7$ pF Max
Freq. = 4.0 MHz @ $C_L = 24$ pF
$R_S = 50$ ohms Max.

Piezoelectric ceramic resonators which have the equivalent specifications may be used instead of crystal oscillators. Follow ceramic resonator manufacturer's suggestions for C_0, C_1, and R_S values.

NOTE: Keep crystal leads and circuit connections as short as possible.

FIGURE 16 – CRYSTAL MOTIONAL ARM PARAMETERS AND SUGGESTED PC BOARD LAYOUT

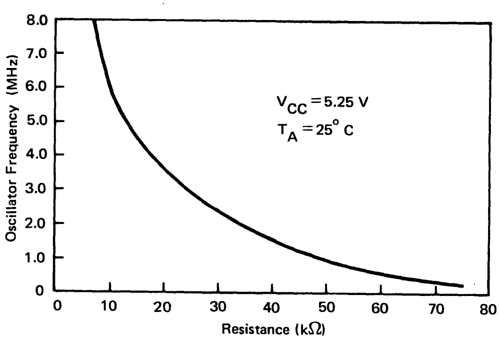


FIGURE 17 – TYPICAL FREQUENCY SELECTION FOR RESISTOR (OSCILLATOR OPTION)

INTERRUPTS

The microcomputers can be interrupted four different ways : through the external interrupt ($\overline{\text{INT}}$) input pin, the internal timer interrupt request, the external port D bit 6 ($\overline{\text{INT2}}$) input pin, or the software interrupt instruction (SWI). When any interrupt occurs : the current instruction (including SWI) is completed, processing is suspended, the present CPU state is pushed onto the stack, the interrupt bit (I) in the condition code register is set, the address of the interrupt routine is obtained from the appropriate interrupt vector address, and the interrupt routine is executed. Stacking the CPU register, setting the I bit, and vector fetching require a total of 11 t_{cyc} periods for completion. A flowchart of the interrupt sequence is shown in Figure 18. The interrupt service routine must end with a return from interrupt (RTI) instruction which allows the MCU to resume processing of the program prior to the interrupt (by unstacking the previous CPU state). Unlike $\overline{\text{RESET}}$, hardware interrupts do not cause the current instruction execution to be halted, but are considered pending until the current instruction execution is complete.

When the current instruction is complete, the processor checks all pending hardware interrupts and if unmasked, proceeds with interrupt processing ; otherwise the next instruction is fetched and executed. Note that masked interrupts are latched for later interrupt service.

If both an external interrupt and a timer interrupt are pending at the end of an instruction execution, the external interrupt is serviced first. The SWI is executed as any other instruction.

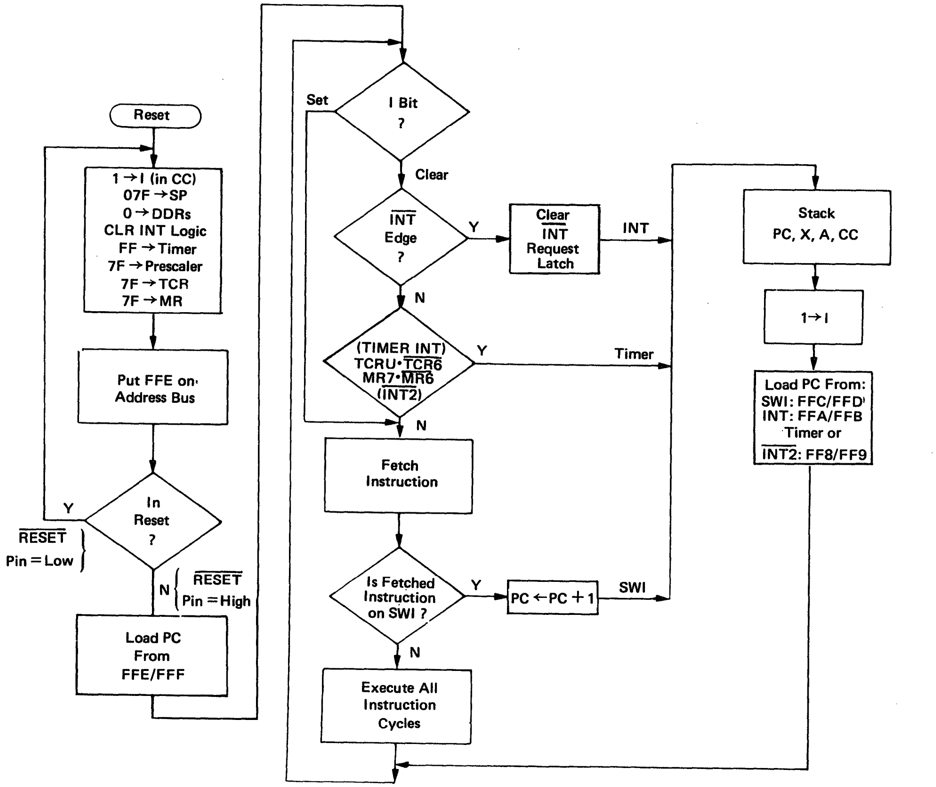


FIGURE 18 – $\overline{\text{RESET}}$ AND INTERRUPT PROCCESSING FLOWCHARD

NOTE

The timer and $\overline{\text{INT2}}$ interrupts share the same vector address. The interrupt routine must determine the source by examining the interrupt request bits (TCR b7 and MR b7). Both TCR b7 and MR b7 can only be written to zero by software.

The external interrupt, $\overline{\text{INT}}$ and $\overline{\text{INT2}}$, are synchronized and then latched on the falling edge of the input signal. The $\overline{\text{INT2}}$ interrupt has an interrupt request bit (bit 7) and a mask bit (bit 6) located in the miscellaneous register (MR). The $\overline{\text{INT2}}$ interrupt is inhibited when the mask bit is set. The $\overline{\text{INT2}}$ is always read as a digital input on port D. The $\overline{\text{INT2}}$ and timer interrupt request bits, if set, cause the MCU to process an interrupt when the condition code I bit is clear.

A sinusoidal input signal (f_{INT} maximum) can be used to generate an external interrupt for use as a zero-crossing detector. This allows applications such as servicing time-of-day routines and engaging/disengaging ac power control devices. Off-chip full wave rectification provides an interrupt at every zero crossing of the ac signal and thereby provides a 2f clock. See Figure 19.

NOTE

The $\overline{\text{INT}}$ (pin 3) is internally biased at approximately 2.2 V due to the internal zero-crossing detection.

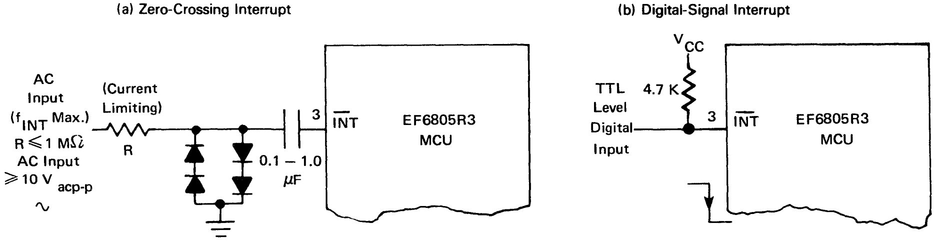
A software interrupt (SWI) is an executable instruction which is executed regardless of the state of the I bit in the condition code register. SWIs are usually used as breakpoints for debugging or as system calls.

A/D CONTROL REGISTER (ACR)

7	6	5	4	3	2	1	0	
	1	1	1	1				$00E

Bit 7 – Conversion Complete Status Flag: Set when conversion is complete; cleared only on a write to ACR.
Readable, not writable.

Bits 2, 1, 0 – A/D input Mux Selection (See Table 2).
Bits 6, 5, 4, 3 read as "1s" – unused bits.


(a) Zero-Crossing Interrupt

(b) Digital-Signal Interrupt

FIGURE 19 – TYPICAL INTERRUPT CIRCUITS

INPUT/OUTPUT CIRCUITRY

There are 32 input/output pins. The $\overline{\text{INT}}$ pin may be polled with branch instructions to provide an additional input pin. All pins on ports A, B, and C are programmable as either inputs or outputs under software control of the corresponding data direction register (DDR). See below I/O port control registers configuration. The port I/O programming is accomplished by writing the corresponding bit in the port DDR to a logic one for output or a logic zero for input. On reset all the DDRs are initialized to a logic zero state, placing the ports in the input mode. The port output registers are not initialized on reset and should be initialized by software before changing the DDRs from input to output. A read operation on a port programmed as an output will read the contents of the output latch regardless of the logic levels at the output pin, due to output loading. Refer to Figure 20.

PORT DATA REGISTER

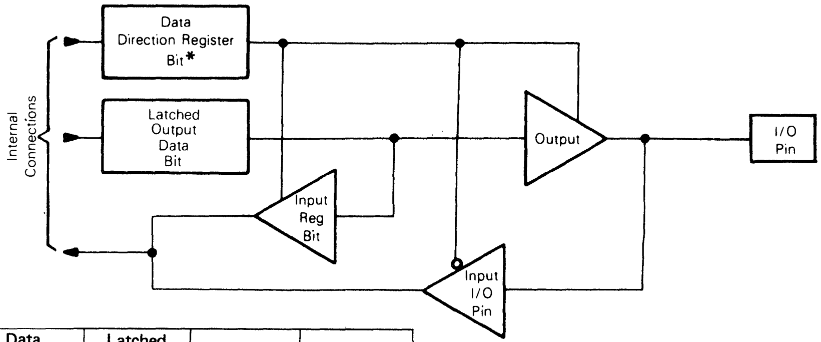
7	0

Port A Addr = $000
Port B Addr = $001
Port C Addr = $002
Port D Addr = $003

PORT DATA DIRECTION REGISTER (DDR)

7	0

(1) Write Only; reads as all "1s"
(2) 1 = Output, 0 = Input. Cleared to 0 by Reset
(3) Port A Addr = $004
Port B Addr = $005
Port C Addr = $006



Data Direction Register Bit	Latched Output Data Bit	Output State	Input To MCU
1	0	0	0
1	1	1	1
0	X	High-Z **	Pin

*DDR is a write-only register and reads as all "1s"
** Ports B, and C are three-state ports.
Port A has optional internal pullup devices to provide CMOS data drive capability. See Electrical Characteristics tables for complete information.

FIGURE 20 – TYPICAL PORT I/O CIRCUITRY

All input/output lines are TTL compatible as both inputs and outputs. Port A lines are CMOS compatible as outputs (mask option) while port B, C, and D lines are CMOS compatible as inputs. Port D lines are input only; thus, there is no corresponding DDR. When programmed as outputs, port B is capable of sinking 10 milliamperes and sourcing 1 milliampere on each pin.

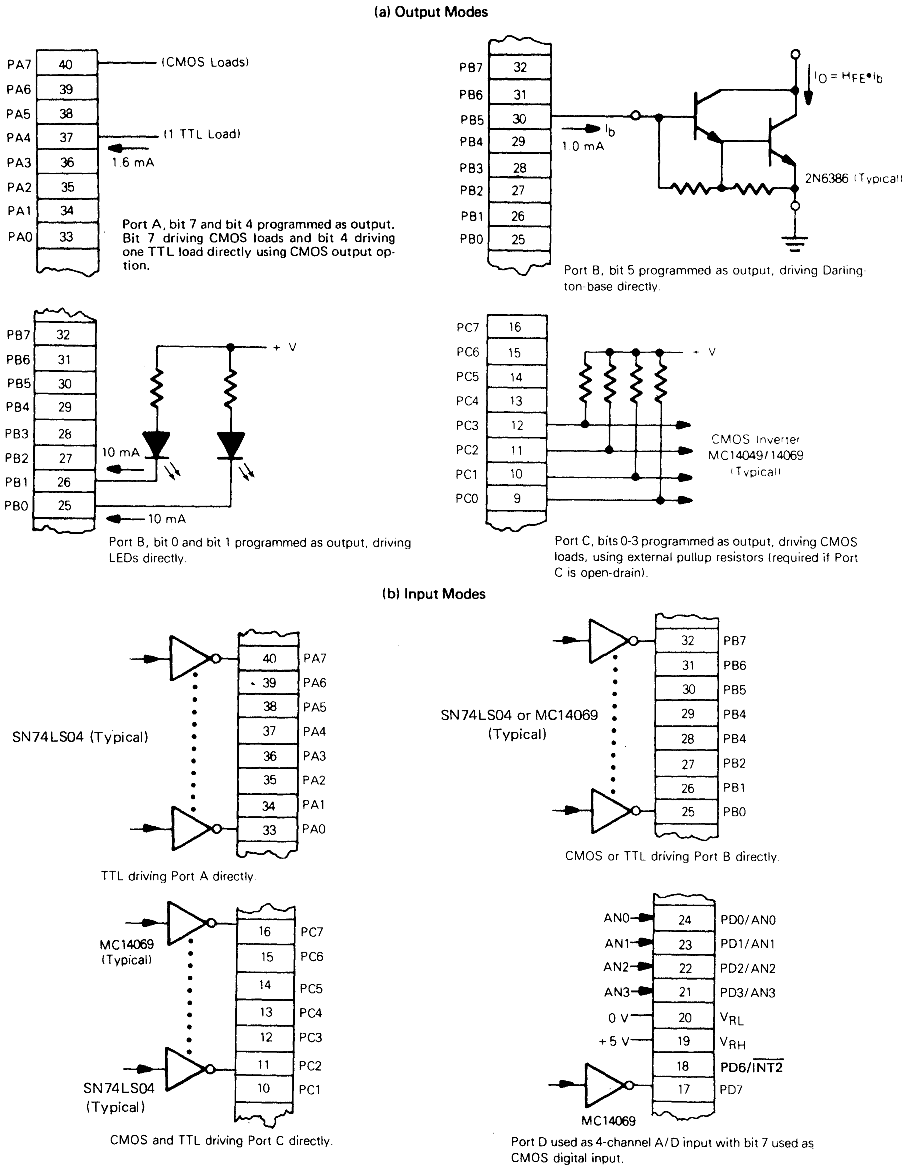
Port D provides the reference voltage, $\overline{\text{INT2}}$, and multiplexed analog inputs. All of these lines are shared with the port D digital inputs. Port D may always be used as digital inputs and may also be used as analog inputs providing V_{RH} and V_{RL} are connected to the appropriate reference voltages. The V_{RL} and V_{RH} lines (PD4 and PD5) are internally connected to the A/D resistor. Analog inputs may be prescaled to attain the V_{RL} and V_{RH} recommended input voltage range.

The address map (Figure 6) gives the addresses of data registers and data direction registers. Figure 21 provides some examples of port connections.

CAUTION

The corresponding DDRs for ports A, B, and C are write-only registers (registers at $004, $005, $006). A read operation on these registers is undefined. Since BSET and BCLR are read-modify-write in function, they cannot be used to set or clear a single DDR bit (all "unaffected" bits would be set). It is recommended that all DDR bits in a port be written using a single-store instruction.

The latched output data bit (see Figure 20) must always be written. Therefore, any write to a port writes all of its data bits even though the port DDR is set to input. This may be used to initialize the data register and avoid undefined outputs ; however, care must be exercised when using read-modify-write instructions, since the data read corresponds to the pin level if the DDR is an input (zero) and corresponds to the latched output data when the DDR is an output (one).



Port A, bit 7 and bit 4 programmed as output. Bit 7 driving CMOS loads and bit 4 driving one TTL load directly using CMOS output option.

Port B, bit 5 programmed as output, driving Darlington-base directly.

Port B, bit 0 and bit 1 programmed as output, driving LEDs directly.

Port C, bits 0-3 programmed as output, driving CMOS loads, using external pullup resistors (required if Port C is open-drain).

TTL driving Port A directly.

CMOS or TTL driving Port B directly.

CMOS and TTL driving Port C directly.

Port D used as 4-channel A/D input with bit 7 used as CMOS digital input.

FIGURE 21 – TYPICAL PORT CONNECTIONS

ANALOG-TO-DIGITAL COUVERTER

The EF6805R2 has an 8-bit analog-to-digital (A/D) converter implemented on the chip using a successive approximation technique, as shown in Figure 22. Up to four external analog inputs, via port D, are connected to the A/D through a multiplexer. Four internal analog channels may be selected for calibration .purposes (V_{RH}-V_{RL}, V_{RH}-V_{RL}/2, V_{RH}-V_{RL}/4, and V_{RL}). The accuracy of these internal channels will not necessarily meet the accuracy specifications of the external channels.

The multiplexer selection is controlled by the A/D control register (ACR) bits 0, 1, and 2 ; see Table 1. This register is cleared during any reset condition.

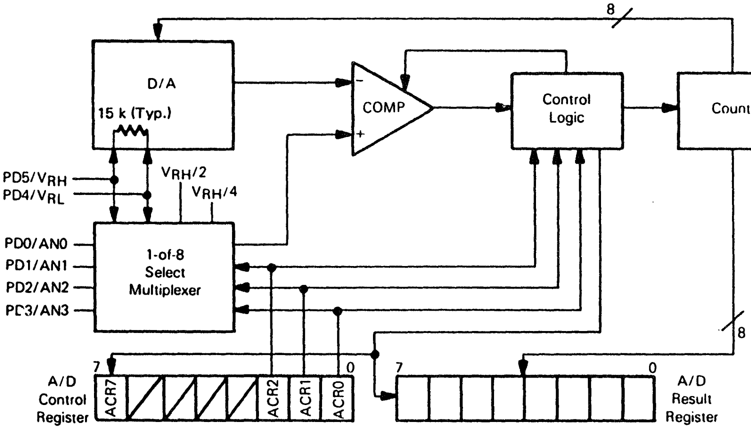


FIGURE 22 – A/D BLOCK DIAGRAM

A/D Control Register			Input Selected	A/D Output (Hex)		
ACR2	ACR1	ACR0		Min	Typ	Max
0	0	0	AN0			
0	0	1	AN1			
0	1	0	AN2			
0	1	1	AN3			
1	0	0	V_{RH}*	FE	FF	FF
1	0	1	V_{RL}*	00	00	01
1	1	0	V_{RH}/4*	3F	40	41
1	1	1	V_{RH}/2*	7F	80	81

*Internal (Calibration) levels

TABLE 1 – A/D INPUT MUX SELECTION

MISCELLANEOUS REGISTER (MR)

7	6					0	
		1	1	1	1	1	1	$00A

MR7 Bit 7 – $\overline{INT2}$ Interrupt Request Bit: Set when falling edge detected on $\overline{INT2}$ pin, must be cleared by software. Cleared to 0 by Reset

MR6 Bit 6 – $\overline{INT2}$ Interrupt Mask Bit: 1 = $\overline{INT2}$ Interrupt masked (disabled). Set to 1 by Reset.

MR Bits 5, 4, 3, 2, 1, 0 – Read as "1s" – unused bits.

A/D RESULT REGISTER (ARR)

7	0	
MSB	LSB	$00F

Whenever the ACR is written, the conversion in progress is aborted, the conversion complete flag (ACR bit 7) is cleared, and the selected input is sampled for five machine cycles and held internally. During these five cycles, the analog input will appear approximately like a 25 picofarad (maximum) capacitor (plus approximately 10 pF for packaging) charging through a 2.6 kiloohm resistor (typical). Refer to Figure 23.

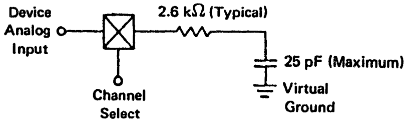


FIGURE 23 – EFFECTIVE ANALOG INPUT IMPEDANCE (DURING SAMPLING ONLY)

The converter operates continuously using 30 machine cycles to complete a conversion of the sampled analog input. When the conversion is complete , the digitized sample of digital value is placed in the A/D result register (ARR), the conversion complete flag is set, the selected input is sampled again, and a new conversion is started.

The A/D is ratiometric. Two reference voltages (V_{RH} and V_{RL}) are supplied to the converter via port D pins. An input voltage equal to V_{RH} converts to $FF (full scale) and an input voltage equal to V_{RL} converts to $00. An input voltage greater than V_{RH} converts to SFF and no overflow indication is provided. Similarly, an input voltage less than V_{RL}, but greater than V_{SS} converts to $00. Maximum and minimum ratings must not be exceeded. For ratiometric conversion, the source of each analog input should use V_{RH} as the supply voltage and be referenced to V_{RL}. To maintain the full accuracy on the A/D, V_{RH} should be equal to or less than V_{DD}, V_{RL} should be equal to or greater than V_{SS} but less than the maximum specification and (V_{RH}-V_{RL}) should be equal to or greater than 4 volts.

The A/D has a built-in 1/2 LSB offset intended to reduce the magnitude of the quantizing error to ± 1/2 LSB, rather than + 0, — 1 LSB with no offset. This implies that, ignoring errors, the transition point from $00 to $01 occurs at 1/2 LSB above V_{RL}. Similarly, the transition from $FE to $FF occurs 1 1/2 LSB below V_{RH}, ideally. Refer to Figure 24 and 25.

On release of reset, the A/D control register (ACR) is cleared therefore after reset, channel zero will be selected and the conversion complete flag will be clear.

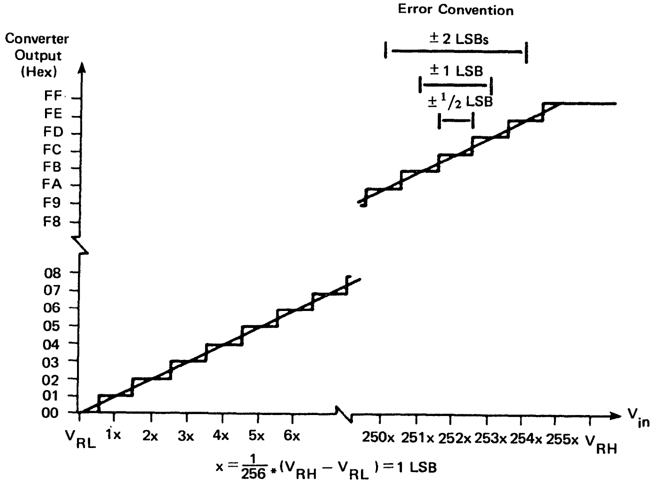


FIGURE 24 – IDEAL CONVERTER TRANSFER CHARACTERISTIC

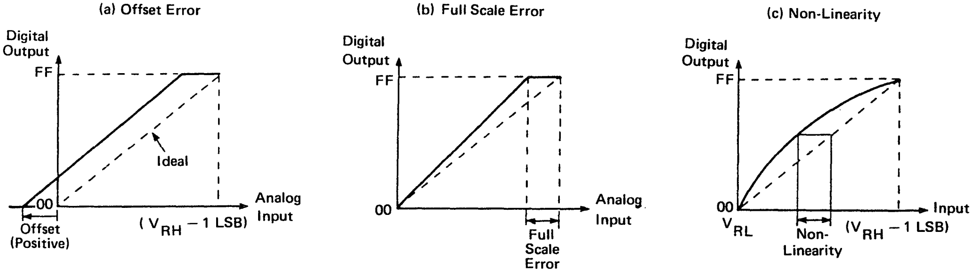


FIGURE 25 – TYPES OF CONVERSION ERRORS

BIT MANIPULATION

The EF6805 R3 has the ability to set or clear any single RAM or I/O bit (except the data direction registers) with a single instruction (BSET, BCLR) (see Caution below). Any bit in page zero can be tested using the BRSET and BRCLR instructions and the program branches as a result of its state. The carry bit equals the value of the bit references by BRSET or BRCLR. The capability to working with any bit in RAM, ROM, or I/O allows the user to have individual flags in RAM or to handle single I/O bits as control lines.

CAUTION

The corresponding data direction registers for ports A, B, and C are write-only registers (locations $004, $005, and $006). A read operation on these registers is undefined. Since BSET and BCLR are read-modify-write functions, they cannot be used to set a data direction register bit (all "unaffected" bits would be set). It is recommended that all data direction register bits in a port be written using a single-store instruction.

The coding examples shown in Figure 26 illustrate the usefulness of the bit manipulation and test instruction. Assume that the microcomputer is to communicate with an external serial device. The external device has a data ready signal, a data output line, and a clock line to clock data one bit at a time, least significant bit first, out of the device. The microcomputer waits until the data is ready, clocks the external device, picks up the data in the carry flag, clears the clock line, and finally accumulates the data bit in a random-access memory location.

```
*       BRSET   2,PORTA, *          WAIT FOR READY

        BSET    1, PORTA            CLOCK NEXT BIT IN
        BRCLR   0, PORTA, NEXT      PICKUP BIT IN C-BIT
NEXT    BCLR    1, PORTA            RETURN CLOCK LINE HIGH
        ASR     RAMLOC              MOVE C-BIT INTO RAM
```

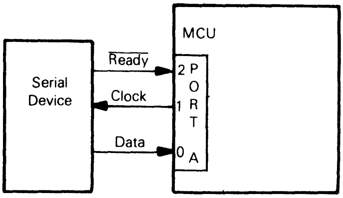


FIGURE 26 - BIT MANIPULATION EXAMPLE

ADDRESSING MODES

The EF6805R3 MCU has ten addressing modes available for use by the programmer. They are explained briefly in the following paragraphs. For additional details and graphical illustrations, refer to the EF6805 Family Users Manual.

The term "effective address" (EA) is used in describing the addressing modes. EA is defined as the address from which the argument for an instruction is fetched or stored.

IMMEDIATE – In the immediate addressing mode, the operand is contained in the byte immediately following the opcode. The immediate addressing mode is used to access constants which do not change during program execution (e.g., a constant used to initialize a loop counter).

DIRECT – In the direct addressing mode, the effective address of the argument is contained in a single byte following the opcode byte. Direct addressing allows the user to directly address the lowest 256 bytes in memory with a single 2-byte instruction. This address area includes all on-chip RAM and I/O registers and 128 bytes of ROM. Direct addressing is an effective use of both memory and time.

EXTENDED – In the extended addressing mode, the effective address of the argument is contained in the two bytes following the opcode. Instructions with extended addressing mode are capable of referencing arguments anywhere in memory with a single 3-byte instruction. When using the assembler, the user need not specify whether an instruction uses direct or extended addressing. The assembler automatically selects the shortest form of the instruction.

RELATIVE – The relative addressing mode is only used in branch instructions. In relative addressing, the contents of the 8-bit signed byte following the opcode (the offset) is added to the PC if, and only if, the branch condition is true. Otherwise, control proceeds to the next instruction. The span of relative addressing is from – 126 to + 129 from the opcode address. The programmer need not worry about calculating the correct offset if he uses the assembler, since it calculates the proper offset and checks to see if it is within the span of the branch.

INDEXED, NO OFFSET – In the indexed, no offset addressing mode, the effective address of the argument is contained in the 8-bit index register. Thus, this addressing mode can access the first 256 memory locations. These instructions are only one byte long. This mode is often used to move a pointer through a table or to hold the address of a frequently referenced RAM or I/O location.

INDEXED, 8-BIT OFFSET – In the indexed, 8-bit offset addressing mode, the effective address is the sum of the contents of the unsigned 8-bit index register and the unsigned byte following the opcode. This addressing mode is useful in selecting the kth element in an n element table. With this 2-byte instruction, k would typically be in X with the address of the beginning of the table in the instruction. As such, tables may begin anywhere within the first 256 addressable locations and could extend as far as location 510 ($1FE is the last location at which the instruction may begin).

INDEXED, 16-BIT OFFSET – In the indexed, 16-bit offset addressing mode, the effective address is the sum of the contents of the unsigned 8-bit index register and the two unsigned bytes following the opcode. This addressing mode can be used in a manner similar to indexed, 8-bit offset except that this 3-byte instruction allows tables to be anywhere in memory. As with direct and extended, the assembler determines the shortest form of indexed addressing.

BIT SET/CLEAR – In the bit set/clear addressing mode, the bit to be set or cleared is part of the opcode, and the byte following the opcode specifies the direct address of the byte in which the specified bit is to be set or cleared. Thus, any read/write bit in the first 256 locations of memory, including I/O, can be selectively set or cleared with a single 2-byte instruction.

CAUTION

The corresponding DDRs for ports A, B, and C are write-only registers (registers at $004, $005, $006). A read operation on these registers is undefined. Since

BSET and BCLR are read-modify-write in function, they cannot be used to set or clear a single DDR bit **(all "unaffected" bits would be set). It is recommended that all DDR bits in a port must be written using a single-store instruction.**

BIT TEST AND BRANCH – The bit test and branch addressing mode is a combination of direct addressing and relative addressing. The bit which is to be tested and condition (set or clear) is included in the opcode, and the address of the byte to be tested is in the single byte immediately following the opcode byte. The signed relative 8-bit offset in the third byte is added to the PC if the specified bit is set or cleared in the specified memory location. This single 3-byte instruction allows the program to branch based on the condition of any readable bit in the first 256 locations of memory. The span of branching is from −125 to +130 from the opcode address. The state of the tested bit is also transferred to the carry bit of the condition code registers.

CAUTION

The corresponding DDRs for ports A, B, and C are write-only registers (registers at $004, $005, $006). A read operation on these registers is undefined. Since BSET and BCLR are read-modify-write in function, they cannot be used to set or clear a single DDR bit **(all "unaffected" bits would be set). It is recommended that all DDR bits in a port must be written using a single-store instruction.**

INHERENT -- In the inherent addressing mode, all the information necessary to execute the instruction is contained in the opcode. Operations specifying only the index register or accumulator, as well as control instructions with no other arguments, are included in this mode. These instructions are one byte long.

INSTRUCTION SET

The EF6805R3 MCU has a set of 59 basic instructions, which when combined with the 10 addressing modes produce 207 usable opcodes. They can be divided into five different types: register/memory, read-modify-write, branch, bit manipulation, and control. The following paragraphs briefly explain each type. All the instructions within a given type are presented in individual tables.

REGISTER/MEMORY INSTRUCTIONS – Most of these instructions use two operands. One operand is either the accumulator or the index register. The other operand is obtained from memory using one of the addressing modes. The jump unconditional (JMP) and jump to subroutine (JSR) instructions have no register operand. Refer to Table 1.

READ-MODIFY-WRITE INSTRUCTIONS – These instructions read a memory location or a register, modify or test its contents, and write the modified value back to memory or to the register; see Caution under Input/Output section. The test for negative or zero (TST) instruction is included in the read-modify-write instruction though it does not perform the write. Refer to Table 2.

BRANCH INSTRUCTIONS – The branch instructions cause a branch from the program when a certain condition is met. Refer to Table 3.

BIT MANIPULATION INSTRUCTIONS – The instructions are used on any bit in the first 256 bytes of the memory; One group either sets or clears. The other group performs the bit test and branch operations. Refer to Table 4.

CAUTION

The corresponding DDRs for ports A, B, and C are write-only registers (registers at $004, $005, $006). A read operation on these registers is undefined. Since BSET and BCLR are read-modify-write in function, they cannot be used to set or clear a single DDR bit (all "unaffected" bits would be set). It is recommended that all DDR bits in a port must be written using a single-store instruction.

CONTROL INSTRUCTION – The control instructions control the MCU operations during program execution. Refer to Table 5.

ALPHABETICAL LISTING – The complete instruction set is given in alphabetical order in Table 6.

OPCODE MAP – Table 8 is an opcode map for the instruction used on the MCU.

TABLE 1 – REGISTER/MEMORY INSTRUCTIONS

		Addressing Modes																	
		Immediate			Direct			Extended			Indexed (No Offset)			Indexed (8-Bit Offset)			Indexed (16-Bit Offset)		
Function	Mnemonic	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles	OP Code	# Bytes	# Cycles
Load A from Memory	LDA	A6	2	2	B6	2	4	C6	3	5	F6	1	4	E6	2	5	D6	3	6
Load X from Memory	LDX	AE	2	2	BE	2	4	CE	3	5	FE	1	4	EE	2	5	DE	3	6
Store A in Memory	STA	—	—		B7	2	5	C7	3	6	F7	1	5	E7	2	6	D7	3	7
Store X in Memory	STX	—	—	—	BF	2	5	CF	3	6	FF	1	5	EF	2	6	DF	3	7
Add Memory to A	ADD	AB	2	2	BB	2	4	CB	3	5	FB	1	4	EB	2	5	DB	3	6
Add Memory and Carry to A	ADC	A9	2	2	B9	2	4	C9	3	5	F9	1	4	E9	2	5	D9	3	6
Subtract Memory	SUB	A0	2	2	B0	2	4	C0	3	5	F0	1	4	E0	2	5	D0	3	6
Subtract Memory from A with Borrow	SBC	A2	2	2	B2	2	4	C2	3	5	F2	1	4	E2	2	5	D2	3	6
AND Memory to A	AND	A4	2	2	B4	2	4	C4	3	5	F4	1	4	E4	2	5	D4	3	6
OR Memory with A	ORA	AA	2	2	BA	2	4	CA	3	5	FA	1	4	EA	2	5	DA	3	6
Exclusive OR Memory with A	EOR	A8	2	2	B8	2	4	C8	3	5	F8	1	4	E8	2	5	D8	3	6
Arithmetic Compare A with Memory	CMP	A1	2	2	B1	2	4	C1	3	5	F1	1	4	E1	2	5	D1	3	6
Arithmetic Compare X with Memory	CPX	A3	2	2	B3	2	4	C3	3	5	F3	1	4	E3	2	5	D3	3	6
Bit Test Memory with A (Logical Compare)	BIT	A5	2	2	B5	2	4	C5	3	5	F5	1	4	E5	2	5	D5	3	6
Jump Unconditional	JMP	—	—	—	BC	2	3	CC	3	4	FC	1	3	EC	2	4	DC	3	5
Jump to Subroutine	JSR	—	—	—	BD	2	7	CD	3	8	FD	1	7	ED	2	8	DD	3	9

TABLE 2 – READ-MODIFY-WRITE INSTRUCTIONS

		Addressing Modes														
		Inherent (A)			Inherent (X)			Direct			Indexed (No Offset)			Indexed (8 Bit Offset)		
Function	Mnemonic	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles
Increment	INC	4C	1	4	5C	1	4	3C	2	6	7C	1	6	6C	2	7
Decrement	DEC	4A	1	4	5A	1	4	3A	2	6	7A	1	6	6A	2	7
Clear	CLR	4F	1	4	5F	1	4	3F	2	6	7F	1	6	6F	2	7
Complement	COM	43	1	4	53	1	4	33	2	6	73	1	6	63	2	7
Negate (2's Complement)	NEG	40	1	4	50	1	4	30	2	6	70	1	6	60	2	7
Rotate Left Thru Carry	ROL	49	1	4	59	1	4	39	2	6	79	1	6	69	2	7
Rotate Right Thru Carry	ROR	46	1	4	56	1	4	36	2	6	76	1	6	66	2	7
Logical Shift Left	LSL	48	1	4	58	1	4	38	2	6	78	1	6	68	2	7
Logical Shift Right	LSR	44	1	4	54	1	4	34	2	6	74	1	6	64	2	7
Arithmetic Shift Right	ASR	47	1	4	57	1	4	37	2	6	77	1	6	67	2	7
Test for Negative or Zero	TST	4D	1	4	5D	1	4	3D	2	6	7D	1	6	6D	2	7

TABLE 3 – BRANCH INSTRUCTIONS

		Relative Addressing Mode		
Function	Mnemonic	Op Code	# Bytes	# Cycles
Branch Always	BRA	20	2	4
Branch Never	BRN	21	2	4
Branch IFF Higher	BHI	22	2	4
Branch IFF Lower or Same	BLS	23	2	4
Branch IFF Carry Clear	BCC	24	2	4
(Branch IFF Higher or Same)	(BHS)	24	2	4
Branch IFF Carry Set	BCS	25	2	4
(Branch IFF Lower)	(BLO)	25	2	4
Branch IFF Not Equal	BNE	26	2	4
Branch IFF Equal	BEQ	27	2	4
Branch IFF Half Carry Clear	BHCC	28	2	4
Branch IFF Half Carry Set	BHCS	29	2	4
Branch IFF Plus	BPL	2A	2	4
Branch IFF Minus	BMI	2B	2	4
Branch IFF Interupt Mask Bit is Clear	BMC	2C	2	4
Branch IFF Interrupt Mask Bit is Set	BMS	2D	2	4
Branch IFF Interrupt Line is Low	BIL	2E	2	4
Branch IFF Interrupt Line is High	BIH	2F	2	4
Branch to Subroutine	BSR	AD	2	8

TABLE 4 – BIT MANIPULATION INSTRUCTIONS

		Addressing Modes					
		Bit Set/Clear			Bit Test and Branch		
Function	Mnemonic	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles
Branch IFF Bit n is set	BRSET n (n = 0 . 7)	—	—	—	2 • n	3	10
Branch IFF Bit n is clear	BRCLR n (n = 0 7)	—	—	—	01 + 2 • n	3	10
Set Bit n	BSET n (n = 0 7)	10 + 2 • n	2	7	—	—	—
Clear bit n	BCLR n (n = 0 ..7)	11 + 2 • n	2	7	—	—	—

TABLE 5 – CONTROL INSTRUCTIONS

		Inherent		
Function	Mnemonic	Op Code	# Bytes	# Cycles
Transfer A to X	TAX	97	1	2
Transfer X to A	TXA	9F	1	2
Set Carry Bit	SEC	99	1	2
Clear Carry Bit	CLC	98	1	2
Set Interrupt Mask Bit	SEi	9B	1	2
Clear Interrupt Mask Bit	CLI	9A	1	2
Software Interrupt	SWI	83	1	11
Return from Subroutine	RTS	81	1	6
Return from Interrupt	RTI	80	1	9
Reset Stack Pointer	RSP	9C	1	2
No-Operation	NOP	9D	1	2

TABLE 6 – INSTRUCTION SET

	Addressing Modes										Condition Code				
Mnemonic	Inherent	Immediate	Direct	Extended	Relative	Indexed (No Offset)	Indexed (8 Bits)	Indexed (16 Bits)	Bit Set/ Clear	Bit Test & Branch	H	I	N	Z	C
ADC		X	X	X		X	X	X			Λ	●	Λ	Λ	Λ
ADD		X	X	X		X	X	X			Λ	●	Λ	Λ	Λ
AND		X	X	X		X	X	X			●	●	Λ	Λ	●
ASL	X		X			X	X				●	●	Λ	Λ	Λ
ASR	X		X			X	X				●	●	Λ	Λ	Λ
BCC					X						●	●	●	●	●
BCLR									X		●	●	●	●	●
BCS					X						●	●	●	●	●
BEQ					X						●	●	●	●	●
BHCC					X						●	●	●	●	●
BHCS					X						●	●	●	●	●
BHI					X						●	●	●	●	●
BHS					X						●	●	●	●	●
BIH					X						●	●	●	●	●
BIL					X						●	●	●	●	●
BIT		X	X	X		X	X	X			●	●	Λ	Λ	●
BLO					X						●	●	●	●	●
BLS					X						●	●	●	●	●
BMC					X						●	●	●	●	●
BMI					X						●	●	●	●	●
BMS					X						●	●	●	●	●
BNE					X						●	●	●	●	●
BPL					X						●	●	●	●	●
BRA					X						●	●	●	●	●
BRN					X						●	●	●	●	●
BRCLR										X	●	●	●	●	Λ
BRSET										X	●	●	●	●	Λ
BSET									X		●	●	●	●	●
BSR					X						●	●	●	●	●
CLL	X										●	●	●	●	O
CLI	X										●	O	●	●	●
CLR	X		X			X	X				●	●	O	1	●
CMP		X	X	X		X	X	X			●	●	Λ	Λ	Λ
COM	X		X			X	X				●	●	Λ	Λ	1
CPX		X	X	X		X	X	X			●	●	Λ	Λ	Λ
DEC	X		X			X	X				●	●	Λ	Λ	●
EOR		X	X	X		X	X	X			●	●	Λ	Λ	●
INC	X		X			X	X				●	●	Λ	Λ	●
JMP			X	X		X	X	X			●	●	●	●	●
JSR			X	X		X	X	X			●	●	●	●	●
LDA		X	X	X		X	X	X			●	●	Λ	Λ	●
LDX		X	X	X		X	X	X			●	●	Λ	Λ	●
LSL	X		X			X	X				●	●	Λ	Λ	Λ
LSR	X		X			X	X				●	●	O	Λ	Λ
NEQ	X		X			X	X				●	●	Λ	Λ	Λ
NOP	X										●	●	●	●	●
ORA		X	X	X		X	X	X			●	●	Λ	Λ	●
ROL	X		X			X	X				●	●	Λ	Λ	Λ
RSP	X										●	●	●	●	●

Condition Code Symbols:

H	Half Carry (From Bit 3)	C	Carry/Borrow
I	Interrupt Mask	Λ	Test and Set if True, Cleared Otherwise
N	Negative (Sign Bit)	●	Not Affected
Z	Zero		

TABLE 6 – INSTRUCTION SET (CONTINUED)

	Addressing Modes										Condition Code				
Mnemonic	Inherent	Immediate	Direct	Extended	Relative	Indexed (No Offset)	Indexed (8 Bits)	Indexed (16 Bits)	Bit Set/ Clear	Bit Test & Branch	H	I	N	Z	C
RTI	X										?	?	?	?	?
RTS	X										●	●	●	●	●
SBC		X	X	X		X	X	X			●	●	Λ	Λ	Λ
SEC	X										●	●	●	●	1
SEI	X										●	1	●	●	●
STA			X	X		X	X	X			●	●	Λ	Λ	●
STX			X	X		X	X	X			●	●	Λ	Λ	●
SUB		X	X	X		X	X	X			●	●	Λ	Λ	Λ
SWI	X										●	1	●	●	●
TAX	X										●	●	●	●	●
TST	X		X			X	X				●	●	Λ	Λ	●
TXA	X										●	●	●	●	●

Condition Code Symbols:

H	Half Carry (From Bit 3)	C	Carry/Borrow
I	Interrupt Mask	Λ	Test and Set if True, Cleared Otherwise
N	Negative (Sign Bit)	●	Not Affected
Z	Zero	?	Load CC Register From Stack

EF6805 HMOS FAMILY

Features	EF6805CT	EF6805P2	EF6805P4	EF6805P6	EF6805R2	EF6805R3	EF6805T2	EF6805U2	EF6805U3	EF6805TV
Technology	HMOS	HMOS	HMOS	HMOS	HMOS	HMOS	HMOS	HMOS	HMOS	HMOS
Number of Pins	40	28	28	28	40	40	28	40	40	40
On-Chip RAM (Bytes)	240	64	112	64	64	112	64	64	112	96
On-Chip User ROM (Bytes)	4096	1100	1100	1796	2048	3776	2508	2048	3776	6144
External Bus	Yes	None	None	None	None	None	None	None	None	None
Bidirectional I/O Lines	29	20	20	20	24	24	19	24	24	32
Unidirectional I/O Lines	None	None	None	None	6 Inputs	6 Inputs	None	8 Inputs	8 Inputs	None
Other I/O Features	Timer,UART	Timer	Timer	Timer	Timer,A/D	Timer,A/D	Timer, PLL	Timer	Timer	Timer,D/A
External Interrupt Inputs	3	1	1	1	2	2	2	2	2	1
STOP and WAIT	No	No	No	No	No	No	No	No	No	No

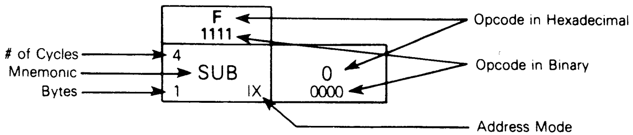
TABLE 7 – 6805 HMOS FAMILY INSTRUCTION SET OPCODE MAP

	Bit Manipulation		Branch	Read-Modify-Write					Control		Register/Memory						
	BTB	BSC	REL	DIR	INH	INH	IX1	IX	INH	INH	IMM	DIR	EXT	IX2	IX1	IX	
Hi / Low	0 0000	1 0001	2 0010	3 0011	4 0100	5 0101	6 0110	7 0111	8 1000	9 1001	A 1010	B 1011	C 1100	D 1101	E 1110	F 1111	Hi / Low
0 0000	10 BRSET0 3 BTB	7 BSET0 2 BSC	4 BRA 2 REL	6 NEG 2 DIR	4 NEG 1 INH	4 NEG 1 INH	7 NEG 2 IX1	6 NEG 1 IX	9 RTI 1 INH		2 SUB 2 IMM	4 SUB 2 DIR	5 SUB 3 EXT	6 SUB 3 IX2	5 SUB 2 IX1	4 SUB 1 IX	0 0000
1 0001	10 BRCLR0 3 BTB	7 BCLR0 2 BSC	4 BRN 2 REL						6 RTS 1 INH		2 CMP 2 IMM	4 CMP 2 DIR	5 CMP 3 EXT	6 CMP 3 IX2	5 CMP 2 IX1	4 CMP 1 IX	1 0001
2 0010	10 BRSET1 3 BTB	7 BSET1 2 BSC	4 BHI 2 REL								2 SBC 2 IMM	4 SBC 2 DIR	5 SBC 3 EXT	6 SBC 3 IX2	5 SBC 2 IX1	4 SBC 1 IX	2 0010
3 0011	10 BRCLR1 3 BTB	7 BCLR1 2 BSC	4 BLS 2 REL	6 COM 2 DIR	4 COMA 1 INH	4 COMX 1 INH	7 COM 2 IX1	6 COM 1 IX	11 SWI 1 INH		2 CPX 2 IMM	4 CPX 2 DIR	5 CPX 3 EXT	6 CPX 3 IX2	5 CPX 2 IX1	4 CPX 1 IX	3 0011
4 0100	10 BRSET2 3 BTB	7 BSET2 2 BSC	4 BCC 2 REL	6 LSR 2 DTR	4 LSRA 1 INH	4 LSRX 1 INH	7 LSR 2 IX1	6 LSR 1 IX			2 AND 2 IMM	4 AND 2 DIR	5 AND 3 EXT	6 AND 3 IX2	5 AND 2 IX1	4 AND 1 IX	4 0100
5 0101	10 BRCLR2 3 BTB	7 BCLR2 2 BSC	4 BCS 2 REL								2 BIT 2 IMM	4 BIT 2 DIR	5 BIT 3 EXT	6 BIT 3 IX2	5 BIT 2 IX1	4 BIT 1 IX	5 0101
6 0110	10 BRSET3 3 BTB	7 BSET3 2 BSC	4 BNE 2 REL	6 ROR 2 DIR	4 RORA 1 INH	4 RORX 1 INH	7 ROR 2 IX1	6 ROR 1 IX			2 LDA 2 IMM	4 LDA 2 DIR	5 LDA 3 EXT	6 LDA 3 IX2	5 LDA 2 IX1	4 LDA 1 IX	6 0110
7 0111	10 BRCLR3 3 BTB	7 BCLR3 2 BSC	4 BEQ 2 REL	6 ASR 2 DIR	4 ASRA 1 INH	4 ASRX 1 INH	7 ASR 2 IX1	6 ASR 1 IX		2 TAX 1 INH		5 STA 2 DIR	6 STA 3 EXT	7 STA 3 IX2	6 STA 2 IX1	5 STA 1 IX	7 0111
8 1000	10 BRSET4 3 BTB	7 BSET4 2 BSC	4 BHCC 2 REL	6 LSL 2 DIR	4 LSLA 1 INH	4 LSLX 1 INH	7 LSL 2 IX1	6 LSL 1 IX		2 CLC 1 INH	2 EOR 2 IMM	4 EOR 2 DIR	5 EOR 3 EXT	6 EOR 3 IX2	5 EOR 2 IX1	4 EOR 1 IX	8 1000
9 1001	10 BRCLR4 3 BTB	7 BCLR4 2 BSC	4 BHCS 2 REL	6 ROL 2 DIR	4 ROLA 1 INH	4 ROLX 1 INH	7 ROL 2 IX1	6 ROL 1 IX		2 SEC 1 INH	2 ADC 2 IMM	4 ADC 2 DIR	5 ADC 3 EXT	6 ADC 3 IX2	5 ADC 2 IX1	4 ADC 1 IX	9 1001
A 1010	10 BRSET5 3 BTB	7 BSET5 2 BSC	4 BPL 2 REL	6 DEC 2 DIR	4 DECA 1 INH	4 DECX 1 INH	7 DEC 2 IX1	6 DEC 1 IX		2 CLI 1 INH	2 ORA 2 IMM	4 ORA 2 DIR	5 ORA 3 EXT	6 ORA 3 IX2	5 ORA 2 IX1	4 ORA 1 IX	A 1010
B 1011	10 BRCLR5 3 BTB	7 BCLR5 2 BSC	4 BMI 2 REL							2 SEI 1 INH	2 ADD 2 IMM	4 ADD 2 DIR	5 ADD 3 EXT	6 ADD 3 IX2	5 ADD 2 IX1	4 ADD 1 IX	B 1011
C 1100	10 BRSET6 3 BTB	7 BSET6 2 BSC	4 BMC 2 REL	6 INC 2 DIR	4 INCA 1 INH	4 INCX 1 INH	7 INC 2 IX1	6 INC 1 IX		2 RSP 1 INH		3 JMP 2 DIR	4 JMP 3 EXT	5 JMP 3 IX2	4 JMP 2 IX1	3 JMP 1 IX	C 1100
D 1101	10 BRCLR6 3 BTB	7 BCLR6 2 BSC	4 BMS 2 REL	6 TST 2 DIR	4 TSTA 1 INH	4 TSTX 1 INH	7 TST 2 IX1	6 TST 1 IX		2 NOP 1 INH	8 BSR 2 REL	7 JSR 2 DIR	8 JSR 3 EXT	9 JSR 3 IX2	8 JSR 2 IX1	7 JSR 1 IX	D 1101
E 1110	10 BRSET7 3 BTB	7 BSET7 2 BSC	4 BIL 2 REL								2 LDX 2 IMM	4 LDX 2 DIR	5 LDX 3 EXT	6 LDX 3 IX2	5 LDX 2 IX1	4 LDX 1 IX	E 1110
F 1111	10 BRCLR7 3 BTB	7 BCLR7 2 BSC	4 BIH 2 REL	6 CLR 2 DIR	4 CLRA 1 INH	4 CLRX 1 INH	7 CLR 2 IX1	6 CLR 1 IX		2 TXA 1 INH		5 STX 2 DIR	6 STX 3 EXT	7 STX 3 IX2	6 STX 2 IX1	5 STX 1 IX	F 1111

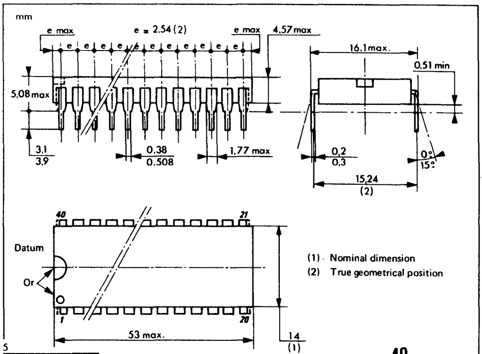
Abbreviations for Address Modes

INH	Inherent
IMM	Immediate
DIR	Direct
EXT	Extended
REL	Relative
BSC	Bit Set/Clear
BTB	Bit Test and Branch
IX	Indexed (No Offset)
IX1	Indexed, 1 Byte (8-Bit) Offset
IX2	Indexed, 2 Byte (16-Bit) Offset

LEGEND

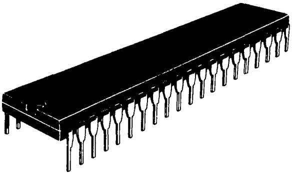


PHYSICAL DIMENSIONS



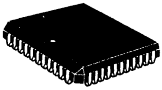
DIN				
A51F CEI	D.A.T.A.	JEDEC	F-119 SITELESC	CB-182

CB-182

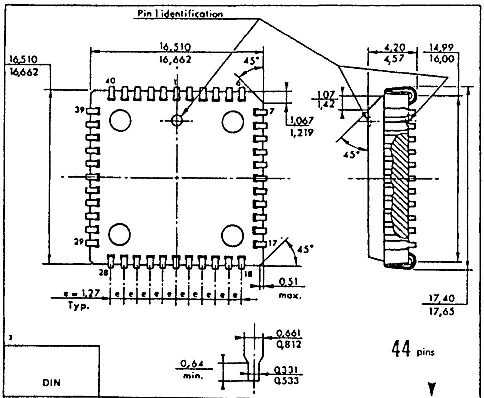


P SUFFIX
PLASTIC PACKAGE

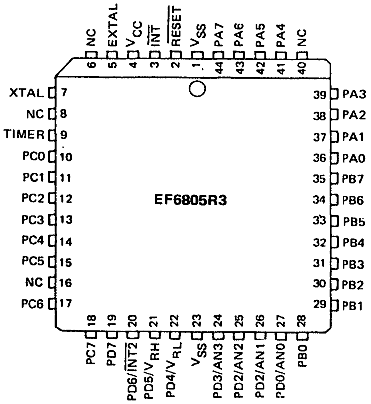
CB-521



FN SUFFIX
PLCC 44



DIN				
CEI	D.A.T.A.	MO-047-AC JEDEC	SITELESC	CB-521



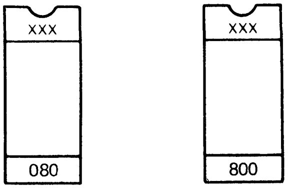
ORDERING INFORMATION

The information required when ordering a custom MCU is listed below. The ROM program may be transmitted to THOMSON SEMICONDUCTEURS on EPROM(s) or an EFDOS/MDOS* disk file.

To initiate a ROM pattern for the MCU, it is necessary to first contact your local THOMSON SEMICONDUCTEURS representative or distributor.

EPROMs

Two 2716 or one 2732 type EPROMs, programmed with the customer program (positive logic sense for address and data), may be submitted for pattern generation. The EPROM must be clearly marked to indicate which EPROM corresponds to which address space. The recommended marking procedure is illustrated below :



XXX = Customer ID

After the EPROM(s) are marked, they should be placed in conductive IC carriers and securely packed. Do not use styrofoam.

VERIFICATION MEDIA

All original pattern media (EPROMs or floppy disk) are filed for contractual purposes and are not returned. A computer listing of the ROM code will be generated and returned along with a listing verification form. The listing should be thoroughly checked and the verification form completed, signed, and returned to THOMSON SEMICONDUCTEURS. The signed verification form constitutes the contractual agreement for creation of the customer mask. If desired, THOMSON SEMICONDUCTEURS will program on blank EPROM from the data file used to create the custom mask and aid in the verification process.

ROM VERIFICATION UNITS (RVUs)

Ten MCUs containing the customer's ROM pattern will be sent for program verification. These units will have been made using the custom mask but are for the purpose of ROM verification only. For expediency they are usually unmarked, packaged in ceramic, and tested only at room temperature and 5 volts. These RVUs are included in the mask charge and are not production parts. The RVUs are thus not guaranteed by THOMSON SEMICONDUCTEURS. Quality Assurance, and should be discarded after verification is completed.

FLEXIBLE DISKS

The disk media submitted must be single-sided, EFDOS/MDOS* compatible floppies.

The customer must write the binary file name and company name on the disk with a felt-tip-pen. The minimum EFDOS/MDOS* system files, as well as the absolute binary object file (Filename .LO type of file) from the 6805 cross assembler, must be on the disk. An object file made from a memory dump using the ROLLOUT command is also acceptable. Consider submitting a source listing as well as the following files : filename .LX (DEVICE/EXORciser loadable format) and filename .SA (ASCII Source Code). These files will of course be kept confidential and are used 1) to speed up the process in-house if any problems arise, and 2) to speed up the user-to-factory interface if the user finds any software errors and needs assistance quickly from THOMSON SEMICONDUCTEURS factory representatives.

EFDOS is THOMSON SEMICONDUCTEURS' Disk Operating System available on development systems such as DEVICE,...

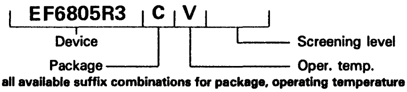
MDOS* is MOTOROLA's Disk Operating System available on development systems such as EXORciser,...

*Requires prior factory approval.

Whenever ordering a custom MCU is required, please contact your local THOMSON SEMICONDUCTEURS representative or THOMSON SEMICONDUCTEURS distributor and/or complete and send the attached "MCU customer ordering sheet" to your local THOMSON SEMICONDUCTEURS representative.

EXORciser is a registered trademark of MOTOROLA Inc.

ORDERING INFORMATION



The table below horizontally shows all available suffix combinations for package, operating temperature and screening level. Other possibilities on request.

DEVICE	PACKAGE					OPER. TEMP			SCREENING LEVEL			
	C	J	P	E	FN	L*	V	T	Std	D	G/B	B/B
EF6805R3			●		●	●	●	●	●	●		

Examples : EF6805R3P, EF6805R3FN, EF6805R3PV, EF6805R3FNV

Package : C : Ceramic DIL, **J :** Cerdip DIL, **P :** Plastic DIL, **E :** LCCC, **FN :** PLCC.
Oper. temp. : L* : 0°C to +70°C, **V :** −40°C to +85°C, T : - 40°C to + 105°C, * : may be omitted.
Screening level : Std : (no-end suffix), **D :** NFC 96883 level D,
G/B : NFC 96883 level G, **B/B :** NFC 96883 level B and MIL-STD-883C level B.

These specifications are subject to change without notice.
Please inquire with our sales offices about the availability of the different products.

Printed in France

Ref.DSEF6805R3

THOMSON
COMPONENTS
SEMICONDUCTEURS

EF6805U2

8-BIT MICROCOMPUTER

ADVANCE INFORMATION

The EF6805U2 Microcomputer Unit (MCU) is a member of the 6805 Family of low-cost single-chip Microcomputers. The 8-bit microcomputer contains a CPU, on-chip CLOCK, ROM, RAM, I/O, and TIMER. It is designed for the user who needs an economical microcomputer with the proven capabilities of the 6800-based instruction set. A comparison of the key features of several members of the 6805 Family of microcomputers is shown at the end of this data sheet. The following are some of the hardware and software highlights of the EF6805U2 MCU.

HARDWARE FEATURES

- 32 TTL/CMOS Compatible I/O Lines
  24 Bidirectional (8 Lines are LED Compatible)
  8 Input-Only
- 2048 Bytes of User ROM
- 64 Bytes of RAM
- Self-Check Mode
- Zero-Crossing Detect/Interrupt
- Internal 8-Bit Timer with 7-Bit Mask Programmable Prescaler and Clock Source
- 5 V Single Supply

SOFTWARE FEATURES

- 10 Powerful Addressing Modes
- Byte Efficient Instruction Set with True Bit Manipulation, Bit Test, and Branch Instructions
- Single Instruction Memory Examine/Change
- Powerful Indexed Addressing for Tables
- Full Set of Conditional Branches
- Memory Usable as Register/Flags
- Complete Development System Support on DEVICE®.

USER SELECTABLE OPTIONS

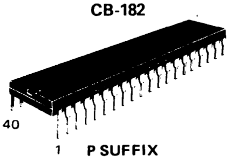
- Internal 8-Bit Timer with Selectable Clock Source (External Timer Input or Internal Machine Clock)
- Timer Prescaler Option (7 Bits, 2^n)
- 8 Bidirectional I/O Lines with TTL or TTL/CMOS Interface Option
- 8 Bidirectional I/O Lines with TTL or Open-Drain Interface Option
- Crystal or Low-Cost Resistor Oscillator Option
- Low Voltage Inhibit Option
- Vectored Interrupts Timer, Software, and External
- User Callable Self-Check Subroutines

DEVICE® is THOMSON SEMICONDUCTEURS' development/emulation tool.

HMOS

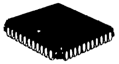
CASES

CB-182



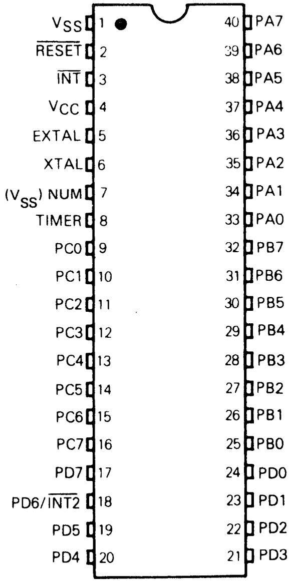
P SUFFIX
PLASTIC PACKAGE

CB-521



L SUFFIX
PLCC44

PIN ASSIGNMENT



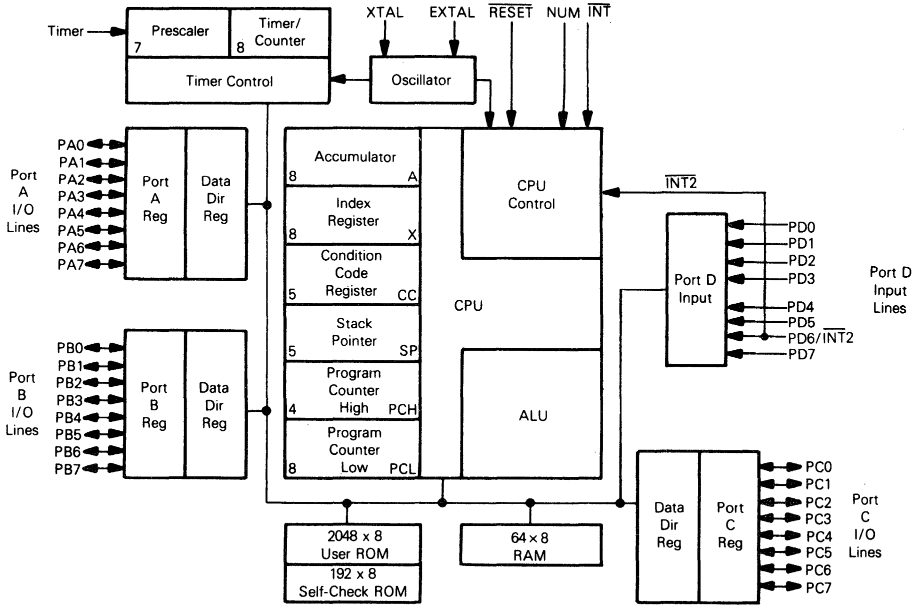


FIGURE 1 – EF6805U2 HMOS MICROCOMPUTER BLOCK DIAGRAM

MAXIMUM RATING

Rating	Symbol	Value	Unit
Supply Voltage	V_{CC}	–0.3 to + 7.0	V
Input Voltage (Except TIMER in Self-Check Mode and Open-Drain Inputs)	V_{in}	–0.3 to + 7.0	V
Input Voltage (Open-Drain Pins, TIMER Pin in Self-Check Mode)	V_{in}	–0.3 to + 15.0	V
Operating Temperature Range V suffix (T_L to T_H) T suffix	T_A	0 to + 70 –40 to + 85 –40 to + 105	°C
Storage Temperature Range	T_{stg}	–55 to + 150	°C
Junction Temperature Plastic Package PLCC	T_J	 150 150	°C

This device contains circuitry to protect the inputs against damage due to high static voltages or electrical fields; however, it is advised that normal precautions be taken to avoid application of any voltage higher than maximum rated voltages to this high-impedance circuit. For proper operation it is recommended that V_{in} and V_{out} be constrained to the range $V_{SS} \leq (V_{in}$ or $V_{out}) \leq V_{CC}$. Reliability of operation is enhanced if unused inputs except EXTAL are tied to an appropriate logic voltage level (e.g., either V_{SS} or V_{CC}).

THERMAL CHARACTERISTICS

Characteristic	Symbol	Value	Unit
Thermal Resistance Plastic PLCC	θ_{JA}	 50 80	°C/W

POWER CONSIDERATIONS

The average chip-junction temperature, T_J, in °C can be obtained from:

$$T_J = T_A + (P_D \bullet \theta_{JA}) \qquad (1)$$

Where:

T_A = Ambient Temperature, °C

θ_{JA} = Package Thermal Resistance, Junction-to-Ambient, °C/W

$P_D = P_{INT} + P_{PORT}$

$P_{INT} = I_{CC} \times V_{CC}$, Watts – Chip Internal Power

P_{PORT} = Port Power Dissipation, Watts – User Determined

For most applications $P_{PORT} \ll P_{INT}$ and can be neglected. P_{PORT} may become significant if the device is configured to drive Darlington bases or sink LED loads.

An approximate relationship between P_D and T_J (if P_{PORT} is neglected) is:

$$P_D = K \div (T_J + 273°C) \qquad (2)$$

Solving equations 1 and 2 for K gives:

$$K = P_D \bullet (T_A + 273°C) + \theta_{JA} \bullet P_D^2 \qquad (3)$$

Where K is a constant pertaining to the particular part. K can be determined from equation 3 by measuring P_D (at equilibrium) for a known T_A. Using this value of K the values of P_D and T_J can be obtained by solving equations (1) and (2) iteratively for any value of T_A.

ELECTRICAL CHARACTERISTICS (V_{CC} = + 5.25 Vdc ± 0.5 Vdc, V_{SS} = 0 Vdc, $T_A = T_L$ to T_H Unless Otherwise Noted)

Characteristic	Symbol	Min	Typ	Max	Unit
Input High Voltage	V_{IH}				V
$\overline{\text{RESET}}$ $(4.75 \leq V_{CC} \leq 5.75)$		4.0	–	V_{CC}	
$(V_{CC} < 4.75)$		$V_{CC} - 0.5$	–	V_{CC}	
$\overline{\text{INT}}$ $(4.75 \leq V_{CC} \leq 5.75)$		4.0	*	V_{CC}	
$(V_{CC} < 4.75)$		$V_{CC} - 0.5$	*	V_{CC}	
All Other (Except Timer)		2.0	–	V_{CC}	
Input High Voltage Timer	V_{IH}				V
Timer Mode		2.0	–	$V_{CC} + 1.0$	
Self-Check Mode		9.0	10.0	15.0	
Input Low Voltage	V_{IL}				V
$\overline{\text{RESET}}$		V_{SS}	–	0.8	
$\overline{\text{INT}}$		V_{SS}	*	1.5	
All Other		V_{SS}	–	0.8	
$\overline{\text{RESET}}$ Hysteresis Voltages (See Figures 10,11 and 12)					V
"Out of Reset"	V_{IRES+}	2.1	–	4.0	
"Into Reset"	V_{IRES-}	0.8	–	2.0	
$\overline{\text{INT}}$ Zero Crossing Input Voltage, Through a Capacitor	V_{INT}	2	–	4	$V_{ac\ p\text{-}p}$
Power Dissipation – (No Port Loading, V_{CC} = 5.75 V)	P_D				mW
$T_A = 0°C$		–	520	740	
$T_A = -40°C$		–	580	800	
Input Capacitance	C_{in}				pF
EXTAL		–	25	–	
All Other		–	10	–	
Low Voltage Recover	V_{LVR}	–	–	4.75	V
Low Voltage Inhibit	V_{LVI}	2.75	3.75	4.70	V
Input Current	I_{in}				µA
TIMER (V_{in} = 0.4 V)		–	–	20	
$\overline{\text{INT}}$ (V_{in} = 2.4 V to V_{CC})		–	20	50	
EXTAL (V_{in} = 2.4 V to V_{CC} - Crystal Option)		–	–	10	
(V_{in} = 0.4 V - Crystal Option)		–	–	–1600	
$\overline{\text{RESET}}$ (V_{in} = 0.8 V) - External Capacitor Charging Current		–4.0	– – –	–40	

*Due to internal biasing this input (when unused) floats to approximately 2.2 V.

SWITCHING CHARACTERISTICS

(V_{CC} =+ 5.25 Vdc ± 0.5 Vdc, V_{SS} = 0 Vdc, T_A = T_L to T_H Unless Otherwise Noted)

Characteristic	Symbol	Min	Typ	Max	Unit
Oscillator Frequency	f_{osc}	0.4	–	4.2	MHz
Cycle Time (4/f_{osc})	t_{cyc}	0.95	–	10	µs
$\overline{INT}$, $\overline{INT2}$, and TIMER Pulse Width (See Interrupt Section)	t_{WL}, t_{WH}	t_{cyc}+ 250	–	–	ns
$\overline{RESET}$ Pulse Width	t_{RWL}	t_{cyc}+ 250	–	–	ns
$\overline{INT}$ Zero-Crossing Detection Input Frequency	f_{INT}	0.03	–	1	kHz
External Clock Input Duty Cycle (EXTAL)	–	40	50	60	%
Crystal Oscillator Start-Up Time *	–	–	–	100	ms

* See Figure 16 for typical crystal parameters

PORT ELECTRICAL CHARACTERISTICS

(V_{CC} =+ 5.25 Vdc ± 0.5 Vdc, V_{SS} = 0 Vdc, T_A = T_L to T_H Unless Otherwise Noted)

Characteristic	Symbol	Min	Typ	Max	Unit
Port A with CMOS Drive Enabled					
Output Low Voltage (I_{Load}= 1.6 mA)	V_{OL}	–	–	0.4	V
Output High Voltage I_{Load}= −100 µA I_{Load}= −10 µA	V_{OH}	 2.4 V_{CC} − 1.0	 – –	 – –	V
Input High Voltage (I_{Load}= −300 µA max.)	V_{IH}	2.0	–	V_{CC}	V
Input Low Voltage (I_{Load}= −500 µs max.)	V_{IL}	V_{SS}	–	0.8	V
High Z State Input Current (V_{in}= 2.0 V to V_{CC})	I_{IH}	–	–	−300	µA
High Z State Input Current (V_{in}= 0.4 V)	I_{IL}	–	–	−500	µA
Port B					
Output Low Voltage I_{Load}= 3.2 mA I_{Load}= 10 mA (Sink)	V_{OL}	 – –	 – –	 0.4 1.0	V
Output High Voltage I_{Load}= −200 µA	V_{OH}	2.4	–	–	V
Darlington Current Drive (Source) V_O= 1.5 V	I_{OH}	−1.0	–	−10	mA
Input High Voltage	V_{IH}	2.0	–	V_{CC}	V
Input Low Voltage	V_{IL}	V_{SS}	–	0.8	V
High Z State Input Current	I_{TSI}	–	<2	10	µA
Port C and Port A with CMOS Drive Disabled					
Output Low Voltage I_{Load}= 1.6 mA	V_{OL}	–	–	0.4	V
Output High Voltage I_{Load}= −100 µA	V_{OH}	2.4	–	–	V
Input High Voltage	V_{IH}	2.0	–	V_{CC}	V
Input Low Voltage	V_{IL}	V_{SS}	–	0.8	V
High Z State Input Current	I_{TSI}	–	<2	10	µs
Port C (Open-Drain Option)					
Input High Voltage	V_{IH}	2.0	–	13.0	V
Input Low Voltage	V_{IL}	V_{SS}	–	0.8	V
Input Leakage Current	I_{LOD}	–	<3	15	µA
Output Low Voltage I_{Load}= 1.6 mA	V_{OL}	–	–	0.4	V
Port D					
Input High Voltage	V_{IH}	2.0	–	V_{CC}	V
Input Low Voltage	V_{IL}	V_{SS}	–	0.8	V
Input Current	I_{in}	–	<1	5	µA

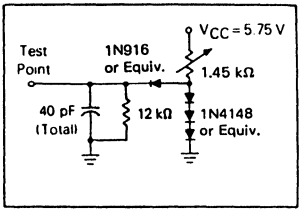


FIGURE 2 – TTL EQUIVALENT TEST LOAD (PORT B)

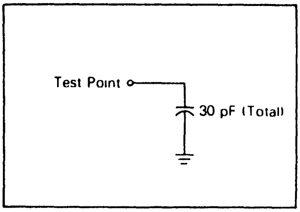


FIGURE 3 – CMOS EQUIVALENT TEST LOAD (PORT A)

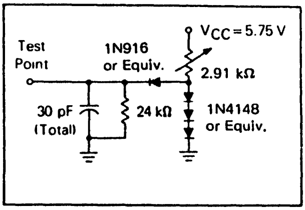


FIGURE 4 – TTL EQUIVALENT TEST LOAD (PORTS A AND C)

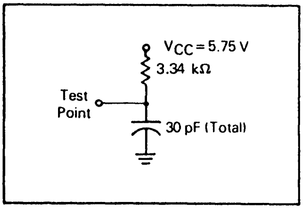


FIGURE 5 – OPEN - DRAIN EQUIVALENT TEST LOAD (PORT C)

SIGNAL DESCRIPTION

The input and output signals for the MCU, shown in Figure 1, are described in the following paragraphs.

V_{CC} AND V_{SS} – Power is supplied to the MCU using these two pins. V_{CC} is power and V_{SS} is the ground connection.

$\overline{\text{INT}}$ – This pin provides the capability for asynchronously applying an external interrupt to the MCU. Refer to Interrupts Section for additional information.

XTAL AND EXTAL – These pins provide control input for the on-chip clock oscillator circuit. A crystal, a resistor, or an external signal, depending on user selectable manufacturing mask option, can be connected to these pins to provide a system clock with various degrees of stability/cost tradeoffs. Lead length and stray capacitance on these two pins should be minimized. Refer to Internal Clock Generator Options Section for recommendations about these inputs.

NUM (NOM USER MODE) - This pin is not for user application and must be connected to V_{SS}.

TIMER – The pin allows an external input to be used to control the internal timer circuitry and also to initiate the self test program. Refer to Timer Section for additional information about the timer circuitry.

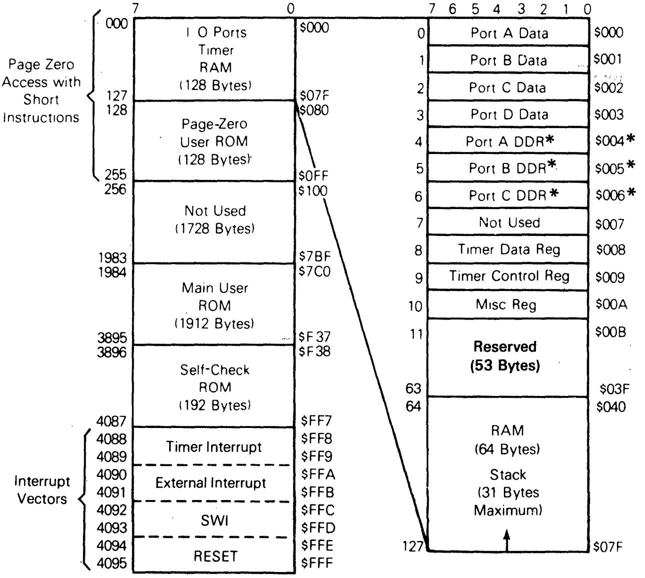
$\overline{\text{RESET}}$ – This pin allows resetting of the MCU at times other than the automatic resetting capability already in the MCU. The MCU can be reset by pulling $\overline{\text{RESET}}$ low. Refer to Resets Section for additional information.

INPUT/OUTPUT LINES (PA0-PA7, PB0-PB7, PC0-PC7, PD0-PD7) – These 32 lines are arranged into four 8-bit ports (A, B, C, and D). Ports A, B, and C are programmable as either inputs or outputs under software control of the data direction registers (DDRs). Port D is for digital Input only and bit 6 may be used for a second interrupt INT2. Refer to Input/Output Section and Interrupts Section for additional information.

MEMORY — The MCU is capable of addressing 4096 bytes of memory and I/O registers with its program counter. The EF6805U2 MCU has implemented 2314 of these bytes. This consists of: 2048 user ROM bytes, 192 self-check ROM bytes, 64 user RAM bytes, 7 port I/O bytes, 2 timer registers, and a miscellaneous register; see Figure 6 for the Address map. The user ROM has been split into three areas. The main user ROM area is from $080 to $OFF and from $7C0 to $F37. The last 8 user ROM locations at the bottom of memory are for the interrupt vectors.

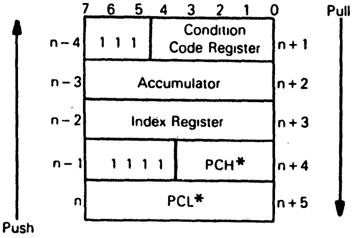
The MCU reserves the first 16 memory locations for I/O features, of which 10 have been implemented. These locations are used for the ports, the port DDRs, the timer and the INT2 miscellaneous register, and the 64 RAM bytes, 31 bytes are shared with the stack area. The stack must be used with care when data shares the stack area.

The shared stack area is used during the processing of an interrupt or subroutine calls to save the contents of the CPU state. The register contents are pushed onto the stack in the order shown in Figure 7. Since the stack pointer decrements during pushes, the low order byte (PCL) of the program.



*Caution: Data Direction Registers (DDRs) are write-only; they read as $FF.

FIGURE 6 – EF6805U2 MCU ADDRESS MAP



*For subroutine calls, only PCH and PCL are stacked

FIGURE 7 – INTERRUPT STACKING ORDER

CENTRAL PROCESSING UNIT

The CPU of the EF6805 Family is implemented independently from the I/O or memory configuration. Consequently, it can be treated as an independent central processor communicating with I/O and memory via internal address, data, and control buses.

REGISTERS

The 6805 Family CPU has five registers available to the programmer. They are shown in Figure 8 and are explained in the following paragraphs.

ACCUMULATOR (A) – The accumulator is a general purpose 8-bit register used to hold operands and results of arithmetic calculations or data manipulations.

INDEX REGISTER (X) – The index register is an 8-bit register used for the indexed addressing mode. It contains an 8-bit value that may be added to an instruction value to create an effective address. The index register can also be used for data manipulations using the read-modify-write instructions. The Index Register may also be used as a temporary storage area.

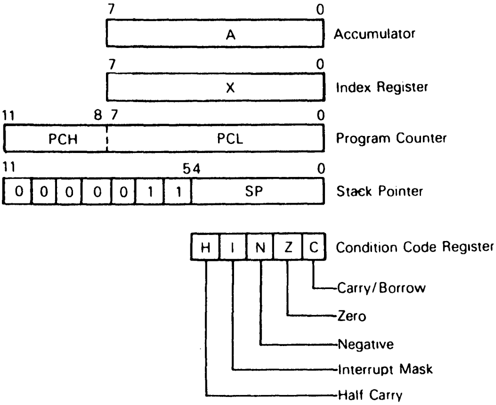


FIGURE 8 – PROGRAMMING MODEL

PROGRAM COUNTER (PC) – The Program Counter is a 12 bit register that contains the address of the next instruction to be executed.

STACK POINTER (SP) – The stack pointer is a 12-bit register that contains the address of the next free location on the stack. During an MCU reset or the reset stack pointer (RSP) instruction, the stack pointer is set to location $07F. The stack pointer is then decremented as data is pushed onto the stack and incremented as data is then pulled from the stack. The seven most significant bits of the stack pointer are permanently set to 0000011. Subroutines and interrupts may be nested down to location $061 (31 bytes maximum) which allows the programmer to use up to 15 levels of subroutine calls (less if interrupts are allowed).

CONDITION CODE REGISTER (CC) – The condition code register is a 5-bit register in which four bits are used to indicate the results of the instruction just executed. These bits can be individually tested by a program and specific action taken as a result of their state. Each bit is explained in the following paragraphs.

Half Carry (H) – Set during ADD and ADC operations to indicate that a carry occurred between bits 3 and 4.

Interrupt (I) – When this bit is set, the timer and external interrupts ($\overline{\text{INT}}$ and $\overline{\text{INT2}}$) are masked (disabled). If an interrupt occurs while this bit is set, the interrupt is latched and is processed as soon as the interrupt bit is cleared.

Negative (N) – When set, this bit indicates that the result of the last arithmetic, logical, or data manipulation was negative (bit 7 in the result is a logical "1").

Zero (Z) – When set, this bit indicates that the result of the last arithmetic, logical, or data manipulation was zero.

Carry/Borrow (C) – When set, this bit indicates that a carry or borrow out of the Arithmetic Logic Unit (ALU) occurred during the last arithmetic operation. This bit is also affected during bit test and branch instructions plus shifts and rotates.

TIMER

The timer circuitry for the MC6805U2 is shown in Figure 10. The 8-bit counter may be loaded under program control and is decremented toward zero by the clock input (or prescaler output). When the timer reaches zero, the timer interrupt request bit (bit 7) in the timer control register (TCR) is set. The timer interrupt can be masked (disabled) by setting the timer interrupt mask bit (bit 6) in the TCR. The interrupt bit (I bit) in the condition code register also prevents a timer interrupt from being processed. The MCU responds to this interrupt by saving the present CPU state on the stack, fetching the timer interrupt vector from locations $FF8 and $FF9 and executing the interrupt routine (see **RESET, CLOCK, AND INTERRUPT STRUCTURE SECTIONS**). *The timer interrupt request bit must be cleared by software.* The TIMER and INT2 share the same interrupt vector. *The interrupt routine must check the request bits to determine the source of the interrupt.*

The clock input to the timer can be from an external source (decrementing of timer counter occurs on a positive transition of the external source) applied to the TIMER input pin, or it can be the internal phase two signal. Three machine cycles are required for a change in state of the TIMER pin to decrement the timer prescaler. The maximum frequency of a signal that can be recognized by the TIMER pin logic is dependent on the parameter labeled t_{WH}. The pin logic that recognizes the high state on the pin must also recognize the low state on the pin in order to "re-arm" the internal logic. Therefore, the period can be calculated as follows (assumes 50/50 duty cycle for a given period):

$$t_{cyc} \times 2 + 250 \text{ ns} = \text{period} = \frac{1}{\text{freq}}$$

The periods is not simply $t_{WH} + t_{WL}$. This computation is allowable, but it does reduce the maximum allowable frequency by defining an unnecessarily longer period (250 nanoseconds times two).

When the phase two signal is used as the source, it can be gated by an input applied to the TIMER input pin allowing the user to easily perform pulse-width measurements. The source of the clock input is one of the mask options that is specified before manufacture of the MCU.

NOTE

For ungated phase two clock input to the timer prescaler, the TIMER pin should be tied to V_{CC}.

A prescaler option, divide by 2^n, can be applied to the clock input that extends the timing interval up to a maximum of 128 counts before decrementing the counter. This prescaling mask option is also specified before manufacture. To avoid truncation errors, the prescaler is cleared when bit 3 of the timer control register is written to a logic one (this bit always reads a logic zero). See Figure 9.

7	6	5	4	3	2	1	0	
		1	1		1	1	1	$009

TCR7 – Timer Interrupt Request Status Bit: Set when TDR goes to zero; must be cleared by software. Cleared to 0 by Reset.

TCR6 – Timer Interrupt Mask Bit: 1 = timer interrupt masked (disabled). Set to 1 by Reset.

TCR3 – Clear prescaler always reads as a 0; clears prescaler when written to a logic "1".

TCR Bits 5, 4, 2, 1, 0 reads "1s" – unused bits.

FIGURE 9 – TIMER CONTROL REGISTER (TCR)

The timer continues to count past zero, falling through to $FF from $00 and then continuing the countdown. Thus, the counter can be read at any time by reading the timer data register (TDR). This allows a program to determine the length of time since a timer interrupt has occurred, and not disturb the counting process.

At power up or reset, the prescaler and counter are initialized with all logic ones; the timer interrupt request bit (bit 7) is cleared and the timer interrupt mask bit (bit 6) is set.

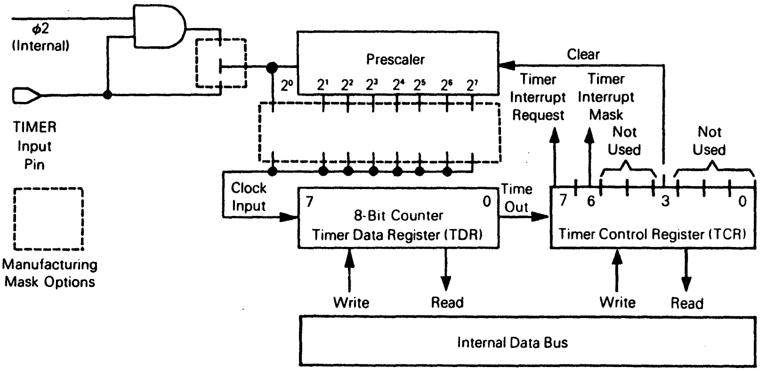


FIGURE 10 – TIMER BLOCK DIAGRAM

SELF-CHECK

The self-check capability of the EF6805U2MCU provides an internal check to determine if the part is functional. Connect the MCU as shown in Figure 11 and monitor the output of Port C bit 3 for an oscillation of approximately 7 Hz. A 10-volt level (through a 10 k resistor) on the timer input, pin 8 and pressing then releasing the RESET button, energizes the ROM-based self-check feature. The self-check program exercises the RAM, ROM, TIMER, interrupts, and I/O ports.

Several of the self-check subroutines can be called by a user program with a JSR or BSR instruction. They are the RAM, ROM, tests. The timer routine may also be called if the timer input is the internal ϕ2 clock.

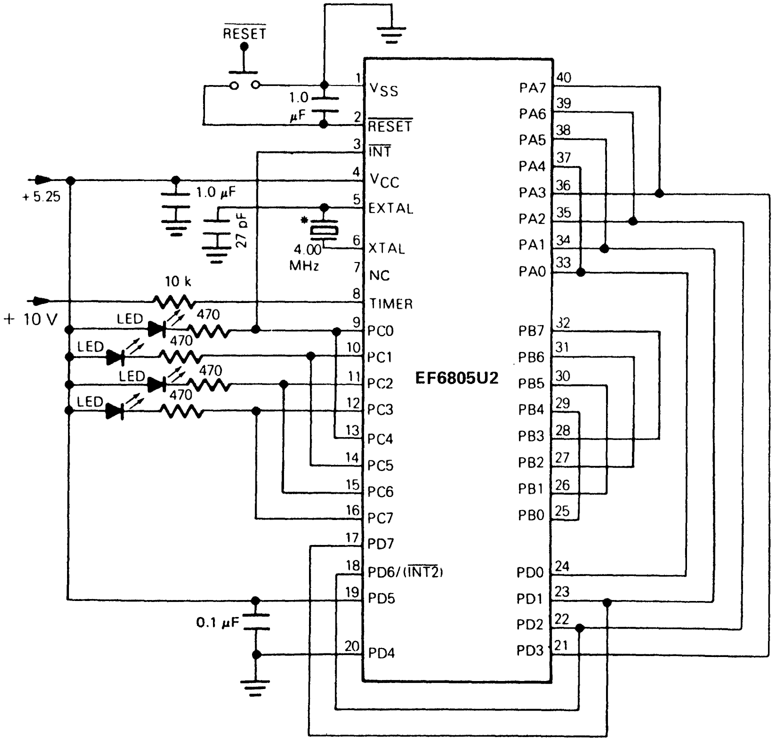
To call those subroutines in customer applications, please contact your local THOMSON SEMICONDUCTEURS sales office in order to obtain the complete description of the self-check program and the entrance/exit conditions.

RAM SELF-CHECK SUBROUTINE – The RAM self-check is called at location $F6F and returns with the Z bit clear if any error is detected ; otherwise the Z bit is set. The walking diagnostic pattern method is used on the EF6805U2.

The RAM test must be called with the stack pointer at $07F. When run, the test checks every RAM cell except for $07F and $07E which are assumed to contain the return address.

The A and X registers and all RAM locations except $07F and $07E are modified.

ROM CHECKSUM SUBROUTINE – The ROM self-check is called at location $F8A. If any error is detected, it returns with the Z bit cleared ; otherwise Z = 1, X = 0 on return, and A is zero if the test passes. RAM location $040 to $043 is overwritten. The checksum is the complement of the execution OR of the contents of the user ROM.



*This connection depends on clock oscillator user selectable mask option. Use jumper if the RC mask option is selected.

FIGURE 11 – SELF-CHECK CONNECTIONS

LED Meanings

PC0	PC1	PC2	PC3	Remarks (1:LED ON; 0:LED OFF)
1	0	1	0	Bad I/O
0	0	1	0	Bad Timer
1	1	0	0	Bad RAM
0	1	0	0	Bad ROM
0	0	0	0	Bad Interrupts or Request Flag
All Flashing				Good Device

Anything else bad Device, Bad Port C, etc.

TIMER SELF-CHECK SUBROUTINE – The timer self-check is called at location $FCF and returns with the Z bit cleared if any error was found; otherwise Z = 1.

In order to work correctly as a user subroutine, the internal φ2 clock must be the clocking source and interrupts must be disabled. Also, on exit, the clock is running and the interrupt mask is not set so the caller must protect from interrupts if necessary.

The A and X register contents are lost. The timer self-check routine counts how many times the clock counts in 128 cycles. The number of counts should be a power of 2 since the prescaler is a power of 2. If not, the timer is probably not counting correctly. The routine also detects a timer which is not running.

RESET

The MCU can be reset three ways : by initial powerup, by the external reset input ($\overline{\text{RESET}}$) and by an optional internal low-voltage detect circuit. The $\overline{\text{RESET}}$ input consists mainly of a Schmitt trigger which senses the $\overline{\text{RESET}}$ line logic level. A typical reset Schmitt trigger hysteresis curve is shown in Figure 12. The Schmitt trigger provides an internal reset voltage if it senses a logical zero on the $\overline{\text{RESET}}$ pin.

Power-On Reset (POR) – An internal reset is generated upon powerup that allows the internal clock generator to stabilize. A delay of t_{RHL} milliseconds is required before allowing the $\overline{\text{RESET}}$ input to go high. Refer to the power and reset timing diagram of Figure 13. Connecting a capacitor to the $\overline{\text{RESET}}$ input (as illustrated in Figure 14) typically provides sufficient delay. During powerup, the Schmitt trigger switches on (removes reset) when $\overline{\text{RESET}}$ rises to V_{IRES+}.

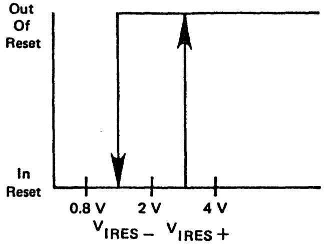


FIGURE 12 – TYPICAL RESET SCHMITT TRIGGER HYSTERESIS

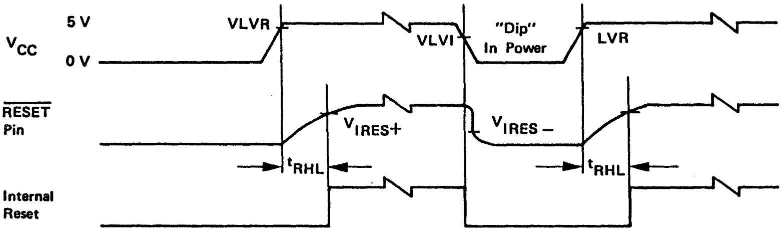


FIGURE 13 – POWER AND RESET TIMING

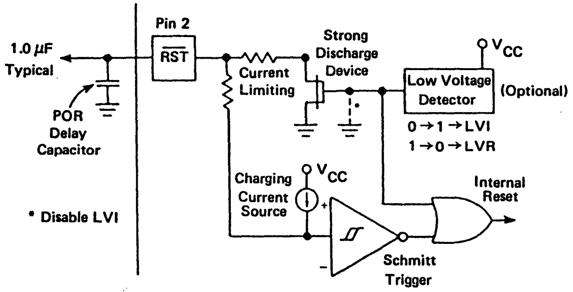


FIGURE 14 – $\overline{\text{RESET}}$ CONFIGURATION

External Reset Input – The MCU will be reset if a logical zero is applied to the $\overline{\text{RESET}}$ input for a period longer than one machine cycle (t_{cyc}). Under this type of reset, the Schmitt trigger switches off at V_{IRES-} to provide an internal reset voltage.

Low-Voltage Inhibit (LVI) – The optional low-voltage detection circuit causes a reset of the MCU if the power supply voltage falls below a certain level (V_{LVI}). The only requirement is that V_{CC} remains at or below the V_{LVI} threshold for one t_{cyc} minimum. In typical applications, the V_{CC} bus filter capacitor will eliminate negative-going voltage glitches of less than one t_{cyc}. The output from the low-voltage detector is connected directly to the internal reset circuitry. It also forces the $\overline{\text{RESET}}$ pin low via a strong discharge device through a resistor. The internal reset will be removed once the power supply voltage rises above a recovery level (V_{LVR}), at which time a normal power-on-reset occurs.

INTERNAL CLOCK GENERATOR OPTIONS

The internal clock generator circuit is designed to require a minimum of external components. A crystal, a resistor, a jumper wire, or an external signal may be used to generate a system clock with various stability/cost trade-offs. The oscillator frequency is internally divided by four to produce the internal system clocks. A manufacturing mask option is used to select crystal or resistor operation.

The different connection methods are shown in Figure 15. Crystal specifications and suggested PC board layouts are given in Figure 16. A resistor selection graph is given in Figure 17.

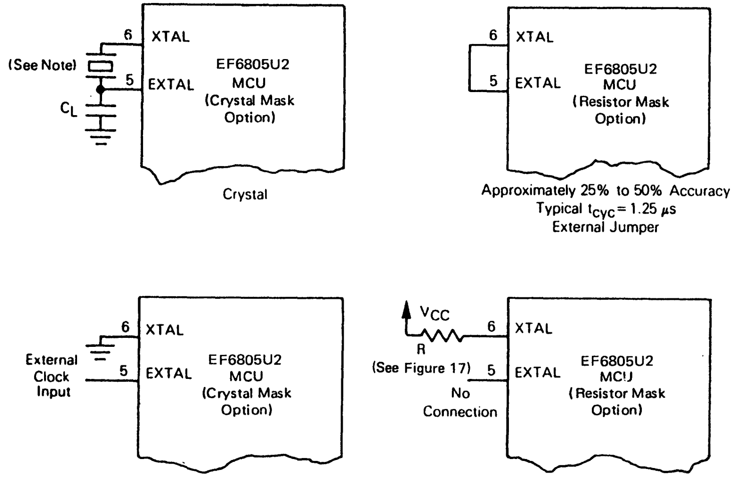
The crystal oscillator start-up time is a function of many variables : crystal parameters (especially R_S), oscillator load capacitances, IC parameters, ambient temperature, and supply voltage. To ensure rapid oscillator start up, neither the crystal characteristics nor the load capacitances should exceed recommendations.

When utilizing the on-board oscillator, the MCU should remain in a reset condition (reset pin voltage below V_{IRES+}) until the oscillator has stabilized at its operating frequency. Several factors are involved in calculating the external reset capacitor required to satisfy this condition: the oscillator start-up voltage, the oscillator stabilization time, the minimum V_{IRES+}, and the reset charging current specification.

Once V_{CC} minimum is reached, the external $\overline{\text{RESET}}$ capacitor will begin to charge at a rate dependent on the capacitor value. The charging current is supplied from V_{CC} through a large resistor, so it appears almost like a constant current source until the reset voltage rises above V_{IRES+}. Therefore, the $\overline{\text{RESET}}$ pin will charge at approximately :

$$(V_{IRES+}) \bullet C_{ext} = I_{RES} \bullet t_{RHL}$$

Assuming the external capacitor is initially discharged.



Crystal

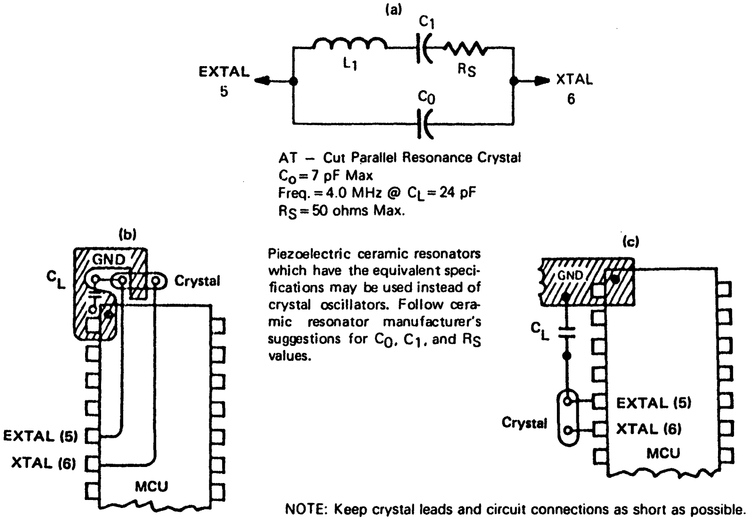
Approximately 25% to 50% Accuracy
Typical t_{cyc} = 1.25 µs
External Jumper

External Clock

Approximately 10% to 25% Accuracy
External Resistor
(Excludes Resistor Tolerance)

NOTE: The recommended C_L value with a 4.0 MHz crystal is 27 pF, maximum, including system distributed capacitance. There is an internal capacitance of approximately 25 pF on the XTAL pin. For crystal frequencies other than 4 MHz, the total capacitance on each pin should be scaled as the inverse of the frequency ratio. For example, with a 2 MHz crystal, use approximately 50 pF on EXTAL and approximately 25 pF on XTAL. The exact value depends on the Motional-Arm parameters of the crystal used.

FIGURE 15 – CLOCK GENERATOR OPTIONS



AT – Cut Parallel Resonance Crystal
$C_0 = 7$ pF Max
Freq. = 4.0 MHz @ $C_L = 24$ pF
$R_S = 50$ ohms Max.

Piezoelectric ceramic resonators which have the equivalent specifications may be used instead of crystal oscillators. Follow ceramic resonator manufacturer's suggestions for C_0, C_1, and R_S values.

NOTE: Keep crystal leads and circuit connections as short as possible.

FIGURE 16 – CRYSTAL MOTIONAL ARM PARAMETERS AND SUGGESTED PC BOARD LAYOUT

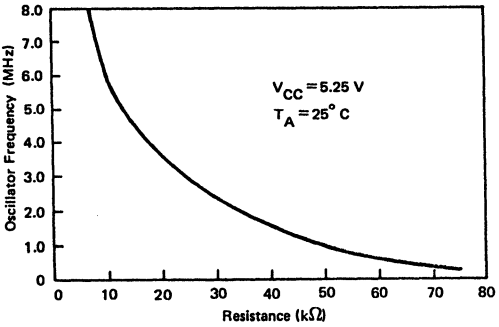


FIGURE 17 – TYPICAL FREQUENCY SELECTION FOR RESISTOR (OSCILLATOR OPTION)

INTERRUPTS

The microcomputers can be interrupted four different ways : through the external interrupt ($\overline{\text{INT}}$) input pin, the internal timer interrupt request, the external port D bit 6 ($\overline{\text{INT2}}$) input pin, or the software interrupt instruction (SWI). When any interrupt occurs : the current instruction (including SWI) is completed, processing is suspended, the present CPU state is pushed onto the stack, the interrupt bit (I) in the condition code register is set, the address of the interrupt routine is obtained from the appropriate interrupt vector address, and the interrupt routine is executed. Stacking the CPU register, setting the I bit, and vector fetching require a total of 11 t_{cyc} periods for completion. A flowchart of the interrupt sequence is shown in Figure 18. The interrupt service routine must end with a return from interrupt (RTI) instruction which allows the MCU to resume processing of the program prior to the interrupt (by unstacking the previous CPU state). Unlike $\overline{\text{RESET}}$, hardware interrupts do not cause the current instruction execution to be halted, but are considered pending until the current instruction execution is complete.

When the current instruction is complete, the processor checks all pending hardware interrupts and if unmasked, proceeds with interrupt processing ; otherwise the next instruction" is fetched and executed. Note that masked interrupts are latched for later interrupt service.

If both an external interrupt and a timer interrupt are pending at the end of an instruction execution, the external interrupt is serviced first. The SWI is executed as any other instruction.

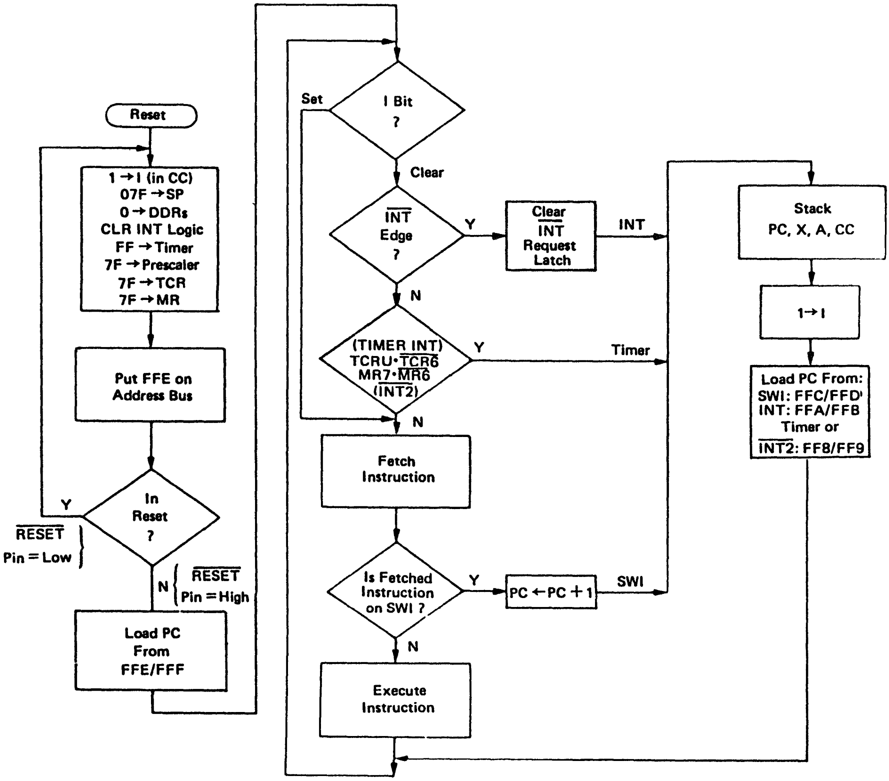

Reset
1 → I (in CC)
07F → SP
0 → DDRs
CLR INT Logic
FF → Timer
7F → Prescaler
7F → TCR
7F → MR
Put FFE on Address Bus
In Reset ?
Y
$\overline{\text{RESET}}$ Pin = Low
N
$\overline{\text{RESET}}$ Pin = High
Load PC From FFE/FFF
Set
I Bit ?
Clear
$\overline{\text{INT}}$ Edge ?
Y
Clear $\overline{\text{INT}}$ Request Latch
INT
N
(TIMER INT) TCRU·$\overline{\text{TCR6}}$ MR7·$\overline{\text{MR6}}$ ($\overline{\text{INT2}}$)
Y
Timer
N
Fetch Instruction
Is Fetched Instruction on SWI ?
Y
PC ← PC + 1
SWI
N
Execute Instruction
Stack PC, X, A, CC
1→ I
Load PC From:
SWI: FFC/FFD
INT: FFA/FFB
Timer or
$\overline{\text{INT2}}$: FF8/FF9

FIGURE 18 – RESET AND INTERRUPT PROCCESSING FLOWCHARD

NOTE

The timer and $\overline{\text{INT2}}$ interrupts share the same vector address. The interrupt routine must determine the source by examining the interrupt request bits (TCR b7 and MR b7). Both TCR b7 and MR b7 can only be written to zero by software.

The external interrupt, $\overline{\text{INT}}$ and $\overline{\text{INT2}}$, are synchronized and then latched on the falling edge of the input signal. The $\overline{\text{INT2}}$ interrupt has an interrupt request bit (bit 7) and a mask bit (bit 6) located in the miscellaneous register (MR). The $\overline{\text{INT2}}$ interrupt is inhibited when the mask bit is set. The $\overline{\text{INT2}}$ is always read as a digital input on port D. The $\overline{\text{INT2}}$ and timer interrupt request bits, if set, cause the MCU to process an interrupt when the condition code I bit is clear.

A sinusoidal input signal (f_{INT} maximum) can be used to generate an external interrupt for use as a zero-crossing detector. This allows applications such as servicing time-of-day routines and engaging/disengaging ac power control devices. Off-chip full wave rectification provides an interrupt at every zero crossing of the ac signal and thereby provides a 2f clock. See Figure 19.

NOTE

The $\overline{\text{INT}}$ (pin 3) is internally biased at approximately 2.2 V due to the internal zero-crossing detection.

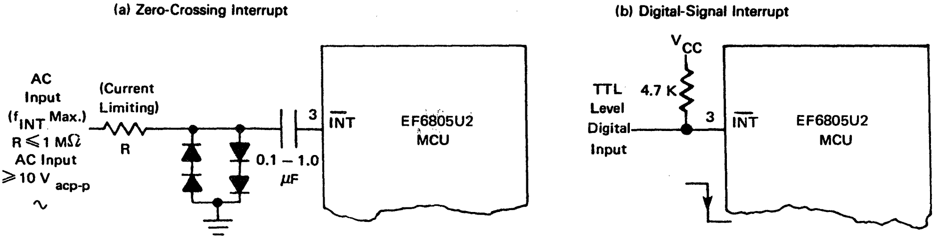
A software interrupt (SWI) is an executable instruction which is executed regardless of the state of the I bit in the condition code register. SWIs are usually used as breakpoints for debugging or as system calls.

A/D CONTROL REGISTER (ACR)

7	6	5	4	3	2	1	0	
	1	1	1	1				$00E

Bit 7 – Conversion Complete Status Flag: Set when conversion is complete; cleared only on a write to ACR.
Readable, not writable.

Bits 2, 1, 0 – A/D input Mux Selection (See Table 2).
Bits 6, 5, 4, 3 read as "1s" – unused bits.


(a) Zero-Crossing Interrupt

(b) Digital-Signal Interrupt

FIGURE 19 – TYPICAL INTERRUPT CIRCUITS

INPUT/OUTPUT CIRCUITRY

There are 32 input/output pins. The $\overline{\text{INT}}$ pin may be polled with branch instructions to provide an additional input pin. All pins on ports A, B, and C are programmable as either inputs or outputs under software control of the corresponding data direction register (DDR). See below I/O port control registers configuration. The port I/O programming is accomplished by writing the corresponding bit in the port DDR to a logic one for output or a logic zero for input. On reset all the DDRs are initialized to a logic zero state, placing the ports in the input mode. The port output registers are not initialized on reset and should be initialized by software before changing the DDRs from input to output. A read operation on a port programmed as an output will read the contents of the output latch regardless of the logic levels at the output pin, due to output loading. Refer to Figure 20.

PORT DATA REGISTER

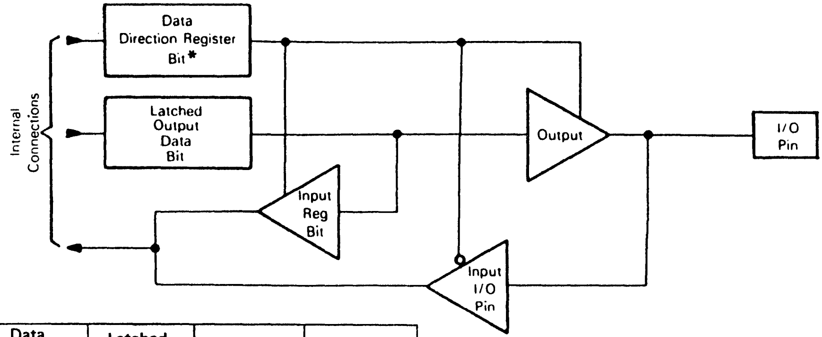
7 … 0

Port A Addr = $000
Port B Addr = $001
Port C Addr = $002
Port D Addr = $003

PORT DATA DIRECTION REGISTER (DDR)

7 … 0

(1) Write Only; reads as all "1s"
(2) 1 = Output, 0 = Input Cleared to 0 by Reset
(3) Port A Addr = $004
Port B Addr = $005
Port C Addr = $006



Data Direction Register Bit	Latched Output Data Bit	Output State	Input To MCU
1	0	0	0
1	1	1	1
0	X	High-Z **	Pin

*DDR is a write-onlv reqister and reads as all "1s".
** Ports B, and C are three-state ports.
Port A has optional internal pullup devices to provide CMOS data drive capability. See Electrical Characteristics tables for complete information.

FIGURE 20 – TYPICAL PORT I/O CIRCUITRY

All input/output lines are TTL compatible as both inputs and outputs. Port A lines are CMOS compatible as outputs (mask option) while port B, C, and D lines are CMOS compatible as inputs. Port D lines are input only; thus, there is no corresponding DDR. When programmed as outputs, port B is capable of sinking 10 milliamperes and sourcing 1 milliampere on each pin.

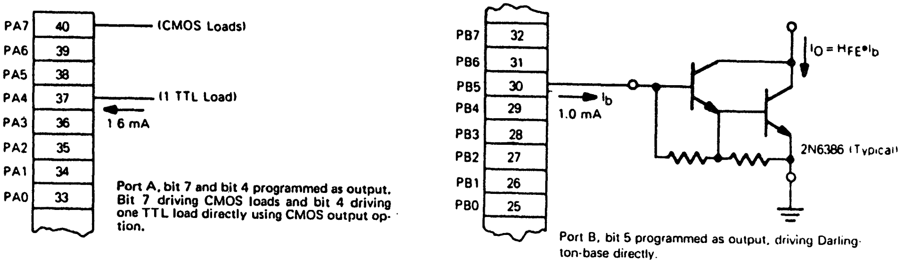
The address map (Figure 6) gives tne aodresses of data registers and data direction registers. Figure 21 provides some examples of port connections.

CAUTION

The corresponding DDRs for ports A, B, and C are write-only registers (registers at $004, $005, $006). A read operation on these registers is undefined. Since BSET and BCLR are read-modify-write in function, they cannot be used to set or clear a single DDR bit (all "unaffected" bits would be set). It is recommended that all DDR bits in a port be written using a single-store instruction.

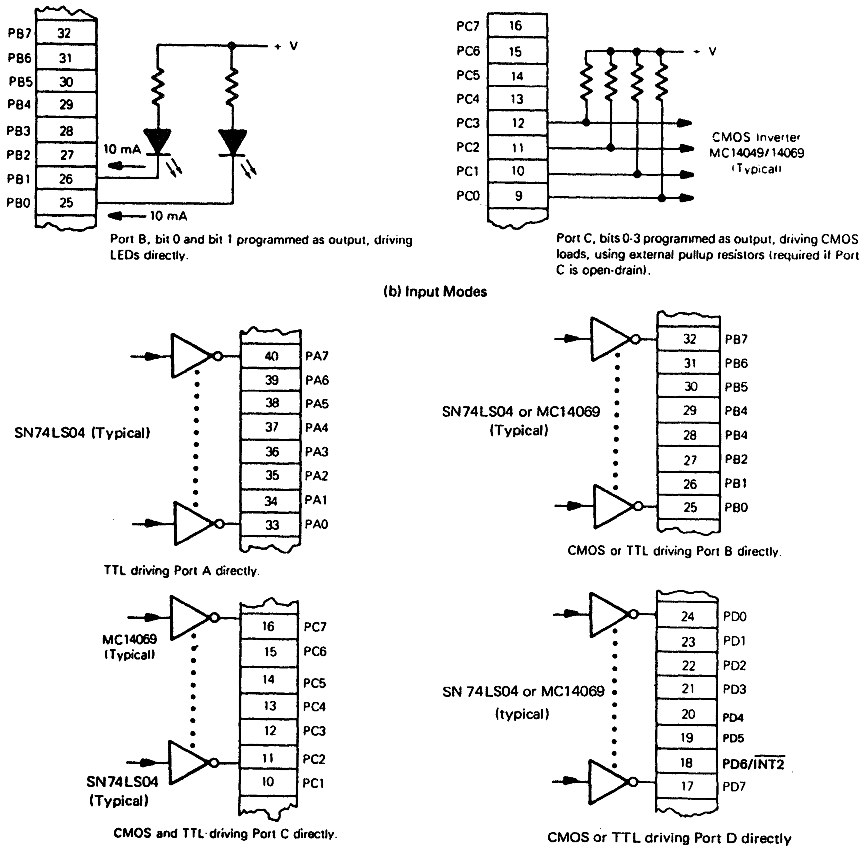
The latched output data bit (see Figure 20) must always be written. Therefore, any write to a port writes all of its data bits even though the port DDR is set to input. This may be used to initialize the data register and avoid undefined outputs ; however, care must be exercised when using read-modify-write instructions, since the data read corresponds to the pin level if the DDR is an input (zero) and corresponds to the latched output data when the DDR is an output (one).

(a) Output Modes



Port A, bit 7 and bit 4 programmed as output. Bit 7 driving CMOS loads and bit 4 driving one TTL load directly using CMOS output option.

Port B, bit 5 programmed as output, driving Darlington-base directly.



Port B, bit 0 and bit 1 programmed as output, driving LEDs directly.

Port C, bits 0-3 programmed as output, driving CMOS loads, using external pullup resistors (required if Port C is open-drain).

(b) Input Modes

TTL driving Port A directly.

CMOS or TTL driving Port B directly.

CMOS and TTL driving Port C directly.

CMOS or TTL driving Port D directly

FIGURE 21 – TYPICAL PORT CONNECTIONS

BIT MANIPULATION

The EF6805U2 as the ability to set or clear any single RAM or I/O bit (except the data direction registers) with a single instruction (BSET, BCLR) (see Caution below). Any bit in page zero can be tested using the BRSET and BRCLR instructions and the program branches as a result of its state. The carry bit equals the value of the bit referencesby BRSET or BRCLR. The capability to working with any bit in RAM, ROM, or I/O allows the user to have individual flags in RAM or to handle single I/O bits as control lines.

CAUTION

The corresponding data direction registers for ports A, B, and C are write-only registers (locations $004, $005, and $006). A read operation on these registers is undefined. Since BSET and BCLR are read-modify-write functions, they cannot be used to set a data direction register bit (all "unaffected" bits would be set). It is recommended that all data direction register bits in a port be written using a single-store instruction.

The coding examples shown in Figure 22 illustrate the usefulness of the bit manipulation and test instruction. Assume that the microcomputer is to communicate with an external serial device. The external device has a data ready signal, a data output line, and a clock line to clock data one bit at a time, least significant bit first,out of the device. The microcomputer waits until the data is ready, clocks the external device, picks up the data in the carry flag, clears the clock line, and finally accumulates the data bit in a random-access memory location.

```
*       BRSET   2,PORTA,*        WAIT FOR READY

        BSET    1, PORTA         CLOCK NEXT BIT IN
        BRCLR   0, PORTA, NEXT   PICKUP BIT IN C-BIT
NEXT    BCLR    1, PORTA         RETURN CLOCK LINE HIGH
        ASR     RAMLOC           MOVE C-BIT INTO RAM
```

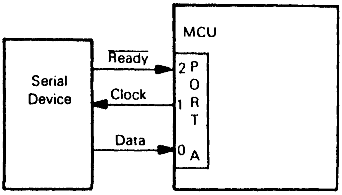


FIGURE 22 – BIT MANIPULATION EXAMPLE

ADDRESSING MODES

The EF6805U2 MCU has ten addressing modes available for use by the programmer. They are explained briefly in the following paragraphs. For additional details and graphical illustrations, refer to the EF6805 Family Users Manual.

The term "effective address" (EA) is used in describing the addressing modes. EA is defined as the address from which the argument for an instruction is fetched or stored.

IMMEDIATE – In the immediate addressing mode, the operand is contained in the byte immediately following the opcode. The immediate addressing mode is used to access constants which do not change during program execution (e.g., a constant used to initialize a loop counter).

DIRECT – In the direct addressing mode, the effective address of the argument is contained in a single byte following the opcode byte. Direct addressing allows the user to directly address the lowest 256 bytes in memory with a single 2-byte instruction. This address area includes all on-chip RAM and I/O registers and 128 bytes of ROM. Direct addressing is an effective use of both memory and time.

EXTENDED – In the extended addressing mode, the effective address of the argument is contained in the two bytes following the opcode. Instructions with extended addressing mode are capable of referencing arguments anywhere in memory with a single 3-byte instruction. When using the assembler, the user need not specify whether an instruction uses direct or extended addressing. The assembler automatically selects the shortest form of the instruction.

RELATIVE – The relative addressing mode is only used in branch instructions. In relative addressing, the contents of the 8-bit signed byte following the opcode (the offset) is added to the PC if, and only if, the branch condition is true. Otherwise, control proceeds to the next instruction. The span of relative addressing is from −126 to +129 from the opcode address. The programmer need not worry about calculating the correct offset if he uses the assembler, since it calculates the proper offset and checks to see if it is within the span of the branch.

INDEXED, NO OFFSET – In the indexed, no offset addressing mode, the effective address of the argument is contained in the 8-bit index register. Thus, this addressing mode can access the first 256 memory locations. These instructions are only one byte long. This mode is often used to move a pointer through a table or to hold the address of a frequently referenced RAM or I/O location.

INDEXED, 8-BIT OFFSET – In the indexed, 8-bit offset addressing mode, the effective address is the sum of the contents of the unsigned 8-bit index register and the unsigned byte following the opcode. This addressing mode is useful in selecting the kth element in an n element table. With this 2-byte instruction, k would typically be in X with the address of the beginning of the table in the instruction. As such, tables may begin anywhere within the first 256 addressable locations and could extend as far as location 510 ($1FE is the last location at which the instruction may begin).

INDEXED, 16-BIT OFFSET – In the indexed, 16-bit offset addressing mode, the effective address is the sum of the contents of the unsigned 8-bit index register and the two unsigned bytes following the opcode. This addressing mode can be used in a manner similar to indexed, 8-bit offset except that this 3-byte instruction allows tables to be anywhere in memory. As with direct and extended, the assembler determines the shortest form of indexed addressing.

BIT SET/CLEAR – In the bit set/clear addressing mode, the bit to be set or cleared is part of the opcode, and the byte following the opcode specifies the direct address of the byte in which the specified bit is to be set or cleared. Thus, any read/write bit in the first 256 locations of memory, including I/O, can be selectively set or cleared with a single 2-byte instruction.

CAUTION

The corresponding DDRs for ports A, B, and C are write-only registers (registers at $004, $005, $006). A read operation on these registers is undefined. Since BSET and BCLR are read-modify-write in function, they cannot be used to set or clear a single DDR bit (all "unaffected" bits would be set). It is recommended that all DDR bits in a port must be written using a single-store instruction.

BIT TEST AND BRANCH – The bit test and branch addressing mode is a combination of direct addressing and relative addressing. The bit which is to be tested and condition (set or clear) is included in the opcode, and the address of the byte to be tested is in the single byte immediately following the opcode byte. The signed relative 8-bit offset in the third byte is added to the PC if the specified bit is set or cleared in the specified memory location. This single 3-byte instruction allows the program to branch based on the condition of any readable bit in the first 256 locations of memory. The span of branching is from − 125 to + 130 from the opcode address. The state of the tested bit is also transferred to the carry bit of the condition code registers.

CAUTION

The corresponding DDRs for ports A, B, and C are write-only registers (registers at $004, $005, $006). A read operation on these registers is undefined. Since BSET and BCLR are read-modify-write in function, they cannot be used to set or clear a single DDR bit (all "unaffected" bits would be set). It is recommended that all DDR bits in a port must be written using a single-store instruction.

INHERENT – In the inherent addressing mode, all the information necessary to execute the instruction is contained in the opcode. Operations specifying only the index register or accumulator, as well as control instructions with no other arguments, are included in this mode. These instructions are one byte long.

INSTRUCTION SET

The EF6805U2 MCU has a set of 59 basic instructions, which when combined with the 10 addressing modes produce 207 usable opcodes. They can be divided into five different types: register/memory, read-modify-write, branch, bit manipulation, and control. The following paragraphs briefly explain each type. All the instructions within a given type are presented in individual tables.

REGISTER/MEMORY INSTRUCTIONS – Most of these instructions use two operands. One operand is either the accumulator or the index register. The other operand is obtained from memory using one of the addressing modes. The jump unconditional (JMP) and jump to subroutine (JSR) instructions have no register operand. Refer to Table 1.

READ-MODIFY-WRITE INSTRUCTIONS – These instructions read a memory location or a register, modify or test its contents, and write the modified value back to memory or to the register; see Caution under Input/Output section. The test for negative or zero (TST) instruction is included in the read-modify-write instruction though it does not perform the write. Refer to Table 2.

BRANCH INSTRUCTIONS – The branch instructions cause a branch from the program when a certain condition is met. Refer to Table 3.

BIT MANIPULATION INSTRUCTIONS – The instructions are used on any bit in the first 256 bytes of the memory; One group either sets or clears. The other group performs the bit test and branch operations. Refer to Table 4.

CAUTION

The corresponding DDRs for ports A, B, and C are write-only registers (registers at $004, $005, $006). A read operation on these registers is undefined. Since BSET and BCLR are read-modify-write in function, they cannot be used to set or clear a single DDR bit (all "unaffected" bits would be set). It is recommended that all DDR bits in a port must be written using a single-store instruction.

CONTROL INSTRUCTION – The control instructions control the MCU operations during program execution. Refer to Table 5.

ALPHABETICAL LISTING – The complete instruction set is given in alphabetical order in Table 6.

OPCODE MAP – Table 7 is an opcode map for the instruction used on the MCU.

TABLE 1 – REGISTER/MEMORY INSTRUCTIONS

		Addressing Modes																	
		Immediate			Direct			Extended			Indexed (No Offset)			Indexed (8-Bit Offset)			Indexed (16-Bit Offset)		
Function	Mnemonic	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles	OP Code	# Bytes	# Cycles
Load A from Memory	LDA	A6	2	2	B6	2	4	C6	3	5	F6	1	4	E6	2	5	D6	3	6
Load X from Memory	LDX	AE	2	2	BE	2	4	CE	3	5	FE	1	4	EE	2	5	DE	3	6
Store A in Memory	STA	–	–		B7	2	5	C7	3	6	F7	1	5	E7	2	6	D7	3	7
Store X in Memory	STX	–	-	-	BF	2	5	CF	3	6	FF	1	5	EF	2	6	DF	3	7
Add Memory to A	ADD	AB	2	2	BB	2	4	CB	3	5	FB	1	4	EB	2	5	DB	3	6
Add Memory and Carry to A	ADC	A9	2	2	B9	2	4	C9	3	5	F9	1	4	E9	2	5	D9	3	6
Subtract Memory	SUB	A0	2	2	B0	2	4	C0	3	5	F0	1	4	E0	2	5	D0	3	6
Subtract Memory from A with Borrow	SBC	A2	2	2	B2	2	4	C2	3	5	F2	1	4	E2	2	5	D2	3	6
AND Memory to A	AND	A4	2	2	B4	2	4	C4	3	5	F4	1	4	E4	2	5	D4	3	6
OR Memory with A	ORA	AA	2	2	BA	2	4	CA	3	5	FA	1	4	EA	2	5	DA	3	6
Exclusive OR Memory with A	EOR	A8	2	2	B8	2	4	C8	3	5	F8	1	4	E8	2	5	D8	3	6
Arithmetic Compare A with Memory	CMP	A1	2	2	B1	2	4	C1	3	5	F1	1	4	E1	2	5	D1	3	6
Arithmetic Compare X with Memory	CPX	A3	2	2	B3	2	4	C3	3	5	F3	1	4	E3	2	5	D3	3	6
Bit Test Memory with A (Logical Compare)	BIT	A5	2	2	B5	2	4	C5	3	5	F5	1	4	E5	2	5	D5	3	6
Jump Unconditional	JMP	-			BC	2	3	CC	3	4	FC	1	3	EC	2	4	DC	3	5
Jump to Subroutine	JSR	--	--		BD	2	7	CD	3	8	FD	1	7	ED	2	8	DD	3	9

TABLE 2 – READ-MODIFY-WRITE INSTRUCTIONS

		Addressing Modes														
		Inherent (A)			Inherent (X)			Direct			Indexed (No Offset)			Indexed (8 Bit Offset)		
Function	Mnemonic	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles
Increment	INC	4C	1	4	5C	1	4	3C	2	6	7C	1	6	6C	2	7
Decrement	DEC	4A	1	4	5A	1	4	3A	2	6	7A	1	6	6A	2	7
Clear	CLR	4F	1	4	5F	1	4	3F	2	6	7F	1	6	6F	2	7
Complement	COM	43	1	4	53	1	4	33	2	6	73	1	6	63	2	7
Negate (2's Complement)	NEG	40	1	4	50	1	4	30	2	6	70	1	6	60	2	7
Rotate Left Thru Carry	ROL	49	1	4	59	1	4	39	2	6	79	1	6	69	2	7
Rotate Right Thru Carry	ROR	46	1	4	56	1	4	36	2	6	76	1	6	66	2	7
Logical Shift Left	LSL	48	1	4	58	1	4	38	2	6	78	1	6	68	2	7
Logical Shift Right	LSR	44	1	4	54	1	4	34	2	6	74	1	6	64	2	7
Arithmetic Shift Right	ASR	47	1	4	57	1	4	37	2	6	77	1	6	67	2	7
Test for Negative or Zero	TST	4D	1	4	5D	1	4	3D	2	6	7D	1	6	6D	2	7

TABLE 3 – BRANCH INSTRUCTIONS

		Relative Addressing Mode		
Function	Mnemonic	Op Code	# Bytes	# Cycles
Branch Always	BRA	20	2	4
Branch Never	BRN	21	2	4
Branch IFF Higher	BHI	22	2	4
Branch IFF Lower or Same	BLS	23	2	4
Branch IFF Carry Clear	BCC	24	2	4
(Branch IFF Higher or Same)	(BHS)	24	2	4
Branch IFF Carry Set	BCS	25	2	4
(Branch IFF Lower)	(BLO)	25	2	4
Branch IFF Not Equal	BNE	26	2	4
Branch IFF Equal	BEQ	27	2	4
Branch IFF Half Carry Clear	BHCC	28	2	4
Branch IFF Half Carry Set	BHCS	29	2	4
Branch IFF Plus	BPL	2A	2	4
Branch IFF Minus	BMI	2B	2	4
Branch IFF Interupt Mask Bit is Clear	BMC	2C	2	4
Branch IFF Interrupt Mask Bit is Set	BMS	2D	2	4
Branch IFF Interrupt Line is Low	BIL	2E	2	4
Branch IFF Interrupt Line is High	BIH	2F	2	4
Branch to Subroutine	BSR	AD	2	8

TABLE 4 – BIT MANIPULATION INSTRUCTIONS

		Addressing Modes					
		Bit Set/Clear			Bit Test and Branch		
Function	Mnemonic	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles
Branch IFF Bit n is set	BRSET n (n = 0....7)	—	—	—	2 • n	3	10
Branch IFF Bit n is clear	BRCLR n (n = 0 7)	—	—	—	01 + 2 • n	3	10
Set Bit n	BSET n (n = 0 7)	10 + 2 • n	2	7	—	—	—
Clear bit n	BCLR n (n = 0...7)	11 + 2 • n	2	7	—	—	—

TABLE 5 – CONTROL INSTRUCTIONS

		Inherent		
Function	Mnemonic	Op Code	# Bytes	# Cycles
Transfer A to X	TAX	97	1	2
Transfer X to A	TXA	9F	1	2
Set Carry Bit	SEC	99	1	2
Clear Carry Bit	CLC	98	1	2
Set Interrupt Mask Bit	SEI	9B	1	2
Clear Interrupt Mask Bit	CLI	9A	1	2
Software Interrupt	SWI	83	1	11
Return from Subroutine	RTS	81	1	6
Return from Interrupt	RTI	80	1	9
Reset Stack Pointer	RSP	9C	1	2
No-Operation	NOP	9D	1	2

TABLE 6 – INSTRUCTION SET

	Addressing Modes										Condition Code				
Mnemonic	Inherent	Immediate	Direct	Extended	Relative	Indexed (No Offset)	Indexed (8 Bits)	Indexed (16 Bits)	Bit Set/ Clear	Bit Test & Branch	H	I	N	Z	C
ADC		X	X	X		X	X	X			Λ	●	Λ	Λ	Λ
ADD		X	X	X		X	X	X			Λ	●	Λ	Λ	Λ
AND		X	X	X		X	X	X			●	●	Λ	Λ	●
ASL	X		X			X	X				●	●	Λ	Λ	Λ
ASR	X		X			X	X				●	●	Λ	Λ	Λ
BCC					X						●	●	●	●	●
BCLR									X		●	●	●	●	●
BCS					X						●	●	●	●	●
BEQ					X						●	●	●	●	●
BHCC					X						●	●	●	●	●
BHCS					X						●	●	●	●	●
BHI					X						●	●	●	●	●
BHS					X						●	●	●	●	●
BIH					X						●	●	●	●	●
BIL					X						●	●	●	●	●
BIT		X	X	X		X	X	X			●	●	Λ	Λ	●
BLO					X						●	●	●	●	●
BLS					X						●	●	●	●	●
BMC					X						●	●	●	●	●
BMI					X						●	●	●	●	●
BMS					X						●	●	●	●	●
BNE					X						●	●	●	●	●
BPL					X						●	●	●	●	●
BRA					X						●	●	●	●	●
BRN					X						●	●	●	●	●
BRCLR										X	●	●	●	●	Λ
BRSET										X	●	●	●	●	Λ
BSET									X		●	●	●	●	●
BSR					X						●	●	●	●	●
CLL	X										●	●	●	●	O
CLI	X										●	O	●	●	●
CLR	X		X			X	X				●	●	O	1	●
CMP		X	X	X		X	X	X			●	●	Λ	Λ	Λ
COM	X		X			X	X				●	●	Λ	Λ	1
CPX		X	X	X		X	X	X			●	●	Λ	Λ	Λ
DEC	X		X			X	X				●	●	Λ	Λ	●
EOR		X	X	X		X	X	X			●	●	Λ	Λ	●
INC	X		X			X	X				●	●	Λ	Λ	●
JMP			X	X		X	X	X			●	●	●	●	●
JSR			X	X		X	X	X			●	●	●	●	●
LDA		X	X	X		X	X	X			●	●	Λ	Λ	●
LDX		X	X	X		X	X	X			●	●	Λ	Λ	●
LSL	X		X			X	X				●	●	Λ	Λ	Λ
LSR	X		X			X	X				●	●	O	Λ	Λ
NEQ	X		X			X	X				●	●	Λ	Λ	Λ
NOP	X										●	●	●	●	●
ORA		X	X	X		X	X	X			●	●	Λ	Λ	●
ROL	X		X			X	X				●	●	Λ	Λ	Λ
RSP	X										●	●	●	●	●

Condition Code Symbols:

H Half Carry (From Bit 3)	C Carry/Borrow
I Interrupt Mask	Λ Test and Set if True, Cleared Otherwise
N Negative (Sign Bit)	• Not Affected
Z Zero	

TABLE 6 – INSTRUCTION SET (CONTINUED)

	Addressing Modes										Condition Code				
Mnemonic	Inherent	Immediate	Direct	Extended	Relative	Indexed (No Offset)	Indexed (8 Bits)	Indexed (16 Bits)	Bit Set/ Clear	Bit Test & Branch	H	I	N	Z	C
RTI	X										?	?	?	?	?
RTS	X										●	●	●	●	●
SBC		X	X	X		X	X	X			●	●	Λ	Λ	Λ
SEC	X										●	●	●	●	1
SEI	X										●	1	●	●	●
STA			X	X		X	X	X			●	●	Λ	Λ	●
STX			X	X		X	X	X			●	●	Λ	Λ	●
SUB		X	X	X		X	X	X			●	●	Λ	Λ	Λ
SWI	X										●	1	●	●	●
TAX	X										●	●	●	●	●
TST	X		X			X	X				●	●	Λ	Λ	●
TXA	X										●	●	●	●	●

Condition Code Symbols:

H Half Carry (From Bit 3)
I Interrupt Mask
N Negative (Sign Bit)
Z Zero

C Carry/Borrow
Λ Test and Set if True, Cleared Otherwise
● Not Affected
? Load CC Register From Stack

EF6805 HMOS FAMILY

Features	EF6805CT	EF6805P2	EF6805P4	EF6805P6	EF6805R2	EF6805R3	EF6805T2	EF6805U2	EF6805U3	EF6805TV
Technology	HMOS	HMOS	HMOS	HMOS	HMOS	HMOS	HMOS	HMOS	HMOS	HMOS
Number of Pins	40	28	28	28	40	40	28	40	40	40
On-Chip RAM (Bytes)	240	64	112	64	64	112	64	64	112	96
On-Chip User ROM (Bytes)	4096	1100	1100	1796	2048	3776	2508	2048	3776	6144
External Bus	Yes	None	None	None	None	None	None	None	None	None
Bidirectional I/O Lines	29	20	20	20	24	24	19	24	24	32
Unidirectional I/O Lines	None	None	None	None	6 Inputs	6 Inputs	None	8 Inputs	8 Inputs	None
Other I/O Features	Timer,UART	Timer	Timer	Timer	Timer,A/D	Timer,A/D	Timer, PLL	Timer	Timer	Timer,D/A
External Interrupt Inputs	3	1	1	1	2	2	2	2	2	1
STOP and WAIT	No	No	No	No	No	No	No	No	No	No

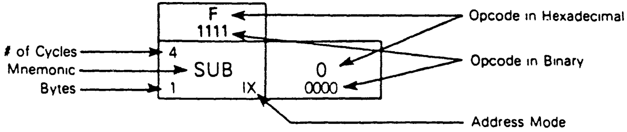
TABLE 7 – 6805 HMOS FAMILY INSTRUCTION SET OPCODE MAP

	Bit Manipulation		Branch	Read-Modify-Write					Control		Register/Memory						
	BTB	BSC	REL	DIR	INH	INH	IX1	IX	INH	INH	IMM	DIR	EXT	IX2	IX1	IX	
Hi Low	0 0000	1 0001	2 0010	3 0011	4 0100	5 0101	6 0110	7 0111	8 1000	9 1001	A 1010	B 1011	C 1100	D 1101	E 1110	F 1111	Hi Low
0 0000	10 BRSET0 3 BTB	7 BSET0 2 BSC	4 BRA 2 REL	6 NEG 2 DIR	4 NEG 1 INH	4 NEG 1 INH	7 NEG 2 IX1	6 NEG 1 IX	9 RTI 1 INH		2 SUB 2 IMM	4 SUB 2 DIR	5 SUB 3 EXT	6 SUB 3 IX2	5 SUB 2 IX1	4 SUB 1 IX	0 0000
1 0001	10 BRCLR0 3 BTB	7 BCLR0 2 BSC	4 BRN 2 REL						6 RTS 1 INH		2 CMP 2 IMM	4 CMP 2 DIR	5 CMP 3 EXT	6 CMP 3 IX2	5 CMP 2 IX1	4 CMP 1 IX	1 0001
2 0010	10 BRSET1 3 BTB	7 BSET1 2 BSC	4 BHI 2 REL								2 SBC 2 IMM	4 SBC 2 DIR	5 SBC 3 EXT	6 SBC 3 IX2	5 SBC 2 IX1	4 SBC 1 IX	2 0010
3 0011	10 BRCLR1 3 BTB	7 BCLR1 2 BSC	4 BLS 2 REL	6 COM 2 DIR	4 COMA 1 INH	4 COMX 1 INH	7 COM 2 IX1	6 COM 1 IX	11 SWI 1 INH		2 CPX 2 IMM	4 CPX 2 DIR	5 CPX 3 EXT	6 CPX 3 IX2	5 CPX 2 IX1	4 CPX 1 IX	3 0011
4 0100	10 BRSET2 3 BTB	7 BSET2 2 BSC	4 BCC 2 REL	6 LSR 2 DIR	4 LSRA 1 INH	4 LSRX 1 INH	7 LSR 2 IX1	6 LSR 1 IX			2 AND 2 IMM	4 AND 2 DIR	5 AND 3 EXT	6 AND 3 IX2	5 AND 2 IX1	4 AND 1 IX	4 0100
5 0101	10 BRCLR2 3 BTB	7 BCLR2 2 BSC	4 BCS 2 REL								2 BIT 2 IMM	4 BIT 2 DIR	5 BIT 3 EXT	6 BIT 3 IX2	5 BIT 2 IX1	4 BIT 1 IX	5 0101
6 0110	10 BRSET3 3 BTB	7 BSET3 2 BSC	4 BNE 2 REL	6 ROR 2 DIR	4 RORA 1 INH	4 RORX 1 INH	7 ROR 2 IX1	6 ROR 1 IX			2 LDA 2 IMM	4 LDA 2 DIR	5 LDA 3 EXT	6 LDA 3 IX2	5 LDA 2 IX1	4 LDA 1 IX	6 0110
7 0111	10 BRCLR3 3 BTB	7 BCLR3 2 BSC	4 BEQ 2 REL	6 ASR 2 DIR	4 ASRA 1 INH	4 ASRX 1 INH	7 ASR 2 IX1	6 ASR 1 IX		2 TAX 1 INH		5 STA 2 DIR	6 STA 3 EXT	7 STA 3 IX2	6 STA 2 IX1	5 STA 1 IX	7 0111
8 1000	10 BRSET4 3 BTB	7 BSET4 2 BSC	4 BHCC 2 REL	6 LSL 2 DIR	4 LSLA 1 INH	4 LSLX 1 INH	7 LSL 2 IX1	6 LSL 1 IX		2 CLC 1 INH	2 EOR 2 IMM	4 EOR 2 DIR	5 EOR 3 EXT	6 EOR 3 IX2	5 EOR 2 IX1	4 EOR 1 IX	8 1000
9 1001	10 BRCLR4 3 BTB	7 BCLR4 2 BSC	4 BHCS 2 REL	6 ROL 2 DIR	4 ROLA 1 INH	4 ROLX 1 INH	7 ROL 2 IX1	6 ROL 1 IX		2 SEC 1 INH	2 ADC 2 IMM	4 ADC 2 DIR	5 ADC 3 EXT	6 ADC 3 IX2	5 ADC 2 IX1	4 ADC 1 IX	9 1001
A 1010	10 BRSET5 3 BTB	7 BSET5 2 BSC	4 BPL 2 REL	6 DEC 2 DIR	4 DECA 1 INH	4 DECX 1 INH	7 DEC 2 IX1	6 DEC 1 IX		2 CLI 1 INH	2 ORA 2 IMM	4 ORA 2 DIR	5 ORA 3 EXT	6 ORA 3 IX2	5 ORA 2 IX1	4 ORA 1 IX	A 1010
B 1011	10 BRCLR5 3 BTB	7 BCLR5 2 BSC	4 BMI 2 REL							2 SEI 1 INH	2 ADD 2 IMM	4 ADD 2 DIR	5 ADD 3 EXT	6 ADD 3 IX2	5 ADD 2 IX1	4 ADD 1 IX	B 1011
C 1100	10 BRSET6 3 BTB	7 BSET6 2 BSC	4 BMC 2 REL	6 INC 2 DIR	4 INCA 1 INH	4 INCX 1 INH	7 INC 2 IX1	6 INC 1 IX		2 RSP 1 INH		3 JMP 2 DIR	4 JMP 3 EXT	5 JMP 3 IX2	4 JMP 2 IX1	3 JMP 1 IX	C 1100
D 1101	10 BRCLR6 3 BTB	7 BCLR6 2 BSC	4 BMS 2 REL	6 TST 2 DIR	4 TSTA 1 INH	4 TSTX 1 INH	7 TST 2 IX1	6 TST 1 IX		2 NOP 1 INH	8 BSR 2 REL	7 JSR 2 DIR	8 JSR 3 EXT	9 JSR 3 IX2	8 JSR 2 IX1	7 JSR 1 IX	D 1101
E 1110	10 BRSET7 3 BTB	7 BSET7 2 BSC	4 BIL 2 REL								2 LDX 2 IMM	4 LDX 2 DIR	5 LDX 3 EXT	6 LDX 3 IX2	5 LDX 2 IX1	4 LDX 1 IX	E 1110
F 1111	10 BRCLR7 3 BTB	7 BCLR7 2 BSC	4 BIH 2 REL	6 CLR 2 DIR	4 CLRA 1 INH	4 CLRX 1 INH	7 CLR 2 IX1	6 CLR 1 IX		2 TXA 1 INH		5 STX 2 DIR	6 STX 3 EXT	7 STX 3 IX2	6 STX 2 IX1	5 STX 1 IX	F 1111

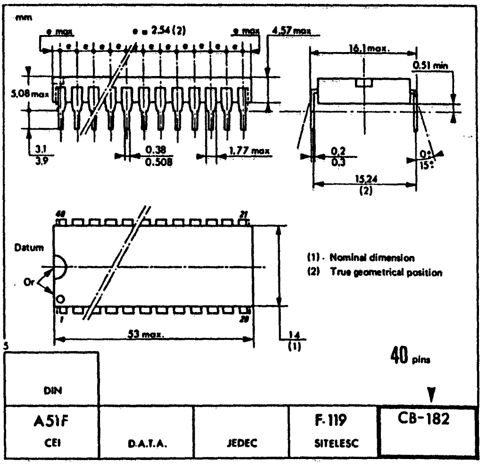
Abbreviations for Address Modes

INH	Inherent
IMM	Immediate
DIR	Direct
EXT	Extended
REL	Relative
BSC	Bit Set/Clear
BTB	Bit Test and Branch
IX	Indexed (No Offset)
IX1	Indexed, 1 Byte (8-Bit) Offset
IX2	Indexed, 2 Byte (16-Bit) Offset

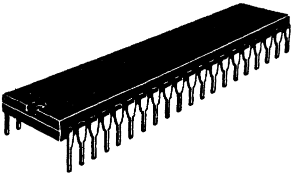
LEGEND



PHYSICAL DIMENSIONS

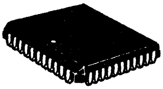


CB-182

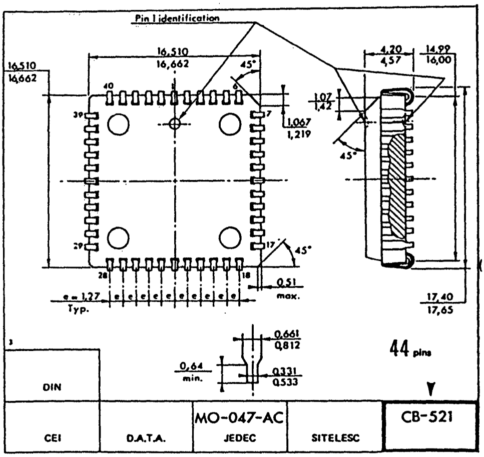


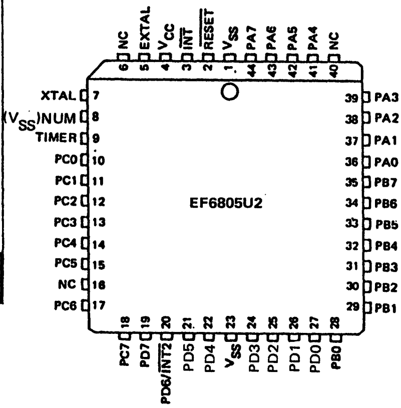
P SUFFIX
PLASTIC PACKAGE

CB-521



FN SUFFIX
PLCC 44





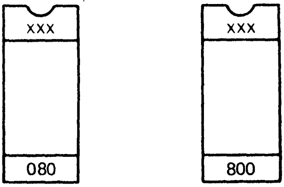
ORDERING INFORMATION

The information required when ordering a custom MCU is listed below. The ROM program may be transmitted to THOMSON SEMICONDUCTEURS on EPROM(s) or an EFDOS/MDOS* disk file.

To initiate a ROM pattern for the MCU, it is necessary to first contact your local THOMSON SEMICONDUCTEURS representative or distributor.

EPROMs

Two 2716 or one 2732 type EPROMs, programmed with the customer program (positive logic sense for address and data), may be submitted for pattern generation. The EPROM must be clearly marked to indicate which EPROM corresponds to which address space. The recommended marking procedure is illustrated below :



XXX = Customer ID

After the EPROM(s) are marked, they should be placed in conductive IC carriers and securely packed. Do not use styrofoam.

VERIFICATION MEDIA

All original pattern media (EPROMs or floppy disk) are filed for contractual purposes and are not returned. A computer listing of the ROM code will be generated and returned along with a listing verification form. The listing should be thoroughly checked and the verification form completed, signed, and returned to THOMSON SEMICONDUCTEURS. The signed verification form constitutes the contractual agreement for creation of the customer mask. If desired, THOMSON SEMICONDUCTEURS will program on blank EPROM from the data file used to create the custom mask and aid in the verification process.

ROM VERIFICATION UNITS (RVUs)

Ten MCUs containing the customer's ROM pattern will be sent for program verification. These units will have been made using the custom mask but are for the purpose of ROM verification only. For expediency they are usually unmarked, packaged in ceramic, and tested only at room temperature and 5 volts. These RVUs are included in the mask charge and are not production parts. The RVUs are thus not guaranteed by THOMSON SEMICONDUCTEURS. Quality Assurance, and should be discarded after verification is completed.

FLEXIBLE DISKS

The disk media submitted must be single-sided, EFDOS/MDOS* compatible floppies.

The customer must write the binary file name and company name on the disk with a felt-tip-pen. The minimum EFDOS/MDOS* system files, as well as the absolute binary object file (Filename .LO type of file) from the 6805 cross assembler, must be on the disk. An object file made from a memory dump using the ROLLOUT command is also acceptable. Consider submitting a source listing as well as the following files : filename .LX (DEVICE/EXORciser loadable format) and filename .SA (ASCII Source Code). These files will of course be kept confidential and are used 1) to speed up the process in-house if any problems arise, and 2) to speed up the user-to-factory interface if the user finds any software errors and needs assistance quickly from THOMSON SEMICONDUCTEURS factory representatives.

EFDOS is THOMSON SEMICONDUCTEURS' Disk Operating System available on development systems such as DEVICE,...

MDOS* is MOTOROLA's Disk Operating System available on development systems such as EXORciser,...

*Requires prior factory approval.

Whenever ordering a custom MCU is required, please contact your local THOMSON SEMICONDUCTEURS representative or THOMSON SEMICONDUCTEURS distributor and/or complete and send the attached "MCU customer ordering sheet" to your local THOMSON SEMICONDUCTEURS representative.

EXORciser is a registered trademark of MOTOROLA Inc.

ORDERING INFORMATION

EF6805U2 P V

Device

Package

Screening level

Oper. temp.

The table below horizontally shows all available suffix combinations for package, operating temperature and screening level. Other possibilities on request.

DEVICE	PACKAGE					OPER. TEMP			SCREENING LEVEL			
	C	J	P	E	FN	L*	V	T	Std	D	G/B	B/B
EF6805U2			●		●	●	●	●	●	●		

Examples : EF6805U2P, EF6805U2FN, EF6805U2PV, EF6805U2FNV.

Package : C : Ceramic DIL, **J :** Cerdip DIL, **P :** Plastic DIL, **E :** LCCC, **FN :** PLCC.
Oper. temp. : L* : 0°C to +70°C, **V :** −40°C to +85°C, **T :** - 40°C to + 105°C, ***:** may be omitted.
Screening level : Std : (no-end suffix), **D :** NFC 96883 level D,
G/B : NFC 96883 level G, **B/B :** NFC 96883 level B and MIL-STD-883C level B.

These specifications are subject to change without notice.
Please inquire with our sales offices about the availability of the different products.

HOMSON
COMPONENTS
EMICONDUCTEURS

EF6805U3

8 BIT MICROCOMPUTER

ADVANCE INFORMATION

The EF6805U3 Microcomputer Unit (MCU) is a member of the 6805 Family of low-cost single-chip Microcomputers. The 8-bit microcomputer contains a CPU, on-chip CLOCK, ROM, RAM, I/O, and TIMER. It is designed for the user who needs an economical microcomputer with the proven capabilities of the 6800-based instruction set. A comparison of the key features of several members of the 6805 Family of Microcomputers is shown at the end of this data sheet. The following are some of the hardware and software highlights of the EF6805U3 MCU.

HARDWARE FEATURES

- 32 TTL/CMOS Compatible I/O Lines
  24 Bidirectional (8 Lines are LED Compatible)
  8 Input-Only
- 3776 Bytes of User ROM
- 112 Bytes of RAM
- Self-Check Mode
- Zero-Crossing Detect/Interrupt
- Internal 8-Bit Timer with 7-Bit Software Programmable Prescaler and Clock Source
- 5 V Single Supply

SOFTWARE FEATURES

- 10 Powerful Addressing Modes
- Byte Efficient Instruction Set with True Bit Manipulation, Bit Test, and Branch Instructions
- Single Instruction Memory Examine/Change
- Powerful Indexed Addressing for Tables
- Full Set of Conditional Branches
- Memory Usable as Register/Flags
- Complete Development System Support on DEVICE®

USER SELECTABLE OPTIONS

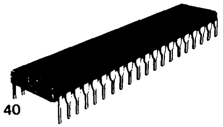
- 8 Bidirectional I/O Lines with TTL or TTL/CMOS Interface Option
- 8 Bidirectional I/O Lines with TTL or Open-Drain Interface Option
- Crystal or Low-Cost Resistor Oscillator Option
- Low Voltage Inhibit Option
- Vectored Interrupts: Timer, Software, and External
- User Callable Self-Check Subroutines

DEVICE® is THOMSON SEMICONDUCTEURS' development/emulation tool.

HMOS

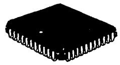
CASES

CB-182



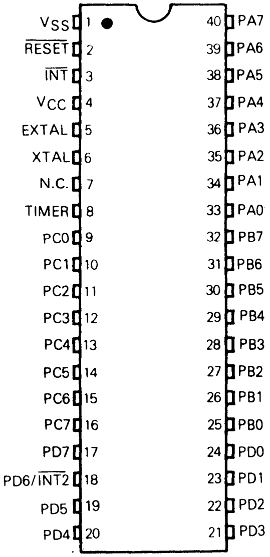
P SUFFIX
PLASTIC PACKAGE

CB-521



FN SUFFIX
PLCC44

PIN ASSIGNMENT



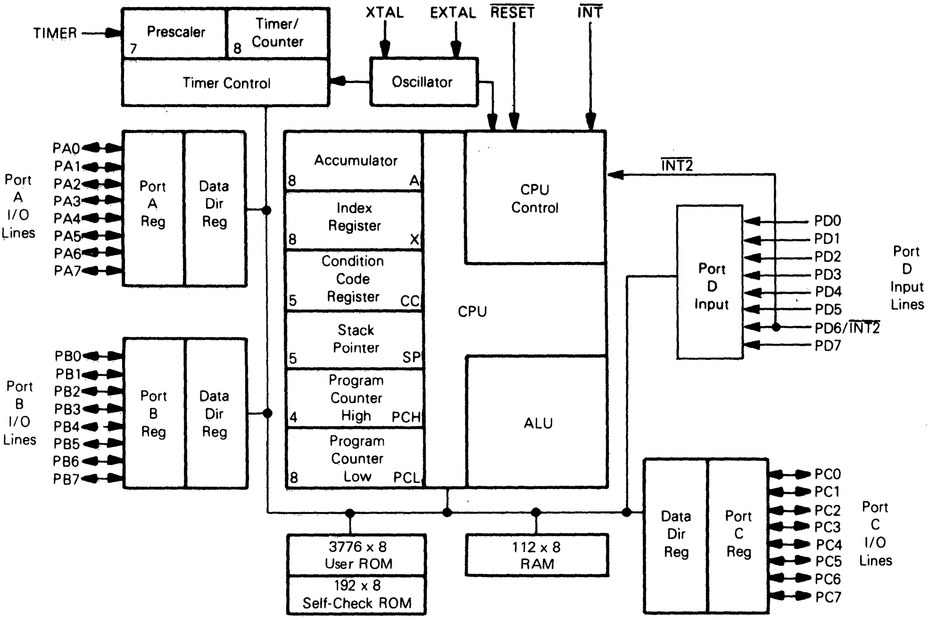


FIGURE 1 - EF6805U3 HMOS MICROCOMPUTER BLOCK DIAGRAM

MAXIMUM RATING

Rating	Symbol	Value	Unit
Supply Voltage	V_{CC}	−0.3 to + 7.0	V
Input Voltage (Except TIMER in Self-Check Mode and Open-Drain Inputs)	V_{in}	−0.3 to + 7.0	V
Input Voltage (Open-Drain Pins, TIMER Pin in Self-Check Mode)	V_{in}	−0.3 to + 15.0	V
Operating Temperature Range (T_L to T_H) V suffix T suffix	T_A	0 to + 70 −40 to + 85 −40 to + 105	°C
Storage Temperature Range	T_{stg}	−55 to + 150	°C
Junction Temperature Plastic Package PLCC	T_J	 150 150	°C

This device contains circuitry to protect the inputs against damage due to high static voltages or electrical fields; however, it is advised that normal precautions be taken to avoid application of any voltage higher than maximum rated voltages to this high-impedance circuit. For proper operation it is recommended that V_{in} and V_{out} be constrained to the range $V_{SS} \leq (V_{in}$ or $V_{out}) \leq V_{CC}$. Reliability of operation is enhanced if unused inputs except EXTAL are tied to an appropriate logic voltage level (e.g., either V_{SS} or V_{CC}).

THERMAL CHARACTERISTICS

Characteristic	Symbol	Value	Unit
Thermal Resistance Plastic PLCC	θ_{JA}	 50 80	°C/W

POWER CONSIDERATIONS

The average chip-junction temperature, T_J, in °C can be obtained from:

$T_J = T_A + (P_D \bullet \theta_{JA})$ (1)

Where:

T_A = Ambient Temperature, °C

θ_{JA} = Package Thermal Resistance, Junction-to-Ambient, °C/W

$P_D = P_{INT} + P_{PORT}$

$P_{INT} = I_{CC} \times V_{CC}$, Watts – Chip Internal Power

P_{PORT} = Port Power Dissipation, Watts – User Determined

For most applications $P_{PORT} \ll P_{INT}$ and can be neglected. P_{PORT} may become significant if the device is configured to drive Darlington bases or sink LED loads.

An approximate relationship between P_D and T_J (if P_{PORT} is neglected) is:

$P_D = K \div (T_J + 273°C)$ (2)

Solving equations 1 and 2 for K gives:

$K = P_D \bullet (T_A + 273°C) + \theta_{JA} \bullet P_D^2$ (3)

Where K is a constant pertaining to the particular part. K can be determined from equation 3 by measuring P_D (at equilibrium) for a known T_A. Using this value of K the values of P_D and T_J can be obtained by solving equations (1) and (2) iteratively for any value of T_A.

ELECTRICAL CHARACTERISTICS (V_{CC} =+ 5.25 Vdc ± 0.5 Vdc, V_{SS} = 0 Vdc, $T_A = T_L$ to T_H Unless Otherwise Noted)

Characteristic	Symbol	Min	Typ	Max	Unit
Input High Voltage $\overline{\text{RESET}}$ $(4.75 \leq V_{CC} \leq 5.75)$ $(V_{CC} < 4.75)$ $\overline{\text{INT}}$ $(4.75 \leq V_{CC} \leq 5.75)$ $(V_{CC} < 4.75)$ All Other (Except Timer)	V_{IH}	 4.0 $V_{CC} - 0.5$ 4.0 $V_{CC} - 0.5$ 2.0	 – – * * –	 V_{CC} V_{CC} V_{CC} V_{CC} V_{CC}	V
Input High Voltage Timer Timer Mode Self-Check Mode	V_{IH}	 2.0 9.0	 – 10.0	 V_{CC} + 1.0 15.0	V
Input Low Voltage $\overline{\text{RESET}}$ $\overline{\text{INT}}$ All Other	V_{IL}	 V_{SS} V_{SS} V_{SS}	 – * –	 0.8 1.5 0.8	V
$\overline{\text{RESET}}$ Hysteresis Voltages (See Figures 10,11 and 12) "Out of Reset" "Into Reset"	 V_{IRES+} V_{IRES-}	 2.1 0.8	 – –	 4.0 2.0	V
$\overline{\text{INT}}$ Zero Crossing Input Voltage, Through a Capacitor	V_{INT}	2	–	4	$V_{ac\ p\text{-}p}$
Power Dissipation – (No Port Loading, V_{CC} = 5.75 V) T_A = 0°C T_A = –40°C	P_D	 – –	 520 580	 740 800	mW
Input Capacitance EXTAL All Other	C_{in}	 – –	 25 10	 – –	pF
Low Voltage Recover	V_{LVR}	–	–	4.75	V
Low Voltage Inhibit	V_{LVI}	2.75	3.75	4.70	V
Input Current TIMER (V_{in} = 0.4 V) $\overline{\text{INT}}$ (V_{in} = 2.4 V to V_{CC}) EXTAL (V_{in} = 2.4 V to V_{CC} - Crystal Option) (V_{in} = 0.4 V - Crystal Option) $\overline{\text{RESET}}$ (V_{in} = 0.8 V) - External Capacitor Charging Current	I_{in}	 – – – – –4.0	 – 20 – – –	 20 50 10 –1600 –40	μA

*Due to internal biasing this input (when unused) floats to approximately 2.2 V.

SWITCHING CHARACTERISTICS (V_{CC} =+ 5.25 Vdc ± 0.5 Vdc, V_{SS} = 0 Vdc, T_A = T_L to T_H Unless Otherwise Noted)

Characteristic	Symbol	Min	Typ	Max	Unit
Oscillator Frequency	f_{osc}	0.4	–	4.2	MHz
Cycle Time (4/f_{osc})	t_{cyc}	0.95	–	10	µs
$\overline{INT}$, $\overline{INT2}$, and TIMER Pulse Width (See Interrupt Section)	t_{WL}, t_{WH}	t_{cyc}+ 250	–	–	ns
$\overline{RESET}$ Pulse Width	t_{RWL}	t_{cyc}+ 250	–	–	ns
$\overline{INT}$ Zero-Crossing Detection Input Frequency	f_{INT}	0.03	–	1	kHz
External Clock Input Duty Cycle (EXTAL)	–	40	50	60	%
Crystal Oscillator Start-Up Time *	–	–	–	100	ms

* See Figure 16 for typical crystal parameters

PORT ELECTRICAL CHARACTERISTICS (V_{CC} =+ 5.25 Vdc ± 0.5 Vdc, V_{SS} = 0 Vdc, T_A = T_L to T_H Unless Otherwise Noted)

Characteristic	Symbol	Min	Typ	Max	Unit
Port A with CMOS Drive Enabled					
Output Low Voltage (I_{Load}= 1.6 mA)	V_{OL}	–	–	0.4	V
Output High Voltage I_{Load}= −100 µA I_{Load}= −10 µA	V_{OH}	 2.4 V_{CC} − 1.0	 – –	 – –	V
Input High Voltage (I_{Load}= −300 µA max.)	V_{IH}	2.0	–	V_{CC}	V
Input Low Voltage (I_{Load}= −500 µs max.)	V_{IL}	V_{SS}	–	0.8	V
High Z State Input Current (V_{in}= 2.0 V to V_{CC})	I_{IH}	–	–	−300	µA
High Z State Input Current (V_{in}= 0.4 V)	I_{IL}	–	–	−500	µA
Port B					
Output Low Voltage I_{Load}= 3.2 mA I_{Load}= 10 mA (Sink)	V_{OL}	 – –	 – –	 0.4 1.0	V
Output High Voltage I_{Load}= −200 µA	V_{OH}	2.4	–	–	V
Darlington Current Drive (Source) V_O= 1.5 V	I_{OH}	−1.0	–	−10	mA
Input High Voltage	V_{IH}	2.0	–	V_{CC}	V
Input Low Voltage	V_{IL}	V_{SS}	–	0.8	V
High Z State Input Current	I_{TSI}	–	<2	10	µA
Port C and Port A with CMOS Drive Disabled					
Output Low Voltage I_{Load}= 1.6 mA	V_{OL}	–	–	0.4	V
Output High Voltage I_{Load}= −100 µA	V_{OH}	2.4	–	–	V
Input High Voltage	V_{IH}	2.0	–	V_{CC}	V
Input Low Voltage	V_{IL}	V_{SS}	–	0.8	V
High Z State Input Current	I_{TSI}	–	<2	10	µs
Port C (Open-Drain Option)					
Input High Voltage	V_{IH}	2.0	–	13.0	V
Input Low Voltage	V_{IL}	V_{SS}	–	0.8	V
Input Leakage Current	I_{LOD}	–	<3	15	µA
Output Low Voltage I_{Load}= 1.6 mA	V_{OL}	–	–	0.4	V
Port D					
Input High Voltage	V_{IH}	2.0	–	V_{CC}	V
Input Low Voltage	V_{IL}	V_{SS}	–	0.8	V
Input Current	I_{in}	–	<1	5	µA

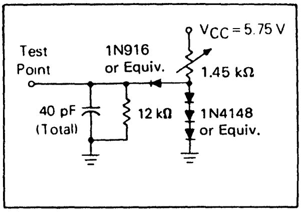


FIGURE 2 – TTL EQUIVALENT TEST LOAD (PORT B)

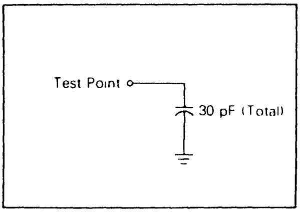


FIGURE 3 – CMOS EQUIVALENT TEST LOAD (PORT A)

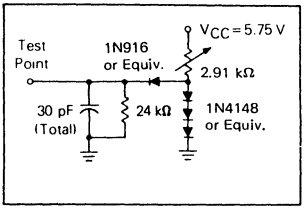


FIGURE 4 – TTL EQUIVALENT TEST LOAD (PORTS A AND C)

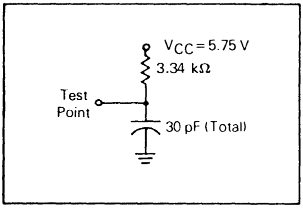


FIGURE 5 – OPEN - DRAIN EQUIVALENT TEST LOAD (PORT C)

SIGNAL DESCRIPTION

The input and output signals for the MCU, shown in Figure 1, are described in the following paragraphs.

V_{CC} AND V_{SS} – Power is supplied to the MCU using these two pins. V_{CC} is power and V_{SS} is the ground connection.

$\overline{\text{INT}}$ – This pin provides the capability for asynchronously applying an external interrupt to the MCU. Refer to Interrupts Section for additional information.

XTAL AND EXTAL – These pins provide control input for the on-chip clock oscillator circuit. A crystal, a resistor, or an external signal, depending on user selectable manufacturing mask option, can be connected to these pins to provide a system clock with various degrees of stability/cost tradeoffs. Lead length and stray capacitance on these two pins should be minimized. Refer to Internal Clock Generator Options Section for recommendations about these inputs.

NOTE: Pin 7 in DIL package/Pin 8 in PLCC package is connected to internal protection.

TIMER – The pin allows an external input to be used to control the internal timer circuitry and also to initiate the self test program. Refer to Timer Section for additional information about the timer circuitry.

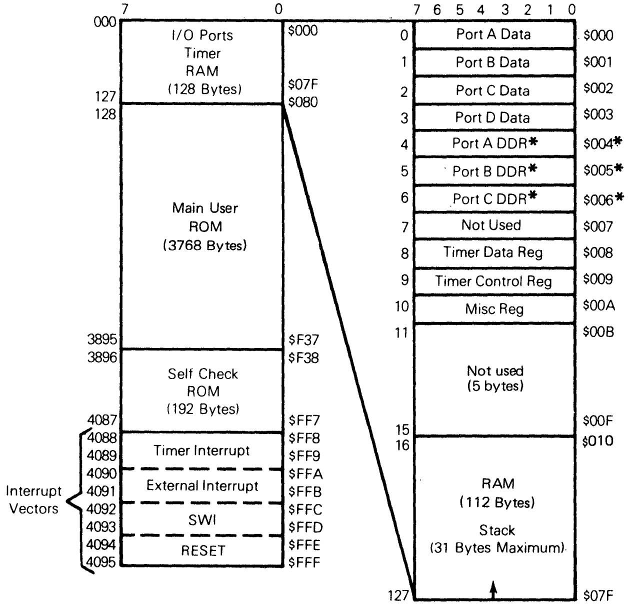
$\overline{\text{RESET}}$ – This pin allows resetting of the MCU at times other than the automatic resetting capability already in the MCU. The MCU can be reset by pulling $\overline{\text{RESET}}$ low. Refer to Resets Section for additional information.

INPUT/OUTPUT LINES (PA0-PA7, PB0-PB7, PC0-PC7, PD0-PD7) – These 32 lines are arranged into four 8-bit ports (A, B, C, and D). Ports A, B, and C are programmable as either inputs or outputs under software control of the data direction registers (DDRs). Port D is for digital input only and bit 6 may be used for a second interrupt $\overline{\text{INT2}}$. Refer to Input/Output Section and Interrupts Section for additional information.

MEMORY – The MCU is capable of addressing 4096 bytes of memory and I/O registers with its program counter. The EF6805U3 MCU has implemented 4090 of these bytes. This consists of : 3776 user ROM bytes, 192 self-check ROM bytes, 112 user RAM bytes, 7 port I/O bytes, 2 timer registers, and a miscellaneous register ; see Figure 6 for the Address map.The user ROM has been split into two areas. The main user ROM area is from **$080 to $F37.** The last 8 user ROM locations at the bottom of memory are for the interrupt vectors.

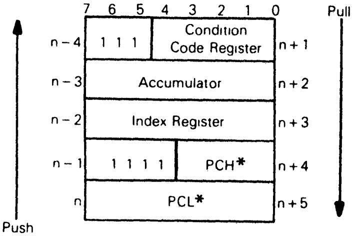
The MCU reserves the first 16 memory locations for I/O features, of which 10 have been implemented. These locations are used for the ports, the port DDRs, the timer and the $\overline{\text{INT2}}$ miscellaneous register, and the 112 RAM bytes, 31 bytes are shared with the stack area. The stack must be used with care when data shares the stack area.

The shared stack area is used during the processing of an interrupt or subroutine calls to save the contents of the CPU state. The register contents are pushed onto the stack in the order shown in Figure 7. Since the stack pointer decrements during pushes, the low order byte (PCL) of the program counter is stacked first, then the high order four bits (PCH) are stacked. This ensures that the program counter is loaded correctly during pulls from the stack since the stack pointer increments when it pulls data from the stack. A subroutine call results in only the program counter (PCL, PCH) contents being pushed onto the stack; the remaining CPU registers are not pushed.



*Caution: Data direction registers (DDRs) are write-only; they read as $FF.

FIGURE 6 – EF6805U3 MCU ADDRESS MAP



*For subroutine calls, only PCH and PCL are stacked

FIGURE 7 – INTERRUPT STACKING ORDER

CENTRAL PROCESSING UNIT

The CPU of the EF6805 Family is implemented independently from the I/O or memory configuration. Consequently, it can be treated as an independent central processor communicating with I/O and memory via internal address, data, and control buses.

REGISTERS

The 6805 Family CPU has five registers available to the programmer. They are shown in Figure 8and are explained in the following paragraphs.

ACCUMULATOR (A) – The accumulator is a general purpose 8-bit register used to hold operands and results of arithmetic calculations or data manipulations.

INDEX REGISTER (X) – The index register is an 8-bit register used for the indexed addressing mode. It contains an 8-bit value that may be added to an instruction value to create an effective address. The index register can also be used for data manipulations using the read-modify-write instructions. The Index Register may also be used as a temporary storage area.

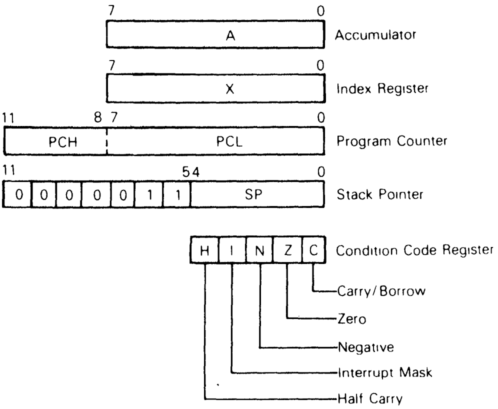


FIGURE 8 – PROGRAMMING MODEL

PROGRAM COUNTER (PC) – The Program Counter is a 12 bit register that contains the address of the next instruction to be executed.

STACK POINTER (SP)– The stack pointer is a 12-bit register that contains the address of the next free location on the stack. During an MCU reset or the reset stack pointer (RSP) instruction, the stack pointer is set to location $07F. The stack pointer is then decremented as data is pushed onto the stack and incremented as data is then pulled from the stack. The seven most significant bits of the stack pointer are permanently set to 0000011. Subroutines and interrupts may be nested down to location $061 (31 bytes maximum) which allows the programmer to use up to 15 levels of subroutine calls (less if interrupts are allowed).

CONDITION CODE REGISTER . (CC) – The condition code register is a 5-bit register in which four bits are used to indicate the results of the instruction just executed. These bits can be individually tested by a program and specific action taken as a result of their state. Each bit is explained in the following paragraphs.

Half Carry (H) – Set during ADD and ADC operations to indicate that a carry occurred between bits 3 and 4.

Interrupt (I) – When this bit is set, the timer and external interrupts ($\overline{\text{INT}}$ and $\overline{\text{INT2}}$) are masked (disabled). If an interrupt occurs while this bit is set, the interrupt is latched and is processed as soon as the interrupt bit is cleared.

Negative (N) – When set, this bit indicates that the result of the last arithmetic, logical, or data manipulation was negative (bit 7 in the result is a logical "1").

Zero (Z) – When set, this bit indicates that the result of the last arithmetic, logical, or data manipulation was zero.

Carry/Borrow (C) – When set, this bit indicates that a carry or borrow out of the Arithmetic Logic Unit (ALU) occurred during the last arithmetic operation. This bit is also affected during bit test and branch instructions plus shifts and rotates.

TIMER

The timer circuitry for the EF6805U3 is shown in Figure 10. The timer contains a single 8-bit software programmable counter with a 7-bit software selectable prescaler. The counter may be preset under program control and decrements toward zero. When the counter decrements to zero, the timer interrupt request bit, i.e., bit 7 of the timer control register (TCR), is set. Then if the timer interrupt is not masked, i.e., bit 6 of the TCR and the I bit in the condition code register are both cleared, the processor receives an interrupt. After completion of the current instruction, the processor proceeds to store the appropriate registers on the stack, and then fetches the **timer interrupt vector from locations $FF8 and $FF9 in** order to begin servicing the interrupt.

The counter continues to count after it reaches zero, allowing the software to determine the number of internal or external input clocks since the timer interrupt request bit was set. The counter may be read at any time by the processor without disturbing the count. The contents of the counter become stable prior to the read portion of a cycle and do not change during the read. The timer interrupt request bit remains set until cleared by the software. If a write occurs before the timer interrupt is serviced, the interrupt is lost. TCR7 may also be used as a scanned status **bit in a non-interrupt mode of operation (TCR6 = 1).**

The prescaler is a 7-bit divider which is used to extend the maximum length of the timer. Bit 0, bit 1, and bit 2 of the TCR are programmed to choose the appropriate prescaler output which is used as the counter input. The processor cannot write into or read from the prescaler ; however, its contents are cleared to all zeros by the write operation into TCR when bit 3 of the written data equals one, which allows for truncation-free counting.

The timer input can be configured for three different operating modes, plus a disable mode, depending on the value written to the TCR4 and TCR5 control bits. For further information see Figure 9.

Timer Input Mode 1 – If TCR5 and TCR4 are both programmed to a zero, the input to the timer is from an internal clock and the external TIMER input is disabled. The internal clock mode can be used for periodic interrupt generation, as well as a reference in frequency and event measurement. The internal clock is the instruction cycle clock.

Timer Input Mode 2 – With TCR5 = 0 and TCR4 = 1, the internal clock and the TIMER input pin are ANDed to form the timer input signal. This mode can be used to measure external pulse widths. The external timer input pulse simply turns on the internal clock for the duration of the pulse widths.

Timer Input Mode 3 – If TCR5 = 1 and TCR4 = 0, then all inputs to the timer are disabled.

Timer Input Mode 4 – If TCR5 = 1 and TCR4 = 1, the internal clock input to the timer is disabled and the TIMER input pin becomes the input to the timer. The external TIMER pin can, in this mode, be used to count external events as well as external frequencies for generating periodic interrupts.

7	6	5	4	3	2	1	0	
TCR7	TCR6	TCR5	TCR4	TCR3*	TCR2	TCR1	TCR0	$009

*Write only (read as zero).

TCR7 – Timer Interrupt Request Bit :
1 - Set when TDR goes to zero, or under program control
0 - Cleared on external Reset, Power-On-Reset, or under Program Control.

TCR6 – Timer Interrupt Mask Bit :
1 - Timer Interrupt masked (disabled) Set on external Reset, Power-On-Reset, or under Program Control
0 - Cleared under Program Control.

TCR5 – External or Internal Clock Source Bit :
1 - External Clock Source. Set on external Reset, Power On-Reset, or under Program Control
0 - Cleared under Program Control.

TCR4 – External Enable Bit :
1 - Enable external TIMER pin. Set on external Reset, Power-On-Reset, or under Program Control.
0 - Cleared under Program Control.

TCR5	TCR4	Result
0	0	Internal clock to timer
0	1	AND of internal clock and TIMER pin to timer
1	0	Input to timer disabled
1	1	TIMER pin to timer

TCR3 – Timer prescaler reset bit : A read of TCR3 always indicates a zero.
1 - Set on external Reset, Power-On-Reset or under Program Control.
0 - Cleared under Program Control.

TCR2, TCR1, and TCR0 – Prescaler address bits :
1 - All set on external Reset, Power-On-Reset or under Program Control.
0 - Cleared under Program Control.

TCR2	TCR1	TCR0	Result
0	0	0	÷1
0	0	1	÷2
0	1	0	÷4
0	1	1	÷8

TCR2	TCR1	TCR0	Result
1	0	0	÷16
1	0	1	÷32
1	1	0	÷64
1	1	1	÷128

FIGURE 9 – TIMER CONTROL REGISTER (TCR)

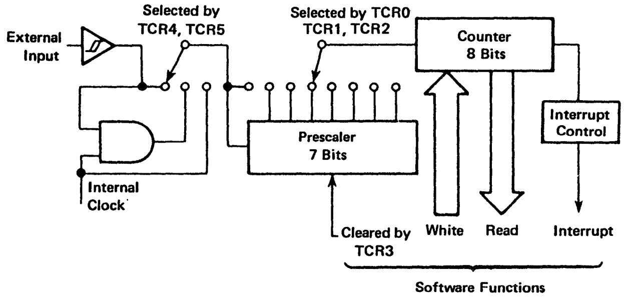


FIGURE 10 – TIMER BLOCK DIAGRAM

NOTES:
1. Prescaler and 8-bit counter are clocked on the failing edge of the internal clock (AS) or external input.
2. Counter is written to during data strobe (DS) and counts down continuously.

SELF-CHECK – The self-check capability of the EF6805U3 MCU provides an internal check to determine if the part is functional. Connect the MCU as shown in Figure 11 and monitor the output of Port C bit 3 for an oscillation of approximately 7 Hz. A 10-volt level (through a 10 k resistor) on the timer input, pin 8 and pressing then releasing the $\overline{\text{RESET}}$ button, energizes the ROM-based sèlf-check feature. The self-check program exercices the RAM, ROM, TIMER, interrupts, and I/O ports.

Several of the self-check subroutines can be called by a user program with a JSR or BSR instruction. They are the RAM, ROM. The timer routine may also be called if the timer input is the internal $\phi 2$ clock.

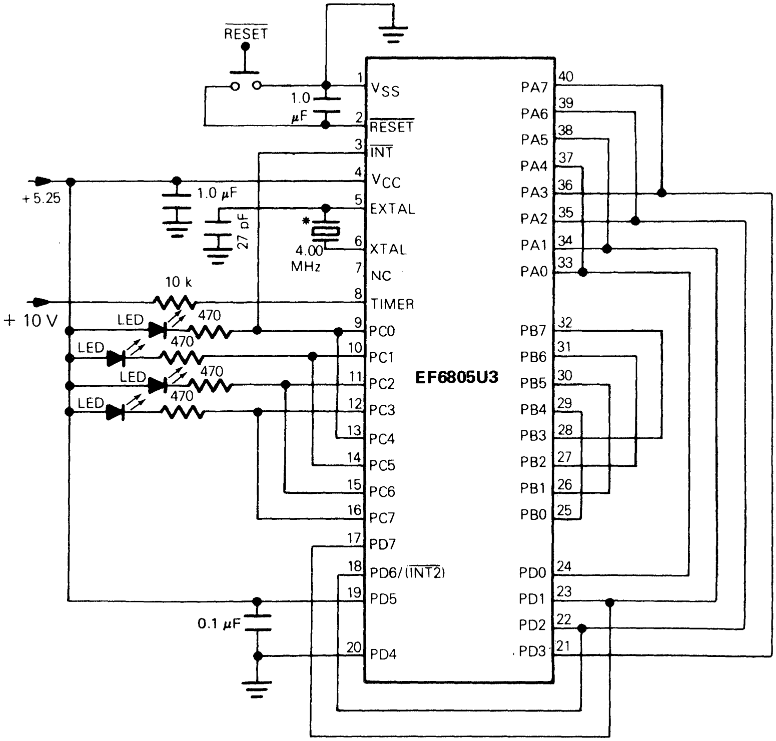
To call those subroutines in customer applications, please contact your local THOMSON SEMICONDUCTEURS sales office in order to obtain the complete description of the self-check program and the entrance/exit conditions.

RAM SELF-CHECK SUBROUTINE – The RAM self-check is called at location $F84 and returns with the Z bit clear if any error is detected ; otherwise the Z bit is set. The RAM test causes each byte to count from 0 up to 0 again with a check after each count.

The RAM test must be called with the stack pointer at $07F and A = 0. When run, the test checks every RAM cell except for $07F and $07E which are assumed to contain the return address.

The A and X registers and all RAM locations except $07F and $07E are modified.

ROM CHECKSUM SUBROUTINE – The ROM self-check is called at location $F95. The A register should be cleared before calling the routine. If any error is detected, it returns with the Z bit cleared ; otherwise Z = 1, X = 0 on return, and A is zero if the test passes. RAM location $040 to $043 is overwritten. The checksum is the complement of the execution OR of the contents of the user ROM.



*This connection depends on clock oscillator user selectable mask option. Use jumper if the RC mask option is selected.

FIGURE 11 – SELF-CHECK CONNECTIONS

LED Meanings

PC0	PC1	PC2	PC3	Remarks [1:LED ON; 0:LED OFF]
1	0	1	0	Bad I/O
0	0	1	0	Bad Timer
1	1	0	0	Bad RAM
0	1	0	0	Bad ROM
0	0	0	0	Bad Interrupts or Request Flag
All Flashing				Good Device

Anything else bad Device, Bad Port C, etc.

TIMER SELF-CHECK SUBROUTINE – The timer self-check is called at location $F6D and returns with the Z bit cleared if any error was found; otherwise Z = 1.

In order to work correctly as a user subroutine, the internal $\phi 2$ clock must be the clocking source and interrupts must be disabled. Also, on exit, the clock is running and the interrupt mask is not set so the caller must protect from interrupts if necessary.

The A and X register contents are lost. This routine sets the prescaler for divide-by-128 and the timer data register is cleared. The X register is configured to count down the same as the timer data register. The two registers are then compared every 128 cycles until they both count down to zero. Any mismatch during the count down is considered as an error. The A and X registers are cleared on exit from the routine.

RESET

The MCU can be reset three ways : by initial powerup, by the external reset input ($\overline{\text{RESET}}$) and by an optional internal low-voltage detect circuit. The $\overline{\text{RESET}}$ input consists mainly of a Schmitt trigger which senses the $\overline{\text{RESET}}$ line logic level. A typical reset Schmitt trigger hysteresis curve is shown in Figure 12. The Schmitt trigger provides an internal reset voltage if it senses a logical zero on the $\overline{\text{RESET}}$ pin.

Power-On Reset (POR) – An internal reset is generated upon powerup that allows the internal clock generator to stabilize. A delay of t_{RHL} milliseconds is required before allowing the $\overline{\text{RESET}}$ input to go high. Refer to the power and reset timing diagram of Figure 13. Connecting a capacitor to the $\overline{\text{RESET}}$ input (as illustrated in Figure 14) typically provides sufficient delay. During powerup, the Schmitt trigger switches on (removes reset) when $\overline{\text{RESET}}$ rises to V_{IRES+}.

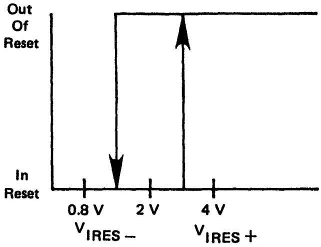


FIGURE 12 – TYPICAL RESET SCHMITT TRIGGER HYSTERESIS

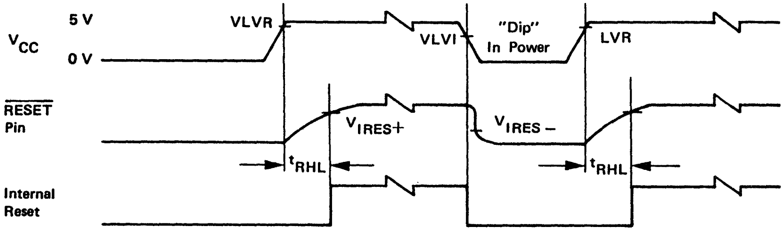


FIGURE 13 – POWER AND RESET TIMING

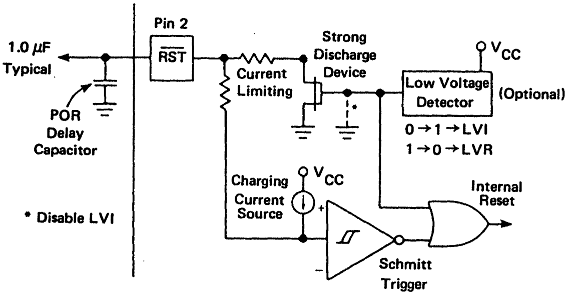


FIGURE 14 – $\overline{\text{RESET}}$ CONFIGURATION

External Reset Input – The MCU will be reset if a logical zero is applied to the $\overline{RESET}$ input for a period longer than one machine cycle (t_{cyc}). Under this type of reset, the Schmitt trigger switches off at V_{IRES-} to provide an internal reset voltage.

Low-Voltage Inhibit (LVI) — The optional low-voltage detection circuit causes a reset of the MCU if the power supply voltage falls below a certain level (V_{LVI}). The only requirement is that V_{CC} remains at or below the V_{LVI} threshold for one t_{cyc} minimum. In typical applications, the V_{CC} bus filter capacitor will eliminate negative-going voltage glitches of less than one t_{cyc}. The output from the low-voltage detector is connected directly to the internal reset circuitry. It also forces the $\overline{RESET}$ pin low via a strong discharge device through a resistor. The internal reset will be removed once the power supply voltage rises above a recovery level (V_{LVR}), at which time a normal power-on-reset occurs.

INTERNAL CLOCK GENERATOR OPTIONS

The internal clock generator circuit is designed to require a minimum of external components. A crystal, a resistor, a jumper wire, or an external signal may be used to generate a system clock with various stability/cost tradeoffs. **The oscillator frequency is internally divided by four to produce the internal system clocks. A manufacturing mask option is used to select crystal or resistor operation.**

The different connection methods are shown in Figure 15. Crystal specifications and suggested PC board layouts are given in Figure 16. A resistor selection graph is given in Figure 17.

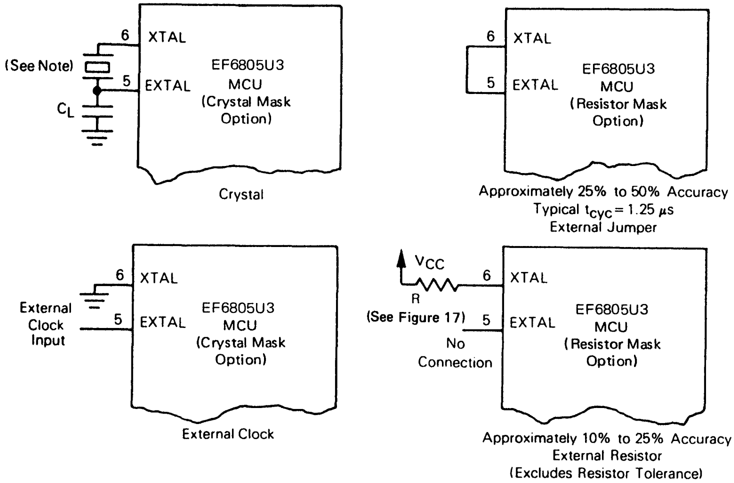
The crystal oscillator start-up time is a function of many variables : crystal parameters (especially R_S), oscillator load capacitances, IC parameters, ambient temperature, and supply voltage. To ensure rapid oscillator start up, neither the crystal characteristics nor the load capacitances should exceed recommendations.

When utilizing the on-board oscillator, the MCU should remain in a reset condition (reset pin voltage below V_{IRES+}) until the oscillator has stabilized at its operating frequency. Several factors are involved in calculating the external reset capacitor required to satisfy this condition: the oscillator start-up voltage, the oscillator stabilization time, the minimum V_{IRES+}, and the reset charging current specification.

Once V_{CC} minimum is reached, the external $\overline{RESET}$ capacitor will begin to charge at a rate dependent on the capacitor value. The charging current is supplied from V_{CC} through a large resistor, so it appears almost like a constant current source until the reset voltage rises above V_{IRES+}. Therefore, the $\overline{RESET}$ pin will charge at approximately :

$$(V_{IRES+})\bullet C_{ext} = I_{RES} \bullet t_{RHL}$$

Assuming the external capacitor is initially discharged.



Crystal

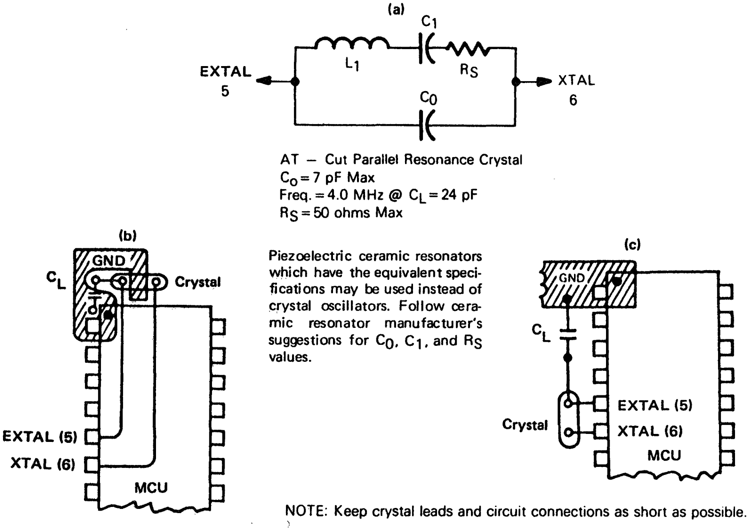
Approximately 25% to 50% Accuracy
Typical t_{cyc} = 1.25 µs
External Jumper

External Clock

Approximately 10% to 25% Accuracy
External Resistor
(Excludes Resistor Tolerance)

NOTE: The recommended C_L value with a 4.0 MHz crystal is 27 pF, maximum, including system distributed capacitance. There is an internal capacitance of approximately 25 pF on the XTAL pin. For crystal frequencies other than 4 MHz, the total capacitance on each pin should be scaled as the inverse of the frequency ratio. For example, with a 2 MHz crystal, use approximately 50 pF on EXTAL and approximately 25 pF on XTAL. The exact value depends on the Motional-Arm parameters of the crystal used.

FIGURE 15 – CLOCK GENERATOR OPTIONS



AT – Cut Parallel Resonance Crystal
$C_0 = 7$ pF Max
Freq. = 4.0 MHz @ $C_L = 24$ pF
$R_S = 50$ ohms Max

Piezoelectric ceramic resonators which have the equivalent specifications may be used instead of crystal oscillators. Follow ceramic resonator manufacturer's suggestions for C_0, C_1, and R_S values.

NOTE: Keep crystal leads and circuit connections as short as possible.

FIGURE 16 – CRYSTAL MOTIONAL ARM PARAMETERS AND SUGGESTED PC BOARD LAYOUT

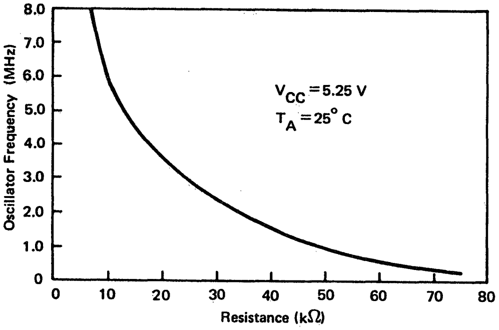


FIGURE 17 – TYPICAL FREQUENCY SELECTION FOR RESISTOR (OSCILLATOR OPTION)

INTERRUPTS

The microcomputers can be interrupted four different ways : through the external interrupt ($\overline{\text{INT}}$) input pin, the internal timer interrupt request, the external port D bit 6 ($\overline{\text{INT2}}$) input pin, or the software interrupt instruction (SWI). When any interrupt occurs : the current instruction (including SWI) is completed, processing is suspended, the present CPU state is pushed onto the stack, the interrupt bit (I) in the condition code register is set, the address of the interrupt routine is obtained from the appropriate interrupt vector address, and the interrupt routine is executed. Stacking the CPU register, setting the I bit, and vector fetching require a total of 11 t_{cyc} periods for completion. A flowchart of the interrupt sequence is shown in Figure 18. The interrupt service routine must end with a return from interrupt (RTI) instruction which allows the MCU to resume processing of the program prior to the interrupt (by unstacking the previous CPU state). Unlike $\overline{\text{RESET}}$, hardware interrupts do not cause the current instruction execution to be halted, but are considered pending until the current instruction execution is complete.

When the current instruction is complete, the processor checks all pending hardware interrupts and if unmasked, proceeds with interrupt processing ; otherwise the next instruction is fetched and executed. Note that masked interrupts are latched for later interrupt service.

If both an external interrupt and a timer interrupt are pending at the end of an instruction execution, the external interrupt is serviced first. The SWI is executed as any other instruction.

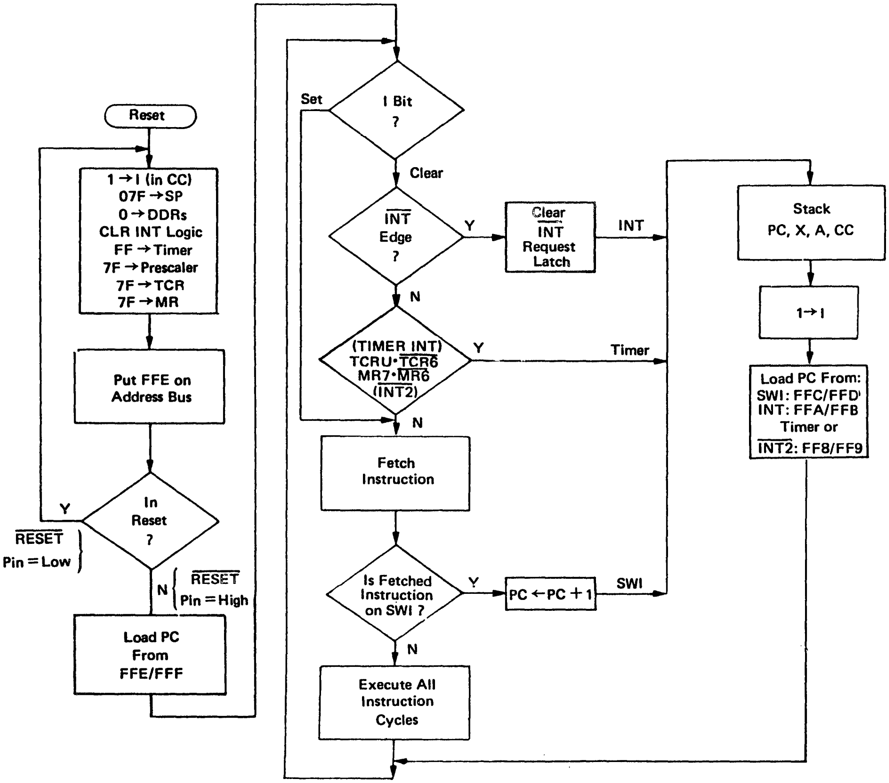


FIGURE 18 – $\overline{\text{RESET}}$ AND INTERRUPT PROCCESSING FLOWCHARD

NOTE

The timer and $\overline{\text{INT2}}$ interrupts share the same vector address. The interrupt routine must determine the source by examining the interrupt request bits (TCR b7 and MR b7). Both TCR b7 and MR b7 can only be written to zero by software.

The external interrupt, $\overline{\text{INT}}$ and $\overline{\text{INT2}}$, are synchronized and then latched on the falling edge of the input signal. The $\overline{\text{INT2}}$ interrupt has an interrupt request bit (bit 7) and a mask bit (bit 6) located in the miscellaneous register (MR). The $\overline{\text{INT2}}$ interrupt is inhibited when the mask bit is set. The $\overline{\text{INT2}}$ is always read as a digital input on port D. The $\overline{\text{INT2}}$ and timer interrupt request bits, if set, cause the MCU to process an interrupt when the condition code I bit is clear.

A sinusoidal input signal (f_{INT} maximum) can be used to generate an external interrupt for use as a zero-crossing detector. This allows applications such as servicing time-of-day routines and engaging/disengaging ac power control devices. Off-chip full wave rectification provides an interrupt at every zero crossing of the ac signal and thereby provides a 2f clock. See Figure 19.

NOTE

The $\overline{\text{INT}}$ (pin 3) is internally biased at approximately 2.2 V due to the internal zero-crossing detection.

A software interrupt (SWI) is an executable instruction which is executed regardless of the state of the I bit in the condition code register. SWIs are usually used as breakpoints for debugging or as system calls.

A/D CONTROL REGISTER (ACR)

7	6	5	4	3	2	1	0	
	1	1	1	1				$00E

Bit 7 – Conversion Complete Status Flag: Set when conversion is complete; cleared only on a write to ACR.
Readable, not writable.

Bits 2, 1, 0 – A/D input Mux Selection (See Table 2).
Bits 6, 5, 4, 3 read as "1s" – unused bits.

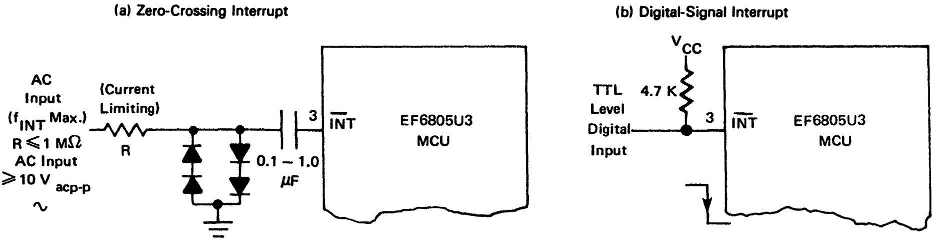


FIGURE 19 – TYPICAL INTERRUPT CIRCUITS

INPUT/OUTPUT CIRCUITRY

There are 32 input/output pins. The $\overline{\text{INT}}$ pin may be polled with branch instructions to provide an additional input pin. All pins on ports A, B, and C are programmable as either inputs or outputs under software control of the corresponding data direction register (DDR). See below I/O port control registers configuration. The port I/O programming is accomplished by writing the corresponding bit in the port DDR to a logic one for output or a logic zero for input. On reset all the DDRs are initialized to a logic zero state, placing the ports in the input mode. The port output registers are not initialized on reset and should be initialized by software before changing the DDRs from input to output. A read operation on a port programmed as an output will read the contents of the output latch regardless of the logic levels at the output pin, due to output loading. Refer to Figure 20.

PORT DATA REGISTER

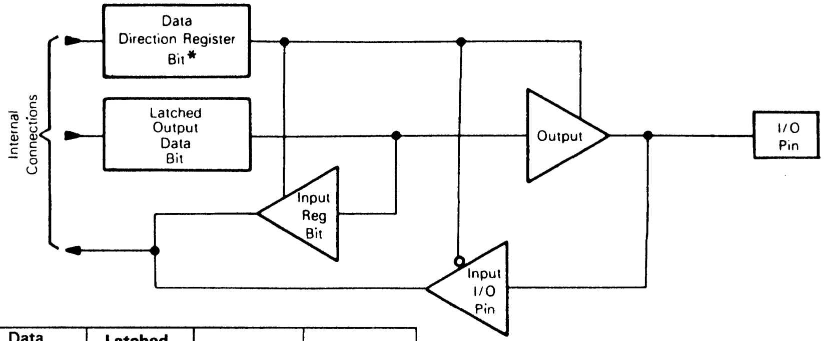
7 … 0

Port A Addr = $000
Port B Addr = $001
Port C Addr = $002
Port D Addr = $003

PORT DATA DIRECTION REGISTER (DDR)

7 … 0

(1) Write Only; reads as all "1s"
(2) 1 = Output, 0 = Input Cleared to 0 by Reset
(3) Port A Addr = $004
Port B Addr = $005
Port C Addr = $006



Data Direction Register Bit	Latched Output Data Bit	Output State	Input To MCU
1	0	0	0
1	1	1	1
0	X	High-Z **	Pin

*DDR is a write-only register and reads as all "1s"
** Ports B, and C are three-state ports.
Port A has optional internal pullup devices to provide CMOS data drive capability. See Electrical Characteristics tables for complete information.

FIGURE 20 – TYPICAL PORT I/O CIRCUITRY

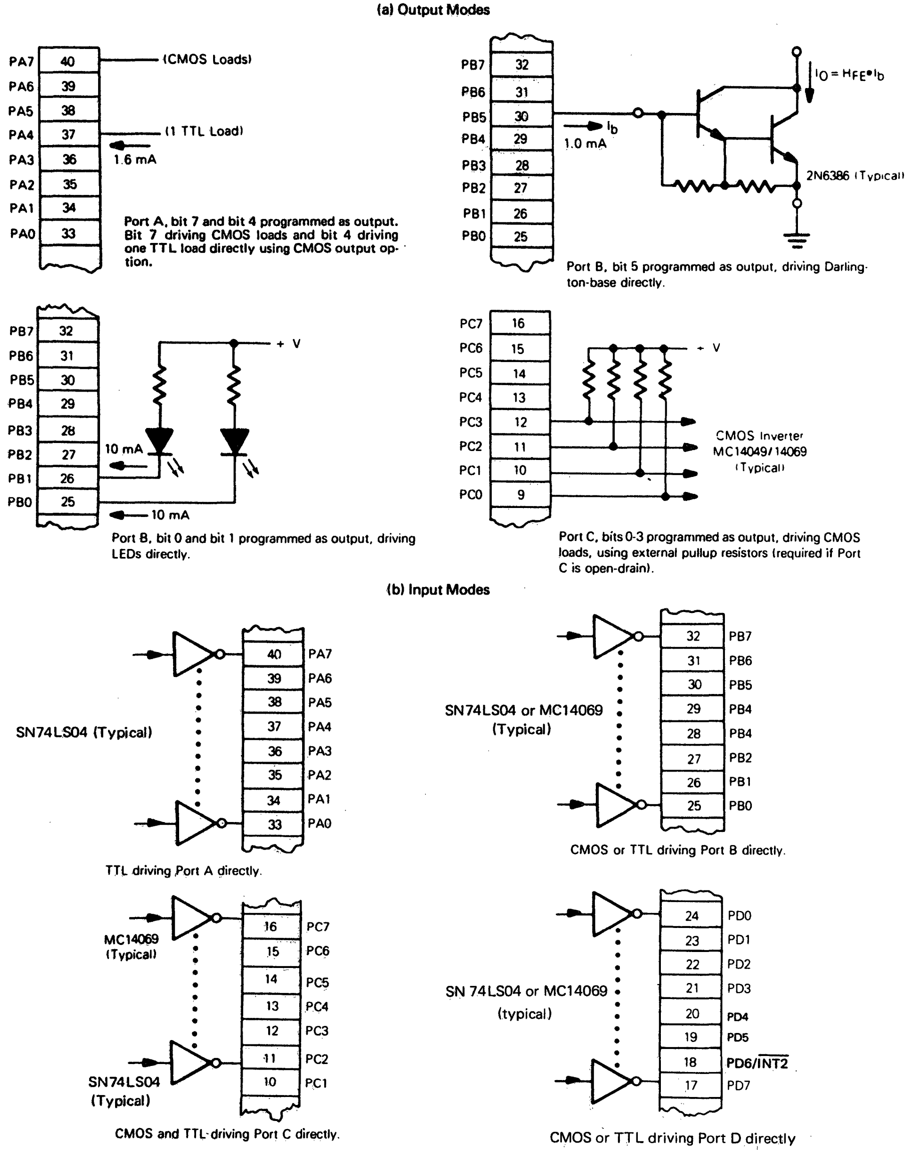
All input/output lines are TTL compatible as both inputs and outputs. Port A lines are CMOS compatible as outputs (mask option) while port B, C, and D lines are CMOS compatible as inputs. Port D lines are input only; thus, there is no corresponding DDR. When programmed as outputs, port B is capable of sinking 10 milliamperes and sourcing 1 milliampere on each pin.

The address map (Figure 6) gives the addresses of data registers and data direction registers. Figure 21 provides some examples of port connections.

CAUTION

The corresponding DDRs for ports A, B, and C are write-only registers (registers at **$004, $005, $006**). A read operation on these registers is undefined. Since BSET and BCLR are read-modify-write in function, they cannot be used to set or clear a single DDR bit (all "unaffected" bits would be set). It is recommended that all DDR bits in a port be written using a single-store instruction.

The latched output data bit (see Figure 20) must always be written. Therefore, any write to a port writes all of its data bits even though the port DDR is set to input. This may be used to initialize the data register and avoid undefined outputs ; however, care must be exercised when using read-modify-write instructions, since the data read corresponds to the pin level if the DDR is an input (zero) and corresponds to the latched output data when the DDR is an output (one).



Port A, bit 7 and bit 4 programmed as output. Bit 7 driving CMOS loads and bit 4 driving one TTL load directly using CMOS output option.

Port B, bit 5 programmed as output, driving Darlington-base directly.

Port B, bit 0 and bit 1 programmed as output, driving LEDs directly.

Port C, bits 0-3 programmed as output, driving CMOS loads, using external pullup resistors (required if Port C is open-drain).

TTL driving Port A directly.

CMOS or TTL driving Port B directly.

CMOS and TTL driving Port C directly.

CMOS or TTL driving Port D directly

FIGURE 21 – TYPICAL PORT CONNECTIONS

BIT MANIPULATION

The EF6805' 3 has the ability to set or clear any single RAM or I/O bit (except the data direction registers) with a single instruction (BSET, BCLR) (see Caution below). Any bit in page zero can be tested using the BRSET and BRCLR instructions and the program branches as a result of its state. The carry bit equals the value of the bit references by BRSET or BRCLR. The capability to working with any bit in RAM, ROM, or I/O allows the user to have individual flags in RAM or to handle single I/O bits as control lines.

CAUTION

The corresponding data direction registers for ports A, B, and C are write-only registers (locations $004, $005, and $006). A read operation on these registers is undefined. Since BSET and BCLR are read-modify-write functions, they cannot be used to set a data direction register bit (all "unaffected" bits would be set). It is recommended that all data direction register bits in a port be written using a single-store instruction.

The coding examples shown in Figure 22 illustrate the usefulness of the bit manipulation and test instruction. Assume that the microcomputer is to communicate with an external serial device. The external device has a data ready signal, a data output line, and a clock line to clock data one bit at a time, least significant bit first out of the device. The microcomputer waits until the data is ready, clocks the external device, picks up the data in the carry flag, clears the clock line, and finally accumulates the data bit in a random-access memory location.

```
*       BRSET    2,PORTA, *          WAIT FOR READY

        BSET     1, PORTA            CLOCK NEXT BIT IN
        BRCLR    0, PORTA, NEXT      PICKUP BIT IN C-BIT
NEXT    BCLR     1, PORTA            RETURN CLOCK LINE HIGH
        ASR      RAMLOC              MOVE C-BIT INTO RAM
```

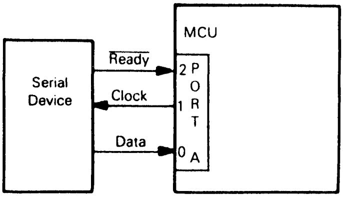


FIGURE 22 – BIT MANIPULATION EXAMPLE

ADDRESSING MODES

The EF6805U3 MCU has ten addressing modes available for use by the programmer. They are explained briefly in the following paragraphs. For additional details and graphical illustrations, refer to the EF6805 Family Users Manual.

The term "effective address" (EA) is used in describing the addressing modes. EA is defined as the address from which the argument for an instruction is fetched or stored.

IMMEDIATE – In the immediate addressing mode, the operand is contained in the byte immediately following the opcode. The immediate addressing mode is used to access constants which do not change during program execution (e.g., a constant used to initialize a loop counter).

DIRECT – In the direct addressing mode, the effective address of the argument is contained in a single byte following the opcode byte. Direct addressing allows the user to directly address the lowest 256 bytes in memory with a single 2-byte instruction. This address area includes all on-chip RAM and I/O registers and 128 bytes of ROM. Direct addressing is an effective use of both memory and time.

EXTENDED – In the extended addressing mode, the effective address of the argument is contained in the two bytes following the opcode. Instructions with extended addressing mode are capable of referencing arguments anywhere in memory with a single 3-byte instruction. When using the assembler, the user need not specify whether an instruction uses direct or extended addressing. The assembler automatically selects the shortest form of the instruction.

RELATIVE – The relative addressing mode is only used in branch instructions. In relative addressing, the contents of the 8-bit signed byte following the opcode (the offset) is added to the PC if, and only if, the branch condition is true. Otherwise, control proceeds to the next instruction. The span of relative addressing is from – 126 to + 129 from the opcode address. The programmer need not worry about calculating the correct offset if he uses the assembler, since it calculates the proper offset and checks to see if it is within the span of the branch.

INDEXED, NO OFFSET – In the indexed, no offset addressing mode, the effective address of the argument is contained in the 8-bit index register. Thus, this addressing mode can access the first 256 memory locations. These instructions are only one byte long. This mode is often used to move a pointer through a table or to hold the address of a frequently referenced RAM or I/O location.

INDEXED, 8-BIT OFFSET – In the indexed, 8-bit offset addressing mode, the effective address is the sum of the contents of the unsigned 8-bit index register and the unsigned byte following the opcode. This addressing mode is useful in selecting the kth element in an n element table. With this 2-byte instruction, k would typically be in X with the address of the beginning of the table in the instruction. As such, tables may begin anywhere within the first 256 addressable locations and could extend as far as location 510 ($1FE is the last location at which the instruction may begin).

INDEXED, 16-BIT OFFSET – In the indexed, 16-bit offset addressing mode, the effective address is the sum of the contents of the unsigned 8-bit index register and the two unsigned bytes following the opcode. This addressing mode can be used in a manner similar to indexed, 8-bit offset except that this 3-byte instruction allows tables to be anywhere in memory. As with direct and extended, the assembler determines the shortest form of indexed addressing.

BIT SET/CLEAR – In the bit set/clear addressing mode, the bit to be set or cleared is part of the opcode, and the byte following the opcode specifies the direct address of the byte in which the specified bit is to be set or cleared. Thus, any read/write bit in the first 256 locations of memory, including I/O, can be selectively set or cleared with a single 2-byte instruction.

CAUTION

The corresponding DDRs for ports A, B, and C are write-only registers (registers at $004, $005, $006). A read operation on these registers is undefined. Since

BSET and BCLR are read-modify-write in function, they cannot be used to set or clear a single DDR bit (all "unaffected" bits would be set). It is recommended that all DDR bits in a port must be written using a single-store instruction.

BIT TEST AND BRANCH – The bit test and branch addressing mode is a combination of direct addressing and relative addressing. The bit which is to be tested and condition (set or clear) is included in the opcode, and the address of the byte to be tested is in the single byte immediately following the opcode byte. The signed relative 8-bit offset in the third byte is added to the PC if the specified bit is set or cleared in the specified memory location. This single 3-byte instruction allows the program to branch based on the condition of any readable bit in the first 256 locations of memory. The span of branching is from –125 to +130 from the opcode address. The state of the tested bit is also transferred to the carry bit of the condition code registers.

CAUTION

The corresponding DDRs for ports A, B, and C are write-only registers (registers at $004, $005, $006). A read operation on these registers is undefined. Since BSET and BCLR are read-modify-write in function, they cannot be used to set or clear a single DDR bit (all "unaffected" bits would be set). It is recommended that all DDR bits in a port must be written using a single-store instruction.

INHERENT – In the inherent addressing mode, all the information necessary to execute the instruction is contained in the opcode. Operations specifying only the index register or accumulator, as well as control instructions with no other arguments, are included in this mode. These instructions are one byte long.

INSTRUCTION SET

The EF6805U3 MCU has a set of 59 basic instructions, which when combined with the 10 addressing modes produce 207 usable opcodes. They can be divided into five different types: register/memory, read-modify-write, branch, bit manipulation, and control. The following paragraphs briefly explain each type. All the instructions within a given type are presented in individual tables.

REGISTER/MEMORY INSTRUCTIONS – Most of these instructions use two operands. One operand is either the accumulator or the index register. The other operand is obtained from memory using one of the addressing modes. The jump unconditional (JMP) and jump to subroutine (JSR) instructions have no register operand. Refer to Table 1.

READ-MODIFY-WRITE INSTRUCTIONS – These instructions read a memory location or a register, modify or test its contents, and write the modified value back to memory or to the register; see Caution under Input/Output section. The test for negative or zero (TST) instruction is included in the read-modify-write instruction though it does not perform the write. Refer to Table 2.

BRANCH INSTRUCTIONS – The branch instructions cause a branch from the program when a certain condition is met. Refer to Table 3.

BIT MANIPULATION INSTRUCTIONS – The instructions are used on any bit in the first 256 bytes of the memory; One group either sets or clears. The other group performs the bit test and branch operations. Refer to Table 4.

CAUTION

The corresponding DDRs for ports A, B, and C are write-only registers (registers at $004, $005, $006). A read operation on these registers is undefined. Since BSET and BCLR are read-modify-write in function, they cannot be used to set or clear a single DDR bit (all "unaffected" bits would be set). It is recommended that all DDR bits in a port must be written using a single-store instruction.

CONTROL INSTRUCTION – The control instructions control the MCU operations during program execution. Refer to Table 5.

ALPHABETICAL LISTING – The complete instruction set is given in alphabetical order in Table 6.

OPCODE MAP – Table 7 is an opcode map for the instruction used on the MCU.

TABLE 1 — REGISTER/MEMORY INSTRUCTIONS

		Addressing Modes																	
		Immediate			Direct			Extended			Indexed (No Offset)			Indexed (8-Bit Offset)			Indexed (16-Bit Offset)		
Function	Mnemonic	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles	OP Code	# Bytes	# Cycles
Load A from Memory	LDA	A6	2	2	B6	2	4	C6	3	5	F6	1	4	E6	2	5	D6	3	6
Load X from Memory	LDX	AE	2	2	BE	2	4	CE	3	5	FE	1	4	EE	2	5	DE	3	6
Store A in Memory	STA	—	—		B7	2	5	C7	3	6	F7	1	5	E7	2	6	D7	3	7
Store X in Memory	STX	—	—	-	BF	2	5	CF	3	6	FF	1	5	EF	2	6	DF	3	7
Add Memory to A	ADD	AB	2	2	BB	2	4	CB	3	5	FB	1	4	EB	2	5	DB	3	6
Add Memory and Carry to A	ADC	A9	2	2	B9	2	4	C9	3	5	F9	1	4	E9	2	5	D9	3	6
Subtract Memory	SUB	A0	2	2	B0	2	4	C0	3	5	F0	1	4	E0	2	5	D0	3	6
Subtract Memory from A with Borrow	SBC	A2	2	2	B2	2	4	C2	3	5	F2	1	4	E2	2	5	D2	3	6
AND Memory to A	AND	A4	2	2	B4	2	4	C4	3	5	F4	1	4	E4	2	5	D4	3	6
OR Memory with A	ORA	AA	2	2	BA	2	4	CA	3	5	FA	1	4	EA	2	5	DA	3	6
Exclusive OR Memory with A	EOR	A8	2	2	B8	2	4	C8	3	5	F8	1	4	E8	2	5	D8	3	6
Arithmetic Compare A with Memory	CMP	A1	2	2	B1	2	4	C1	3	5	F1	1	4	E1	2	5	D1	3	6
Arithmetic Compare X with Memory	CPX	A3	2	2	B3	2	4	C3	3	5	F3	1	4	E3	2	5	D3	3	6
Bit Test Memory with A (Logical Compare)	BIT	A5	2	2	B5	2	4	C5	3	5	F5	1	4	E5	2	5	D5	3	6
Jump Unconditional	JMP	—	—	--	BC	2	3	CC	3	4	FC	1	3	EC	2	4	DC	3	5
Jump to Subroutine	JSR	—	--		BD	2	7	CD	3	8	FD	1	7	ED	2	8	DD	3	9

TABLE 2 — READ-MODIFY-WRITE INSTRUCTIONS

		Addressing Modes														
		Inherent (A)			Inherent (X)			Direct			Indexed (No Offset)			Indexed (8 Bit Offset)		
Function	Mnemonic	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles
Increment	INC	4C	1	4	5C	1	4	3C	2	6	7C	1	6	6C	2	7
Decrement	DEC	4A	1	4	5A	1	4	3A	2	6	7A	1	6	6A	2	7
Clear	CLR	4F	1	4	5F	1	4	3F	2	6	7F	1	6	6F	2	7
Complement	COM	43	1	4	53	1	4	33	2	6	73	1	6	63	2	7
Negate (2's Complement)	NEG	40	1	4	50	1	4	30	2	6	70	1	6	60	2	7
Rotate Left Thru Carry	ROL	49	1	4	59	1	4	39	2	6	79	1	6	69	2	7
Rotate Right Thru Carry	ROR	46	1	4	56	1	4	36	2	6	76	1	6	66	2	7
Logical Shift Left	LSL	48	1	4	58	1	4	38	2	6	78	1	6	68	2	7
Logical Shift Right	LSR	44	1	4	54	1	4	34	2	6	74	1	6	64	2	7
Arithmetic Shift Right	ASR	47	1	4	57	1	4	37	2	6	77	1	6	67	2	7
Test for Negative or Zero	TST	4D	1	4	5D	1	4	3D	2	6	7D	1	6	6D	2	7

TABLE 3 — BRANCH INSTRUCTIONS

		Relative Addressing Mode		
Function	Mnemonic	Op Code	# Bytes	# Cycles
Branch Always	BRA	20	2	4
Branch Never	BRN	21	2	4
Branch IFF Higher	BHI	22	2	4
Branch IFF Lower or Same	BLS	23	2	4
Branch IFF Carry Clear	BCC	24	2	4
(Branch IFF Higher or Same)	(BHS)	24	2	4
Branch IFF Carry Set	BCS	25	2	4
(Branch IFF Lower)	(BLO)	25	2	4
Branch IFF Not Equal	BNE	26	2	4
Branch IFF Equal	BEQ	27	2	4
Branch IFF Half Carry Clear	BHCC	28	2	4
Branch IFF Half Carry Set	BHCS	29	2	4
Branch IFF Plus	BPL	2A	2	4
Branch IFF Minus	BMI	2B	2	4
Branch IFF Interupt Mask Bit is Clear	BMC	2C	2	4
Branch IFF Interrupt Mask Bit is Set	BMS	2D	2	4
Branch IFF Interrupt Line is Low	BIL	2E	2	4
Branch IFF Interrupt Line is High	BIH	2F	2	4
Branch to Subroutine	BSR	AD	2	8

TABLE 4 — BIT MANIPULATION INSTRUCTIONS

		Addressing Modes					
		Bit Set/Clear			Bit Test and Branch		
Function	Mnemonic	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles
Branch IFF Bit n is set	BRSET n (n = 0...7)	—	—	—	2 • n	3	10
Branch IFF Bit n is clear	BRCLR n (n = 0...7)	—	—	—	01 + 2 • n	3	10
Set Bit n	BSET n (n = 0...7)	10 + 2 • n	2	7	—	—	—
Clear bit n	BCLR n (n = 0....7)	11 + 2 • n	2	7	—	—	—

TABLE 5 — CONTROL INSTRUCTIONS

		Inherent		
Function	Mnemonic	Op Code	# Bytes	# Cycles
Transfer A to X	TAX	97	1	2
Transfer X to A	TXA	9F	1	2
Set Carry Bit	SEC	99	1	2
Clear Carry Bit	CLC	98	1	2
Set Interrupt Mask Bit	SEI	9B	1	2
Clear Interrupt Mask Bit	CLI	9A	1	2
Software Interrupt	SWI	83	1	11
Return from Subroutine	RTS	81	1	6
Return from Interrupt	RTI	80	1	9
Reset Stack Pointer	RSP	9C	1	2
No-Operation	NOP	9D	1	2

TABLE 6 — INSTRUCTION SET

	Addressing Modes										Condition Code				
Mnemonic	Inherent	Immediate	Direct	Extended	Relative	Indexed (No Offset)	Indexed (8 Bits)	Indexed (16 Bits)	Bit Set/Clear	Bit Test & Branch	H	I	N	Z	C
ADC		X	X	X		X	X	X			∧	●	∧	∧	∧
ADD		X	X	X		X	X	X			∧	●	∧	∧	∧
AND		X	X	X		X	X	X			●	●	∧	∧	●
ASL	X		X			X	X				●	●	∧	∧	∧
ASR	X		X			X	X				●	●	∧	∧	∧
BCC					X						●	●	●	●	●
BCLR									X		●	●	●	●	●
BCS					X						●	●	●	●	●
BEQ					X						●	●	●	●	●
BHCC					X						●	●	●	●	●
BHCS					X						●	●	●	●	●
BHI					X						●	●	●	●	●
BHS					X						●	●	●	●	●
BIH					X						●	●	●	●	●
BIL					X						●	●	●	●	●
BIT		X	X	X		X	X	X			●	●	∧	∧	●
BLO					X						●	●	●	●	●
BLS					X						●	●	●	●	●
BMC					X						●	●	●	●	●
BMI					X						●	●	●	●	●
BMS					X						●	●	●	●	●
BNE					X						●	●	●	●	●
BPL					X						●	●	●	●	●
BRA					X						●	●	●	●	●
BRN					X						●	●	●	●	●
BRCLR										X	●	●	●	●	∧
BRSET										X	●	●	●	●	∧
BSET									X		●	●	●	●	●
BSR					X						●	●	●	●	●
CLL	X										●	●	●	●	O
CLI	X										●	O	●	●	●
CLR	X		X			X	X				●	●	O	1	●
CMP		X	X	X		X	X	X			●	●	∧	∧	∧
COM	X		X			X	X				●	●	∧	∧	1
CPX		X	X	X		X	X	X			●	●	∧	∧	∧
DEC	X		X			X	X				●	●	∧	∧	●
EOR		X	X	X		X	X	X			●	●	∧	∧	●
INC	X		X			X	X				●	●	∧	∧	●
JMP			X	X		X	X	X			●	●	●	●	●
JSR			X	X		X	X	X			●	●	●	●	●
LDA		X	X	X		X	X	X			●	●	∧	∧	●
LDX		X	X	X		X	X	X			●	●	∧	∧	●
LSL	X		X			X	X				●	●	∧	∧	∧
LSR	X		X			X	X				●	●	O	∧	∧
NEQ	X		X			X	X				●	●	∧	∧	∧
NOP	X										●	●	●	●	●
ORA		X	X	X		X	X	X			●	●	∧	∧	●
ROL	X		X			X	X				●	●	∧	∧	∧
RSP	X										●	●	●	●	●

Condition Code Symbols:

H	Half Carry (From Bit 3)	C	Carry/Borrow
I	Interrupt Mask	∧	Test and Set if True, Cleared Otherwise
N	Negative (Sign Bit)	•	Not Affected
Z	Zero		

TABLE 6 – INSTRUCTION SET (CONTINUED)

	Addressing Modes										Condition Code				
Mnemonic	Inherent	Immediate	Direct	Extended	Relative	Indexed (No Offset)	Indexed (8 Bits)	Indexed (16 Bits)	Bit Set/ Clear	Bit Test & Branch	H	I	N	Z	C
RTI	X										?	?	?	?	?
RTS	X										●	●	●	●	●
SBC		X	X	X		X	X	X			●	●	Λ	Λ	Λ
SEC	X										●	●	●	●	1
SEI	X										●	1	●	●	●
STA			X	X		X	X	X			●	●	Λ	Λ	●
STX			X	X		X	X	X			●	●	Λ	Λ	●
SUB		X	X	X		X	X	X			●	●	Λ	Λ	Λ
SWI	X										●	1	●	●	●
TAX	X										●	●	●	●	●
TST	X		X			X	X				●	●	Λ	Λ	●
TXA	X										●	●	●	●	●

Condition Code Symbols:

H	Half Carry (From Bit 3)	C	Carry/Borrow
I	Interrupt Mask	Λ	Test and Set if True, Cleared Otherwise
N	Negative (Sign Bit)	●	Not Affected
Z	Zero	?	Load CC Register From Stack

EF6805 HMOS FAMILY

Features	EF6805CT	EF6805P2	EF6805P4	EF6805P6	EF6805R2	EF6805R3	EF6805T2	EF6805U2	EF6805U3	EF6805TV
Technology	HMOS	HMOS	HMOS	HMOS	HMOS	HMOS	HMOS	HMOS	HMOS	HMOS
Number of Pins	40	28	28	28	40	40	28	40	40	40
On-Chip RAM (Bytes)	240	64	112	64	64	112	64	64	112	96
On-Chip User ROM (Bytes)	4096	1100	1100	1796	2048	3776	2508	2048	3776	6144
External Bus	Yes	None	None	None	None	None	None	None	None	None
Bidirectional I/O Lines	29	20	20	20	24	24	19	24	24	32
Unidirectional I/O Lines	None	None	None	None	6 Inputs	6 Inputs	None	8 Inputs	8 Inputs	None
Other I/O Features	Timer,UART	Timer	Timer	Timer	Timer,A/D	Timer,A/D	Timer, PLL	Timer	Timer	Timer,D/A
External Interrupt Inputs	3	1	1	1	2	2	2	2	2	1
STOP and WAIT	No	No	No	No	No	No	No	No	No	No

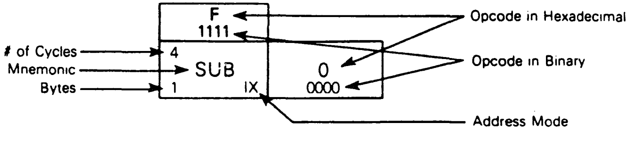
TABLE 7 – 6805 HMOS FAMILY INSTRUCTION SET OPCODE MAP

	Bit Manipulation		Branch	Read-Modify-Write					Control		Register/Memory						
	BTB	BSC	REL	DIR	INH	INH	IX1	IX	INH	INH	IMM	DIR	EXT	IX2	IX1	IX	
Hi Low	0 0000	1 0001	2 0010	3 0011	4 0100	5 0101	6 0110	7 0111	8 1000	9 1001	A 1010	B 1011	C 1100	D 1101	E 1110	F 1111	Hi Low
0 0000	10 BRSET0 3 BTB	7 BSET0 2 BSC	4 BRA 2 REL	6 NEG 2 DIR	4 NEG 1 INH	4 NEG 1 INH	7 NEG 2 IX1	6 NEG 1 IX	9 RTI 1 INH		2 SUB 2 IMM	4 SUB 2 DIR	5 SUB 3 EXT	6 SUB 3 IX2	5 SUB 2 IX1	4 SUB 1 IX	0 0000
1 0001	10 BRCLR0 3 BTB	7 BCLR0 2 BSC	4 BRN 2 REL						6 RTS 1 INH		2 CMP 2 IMM	4 CMP 2 DIR	5 CMP 3 EXT	6 CMP 3 IX2	5 CMP 2 IX1	4 CMP 1 IX	1 0001
2 0010	10 BRSET1 3 BTB	7 BSET1 2 BSC	4 BHI 2 REL								2 SBC 2 IMM	4 SBC 2 DIR	5 SBC 3 EXT	6 SBC 3 IX2	5 SBC 2 IX1	4 SBC 1 IX	2 0010
3 0011	10 BRCLR1 3 BTB	7 BCLR1 2 BSC	4 BLS 2 REL	6 COM 2 DIR	4 COMA 1 INH	4 COMX 1 INH	7 COM 2 IX1	6 COM 1 IX	11 SWI 1 INH		2 CPX 2 IMM	4 CPX 2 DIR	5 CPX 3 EXT	6 CPX 3 IX2	5 CPX 2 IX1	4 CPX 1 IX	3 0011
4 0100	10 BRSET2 3 BTB	7 BSET2 2 BSC	4 BCC 2 REL	6 LSR 2 DTR	4 LSRA 1 INH	4 LSRX 1 INH	7 LSR 2 IX1	6 LSR 1 IX			2 AND 2 IMM	4 AND 2 DIR	5 AND 3 EXT	6 AND 3 IX2	5 AND 2 IX1	4 AND 1 IX	4 0100
5 0101	10 BRCLR2 3 BTB	7 BCLR2 2 BSC	4 BCS 2 REL								2 BIT 2 IMM	4 BIT 2 DIR	5 BIT 3 EXT	6 BIT 3 IX2	5 BIT 2 IX1	4 BIT 1 IX	5 0101
6 0110	10 BRSET3 3 BTB	7 BSET3 2 BSC	4 BNE 2 REL	6 ROR 2 DIR	4 RORA 1 INH	4 RORX 1 INH	7 ROR 2 IX1	6 ROR 1 IX			2 LDA 2 IMM	4 LDA 2 DIR	5 LDA 3 EXT	6 LDA 3 IX2	5 LDA 2 IX1	4 LDA 1 IX	6 0110
7 0111	10 BRCLR3 3 BTB	7 BCLR3 2 BSC	4 BEQ 2 REL	6 ASR 2 DIR	4 ASRA 1 INH	4 ASRX 1 INH	7 ASR 2 IX1	6 ASR 1 IX		2 TAX 1 INH		5 STA 2 DIR	6 STA 3 EXT	7 STA 3 IX2	6 STA 2 IX1	5 STA 1 IX	7 0111
8 1000	10 BRSET4 3 BTB	7 BSET4 2 BSC	4 BHCC 2 REL	6 LSL 2 DIR	4 LSLA 1 INH	4 LSLX 1 INH	7 LSL 2 IX1	6 LSL 1 IX		2 CLC 1 INH	2 EOR 2 IMM	4 EOR 2 DIR	5 EOR 3 EXT	6 EOR 3 IX2	5 EOR 2 IX1	4 EOR 1 IX	8 1000
9 1001	10 BRCLR4 3 BTB	7 BCLR4 2 BSC	4 BHCS 2 REL	6 ROL 2 DIR	4 ROLA 1 INH	4 ROLX 1 INH	7 ROL 2 IX1	6 ROL 1 IX		2 SEC 1 INH	2 ADC 2 IMM	4 ADC 2 DIR	5 ADC 3 EXT	6 ADC 3 IX2	5 ADC 2 IX1	4 ADC 1 IX	9 1001
A 1010	10 BRSET5 3 BTB	7 BSET5 2 BSC	4 BPL 2 REL	6 DEC 2 DIR	4 DECA 1 INH	4 DECX 1 INH	7 DEC 2 IX1	6 DEC 1 IX		2 CLI 1 INH	2 ORA 2 IMM	4 ORA 2 DIR	5 ORA 3 EXT	6 ORA 3 IX2	5 ORA 2 IX1	4 ORA 1 IX	A 1010
B 1011	10 BRCLR5 3 BTB	7 BCLR5 2 BSC	4 BMI 2 REL							2 SEI 1 INH	2 ADD 2 IMM	4 ADD 2 DIR	5 ADD 3 EXT	6 ADD 3 IX2	5 ADD 2 IX1	4 ADD 1 IX	B 1011
C 1100	10 BRSET6 3 BTB	7 BSET6 2 BSC	4 BMC 2 REL	6 INC 2 DIR	4 INCA 1 INH	4 INCX 1 INH	7 INC 2 IX1	6 INC 1 IX		2 RSP 1 INH		3 JMP 2 DIR	4 JMP 3 EXT	5 JMP 3 IX2	4 JMP 2 IX1	3 JMP 1 IX	C 1100
D 1101	10 BRCLR6 3 BTB	7 BCLR6 2 BSC	4 BMS 2 REL	6 TST 2 DIR	4 TSTA 1 INH	4 TSTX 1 INH	7 TST 2 IX1	6 TST 1 IX		2 NOP 1 INH	8 BSR 2 REL	7 JSR 2 DIP	8 JSR 3 EXT	9 JSR 3 IX2	8 JSR 2 IX1	7 JSR 1 IX	D 1101
E 1110	10 BRSET7 3 BTB	7 BSET7 2 BSC	4 BIL 2 REL								2 LDX 2 IMM	4 LDX 2 DIR	5 LDX 3 EXT	6 LDX 3 IX2	5 LDX 2 IX1	4 LDX 1 IX	E 1110
F 1111	10 BRCLR7 3 BTB	7 BCLR7 2 BSC	4 BIH 2 REL	6 CLR 2 DIR	4 CLRA 1 INH	4 CLRX 1 INH	7 CLR 2 IX1	6 CLR 1 IX		2 TXA 1 INH		5 STX 2 DIR	6 STX 3 EXT	7 STX 3 IX2	6 STX 2 IX1	5 STX 1 IX	F 1111

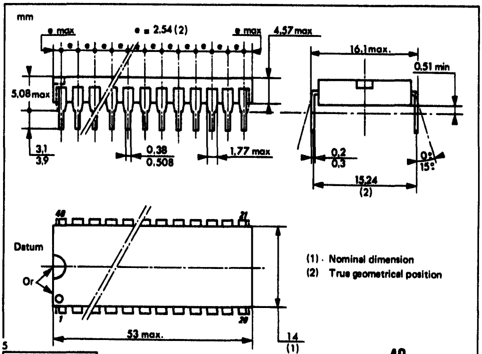
Abbreviations for Address Modes

INH	Inherent
IMM	Immediate
DIR	Direct
EXT	Extended
REL	Relative
BSC	Bit Set/Clear
BTB	Bit Test and Branch
IX	Indexed (No Offset)
IX1	Indexed, 1 Byte (8-Bit) Offset
IX2	Indexed, 2 Byte (16-Bit) Offset

LEGEND



PHYSICAL DIMENSIONS

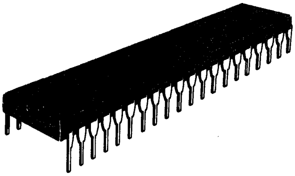


(1) · Nominal dimension
(2) True geometrical position

40 pins

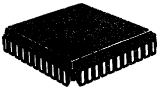
DIN				
A51F CEI	D.A.T.A.	JEDEC	F-119 SITELESC	CB-182

CB-182

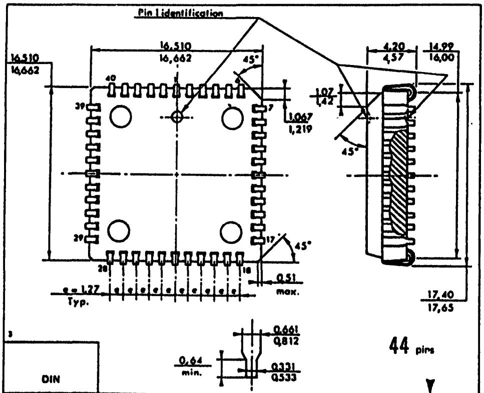


P SUFFIX
PLASTIC PACKAGE

CB-521

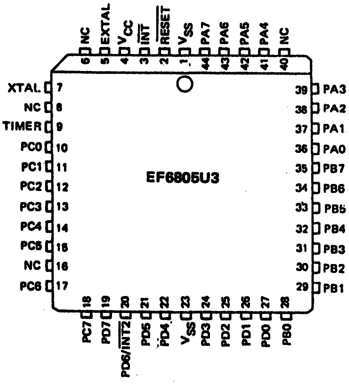


FN SUFFIX
PLCC44



44 pins

DIN				
CEI	D.A.T.A.	MO-047-AC JEDEC	SITELESC	CB-521



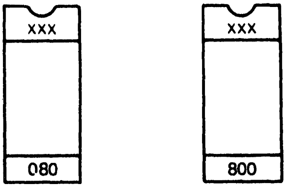
ORDERING INFORMATION

The information required when ordering a custom MCU is listed below. The ROM program may be transmitted to THOMSON SEMICONDUCTEURS on EPROM(s) or an EFDOS/MDOS* disk file.

To initiate a ROM pattern for the MCU, it is necessary to first contact your local THOMSON SEMICONDUCTEURS representative or distributor.

EPROMs

Two 2716 or one 2732 type EPROMs, programmed with the customer program (positive logic sense for address and data), may be submitted for pattern generation. The EPROM must be clearly marked to indicate which EPROM corresponds to which address space. The recommended marking procedure is illustrated below :



XXX = Customer ID

After the EPROM(s) are marked, they should be placed in conductive IC carriers and securely packed. Do not use styrofoam.

VERIFICATION MEDIA

All original pattern media (EPROMs or floppy disk) are filed for contractual purposes and are not returned. A computer listing of the ROM code will be generated and returned along with a listing verification form. The listing should be thoroughly checked and the verification form completed, signed, and returned to THOMSON SEMICONDUCTEURS. The signed verification form constitutes the contractual agreement for creation of the customer mask. If desired, THOMSON SEMICONDUCTEURS will program on blank EPROM from the data file used to create the custom mask and aid in the verification process.

ROM VERIFICATION UNITS (RVUs)

Ten MCUs containing the customer's ROM pattern will be sent for program verification. These units will have been made using the custom mask but are for the purpose of ROM verification only. For expediency they are usually unmarked, packaged in ceramic, and tested only at room temperature and 5 volts. These RVUs are included in the mask charge and are not production parts. The RVUs are thus not guaranteed by THOMSON SEMICONDUCTEURS. Quality Assurance, and should be discarded after verification is completed.

FLEXIBLE DISKS

The disk media submitted must be single-sided, EFDOS/MDOS* compatible floppies.

The customer must write the binary file name and company name on the disk with a felt-tip-pen. The minimum EFDOS/MDOS* system files, as well as the absolute binary object file (Filename .LO type of file) from the 6805 cross assembler, must be on the disk. An object file made from a memory dump using the ROLLOUT command is also acceptable. Consider submitting a source listing as well as the following files : filename .LX (DEVICE/EXORciser loadable format) and filename .SA (ASCII Source Code). These files will of course be kept confidential and are used 1) to speed up the process in-house if any problems arise, and 2) to speed up the user-to-factory interface if the user finds any software errors and needs assistance quickly from THOMSON SEMICONDUCTEURS factory representatives.

EFDOS is THOMSON SEMICONDUCTEURS' Disk Operating System available on development systems such as DEVICE,...

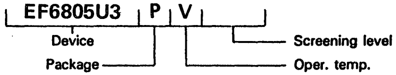
MDOS* is MOTOROLA's Disk Operating System available on development systems such as EXORciser,...

*Requires prior factory approval.

Whenever ordering a custom MCU is required, please contact your local THOMSON SEMICONDUCTEURS representative or THOMSON SEMICONDUCTEURS distributor and/or complete and send the attached "MCU customer ordering sheet" to your local THOMSON SEMICONDUCTEURS representative.

EXORciser is a registered trademark of MOTOROLA Inc.

ORDERING INFORMATION



The table below horizontally shows all available suffix combinations for package, operating temperature and screening level. Other possibilities on request.

DEVICE	PACKAGE					OPER. TEMP			SCREENING LEVEL			
	C	J	P	E	FN	L*	V	T	Std	D	G/B	B/B
EF6805U3			●		●	●	●	●	●	●		

Examples : EF6805U3P, EF6805U3FN, EF6805U3PV, EF6805U3FNV

Package : C : Ceramic DIL, **J :** Cerdip DIL, **P :** Plastic DIL, **E :** LCCC, **FN :** PLCC.
Oper. temp. : L* : 0°C to +70°C, **V :** −40°C to +85°C, **T** — 40°C to + 105°C, *: may be omitted.
Screening level : Std : (no-end suffix), **D :** NFC 96883 level D,
G/B : NFC 96883 level G, **B/B :** NFC 96883 level B and MIL-STD-883C level B.

These specifications are subject to change without notice.
Please inquire with our sales offices about the availability of the different products.

THOMSON COMPONENTS SEMICONDUCTEURS

MK2870

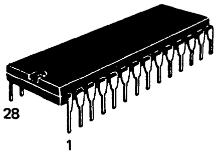
3870 SINGLE CHIP MICRO FAMILY

MK2870 FEATURES

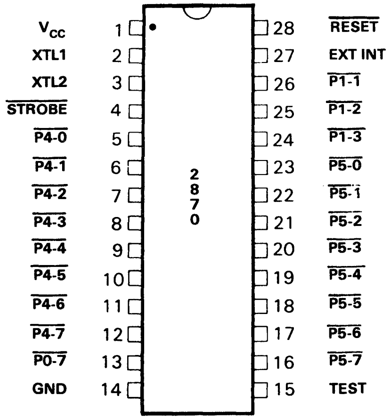
- 28-pin version of the industry standard MK3870 single chip microcomputer
- Available with 1K bytes of mask programmable ROM memory
- 64 bytes scratchpad RAM
- 20 bits TTL compatible I/O
- Programmable binary timer
  Interval timer mode
  Pulse width measurement mode
  Event counter mode
- External interrupt input
- Crystal, LC, RC, or external time base options
- Low power (275 mW typ.)
- Single +5 volt supply

NMOS

CASE



PIN ASSIGNMENT


V_{CC} 1
XTL1 2
XTL2 3
$\overline{\text{STROBE}}$ 4
$\overline{\text{P4-0}}$ 5
$\overline{\text{P4-1}}$ 6
$\overline{\text{P4-2}}$ 7
$\overline{\text{P4-3}}$ 8
$\overline{\text{P4-4}}$ 9
$\overline{\text{P4-5}}$ 10
$\overline{\text{P4-6}}$ 11
$\overline{\text{P4-7}}$ 12
$\overline{\text{P0-7}}$ 13
GND 14
2870
28 $\overline{\text{RESET}}$
27 EXT INT
26 $\overline{\text{P1-1}}$
25 $\overline{\text{P1-2}}$
24 $\overline{\text{P1-3}}$
23 $\overline{\text{P5-0}}$
22 $\overline{\text{P5-1}}$
21 $\overline{\text{P5-2}}$
20 $\overline{\text{P5-3}}$
19 $\overline{\text{P5-4}}$
18 $\overline{\text{P5-5}}$
17 $\overline{\text{P5-6}}$
16 $\overline{\text{P5-7}}$
15 TEST

PIN NAME	DESCRIPTION	TYPE
$\overline{\text{P0-7}}$	I/O Port 0 Bit 7	Bidirectional
$\overline{\text{P1-1}}$ -- $\overline{\text{P1-3}}$	I/O Port 1 Bits 1-3	Bidirectional
$\overline{\text{P4-0}}$ -- $\overline{\text{P4-7}}$	I/O Port 4	Bidirectional
$\overline{\text{P5-0}}$ -- $\overline{\text{P5-7}}$	I/O Port 5	Bidirectional
$\overline{\text{STROBE}}$	Ready Strobe	Output
EXT INT	External Interrupt	Input
$\overline{\text{RESET}}$	External Reset	Input
TEST	Test Line	Input
XTL 1, XTL 2	Time Base	Input
V_{CC}, GND	Power Supply Lines	Input

FEBRUARY 1987 1/18

MK2870 BLOCK DIAGRAM

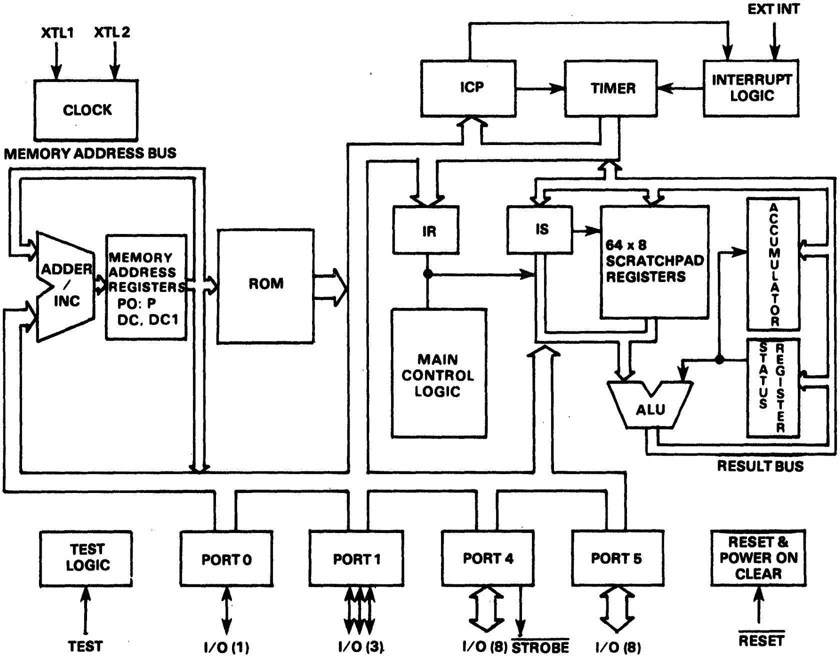


Figure 1

GENERAL DESCRIPTION

The MK2870 is the 28-pin version of the industry standard Mostek MK3870 single chip microcomputer. It is offered as a low cost device which can be used in those applications that do not require the entire I/O capability of the 40 pin MK3870. The compact 28-pin package makes the MK2870 ideally suited for applications where PC board space is a premium.

The MK2870 can execute more than 70 instructions, and is completely software compatible with the rest of the devices in the 3870 family. The MK2870 features 1 K bytes of ROM. The MK2870 also features 64 bytes of scratchpad RAM, a programmable binary timer, and 20 bits of I/O.

The programmable binary timer operates by itself in the interval timer mode or in conjunction with the external interrupt input in the pulse width measurement and the event counter modes of operation. Two sources of vectored, prioritized interrupt are provided with the binary timer and the external interrupt input. The user has the option of specifying one of four clock sources for the MK2870: Crystal, LC, RC, or external clock. In addition, the user can specify either a ±10% power supply tolerance or a ±5% power supply tolerance.

FUNCTIONAL PIN DESCRIPTION

$\overline{\text{P0-7}}$, $\overline{\text{P1-1}}$--$\overline{\text{P1-3}}$, $\overline{\text{P4-0}}$--$\overline{\text{P4-7}}$, and $\overline{\text{P5-0}}$--$\overline{\text{P5-7}}$ are 20 lines which can bɔ individually used as either TTL compatible inputs or as latched outputs.

$\overline{\text{STROBE}}$ is a ready strobe associated with I/O Port 4. This pin, which is normally high, provides a single low pulse after valid data is present on the $\overline{\text{P4-0}}$--$\overline{\text{P4-7}}$ pins during an output instruction.

$\overline{\text{RESET}}$ may be used to externally reset the MK2870. When pulled low the MK2870 will reset. When then allowed to go high the MK2870 will begin program execution at program location H '000'.

EXT INT is the external interrupt input. Its active state is software programmable. This input is also used in conjunction with the timer for pulse width measurement and event counting.

XTL 1 and XTL 2 are the time base inputs to which a crystal, LC network, RC network, or an external single-phase clock may be connected. The time base network must be specified when ordering a mask ROM MK2870.

TEST is an input, used only in testing the MK2870. For normal circuit functionality this pin may be left unconnected, but it is recommended that TEST be grounded.

V_{CC} is the power supply input (single +5 V).

MK2870 ARCHITECTURE

The basic functional elements of the MK2870 are shown in Figure 1 . A programming model is shown in Figure 2. The MK2870 is instruction set compatible. The unique features of the MK2870 are discussed in the following sections. The user is referred to the 3870 Family Technical Manual for a thorough discussion of the architecture, instruction set, and other features.

MK2870 MAIN MEMORY

There are four address registers used to access main memory. These are the Program Counter (PO), the Stack Register (P), the Data Counter (DC), and the Auxiliary Data Counter (DC1). The Program Counter is used to address instructions or immediate operands. The Stack Register is used to save the contents of the Program Counter during an interrupt or subroutine call. Thus, the Stack Register contains the return address at which processing is to resume upon completion of the subroutine or interrupt routine. The Data Counter is used to address data tables. This register is auto-incrementing. Of the two data counters, only Data Counter (DC), can access the ROM. However, the SDC instruction allows the Data Counter and Auxiliary Data Counter to be exchanged.

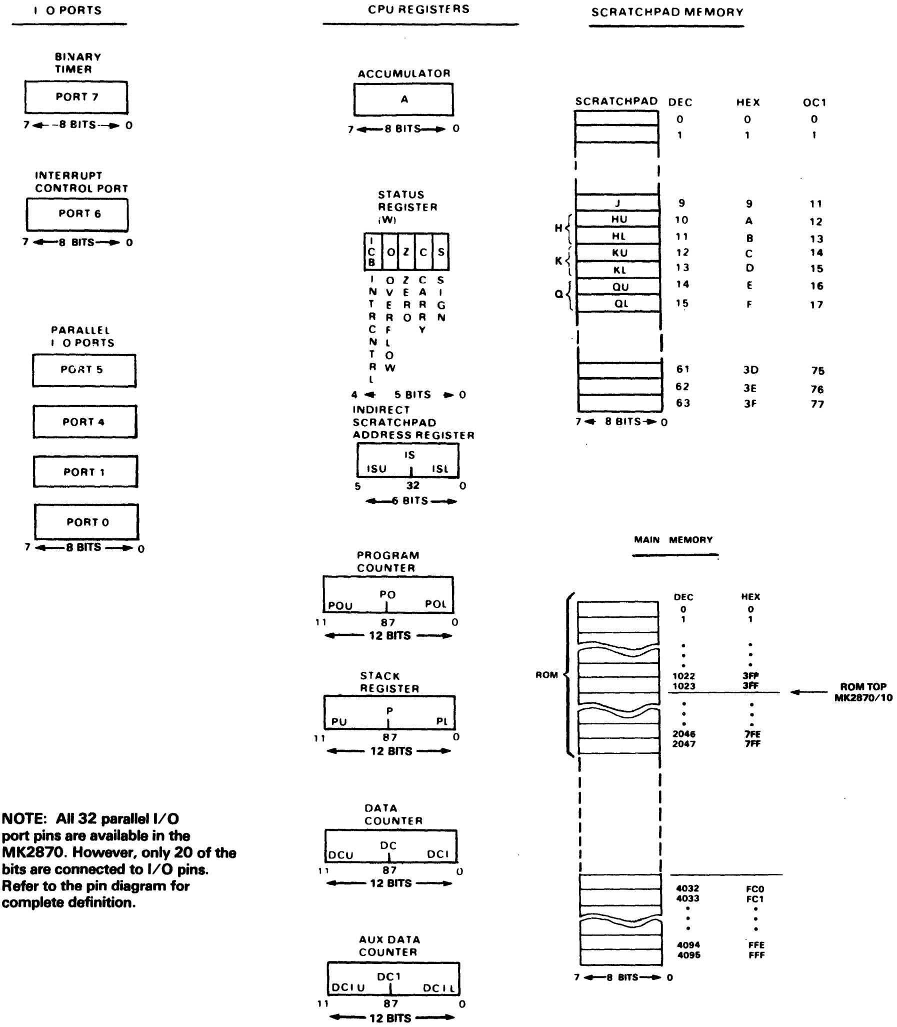
Figure 3 shows the amounts of ROM and executable RAM in the MK2870/10 pin configuration.

I/O PORTS

The MK2870 provides four, 8 bit bidirectional Input/Output ports. These are ports 0, 1, 4, 5. However, only 20 of the bits are connected to I/O pins. The remaining bits are storage elements. In addition, the Interrupt Control Port is addressed as Port 6 and the binary timer is addressed as Port 7. The programming of Ports 6 and 7 and the bidirectional I/O pin are covered in the 3870 Family Technical Manual. The schematic of an I/O pin and available output drive options are shown in Figure 4.

An output ready strobe is associated with Port 4. This flag may be used to signal a peripheral device that the MK2870 has just completed an output of new data to Port 4. The strobe provides a single low pulse shortly after the output operation is completely finished, so either edge may be used to signal the peripheral. $\overline{\text{STROBE}}$ may also be used as an input strobe to Port 4 after completing the input operation.

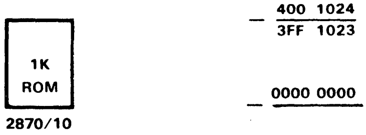
MK2870 PROGRAMMABLE REGISTERS, PORTS, AND MEMORY MAP



NOTE: All 32 parallel I/O port pins are available in the MK2870. However, only 20 of the bits are connected to I/O pins. Refer to the pin diagram for complete definition.

Figure 2

MK2870 MAIN MEMORY SIZE AND TYPE



This device contains 64 bytes of scratchpad RAM.

NOTE:

> Data derived from addressing any locations other than those within a part's specified ROM space or RAM space (if any) is not tested nor is it guaranteed. Users should refrain from entering this area of the memory map.

Device	Scratchpad RAM Size (Decimal)	Address Register Size (P0, P, DC, DC1)	ROM Size (Decimal)	Executable RAM Size (Decimal)
MK2870/10	64 bytes	12 bits	1024 bytes	0 bytes

Figure 3

To use a port pin as an input, the large transistor which pulls the pin to V_{SS} must be turned off. This is accomplished by writing a '0' to that bit of the port. This applies to Ports 0, 1, 4, and 5 only.

MK2870 TIME BASE OPTIONS

The MK2870 contains an on-chip oscillator circuit which provides an internal clock. The frequency of the oscillator circuit is set from the external time base network. The time base for the MK2870 may originate from one of four sources:

1) Crystal
2) LC Network
3) RC Network
4) External Clock

The type of network which is to be used with the mask ROM MK2870 must be specified at the time when mask ROM devices are ordered.

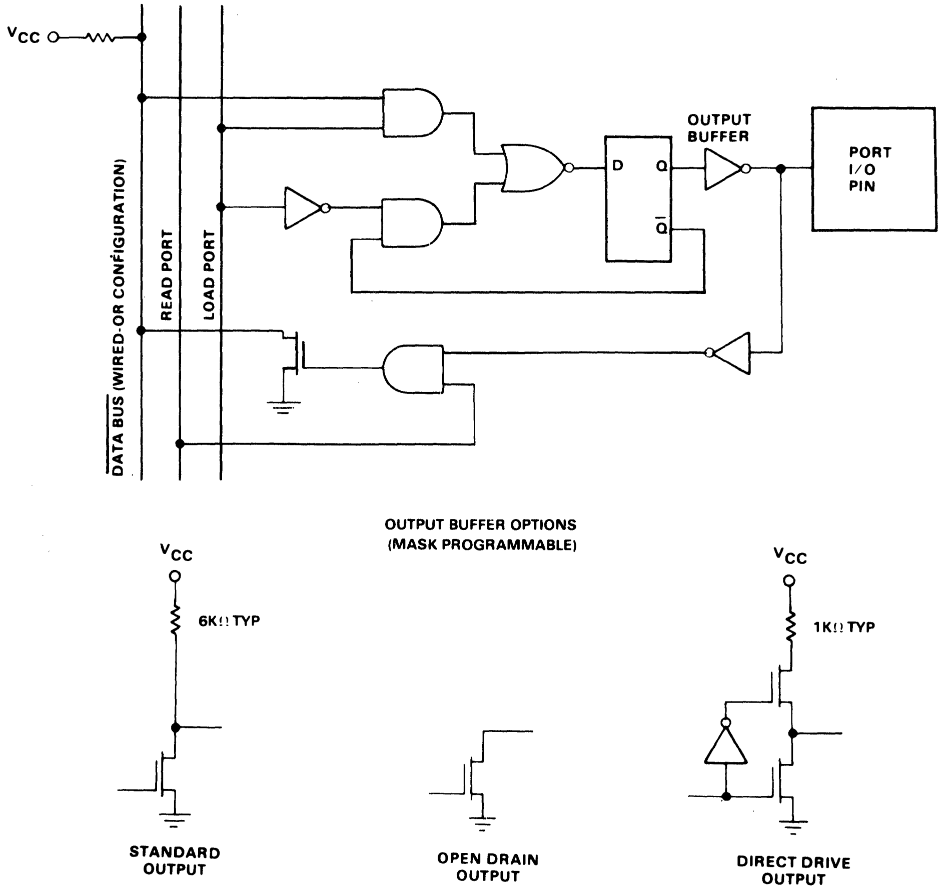
The specifications for the four configurations are given in the following text. There is an internal capacitor between XTL 1 and GND and an internal capacitor between XTL 2 and GND. Thus, external capacitors are not necessarily required. In all clock modes the external time base frequency is divided by two to form the internal PHI clock.

CRYSTAL SELECTION

The use of a crystal as the time base is highly recommended as the frequency stability and reproducability from system to system is unsurpassed. The MK2870 has an internal divide by two to allow the use of inexpensive and widely available TV Color Burst Crystals (3.58 MHz). Figure 6 lists the required crystal parameters for use with the MK2870. The Crystal Mode time base configuration is shown in Figure5.

Through careful buffering of the XTL 1 pin it may be possible to amplify this waveform and distribute it to other devices. However, Mostek recommends that a separate active device (such as a 7400 series TTL gate) be used to oscillate the crystal and the waveform from that oscillator be buffered and supplied to all devices, including the MK2870, in the event that a single crystal is to provide the time base for more than just a single MK2870.

I/O PIN CONCEPTUAL DIAGRAM WITH OUTPUT BUFFER OPTIONS



Ports 0 and 1 are Standard Output type only.

Ports 4 and 5 may both be any of the three output options (mask programmable bit by bit)

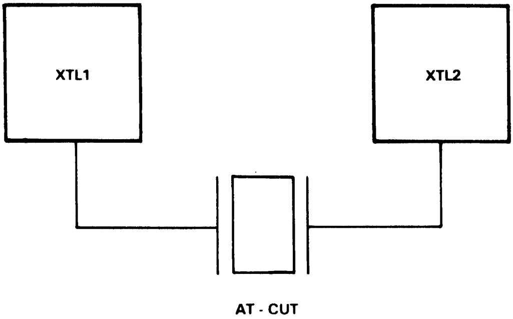
The $\overline{\text{STROBE}}$ output is always configured similar to a Direct Drive Output except that it is capable of driving 3 TTL loads.

$\overline{\text{RESET}}$ and EXT INT may have standard 6K Ω (typical) pull-up or may have no pull-up (mask programmable).

When Direct Drive option is selected, it should be used as an Output only (not as an input).

Figure 4

CRYSTAL MODE CONNECTION



NOTE: Lead lengths from the crystal to the 2870 pins should be kept reasonably short to reduce stray capacitance load.

Figure 5

CRYSTAL PARAMETERS

a) Parallel resonance, fundamental mode AT-Cut
b) Shunt capacitance (C_0) = 7 pf max.
c) Series resistance (R_s) = See table
d) Holder = See table below.

Frequency	Series Resistance	Holder
f = 2-2.7 MHz	Rs = 300 ohms max	HC-6 HC-33
f = 2.8-4 MHz	Rs = 150 ohms max	HC-6 HC-18* HC-25* HC-33

*This holder may not be available at frequencies near the lower end of this range.

Figure 6

While a ceramic resonator may work with the MK2870 crystal oscillator, it was not designed specifically to support the use of this component. Thus, Mostek does not support the use of a ceramic resonator either through proper testing, parametric specification, or applications support.

LC NETWORK

The LC time base configuration can be used to provide a less expensive time base for the MK2870 than can be provided with a crystal. However, the LC configuration is much less accurate than is the crystal configuration. The LC time base configuration is shown in Figure 7. Also shown in the figure are the specified parameters for the LC components, along with the formula for calculating the resulting time base frequency. The minimum value of the inductor which is required for proper operation of the LC time base network is 0.1 millihenries. The inductor must be a Q factor which is no less than 40. The value of C is derived from C external, the internal capacitance of the MK2870 C_{XTL}, and the stray capacitances, C_{S1} and C_{S2}. C_{XTL} is the capacitance looking into the internal two port network at XTL 1 and XTL 2. C_{XTL} is listed under the "Capacitance" section of the Electrical Specifications. C_{S1} and C_{S2} are stray capacitances from XTL 1 to ground and from XTL 2 to ground, respectively. C external should also include the stray shunt capacitance across the inductor. This is typically in the 3 to 5 pf range and significant error can result if it is not included in the frequency calculation.

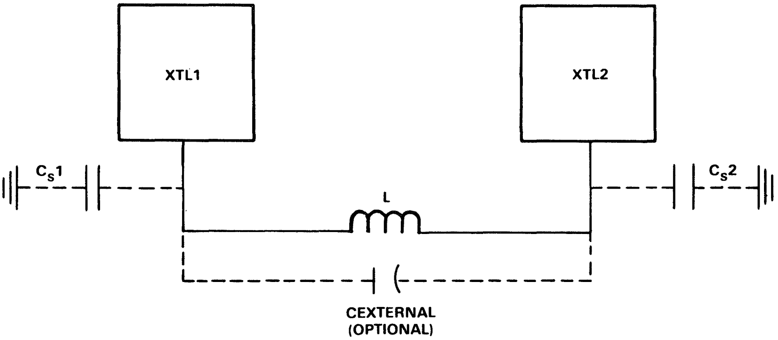
Variation in time base frequency with the LC network can arise from one of four sources: 1) Variation in the value of the inductor. 2) Variation in the value of the external capacitor. 3) Variation in the value of the internal capacitance of the MK2870 at XTL 1 and XTL 2 and 4)

Variation in the amount of stray capacitance which exists in the circuit. Therefore, the actual frequency which is qenerated by the LC circuit is within a range of possible rrequencies, where the range of frequencies is determined by the worst case variation in circuit parameters. The des[illegible]ner must select component values such that the range of possible frequencies with the LC mode does not go outside of the specified operating frequency range for the MK2870.

RC CLOCK CONFIGURATION

The time base for the MK2870 may be provided from an RC network tied to the XTL 2 pin, when XTL 1 is grounded. A schematic picturing the RC clock configuration is shown in Figure 8 . The RC time base configuration is intended to provide an inexpensive time base source for applications in which timing is not critical. Some users have elected to tune each unit using a variable resistor or external capacitor thus reducing the variation in frequency. However, for increased time base accuracy Mostek recommends the use of the Crystal or LC time base configuration. Figure 9 illustrates a curve which gives the resulting operating frequency for a particular RC value. The x-axis represents the product of the value of the resistor times the value of the capacitor. Note that three curves are actually shown. The curve in the middle represents the nominal frequency obtained for a given value of RC. A maximum curve and a minimum curve for different types of MK2870 devices are also shown in the diagram.

LC MODE CONNECTION



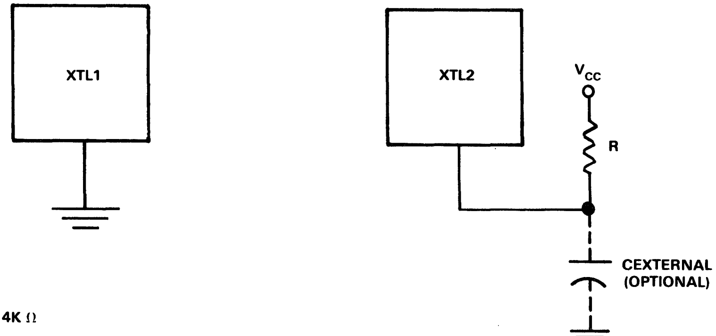
NOTE: The LC options uses the same mask option as the crystal option.

$$f = \frac{1}{2\pi\sqrt{LC}}$$

Figure 7

RC MODE CONNECTION

RC MODE



MINIMUM R = 4K Ω

C = 26.5 pF + 2.6pF + Cexternal

Figure 8

FREQUENCY VS. RC

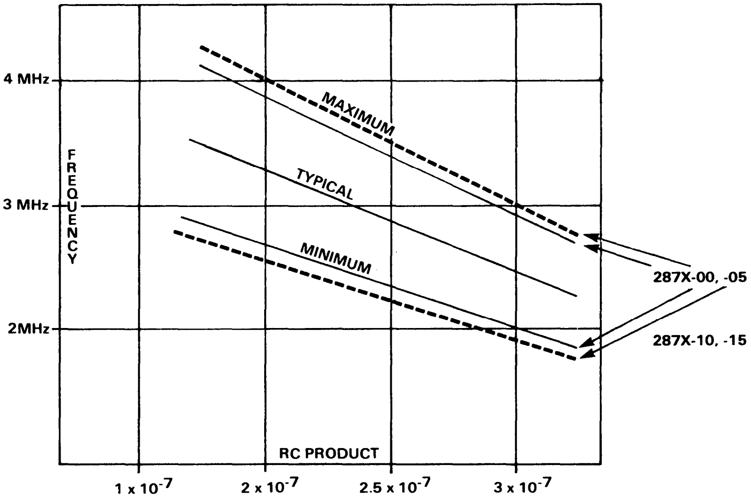


Figure 9

The designer must select the RC product such that a frequency of less than 2 MHz is not possible taking into account the maximum possible RC product and using the minimum curve shown in Figure 9 . Also, the RC product must not allow a frequency of more than 4 MHz taking into account the minimum possible R and C and using the Maximum curve shown below. Temperature induced variations in the external components should be considered in calculating the RC Product.

Frequency variation from unit to unit due to switching speed and level at constant temperature and V_{CC} = + or -5 percent.

Frequency variation due to V_{CC} with all other parameters constant with respect to +5 V = +7 percent to -4 percent on all devices.

Frequency variation due to temperature with respect to 25 C (all other parameters constant) is as follows:

PART #	VARIATION
287X-00, -05	+6 percent to - 9 percent
287X-10, -15	+9 percent to -12 percent

Variations in frequency due to variations in RC components may be calculated as follows:

Maximum RC = (R max) (C external max + C_{XTL} max)

Minimum RC = (R min) (C external min + C_{XTL} min)

Typical RC = (R typ) (C external typ + $\frac{\{C_{XTL} \text{ max} + C_{XTL} \text{ min}\}}{2}$)

Positive Freq. Variation = $\frac{\text{RC typical - RC minimum}}{\text{RC typical}}$

Netative Freq. Variation = $\frac{\text{RC maximum - RC typical}}{\text{RC typical}}$
due to RC Components

Total frequency variation due to all factors:

287X-00, -05	287X-10, -15
+18 percent plus positive frequency variation due to RC components	= +21 percent plus positive frequency variation due to RC components
= -18 percent minus negative frequency variation due to RC components	= -21 percent minus negative frequency variation due to RC components

Total frequency variation due to V_{CC} and temperature of a unit tuned to frequenc at +5 V V_{CC}, 25 C

287X-00, -05	287X-10, -15
= + 13 percent	= +16 percent

EXTERNAL MODE CONNECTION

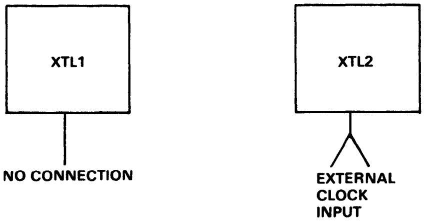


Figure 10

EXTERNAL CLOCK CONFIGURATION

The connection for the external clock time base configuration is shown in Figure10 . Refer to the DC Characteristics section for proper input levels and current requirements.

Refer to the Capacitance section for input capacitance.

ELECTRICAL SPECIFICATIONS
MK2870

OPERATING VOLTAGES AND TEMPERATURES

Dash Number Suffix	Operating Voltage V_{CC}	Operating Temperature T_A
— 00	+5 V ±10%	0°C - 70°C
— 05	+5 V ± 5%	0°C - 70°C
— 10	+5 V ± 10%	-40°C - +85°C
— 15	+5 V ± 5%	-40°C - +85°C

See Ordering Information for explanation of part numbers.

MAXIMUM RATINGS*

	-00, -05	-10, -15
Temperature Under Bias	-20°C to +85°C	-50°C to +100°C
Storage Temperature	-65°C to +150°C	-65°C to +150°C
Voltage on any Pin With Respect to Ground (Except open drain pins and TEST)	-1.0 V to +7 V	-1.0 V to + 7 C
Voltage on TEST with Respect to Ground	-1.0 V to +9 V	-1.0 V to +9 V
Voltage on Open Drain Pins With Respect to Ground	-1.0 V to +13.5 V	-1.0 V to +13.5 V
Power Dissipation	1.5 W	1.5 W
Power Dissipation by any one I/O pin	60 mW	60 mW
Power Dissipation by all I/O pins	600 mW	600 mW

*Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other condition above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

AC CHARACTERISTICS

T_A, V_{CC} within specified operating range.
I/O power dissipation ≤ 100 mW (Note 2)

SIGNAL	SYM	PARAMETER	-00, -05 MIN	-00, -05 MAX	-10, -15 MIN	-10, -15 MAX	UNIT	NOTES
XTL 1 XTL 2	t_o	Time Base Period, all clock modes	250	500	250	500	ns	4 MHz - 2 MHz
	$t_{ex(H)}$	External clock pulse width high	90	400	100	390	ns	
	$t_{ex(L)}$	External clock pulse width low	100	400	110	390	ns	
Φ	t_Φ	Internal Φ clock	$2t_O$		$2t_O$			
WRITE	t_w	Internal WRITE Clock period	4tΦ 6tΦ		4tΦ 6tΦ			Short Cycle Long Cycle
I/O	$t_{dI/O}$	Output delay from internal WRITE clock	0	1000	0	1200	ns	50 pF plus one TTL load
	$t_{sI/O}$	Input setup time to internal WRITE clock	1000		1200		ns	
$\overline{\text{STROBE}}$	$t_{I/O\text{-}s}$	Output valid to $\overline{\text{STROBE}}$ delay	3tΦ -1000	3tΦ +250	3tΦ -1200	3tΦ +300	ns	I/O load = 50 pF + 1 TTL load
	t_{sL}	$\overline{\text{STROBE}}$ low time	8tΦ -250	12tΦ +250	8tΦ -300	12tΦ +300	ns	$\overline{\text{STROBE}}$ load = 50 pF + 3TTL loads
$\overline{\text{RESET}}$	t_{RH}	$\overline{\text{RESET}}$ hold time, low	6tΦ +750		6tΦ +1000		ns	
	t_{RPOC}	$\overline{\text{RESET}}$ hold time, low for power clear	power supply rise time + 0.1		power supply rise time + .15		ms	
EXT INT	t_{EH}	EXT INT hold time in active and inactive state	6tΦ +750		6tΦ +1000		ns	To trigger interrupt
			2tΦ		2tΦ		ns	To trigger timer

DC CHARACTERISTICS

T_A, V_{CC} within specified operating range
I/O power dissipation ≤ 100 mW (Note 2)

		-00, -05		-10, -15			
SYMBOL	**PARAMETER**	**MIN**	**MAX**	**MIN**	**MAX**	**UNIT**	**DEVICE**
I_{CC}	Average Power Supply Current		85		110	mA	MK2870/10 Outputs Open
P_D	Power Dissipation		400		525	mW	MK2870/10 Outputs Open
V_{IHEX}	External Clock input high level	2.4	5.8	2.4	5.8	V	
V_{ILEX}	External Clock input low level	–.3	.6	–.3	.6	V	
I_{IHEX}	External Clock input high current		100		130	µA	$V_{IHEX}=V_{CC}$
I_{ILEX}	External Clock input low current		-100		-130	µA	$V_{ILEX}=V_{SS}$
$V_{IHI/O}$	Input high level, I/O pins	2.0	5.8	2.0	5.8	V	Standard pull-up
		2.0	13.2	2.0	13.2	V	Open drain (1)
V_{IHR}	Input high level, $\overline{\text{RESET}}$	2.0	5.8	2.2	5.8	V	Standard pull-up
		2.0	13.2	2.2	13.2	V	No Pull-up
V_{IHEI}	Input high level, EXT INT	2.0	5.8	2.2	5.8	V	Standard pull-up
		2.0	13.2	2.2	13.2	V	No Pull-up
V_{IL}	Input low level	–.3	.8	–.3	.7	V	(1)
I_{IL}	Input low current, all pins with standard pull-up resistor		-1.6		-1.9	mA	$V_{IN} = 0.4$ V
I_L	Input leakage current, open drain pins, and inputs with no pull-up resistor		+10 -5		+18 -8	µA µA	$V_{OH} = 13.2$ V $V_{IN} = 0.0$ V
I_{OH}	Output high current pins with standard pull-up resistor	-100 -30		-89 -25		µA µA	$V_{OH} = 2.4$ V $V_{OH} = 3.9$ V

DC CHARACTERISTICS (cont.)

T_A, V_{CC} within specified operating range, I/O power dissipation ≥ 100 mW (Note 2)

SYM	PARAMETER	-00, -05 MIN	-00, -05 MAX	-10, -15 MIN	-10, -15 MAX	UNIT	CONDITIONS
I_{OHDD}	Output high current, direct drive pins	-100 -1.5	 -8.5	-80 -1.3	 -11	µA mA mA	V_{OH} = 2.4 V V_{OH} = 1.5 V V_{OH} = 0.7 V
I_{OHS}	$\overline{\text{STROBE}}$ Output High current	-300		-270		µA	V_{OH} = 2.4 V
I_{OL}	Output low current	1.8		1.65		mA	V_{OL} = 0.4 V
I_{OLS}	$\overline{\text{STROBE}}$ Output Low current	5.0		4.5		mA	V_{OL} = 0.4 V

TIMER AC CHARACTERISTICS

Definitions:
Error = Indicated time value - actual time value

tpsc = tΦ x Prescale Value

Interval Timer Mode:

Single interval error, free running (Note 3) .... ±6tΦ
Cumulative interval error, free running (Note 3) 0
Error between two Timer reads (Note 2) .... ±(tpsc + tΦ)
Start Timer to stop Timer error (Notes 1, 4).................... +tΦ to - (tpsc + tΦ)
Start Timer to read Timer error (Notes 1, 2)....................-5t Φ to - (tpsc + 7tΦ)
Start Timer to interrupt request error (Notes 1, 3) ....-2tΦ to -8 tΦ
Load Timer to stop Timer error (Note 1) ....+ tΦ to - (tpsc + 2tΦ)
Load Timer to read Timer error (Notes 1, 2).................... -5tΦ ± to - (tpsc + 8tΦ)
Load Timer to interrupt request error (Notes 1, 3) .... -2tΦ to -9tΦ

Pulse Width Measurement Mode:

Measurement accuracy (Note 4) ....+ tΦ to - (tpsc + 2tΦ)
Minimum pulse width of EXT INT pin.................... 2tΦ

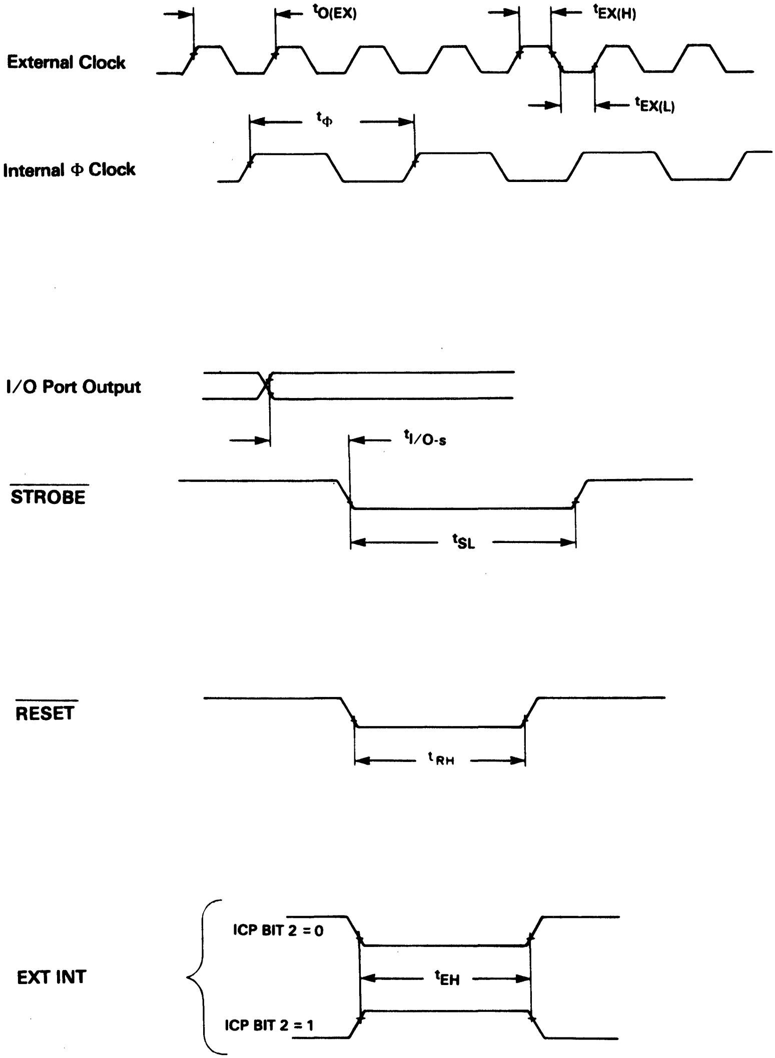
Event Counter Mode:

Minimum active time of EXT INT pin 2tΦ
Minimum inactive time of EXT INT pin.................... 2tΦ

NOTES:

1. All times which entail loading, starting, or stopping the Timer are referenced from the end of the last machine cycle of the OUT or OUTS instruction.
2. All times which entail reading the Timer are referenced from the end of the last machine cycle of the IN or INS instruction.
3. All times which entail the generation of an interrupt request are referenced from the start of the machine cycle in which the appropriate interrupt request latch is set. Additional time may elapse if the interrupt request occurs during a privileged or multicycle instruction.
4. Error may be cumulative if operation is repetitively performed.

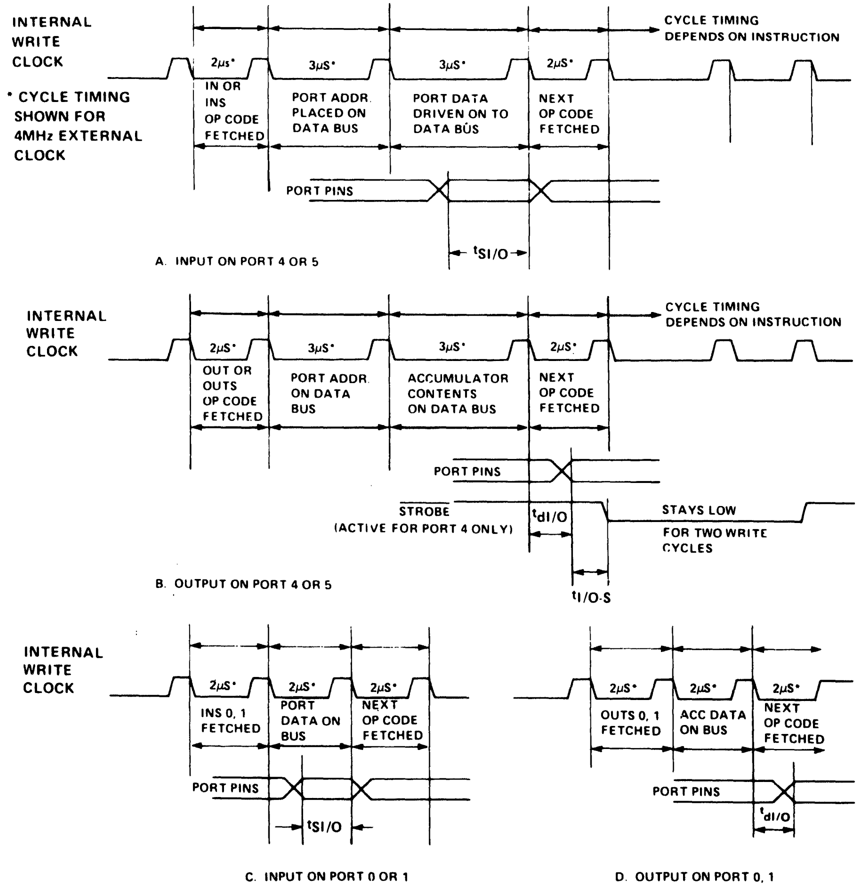
AC TIMING DIAGRAM



Note: All AC measurements are referenced to V_{IL} max., V_{IH} min., V_{OL} (.8v), or V_{OH} (2.0v).

Figure 11

INPUT/OUTPUT AC TIMING



A. INPUT ON PORT 4 OR 5

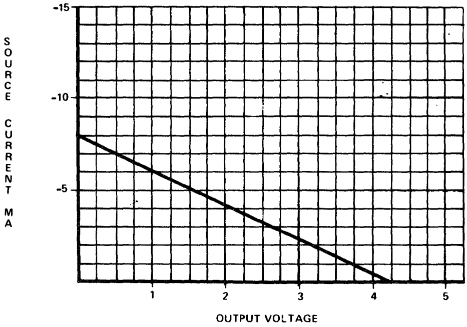
B. OUTPUT ON PORT 4 OR 5

C. INPUT ON PORT 0 OR 1

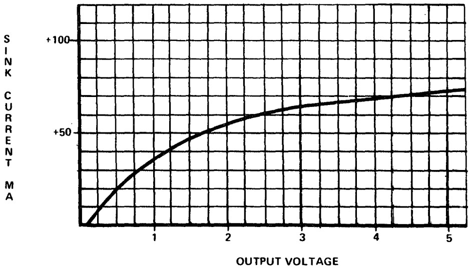
D. OUTPUT ON PORT 0, 1

Figure 12

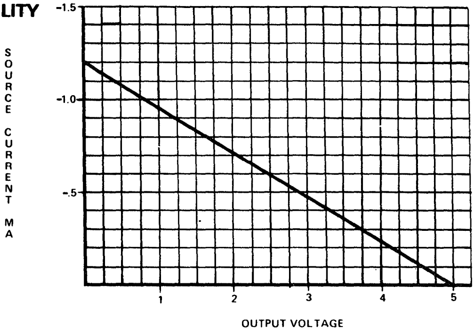
STROBE SOURCE CAPABILITY
(TYPICAL AT V_{CC} = 5 V, T_A = 25°C)
Figure 13



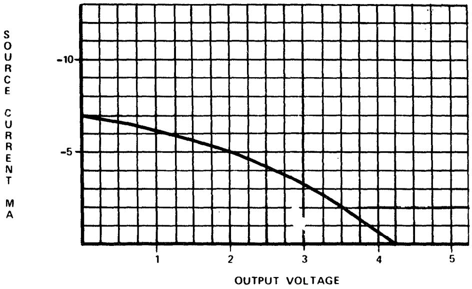
STROBE SINK CAPABILITY
(TYPICAL AT V_{CC} = 5 V, T_A = 25°C)
Figure 14



STANDARD I/O PORT SOURCE CAPABILITY
(TYPICAL AT V_{CC} = 5 V, T_A = 25°C)
Figure 15



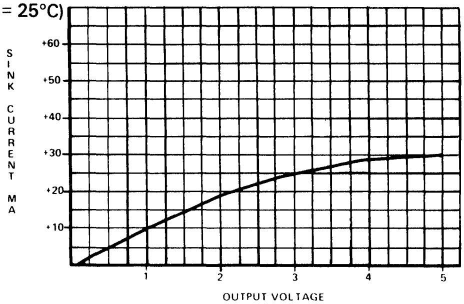
DIRECT DRIVE I/O PORT SOURCE CAPABILITY
(TYPICAL AT V_{CC} = 5 V, T_A = 25°C)
Figure 16



I/O PORT SINK CAPABILITY

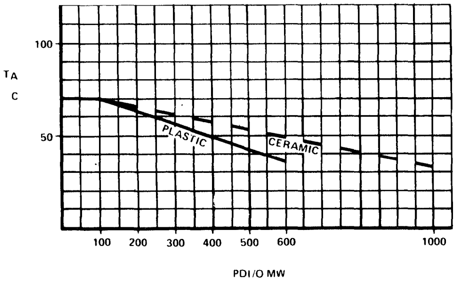
(TYPICAL AT V_{CC} = 5 V, $T_{A'}$ = 25°C)

Figure 17



MAXIMUM OPERATING TEMPERATURE VS. I/O POWER DISSIPATION

Figure 18



ORDERING INFORMATION

There are two types of part numbers for the 2870 family of devices. The generic part number describes the basic device type, the amount of ROM and Executable RAM, the desired package type, temperature range, and power supply tolerance. For each customer specific code, additional information defining I/O options and oscillator options will be combined with the information described in the generic part number to define a customer/code specific device order number.

GENERIC PART NUMBER

An example of the generic part number is shown below.

MK 2870 / 1 0 N - 1 0

Power Supply Tolerance	0 = 5 V ± 10% 5 = 5 V ± 5%
Operating Temperature Range	0 = 0°C - +70°C 1 = -40°C - +85°C
Package type	N = Plastic
Executable RAM Designator	0 = None 2 = 64 bytes
ROM Designator	1 = 1K bytes 2 = 2K bytes
Basic Device Type	

DEVICE ORDER NUMBER

An example of the device order number is shown below.

MK 80007 N - 0 5

Power Supply Tolerance	0 = 5 V ± 10% 5 = 5 V ± 5%
Operating Temperature Range	0 = 0°C - +70°C 1 = -40°C - +85°C
Package Type	N = Plastic
Customer/Code Specific Number	

The Customer/Code specific number defines the ROM bit pattern, I/O configuration, oscillator type, and generic part type to be used to satisfy the requirements of a particular customer purchase order. For further information on the ordering of mask ROM devices, the customer should refer to the 3870 Family Technical Manual.

These specifications are subject to change without notice.
Please inquire with our sales offices about the availability of the different products.

HOMSON
COMPONENTS
EMICONDUCTEURS

MK3870 • MK38P70

3870 SINGLE CHIP MICRO FAMILY

MK3870 FEATURES

- Available with 1K, 2K, or 4K bytes of mask programmable ROM memory
- 64 bytes scratchpad RAM
- Available with 64 bytes executable RAM
- 32 bits (4 ports) TTL compatible I/O
- Programmable binary timer
  Interval timer mode
  Pulse width measurement mode
  Event counter mode
- External interrupt input
- Crystal, LC, RC, or external time base options
- Low power (275 mW typ.)
- Single +5 volt supply

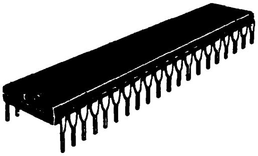
MK38P70 FEATURES

- EPROM version of MK3870
- Piggyback PROM (P-PROM)™ package
- Accepts 24 pin or 28 pin EPROM memories
- Identical pinout as MK3870
- In-Socket emulation of MK3870

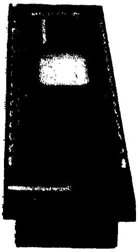
NMOS

CASES

MK3870



MK38P70



FEBRUARY 1987 1/28

MK3870 BLOCK DIAGRAM

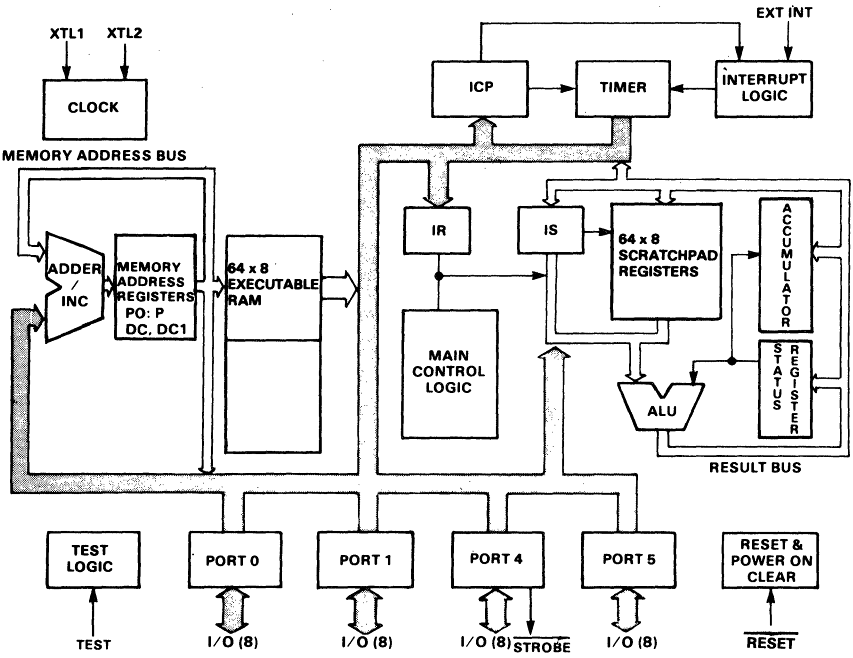
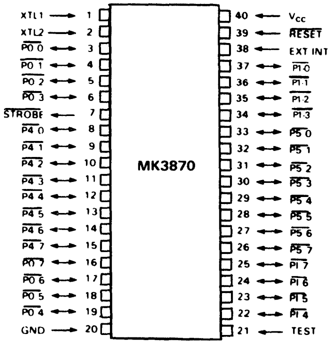
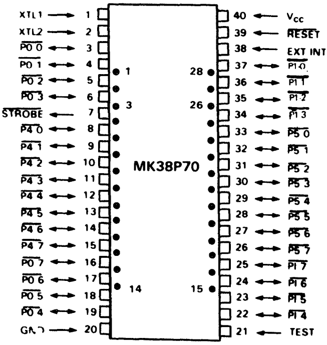


Figure 1

PIN ASSIGNMENTS



PIN NAME	DESCRIPTION	TYPE
P0-0 -- P0-7	I/O Port 0	Bidirectional
P1-0 -- P1-7	I/O Port 1	Bidirectional
P4-0 -- P4-7	I/O Port 4	Bidirectional
P5-0 -- P5-7	I/O Port 5	Bidirectional
STROBE	Ready Strobe	Output
EXT INT	External Interrupt	Input
RESET	External Reset	Input
TEST	Test Line	Input
XTL 1, XTL 2	Time Base	Input
V_{CC}, GND	Power Supply Lines	Input



GENERAL DESCRIPTION

The MK3870 is a complete 8-bit microcomputer on a single MOS integrated circuit. The MK3870 can execute a set of more than 70 instructions, and is completely software compatible with the rest of the devices in the 3870 family. The MK3870 features 1-4K bytes of ROM and optional additional executable RAM depending on the specific part type designated by a slash number suffix. The MK3870 also features 64 bytes of scratchpad RAM, a programmable binary timer, and 32 bits of I/O.

The programmable binary timer operates by itself in the interval timer mode or in conjunction with the external interrupt input in the pulse width measurement and the event counter modes of operation. Two sources of vectored, prioritized interrupt are provided with the binary timer and the external interrupt input. The user has the option of specifying one of four clock sources for the MK3870 and MK38P70: Crystal, LC, RC, or external clock. In addition, the user can specify either a ±10% power supply tolerance or a ±5% power supply tolerance.

The MK38P70 microcomputer is the PROM based version of the MK3870. It is called the piggyback PROM (P-PROM)™ because of its packaging concept. This concept allows a standard 24-pin or 28 pin EPROM to be mounted directly on top of the microcomputer itself. The EPROM can be removed and reprogrammed as required with a standard PROM programmer. The MK38P70 retains exactly the same pinout and architectural features as other members of the 3870 family. The MK38P70 is discussed in more detail in a later section.

FUNCTIONAL PIN DESCRIPTION

$\overline{\text{P0-0}}$-- $\overline{\text{P0-7}}$, $\overline{\text{P1-0}}$--$\overline{\text{P1-7}}$, $\overline{\text{P4-0}}$--$\overline{\text{P4-7}}$, and $\overline{\text{P5-0}}$--$\overline{\text{P5-7}}$ are 32 lines which can be individually used as either TTL compatible inputs or as latched outputs.

$\overline{\text{STROBE}}$ is a ready strobe associated with I/O Port 4. This pin, which is normally high, provides a single low pulse after valid data is present on the $\overline{\text{P4-0}}$--$\overline{\text{P4-7}}$ pins during an output instruction.

$\overline{\text{RESET}}$ may be used to externally reset the MK3870. When pulled low the MK3870 will reset. When then allowed to go high the MK3870 will begin program execution at program location H '000'.

EXT INT is the external interrupt input. Its active state is software programmable. This input is also used in conjunction with the timer for pulse width measurement and event counting.

XTL 1 and XTL 2 are the time base inputs to which a crystal, LC network, RC network, or an external single-phase clock may be connected. The time base network must be specified when ordering a mask ROM MK3870. The MK38P70 will operate with any of the four configurations.

TEST is an input, used only in testing the MK3870. For normal circuit function this pin may be left unconnected, but it is recommended that TEST be grounded. On MK38P70 devices, the TEST must be grounded.

V_{CC} is the power supply input (single +5v).

MK3870 ARCHITECTURE

The basic functional elements of the MK3870 are shown in Figure 1. A programming model is shown in Figure 2. The architecture is common to all members of the 3870 family. All 3870 devices are instruction set compatible and differ only in amount and type of ROM, RAM, and I/O. The unique features of the MK3870 are discussed in the following sections. The user is referred to the 3870 Family Technical Manual for a thorough discussion of the architecture, instruction set, and other features which are common to all 3870 family devices.

MK3870 MAIN MEMORY

There are four address registers used to access main memory. These are the Program Counter (PO), the Stack Register (P), the Data Counter (DC), and the Auxiliary Data Counter (DC1). The Program Counter is used to address instructions or immediate operands. The Stack Register is used to save the contents of the Program Counter during an interrupt or subroutine call. Thus, the Stack Register contains the return address at which processing is to resume upon completion of the subroutine or interrupt routine. The Data Counter is used to address data tables. This register is auto-incrementing. Of the two data counters, only Data Counter (DC), can access the ROM. However, the XDC instruction allows the Data Counter and Auxiliary Data Counter to be exchanged.

The graph in Figure 3 shows the amounts of ROM and executable RAM for every available slash number in the MK3870 pin configuration.

EXECUTABLE RAM

The upper bytes of the address space in some of the MK3870 devices are RAM memory. As with the ROM memory, the RAM may be addressed by the PO and the DC address registers. The executable RAM may be addressed by all MK3870 instructions which address Main Memory.

MK3870 PROGRAMMABLE REGISTERS, PORTS, AND MEMORY MAP

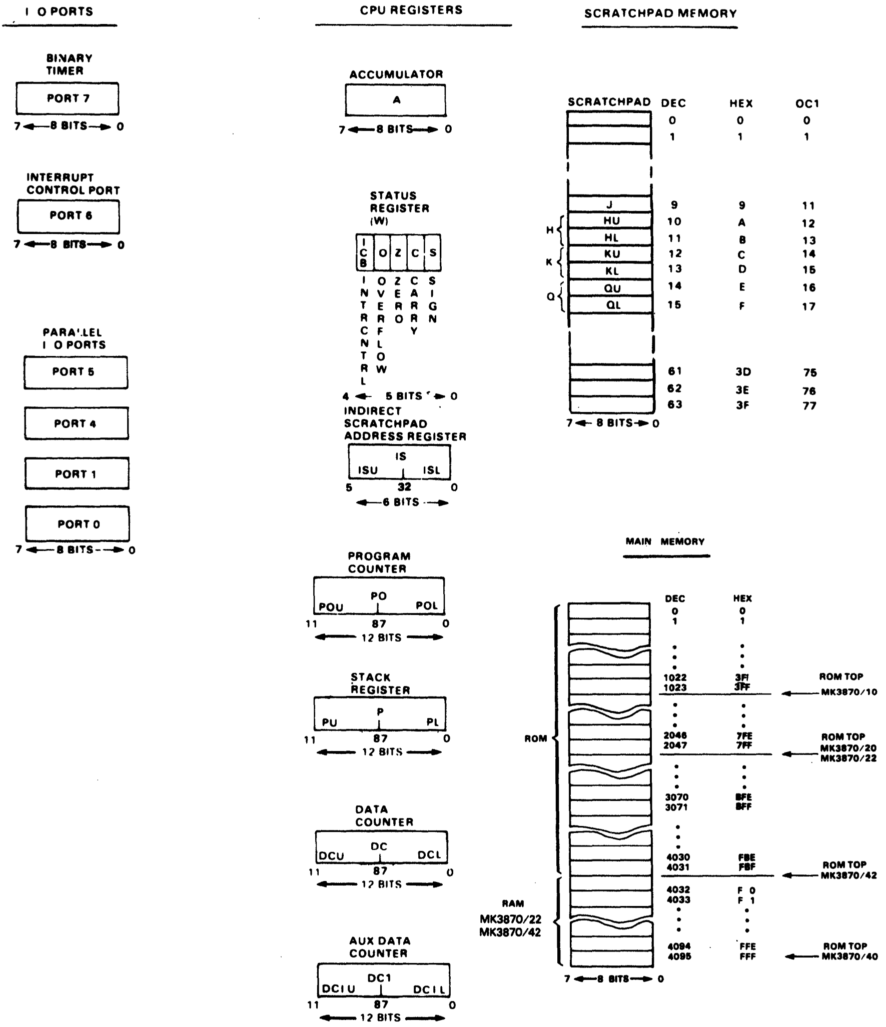
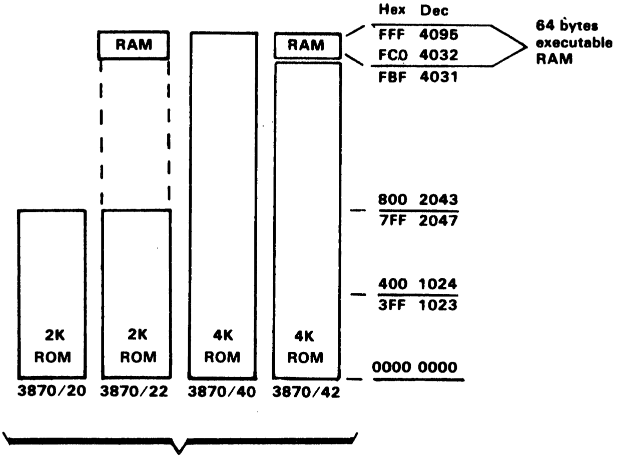


Figure 2

MK3870 MAIN MEMORY SIZES AND TYPES BY SLASH NUMBERS



All devices contain 64 bytes of scratchpad RAM.

NOTE: Data derived from addressing any locations other than those within a part's specified ROM space or RAM space (if any) are not tested nor are the data guaranteed. Users should refrain from entering this area of the memory map.

Device	Scratchpad RAM Size (Decimal)	Address Register Size (P0, P, DC, DC1)	ROM Size (Decimal)	Executable RAM Size
MK3870/20*	64 bytes	12 bits	2048 bytes	0 bytes
MK3870/22	64 bytes	12 bits	2048 bytes	64 bytes
MK3870/40	64 bytes	12 bits	4096 bytes	0 bytes
MK3870/42	64 bytes	12 bits	4032 bytes	64 bytes

*The MK3870/20 is equivalent to the original 3870 device in memory size; however, the original 3870 had an 11-bit Address Register. The original 3870 with 11-bit Address Register is available where required. Consult the section describing ROM Code Ordering Information for additional information.

Figure 3

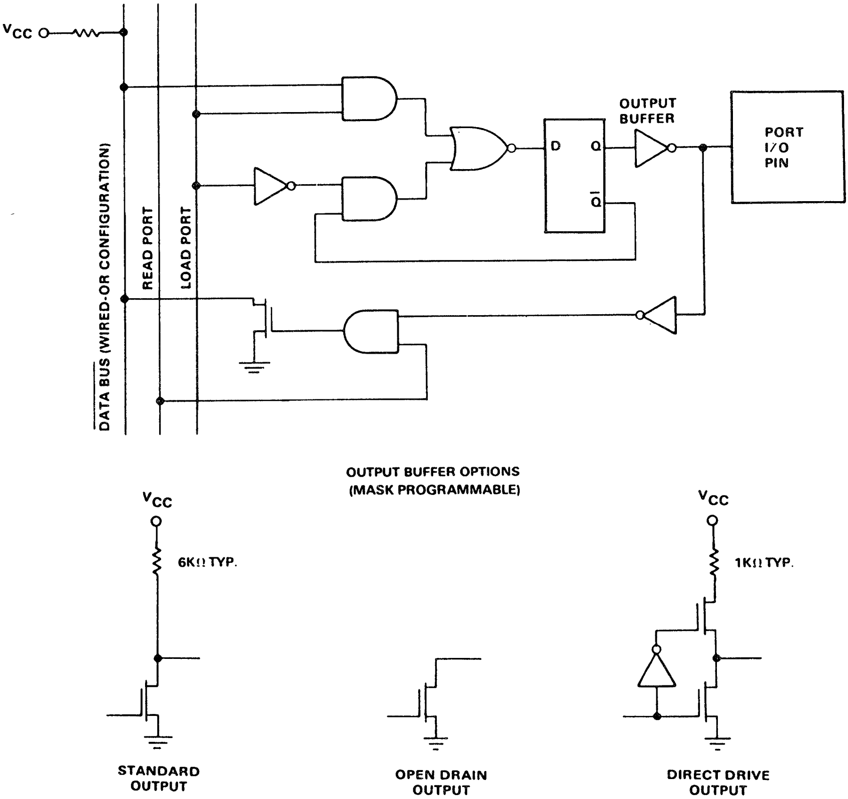
Additionally, the MK3870 may execute an instruction sequence which resides in the executable RAM. Note this cannot be done with the scratchpad RAM memory, which is the reason the term "executable RAM" is given to this additional memory.

I/O PORTS

The MK3870 provides four, 8 bit bidirectional Input/Output ports. These are ports 0, 1, 4, 5. In addition, the Interrupt Control Port is addressed as Port 6 and the binary timer is addressed as Port 7. The programming of Ports 6 and 7 and the bidirectional I/O pin are covered in the 3870 Famil Technical Manual. The schematic of an I/O pin anc available output drive options are shown in Figure 4.

An output ready strobe is associated with Port 4. This fla may be used to signal a peripheral device that the MK387 has just completed an output of new data to Port 4. Th strobe provides a single low pulse shortly after the outpu operation is completely finished, so either edge may be use to signal the peripheral. $\overline{\text{STROBE}}$ may also be used as a input strobe to Port 4 after completing the input operatio

I/O PIN CONCEPTUAL DIAGRAM WITH OUTPUT BUFFER OPTIONS



Ports 0 and 1 are Standard Output type only.

Ports 4 and 5 may both be any of the three output options (mask programmable bit by bit)

The $\overline{\text{STROBE}}$ output is always configured similar to a Direct Drive Output except that it is capat e of driving 3 TTL loads.

$\overline{\text{RESET}}$ and EXT INT may have standard 6KΩ (typical) pull-up or may have no pull-up (mask programmable).

$\overline{\text{RESET}}$ and EXT INT do not have internal pull up on the MK38P70.

Figure 4

MK38P70 GENERAL DESCRIPTION

The MK38P70 is the EPROM version of the MK3870. It retains an identical pinout with the MK3870, which is documented in the section of this data sheet entitled "FUNCTIONAL PIN DESCRIPTION". The MK38P70 is housed in two packages which incorporate a 28 or 24-pin socket located directly on top of the package. A number of standard EPROMs may be plugged into this socket.

The MK38P70 can act as an emulator for the purpose of verification of user code prior to the ordering of mask ROM MK3870 devices. Thus, the MK38P70 eliminates the need for emulator board products. In addition, several MK38P70s can be used in prototype systems in order to test design concepts in field service before committing to high-volume production with mask ROM MK3870s. The compact size of the MK38P70/EPROM combination allows the packaging of such prototype systems to be the same as that used in production. Finally, in low-volume applications, the MK38P70 can be used as the actual production device.

Most of the material which has been presented for the MK3870 in this document applies to the MK38P70. This includes the description of the pin configuration, architecture, programming model, and I/O ports. Additional information is presented in the following sections.

MK38P70 BLOCK DIAGRAM

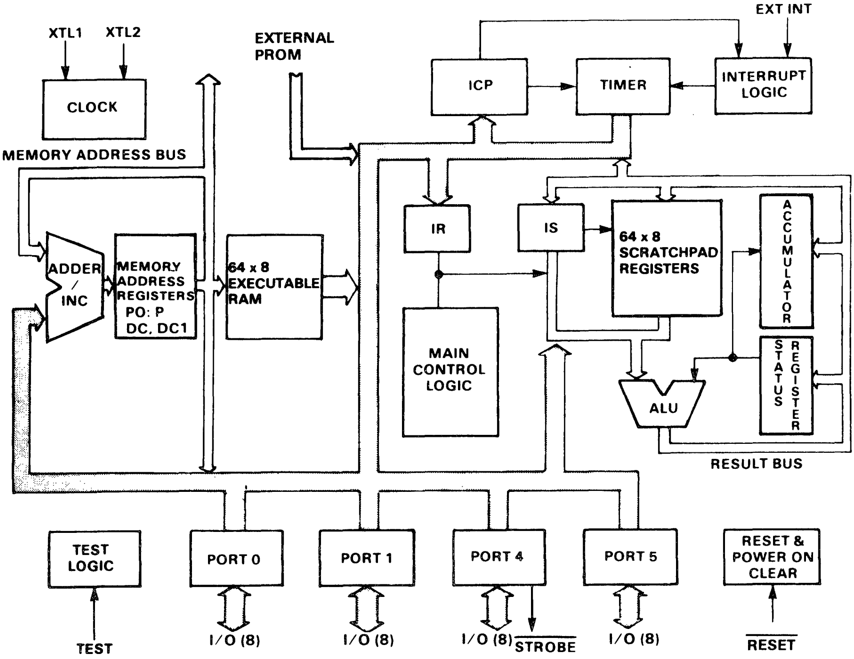


Figure 5

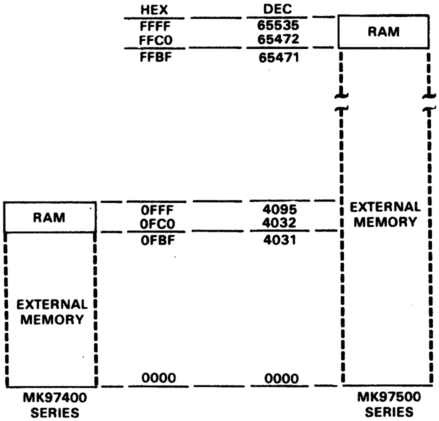
MK38P70 MAIN MEMORY

There are two basic versions of the MK38P70. These are the 97400 series and the 97500 series. The 97400 series parts have twelve bit address capability thus a total 4K memory map like the MK3870 ROM devices. The 97500 series has 16 bit address capability. The 97400 series accepts 24 pin EPROMs, and the 97500 series accepts 28 pin EPROMs.

As can be seen from Figure 6, both the 97400 series and the 97500 series contain on-chip RAM in the upper portion of their memory maps and no on-chip ROM. Instead of on-chip ROM, address and dat lines are brought out to the 28 pin socket located directly on top of the 40 pin package so that external memory devices (principally EPROMs) are addressed.

By using an external EPROM, the 38P70 may be used to

MK38P70 MAIN MEMORY MAP



MK38P70 TYPE	SCRATCHPAD RAM SIZE (DECIMAL)	ADDRESS REGISTER SIZE	EXTERNALLY ADDRESSABLE MEMORY SIZE	INTERNAL EXECUTABLE RAM SIZE
97400 Series	64 bytes	12 bits	4032 bytes	64 bytes
97500 Series	64 bytes	16 bits	65472 bytes	64 bytes

MK38P70 devices have no internal ROM memory.

Figure 6

emulate the 3870 ROM devices. The 97400 series can directly emulate the following devices.

MK3870/10
MK3870/20
MK3870/22
MK3870/42

The MK3870/40 cannot be emulated exactly by the 97400 series because the 97400 devices have the 64 bytes of RAM in the upper memory map while the 3870/40 provides ROM memory in this address space.

Besides the difference in the size of the address registers, 97500 series can also emulate many of the 3870 ROM devices. This difference in address capability should not cause any functional difference as long as normal programming practice is used. That is, as long as address roll-over or automatic truncation is not used. One such usage would be an end around branch (branching forward at upper memory to get to lower memory). Another case would be in using automatic truncation of data loaded into the 12 bit address registers on the ROM devices. For example, to access some particular location (03FF hex for example) via the data counter, one could load that address into DC using the DCI instruction. The instruction

DCI '73FF'

would cause 3FF to be loaded into the DC of the 3870 ROM device because the upper bits of the DC (bits 12-15) do no exist. If that instruction was followed by the LM instruction the data stored at location 3FF would be obtained. Th 97500 series devices would not truncate the 73FF address to 3FF. As previously stated, this type of programming i generally not done and thus the 97500 devices can be used to emulate the following devices directly.

MK3870/10
MK3870/20
MK3870/40

The 97500 series can also be used to emulate th remainder of the 3870 devices as long as one accounts fo the difference in the location of the RAM memory. In th 97500 devices, RAM is located at FFC0 through FFFF While in 3870 devices this RAM (when it exists) is located a 0FC0 through 0FFF. When this minor difference i accounted for, the 97500 series will also emulate th following devices.

MK3870/22
MK3870/42

MK38P70 EPROM SOCKET

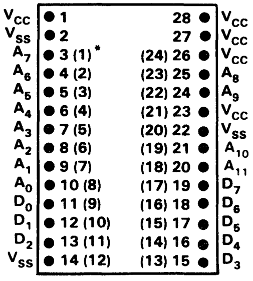
A 28 or 24 pin socket is located on top of the 40 pin package, depending on the specific device. A 28 pin top socket array was used so that the same package could be used for all 38P70 devices but could accommodate both 24 pin and 28 pin. Due to pin-out differences between various common memory devices, several different versions of the MK38P70 are provided with differing signals connected to particular pins on the socket. Figure 7 shows the various options available. When 24 pin memories are used, they are inserted so that pin 1 of the memory device is plugged into pin 3 of the array (the 24 pin memory is lower justified in the 28 pin array).

MK38P70 I/O PORTS

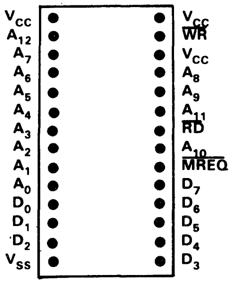
For custom 3870 ROM codes, the user is given a bit by bit selection of I/O options on I/O ports 4 and 5. Additionally, the user has the option of selecting whether or not either $\overline{\text{RESET}}$ or EXT INT has an internal pull-up resistor. This flexibility allows about 172 million possible variations in I/O port and $\overline{\text{RESET}}$ and EXT INT configurations. Obviously, it is not practical to offer this variety in an "off the shelf" product like the 38P70. Thus a few variations are offered which still give some flexibility to the designer. The available I/O options are also shown in Figure 7.

MK38P70 VERSIONS

DEVICE	PORT 4 I/O TYPE	PORT 5 I/O TYPE	SUPPORTS THESE MEMORY DEVICES	TOP 28 PIN SOCKET ARRAY WIRING VERSION
MK97400	TTL	TTL	2716, 2516, 2532, 2758 MK34000 ROM	A
MK97410	Open Drain	Open Drain	2716, 2516, 2532, 2758 MK34000 ROM	A
MK97501	TTL	TTL	2764, 2732, MK37000 ROM MK34000 ROM	B
MK97521	TTL	Open Drain	2764, 2732, MK37000 ROM MK34000 ROM	B



VERSION A



VERSION B

*NOTE: On Version A packages, sockets are provided in only 24 of the 28 possible locations for a 24 pin EPROM.

Figure 7

28 PIN SOCKET SIGNALS

The 40 package pins are the identical signals that are provided with the MK3870 ROM devices. In addition to these 40 inputs and outputs, various other signals are implemented on the 38P70 die which are available for connection to the top socket. Depending upon the particular version, some subset of these signals are connected to the 28 or 24 pin socket. These signals are described below.

A_0 - A_{11} (97400 Series)
A_0 - A_{15} (97500 Series)

These are the address buses. They are always outputs and a new address will appear on this bus during each machine cycle. Normally this is the address of op-codes or operands, but there are machine cycles wherein no op-code or operand is required by the CPU. During these cycles, an address is still provided but the data that may be read from that address is not used.

D_0 - D_7 (97400 and 97500 Series)

This is the bi-directional data bus for the external memory. Normally these lines are high impedance inputs. During op-code or operand reads, they receive data from the external memory and conduct it onto the internal 38P70 data bus. During those cycles wherein the operation is strictly internal to the 38P70, they remain hi-z inputs. Data may be presented to the 38P70 by an external memory device but it is not conducted onto the internal data bus. This includes machine cycles wherein op-codes or operands are read from the internal executable RAM. During the operand write machine cycle that occurs in the ST (store) instruction, they become push-pull outputs to conduct data to be written out to the external memory. However, if data is written to the internal executable RAM, this transaction is strictly internal and thus the data bus lines remain in their hi-z state. It, therefore, depends upon the address as to whether this bus becomes an active output bus or remains high impedance. If the address of the operand is not within the internal executable RAM space when a ST instruction is executed, D_0 - D_7 will become active outputs at the appropriate time, or else they will remain in the hi-z state. The 97400 devices do not provide a RD (read) control signal, nor is this signal provided on all versions of the 97500 series. Thus if a ST is executed with the operand address being that of external memory, that memory may access data and drive it onto D_0 - D_7 while the 38P70 is also driving data onto D_0 -D_7 and a bus conflict will result. This condition should be avoided; thus the user should note whether or not his external memory will drive D_0 - D_7 in this event. If it will drive D_0 - D_7, an ST with that operand address should be avoided. In general, one would not normally execute a writ to a memory location where there is ROM or EPRON memory instead of RAM. However, some 3870 users hav found the ST instruction useful even in devices like th 3870/20 which have no executable RAM. In this case causes the data counter to increment (to perhaps totaliz some event) but otherwise does nothing as one cann write the internal ROM. No internal conflicts will occur one attempts to write a 3870 ROM location. Most 9750 versions place a $\overline{RD}$ (read, active low) signal on the to socket pin which matches the $\overline{OE}$ (output enable, active low input on most memories. Since $\overline{RD}$ will remain high durin an operand write, the external memory would not have it data outputs enabled and no conflict will occur.

$\overline{MREQ}$ (97500 Series Only)

This is an active low output which occurs during eac machine cycle. It goes high at the start of each cycle the goes low for the remainder of the cycle.

$\overline{RD}$ (97500 Series Only)

This is the active low read output which goes high at the start of each cycle then goes low if data (op-codes o operands) are to be read from external memory. During cycles wherein a strictly internal operation occurs, $\overline{RD}$ wil remain high. It will also remain high during an operan write cycle.

$\overline{WR}$ (97500 Series Only)

This is the active low write control output. It is normally hig but will go low then return high during an operand write the address is not that of internal executable RAM.

$\overline{FETCH}$ (97500 Series Only)

This is the active low fetch status signal which signals tha an op-code fetch occurred during that cycle. It is generate for use of the 97500 as a development system componen

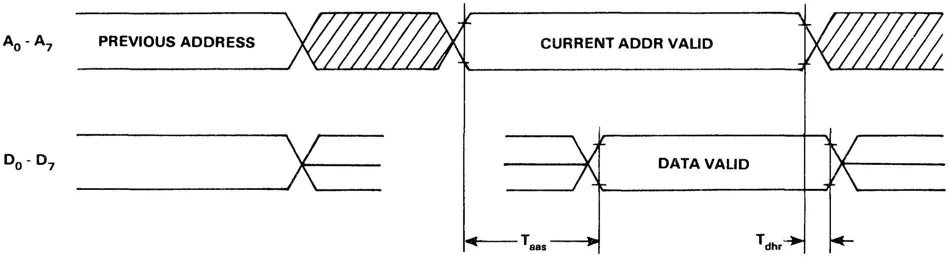
It will go low during all op-code fetches whether fro internal or external memory.

38P70 EXTERNAL MEMORY TIMING

The following Figures show the relative waveforms for th signals used to interface with external memory. The timin parameters are labeled. Their values are given in the A.C Characteristics section of the Electrical Specifications.

97400 SERIES TIMING

Read Cycle

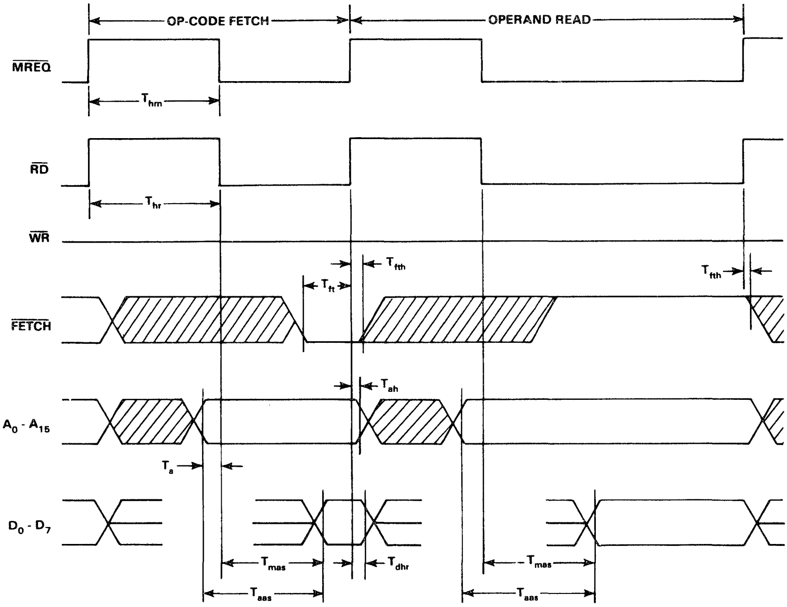


▨ = Actively driven by 38P70 but not necessarily in valid state.

▭ = Not actively driven by the 38P70. May be actively driven by external memory but does not have to be in a valid state or may be placed in hi-z state by external memory.

Figure 8

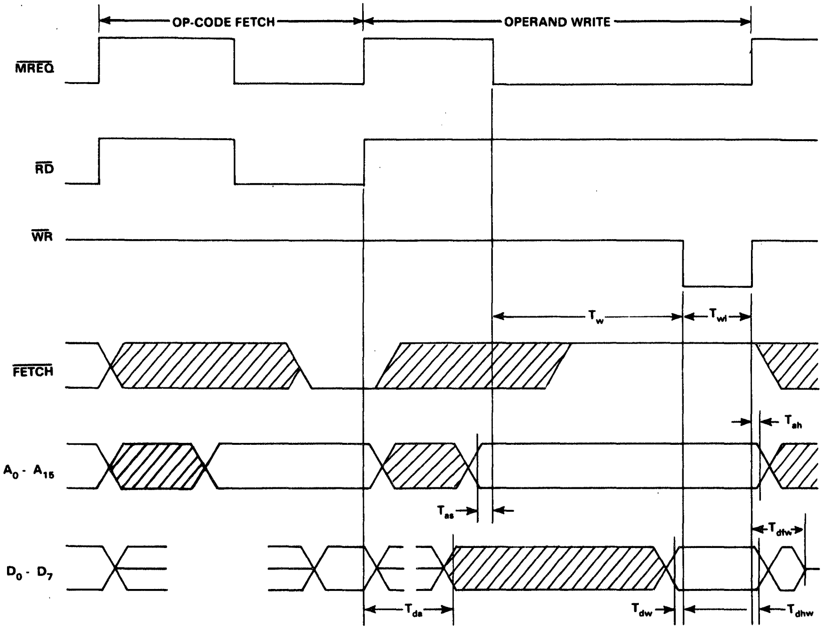
97500 SERIES TIMING



A. OP-CODE AND OPERAND READ
FROM EXTERNAL MEMORY

Figure 9

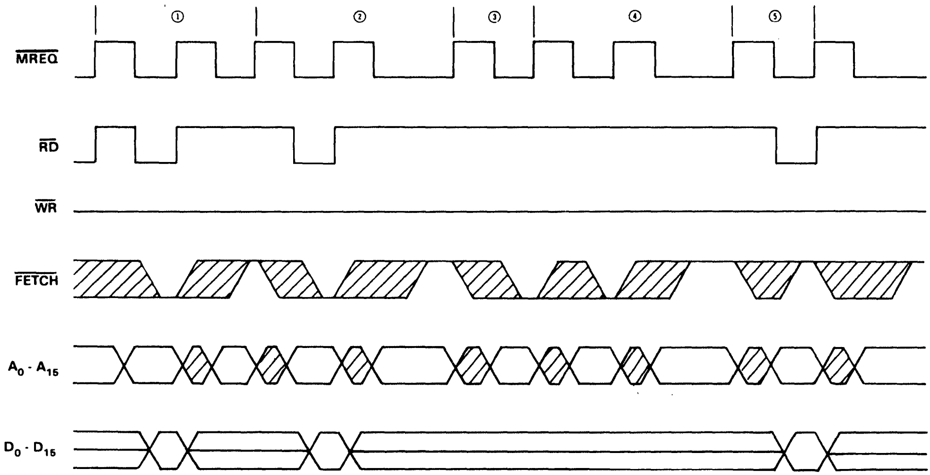
97500 SERIES TIMING (Continued)



B. OPERAND WRITE TO EXTERNAL MEMORY

Figure 9

97500 SERIES TIMING (Continued)



C. EXAMPLES OF VARIOUS CYCLES

① Op-code fetch from external memory followed by an internal cycle (short cycle).
② Op-code fetch from external memory followed by an internal cycle (long cycle) such as an operand read or write internal executable RAM.
③ Op-code fetch from internal executable RAM.
④ Op-code fetch from internal executable RAM followed by a internal cycle (long cycle) such as an operand read or write of internal executable RAM.
⑤ First cycle of an interrupt acknowledge. Had an interrupt not occurred, this would have been an op-code fetch. If it would have been an external op-code fetch, RD will still go low but FETCH will not indicate a fetch cycle. Externally this would appear to be an operand read except that it occurs in a short cycle and all real operand reads occur in a long cycle.

Figure 9

3870 TIME BASE OPTIONS

The 3870 contains an on-chip oscillator circuit which provides an internal clock. The frequency of the oscillator circuit is set from the external time base network. The time base for the 3870 may originate from one of four sources:

1) Crystal
2) LC Network
3) RC Network
4) External Clock

The type of network which is to be used with the mask ROM MK3870 must be specified at the time when mask ROM devices are ordered. However, the MK38P70 may operate with any of the four configurations so that it may emulate any configuration used with a mask ROM device.

The specifications for the four configurations are given in the following text. There is an internal 26 pF capacitor between XTL 1 and GND and an internal 26 pF capacitor between XTL 2 and GND. Thus, external capacitors are not necessarily required. In all external clock modes the external time base frequently is divided by two to form the internal PHI clock.

CRYSTAL SELECTION

The use of a crystal as the time base is highly recommended as the frequency stability and reproduction from system to system is unsurpassed. The 3870 has an internal divide by two to allow the use of inexpensive and widely available TV Color Burst Crystals (3.58 MHz). Figure 11 lists the required crystal parameters for use with the 3870. The Crystal Mode time base configuration is shown in Figure 10.

Through careful buffering of the XTL1 pin it may be possible to amplify this waveform and distribute it to other devices. However, Mostek recommends that a separate active device (such as a 7400 series TTL gate) be used to oscillate the crystal and that the waveform from that oscillator be buffered and supplied to all devices, including the 3870, if a single crystal is to provide the time base for more than just a single 3870.

While a ceramic resonator may work with the 3870 crystal oscillator, it was designed specifically to support the use of this component. Thus, Mostek does not support the use of a ceramic resonator either through proper testing, parametric specification, or applications support.

LC NETWORK

The LC time base configuration can be used to provide a les expensive time base for the 3870 than can be provided wit a crystal. However, the LC configuration is much les accurate than is the crystal configuration. The LC time bas configuration is shown in Figure 12. Also shown in th figure are the specified parameters for the LC component: along with the formula for calculating the resulting tim base frequency. The minimum value of the inductor whic is required for proper operation of the LC time base networ is 0.1 millihenries. The inductor must have a Q factor whic is no less than 40. The value of C is derived from C externa the internal capacitance of the 3870, C_{XTL}, and the stra

CRYSTAL MODE CONNECTION

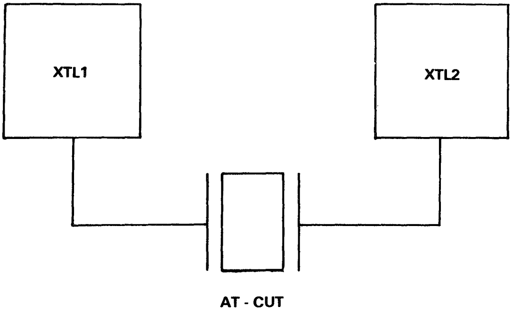


Figure 10

CRYSTAL PARAMETERS

a) Parallel resonance, fundamental mode AT-Cut
b) Shunt capacitance (C_O) = 7 pf max.
c) Series resistance (R_S) = See table
d) Holder = See table below.

Frequency	Series Resistance	Holder
f = 2-2.7 MHz	Rs = 300 ohms max	HC-6 HC-33
f = 2.8-4 MHz	Rs = 150 ohms max	HC-6 HC-18* HC-25* HC-33

*This holder may not be available at frequencies near the lower end of this range.

Figure 11

LC MODE CONNECTION

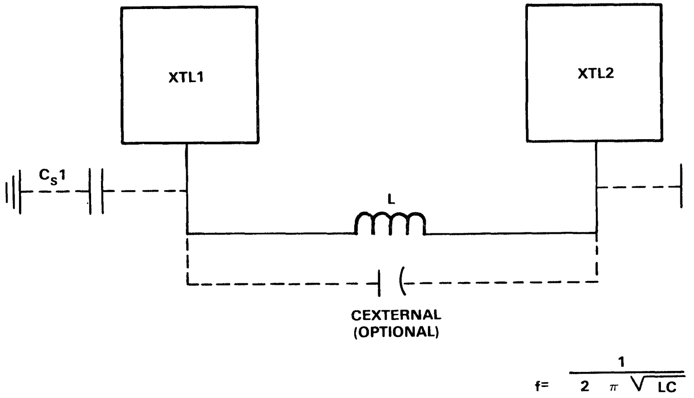


Figure 12

capacitances, C_{S1} and C_{S2}. C_{XTL} is the capacitance looking into the internal two port network at XTL1 and XTL2. C_{XTL} is listed under the "Capacitance" section of the Electrical Specifications. C_{S1} and C_{S2} are stray capacitances from XTL1 to ground and from XTL2 to ground, respectively. C external should also include the stray shunt capacitance across the inductor. This is typically in the 3 to 5 pf range and significant error can result if it is not included in the frequency calculation.

Variation in time base frequency with the LC network can arise from one of four sources: 1) Variation in the value of the inductor. 2) Variation in the value of the external capacitor. 3) Variation in the value of the internal capacitance of the 3870 at XTL1 and XTL2 and 4) Variation in the amount of stray capacitance which exists in the circuit. Therefore, the actual frequency which is generated by the LC circuit is within a range of possible frequencies, where the range of frequencies is determined by the worst case variation in circuit parameters. The designer must select component values such that the range of possible frequencies with the LC mode does not go outside of the specified operating frequency range for the 3870.

RC CLOCK CONFIGURATION

The time base for the 3870 may be provided from an RC network tied to the XTL2 pin, when XTL1 is grounded. A schematic picturing the RC clock configuration is shown in Figure 13. The RC time base configuration is intended to provide an inexpensive time base source for applications in which timing is not critical. Some users have elected to tune each unit using a variable resistor or external capacitor thus reducing the variation in frequency. However, for increased time base accuracy, Mostek recommends the use of the Crystal or LC time base configuration. Figure 14 illustrates a curve which gives the resulting operating frequency for a particular RC value. The x-axis represents the product of the value of the resistor times the value of the capacitor. Note that three curves are actually shown. The curve in the middle represents the nominal frequency obtained for a given value of RC. A maximum curve and a minimum curve for different types of 3870 devices are also shown in the diagram.

The designer must select the RC product such that a frequency of less than 2 MHz is not possible, taking into account the maximum possible RC product and using the minimum curve shown in Figure 14 below. Also, the RC product must not allow a frequency of more than 4 MHz, taking into account the minimum possible R and C and using the Maximum curve shown below. Temperature induced variations in the external components should be considered in calculating the RC product.

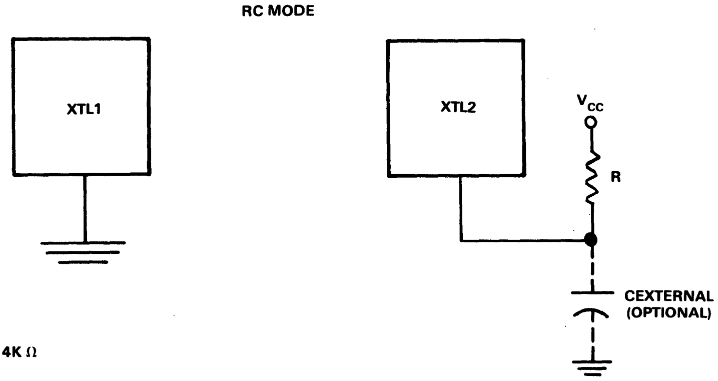
Frequency variation from unit to unit owing to switching speed and level at constant temperature and V_{CC} = + or – 5 percent.

Frequency variation due to V_{CC} with all other parameters constant with respect to +5V = +7 percent to –4 percent on all devices.

Frequency variation due to temperature with respect to 25 C (all other parameterء constant) is as follows:

PART #	VARIATION
387X-00, -05	+6 percent to – 9 percent
387X-10, -15	+9 percent to -12 percent

RC MODE CONNECTION



MINIMUM R = 4K Ω

C = 26.5 pF ± 2.6pF + Cexternal

Figure 13

FREQUENCY VS. RC

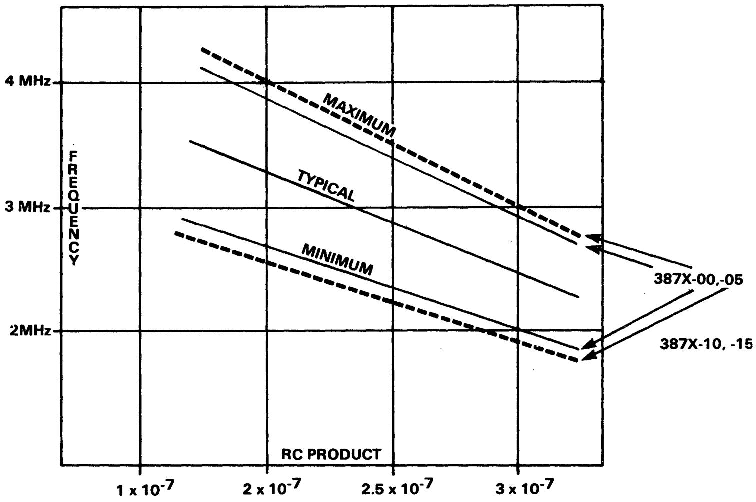


Figure 14

Variations in frequency due to variations in RC components may be calculated as follows:

Maximum RC = (R max) (C external max + C_{XTL} max)

Minimum RC = (R min) (C external min + C_{XTL} min)

$$\text{Typical RC} = (\text{R typ})\left(\text{C external typ} + \frac{\{C_{XTL}\ \text{max} + C_{XTL}\ \text{min}\}}{2}\right)$$

$$\text{Positive Freq. Variation} = \frac{\text{RC typical - RC minimum}}{\text{RC typical}}$$

$$\begin{array}{l}\text{Negative Freq. Variation} \\ \text{due to RC Components}\end{array} = \frac{\text{RC maximum - RC typical}}{\text{RC typical}}$$

Total frequency variation due to all factors:

387X-00, -05	387X-10, -15
= +18 percent plus positive frequency variation due to RC components	= +21 percent plus positive frequency variation due to RC components
= –18 percent minus negative frequency variation due to RC components	= –21 percent minus negative frequency variation due to RC components

Total frequency variation due to V_{CC} and temperature of a unit tuned to frequency at +5V V_{CC}, 25 C

387X-00, -05	387X-10, -15
= + 13 percent	= + 16 percent

EXTERNAL CLOCK CONFIGURATION

The connection for the external clock time base configuration is shown in Figure 15. Refer to the DC Characteristics section for proper input levels and current requirements.

Refer to the Capacitance section of the appropriate 3870 Family device data sheet for input capacitance.

EXTERNAL MODE CONNECTION

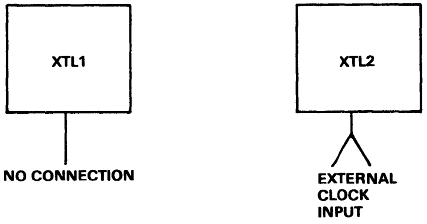


Figure 15

ELECTRICAL SPECIFICATIONS
MK3870, MK38P70

OPERATING VOLTAGES AND TEMPERATURES

Dash Number Suffix	Operating Voltage V_{CC}	Operating Temprature T_A
— 00	+5V ± 10%	0°C - 70°C
— 05	+5V ± 5%	0°C - 70°C
— 10	+5V ± 10%	-40°C - +85°C
— 15	+5V ± 5%	-40°C - +85°C

See Ordering Information for explanation of part numbers.

MAXIMUM RATINGS*

	-00, -05	-10, -15
Temperature Under Bias	-20°C to +85°C	-50°C to +100°C
Storage Temperature	-65°C to +150°C	-65°C to +150°C
Voltage on any Pin With Respect to Ground (Except open drain pins and TEST)	-1.0V to +7V	-1.0V to +7V
Voltage on TEST with Respect to Ground	-1.0V to +9V	-1.0V to +9V
Voltage on Open Drain Pins With Respect to Ground	-1.0V to +13.5V	-1.0V to +13.5V
Power Dissipation	1.5W	1.5W
Power Dissipation by any one I/O pin	60mW	60mW
Power Dissipation by all I/O pins	600mW	600mW

*Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other condition above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

AC CHARACTERISTICS
T_A, V_{CC} within specified operating range.
I/O power dissipation ≤ 100mW (Note 2)

			-00,-05		-10,-15			
SIGNAL	SYM	PARAMETER	MIN	MAX	MIN	MAX	UNIT	NOTES
XTL1 XTL2	t_o	Time Base Period, all clock modes	250	500	250	500	ns	4MHz-2MHz
	$t_{ex(H)}$	External clock pulse width high	90	400	100	390	ns	
	$t_{ex(L)}$	External clock pulse width low	100	400	110	390	ns	
Φ	t_Φ	Internal Φ clock	$2t_0$		$2t_0$			
WRITE	t_W	Internal WRITE Clock period	4tΦ 6tΦ		4tΦ 6tΦ			Short Cycle Long Cycle
I/O	$t_{dI/O}$	Output delay from internal WRITE clock	0	1000	0	1200	ns	50pF plus one TTL load
	$t_{sI/O}$	Input setup time to internal WRITE clock	1000		1200		ns	
$\overline{\text{STROBE}}$	$t_{I/O\text{-}s}$	Output valid to $\overline{\text{STROBE}}$ delay	3tΦ -1000	3tΦ +250	3tΦ -1200	3tΦ +300	ns	I/O load = 50pF + 1 TTL load
	t_{sL}	$\overline{\text{STROBE}}$ low time	8tΦ -250	12tΦ +250	8tΦ -300	12tΦ +300	ns	$\overline{\text{STROBE}}$ load= 50pF + 3TTL loads
$\overline{\text{RESET}}$	t_{RH}	$\overline{\text{RESET}}$ hold time, low	6tΦ +750		6tΦ +1000		ns	
	t_{RPOC}	$\overline{\text{RESET}}$ hold time, low for power clear	power supply rise time ·0 1		power supply rise time ·.15		ms	
EXT INT	t_{EH}	EXT INT hold time in active and inactive state	6tΦ +750		6tΦ +1000		ns	To trigger interrupt
			2tΦ		2tΦ		ns	To trigger timer

AC CHARACTERISTICS FOR MK38P70 Signals brought to top 28 pin socket.
T_A, V_{CC} within specified operating range.
I/O Power Dissipation ≤ 100 mW (Note 2)
97400 Series (See Note 3)

SYMBOL	PARAMETER	-00, -05		-10, -15		UNIT	CONDITION
		MIN	MAX	MIN	MAX		
t_{aas}	External memory required access time from A_0-A_{11} stable	3tΦ –850		3tΦ –850		ns	C_L A_0-A_{11} = 50 pF

97500 Series (See Note 3)

SIGNAL	SYMBOL	PARAMETER	-00, -05		-10, -15		UNITS	CONDITION
			MIN	MAX	MIN	MAX		
$\overline{\text{MREQ}}$	T_{hm}	$\overline{\text{MREQ}}$ high time	2tΦ –100		2tΦ –100		ns	Load = 50 pF + 1 TTL load
$\overline{\text{RD}}$	T_{hr}	$\overline{\text{RD}}$ high time	2tΦ –100		2tΦ –100		ns	Load = 50 pF + 1 TTL load
$\overline{\text{WR}}$	T_w	$\overline{\text{WR}}$ low from $\overline{\text{MREQ}}$ low	3 to –200	3 to +100	3 to –200	3 to +100	ns	Load = 50 pF + 1 TTL load
	T_{wl}	$\overline{\text{WR}}$ low time	to –100	to +100	to –100	to +100	ns	
$\overline{\text{FETCH}}$	T_{ft}	$\overline{\text{FETCH}}$ stable prior to rising $\overline{\text{MREQ}}$	650	650	650	650	ns	Load = 50 pF + 1 TTL Load
	T_{fh}	$\overline{\text{FETCH}}$ hold time after $\overline{\text{MREQ}}$ high	20		20		ns	Load = 20 pF
A_0 - A_{15}	T_a	Address stable prior to $\overline{\text{RD}}$ or $\overline{\text{MREQ}}$ falling	tΦ –400		tΦ –400		ns	Load = 50 pF + 1 TTL load
	T_{ah}	Address hold time after $\overline{\text{MREQ}}$, $\overline{\text{RD}}$, or $\overline{\text{WR}}$ high	15		15		ns	Load = 20 pF
D_0 - D_7	T_{aas}	External memory required access time from	3tΦ –850		3tΦ –850		ns	
	T_{mas}	External memory required access time from $\overline{\text{MREQ}}$ or $\overline{\text{RD}}$ low	2tΦ –450		2tΦ –450		ns	
	T_{dhr}	Required data hold time after $\overline{\text{MREQ}}$ rising	0		0		ns	
	T_{da}	Data bus active after $\overline{\text{MREQ}}$ or $\overline{\text{RD}}$ high	tΦ		tΦ			
	T_{dw}	Data stable prior to $\overline{\text{WR}}$ falling	5tΦ –2250		5tΦ –2250		ns	Load = 50 pF + 1 TTL load
	T_{dhr}	Data hold after $\overline{\text{WR}}$ high	15		15		ns	Load = 20 pF
	T_{dfw}	Data bus delay to float after $\overline{\text{MREQ}}$ rising		200		200	ns	

CAPACITANCE

T_A = 25°C
All Part Numbers

SYM	PARAMETER	MIN	MAX	UNIT	NOTES
C_{IN}	Input capacitance		10	pF	unmeasured pins grounded
C_{XTL}	Input capacitance; XTL1, XTL2	23.5	29.5	pF	

DC CHARACTERISTICS

T_A, V_{CC} within specified operating range
I/O power dissipation ≤ 100 mW (Note 2)

		-00, -05		-10, -15			
SYMBOL	PARAMETER	MIN	MAX	MIN	MAX	UNIT	DEVICE
I_{CC}	Average Power Supply Current		85		110	mA	MK3870/10 Outputs Open
			85		110	mA	MK3870/20 Outputs Open
			94		125	mA	MK3870/22 Outputs Open
			100		130	mA	MK3870/40 Outputs Open
			100		130	mA	MK3870/42 Outputs Open
			125		150	mA	MK38P70/X02 No EPROM, Outputs Open

DC CHARACTERISTICS (cont.)

SYMBOL	PARAMETER	-00, -05		-10, -15		UNIT	DEVICE
		MIN	MAX	MIN	MAX		
P_D	Power Dissipation		400		525	mW	MK3870/10 Outputs Open
			400		525	mW	MK3870/20 Outputs Open
			440		575	mW	MK3870/22 Outputs Open
			475		620	mW	MK3870/40 Outputs Open
			475		620	mW	MK3870/42 Outputs Open
			600		750	mW	MK38P70/X02 No EPROM, Outputs Open

DC CHARACTERISTICS (cont.)

T_A, V_{CC} within specified operating range, I/O power dissipation ≤ 100mW (Note 2)

SYM	PARAMETER	-00,-05		-10,-15		UNIT	CONDITIONS
		MIN	MAX	MIN	MAX		
V_{IHEX}	External Clock input high level	2.4	5.8	2.4	5.8	V	
V_{ILEX}	External Clock input low level	-.3	.6	-.3	.6	V	
I_{IHEX}	External Clock input high current		100		130	µA	$V_{IHEX}=V_{CC}$
I_{ILEX}	External Clock input low current		-100		-130	µA	$V_{ILEX}=V_{SS}$
$V_{IHI/O}$	Input high level, I/O pins	2.0	5.8	2.0	5.8	V	Standard pull-up
		2.0	13.2	2.0	13.2	V	Open drain (1)
V_{IHR}	Input high level, $\overline{\text{RESET}}$	2.0	5.8	2.2	5.8	V	Standard pull-up
		2.0	13.2	2.2	13.2	V	No Pull-up
V_{IHEI}	Input high level, EXT INT	2.0	5.8	2.2	5.8	V	Standard pull-up
		2.0	13.2	2.2	13.2	V	No Pull-up
V_{IL}	Input low level	-.3	.8	-.3	.7	V	(1)
I_{IL}	Input low current, all pins with standard pull-up resistor		-1.6		-1.9	mA	V_{IN}=0.4V
I_L	Input leakage current, open drain pins, and inputs with no pull-up resistor		+10		+18	µA	V_{IN}=13.2V
			-5		-8	µA	V_{IN}=0.0V
I_{OH}	Output high current pins with standard pull-up resistor	-100		-89		µA	V_{OH}=2.4V
		-30		-25		µA	V_{OH}=3.9V
I_{OHDD}	Output high current, direct drive pins	-100		-80		µA	V_{OH}=2.4V
		-1.5		-1.3		mA	V_{OH}=1.5V
			-8.5		-11	mA	V_{OH}=0.7V
I_{OHS}	$\overline{\text{STROBE}}$ Output High current	-300		-270		µA	V_{OH} = 2.4V
I_{OL}	Output low current	1.8		1.65		mA	V_{OL}=0.4V
I_{OLS}	$\overline{\text{STROBE}}$ Output Low current	5.0		4.5		mA	V_{OL}=0.4V

DC CHARACTERISTICS FOR MK38P70

Signals brought to top 25 pin socket
T_A, V_{CC} within specified range
I/O Power Dissipation ≤ 100 mW (Note 2)

97400, 97500 Series

SYMBOL	PARAMETER	-00, -05		-10, -15		UNIT	CONDITION
		MIN	MAX	MIN	MAX		
V_{IH}	Input high level (D_0 - D_7)	2.2	V_{CC} +.3	2.2	V_{CC} +.3	V	D_0 - D_7 in Hi-z input mode
V_{IL}	Input low level (D_0 - D_7)	V_{SS} -.3	.8	V_{SS} -.3	.7	V	
I_L	Input Leakage (D_0 - D_7)		±10		±15	μA	
V_{OH}	Output high level (all outputs and D_0-D_7 in output mode)	2.4		2.4		V	
V_{OL}	Output low level (all outputs and D_0-D_7 in output mode)		.4		.4	V	
I_{OH}	Output source current (all outputs and D_0-D_7 in output mode)	-100		-90		μA	V_{OH} = 2.4 V
I_{OL}	Output sink current (all outputs and D_0-D_7 in output mode)	1.8		1.65		mA	V_{OL} = .4 V
R_{CC}	Package resistance from device pin 40 to top socket V_{CC} pin(s)					Ω	Pin 28, 27, or 26 when V_{CC}
						Ω	Pin 1 if V_{CC}
						Ω	Pin 23 if V_{CC}
R_{SS}	Package resistance from device pin 20 to top socket V_{SS} pin(s)					Ω	Pin 14 when V_{SS}
						Ω	Pin 2 or 22 when V_{SS}
I_{CC}	Supply current available from top socket V_{CC} pin(s)		-185		-185	mA	ΣI pin 28 27, 26 when V_{CC}
			-20		-20	mA	Pin 1 if V_{CC}
			-10		-10	mA	Pin 23 if V_{CC}
I_{SS}	Supply current available from top socket V_{SS} pin(s)		190		190	mA	Pin 14 if V_{SS}
			2		2	mA	Pin 2 if V_{SS}
			2		2	mA	Pin 22 if V_{SS}

NOTES:

1. RESET and EXT INT have internal Schmit triggers giving minimum .2 V hysteresis.
2. Power dissipation for I/O pins is calcualted by $\Sigma(V_{CC} - V_{IL})(|I_{IL}|) = \Sigma(V_{CC} - V_{OH})(|I_{OH}|) = \Sigma(V_{OH})(I_{OL})$
3. AC timing for external memory signals on 38P70 are measured from either the .8 or 2.0 volt points as applicable. High means at or above 2.0 volts. Low means at or below .8 volts. Stable means high or low as appropriate. Rising means signal is no longer below .8 volts. Falling means signal is no longer above 2.0 volts. Hold times on outputs assume full rated load on reference signal and 20 pf load on specified signal. For 97400 series, only applicable specification is Taas as no other signals are available to reference to other than A_0-A_{11}.

TIMER AC CHARACTERISTICS

Definitions:
Error = Indicated time value - actual time value

tpsc = tΦ x Prescale Value

Interval Timer Mode:

Single interval error, free running (Note 3) .... ±6tΦ
Cumulative interval error, free running (Note 3) 0

Error between two Timer reads (Note 2)	± (tpsc + tΦ)
Start Timer to stop Timer error (Notes 1, 4)	+ tΦ to - (tpsc + tΦ)
Start Timer to read Timer error (Notes 1, 2)	-5tΦ to - (tpsc + 7tΦ)
Start Timer to interrupt request error (Notes 1, 3)	-2tΦ to -8tΦ
Load Timer to stop Timer error (Note 1)	+ tΦ to - (tpsc + 2tΦ)
Load Timer to read Timer error (Notes 1, 2)	-5t ± to - (tpsc + 8t ±)
Load Timer to interrupt request error (Notes 1, 3)	-2tΦ to -9tΦ

Pulse Width Measurement Mode:

Measurement accuracy (Note 4)	+ tΦ to -(tpsc + 2tΦ)
Minimum pulse width of EXT INT pin	2tΦ

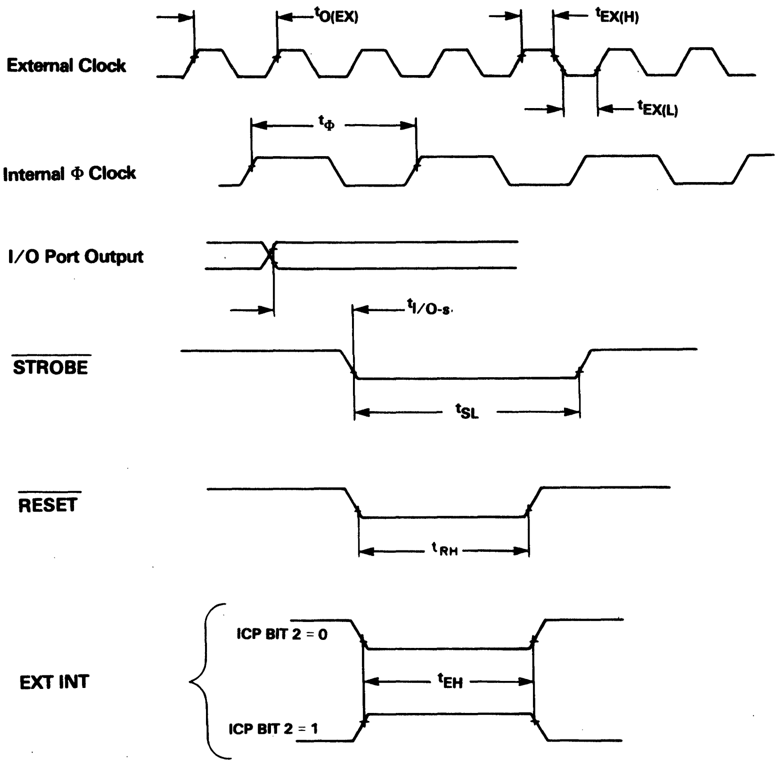
Event Counter Mode:

Minimum active time of EXT INT pin	2tΦ
Minimum inactive time of EXT INT pin	2tΦ

Notes:

1. All times which entail loading, starting, or stopping the Timer are referenced from the end of the last machine cycle of the OUT or OUTS instruction.
2. All times which entail reading the Timer are referenced from the end of the last machine cycle of the IN or INS instruction.
3. All times which entail the generation of an interrupt request are referenced from the start of the machine cycle in which the appropriate interrupt request latch is set. Additional 'ime may elapse if the interrupt request occurs during a privileged or multicycle instruction.
4. Error may be cumulative if operation is repetitively performed.

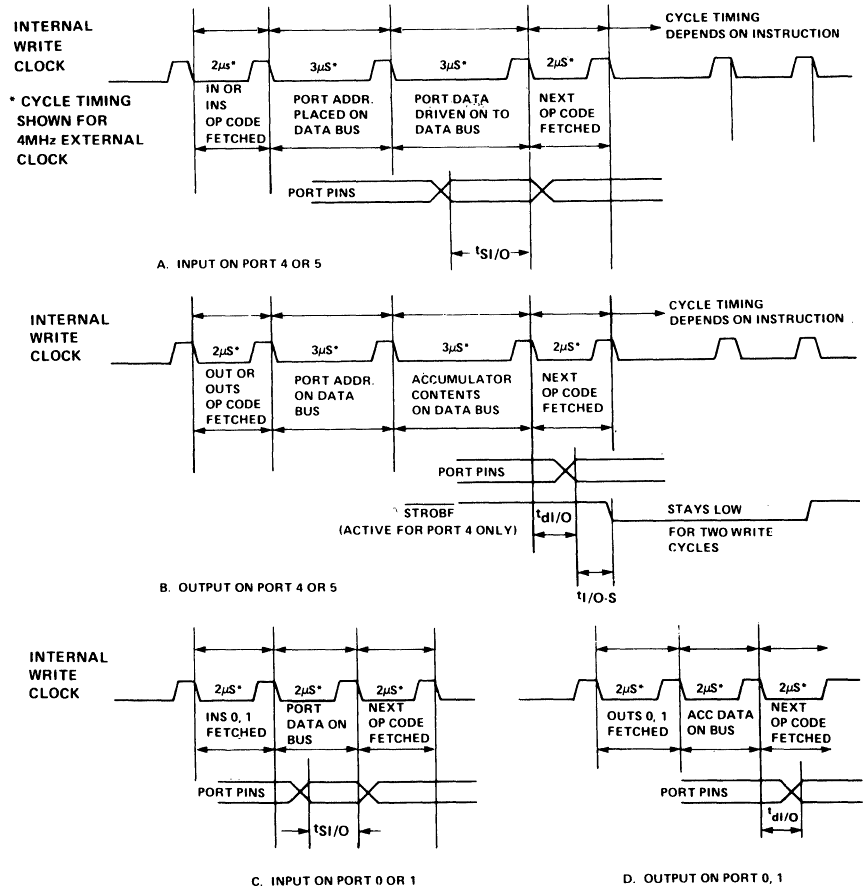
AC TIMING DIAGRAM



Note: All AC measurements are referenced to V_{IL} max., V_{IH} min., V_{OL} (.8v), or V_{OH} (2.0v).

Figure 16

INPUT/OUTPUT AC TIMING



A. INPUT ON PORT 4 OR 5

B. OUTPUT ON PORT 4 OR 5

C. INPUT ON PORT 0 OR 1

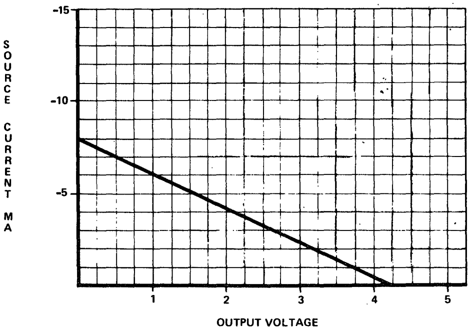
D. OUTPUT ON PORT 0, 1

Figure 17

STROBE SOURCE CAPABILITY

(TYPICAL AT V_{CC} = 5 V, T_A = 25°C)

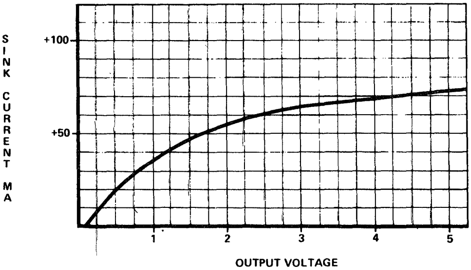
Figure 18



STROBE SINK CAPABILITY

(TYPICAL AT V_{CC} = 5 V, T_A = 25°C)

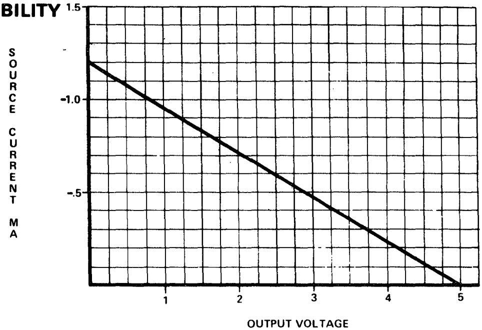
Figure 19



STANDARD I/O PORT SOURCE CAPABILITY

(TYPICAL AT V_{CC} = 5 V, T_A = 25°C)

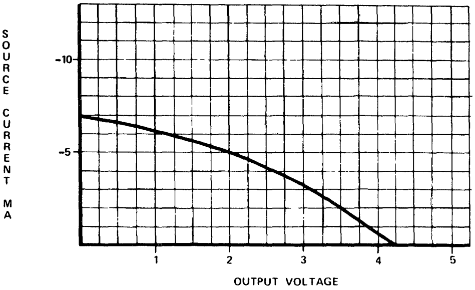
Figure 20



DIRECT DRIVE I/O PORT SOURCE CAPABILITY

(TYPICAL AT V_{CC} = 5 V, T_A = 25°C)

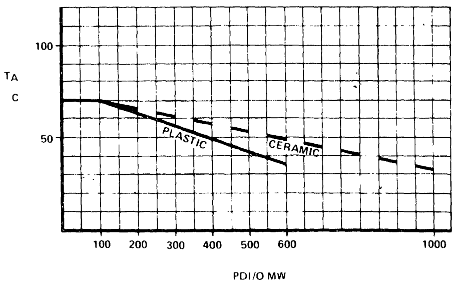
Figure 21



I/O PORT SINK CAPABILITY

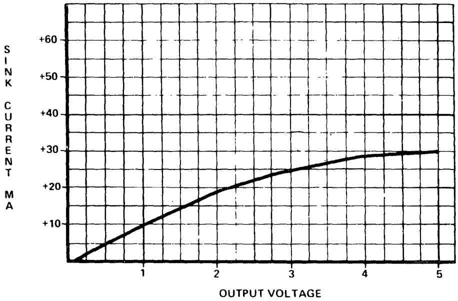
(TYPICAL AT V_{CC} = 5 V, T_A = 25°C)

Figure 22



MAXIMUM OPERATING TEMPERATURE VS. I/O POWER DISSIPATION

Figure 23

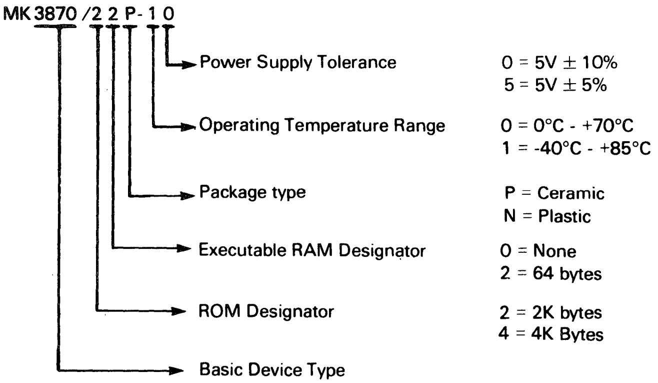


ORDERING INFORMATION

There are two types of part numbers for the 3870 family of devices. The generic part number describes the basic device type, the amount of ROM and Executable RAM, the desired package type, temperature range, and power supply tolerance. For each customer specific code, additional information defining I/O options and oscillator options will be combined with the information described in the generic part number to define a customer/code specific device order number. Note: the specific device order number will be used to differentiate between the MK3870/20 with 12-bit Address Registers and the original 3870 with 11-bit Address Register, as mentioned in an earlier section.

GENERIC PART NUMBER

An example of the generic part number is shown below.


MK3870 /2 2 P- 1 0

Power Supply Tolerance: 0 = 5V ± 10%, 5 = 5V ± 5%

Operating Temperature Range: 0 = 0°C - +70°C, 1 = -40°C - +85°C

Package type: P = Ceramic, N = Plastic

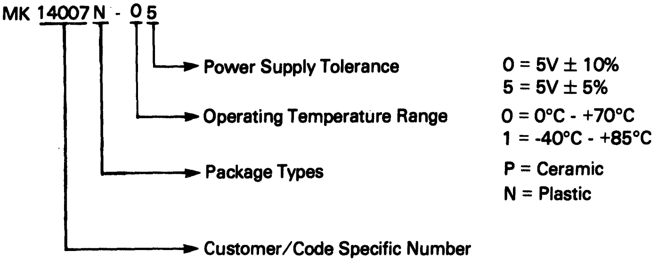
Executable RAM Designator: 0 = None, 2 = 64 bytes

ROM Designator: 2 = 2K bytes, 4 = 4K Bytes

Basic Device Type

DEVICE ORDER NUMBER

An example of the device order number is shown below.



MK 14007 N - 0 5

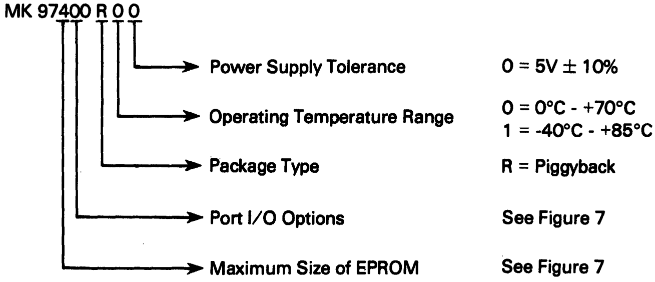
Power Supply Tolerance	0 = 5V ± 10% 5 = 5V ± 5%
Operating Temperature Range	0 = 0°C - +70°C 1 = -40°C - +85°C

Package Types	P = Ceramic
	N = Plastic

Customer/Code Specific Number	

The Customer/Code specific number defines the ROM bit pattern, I/O configuration, oscillator type, and generic part type to be used to satisfy the requirements of a particular customer purchase order. For further information on the ordering of mask ROM devices, the customer should refer to the 3870 Family Technical Manual.

Examples of a 38P70 device order number is shown below.



MK 97400 R 0 0

Power Supply Tolerance	0 = 5V ± 10%
Operating Temperature Range	0 = 0°C - +70°C 1 = -40°C - +85°C
Package Type	R = Piggyback
Port I/O Options	See Figure 7
Maximum Size of EPROM	See Figure 7

These specifications are subjet to change without notice.
Please inquire with our sales offices about the availability of the different products.

Printed in France

Ref.DSMK3870P70

THOMSON
COMPONENTS
SEMICONDUCTEURS

MK3873 • MK38P73

3870 SINGLE CHIP MICRO FAMILY

MK3873 FEATURES

- Available with 2K byte mask programmable ROM
- Software compatible with 3870 instruction set
- 64 byte scratchpad RAM
- 64 byte Executable RAM
- 29 bits (4 ports) TTL compatible parallel I/O
- Serial Input/Output port
  External or Internal Serial Port Clock
  Transmit and Receive registers double buffered
  Internal Baud rate generator
  Synchronous or Asynchronous serial I/O
  Data rates to 9600 bits per second (ASYNC)
  I/O pins dedicated as SERIAL IN, SERIAL OUT, and SERIAL CLOCK
  Variable duty cycle waveform generation
- Vectored interrupts
- Programmable binary timer
  Internal timer mode
  Pulse width measurement mode
  Event counter mode
- External Interrupt
- Crystal, LC, RC or external time base options available
- Low power (325 mW typ.)
- Single +5V power supply
- Pinout compatible with the 3870 Family members

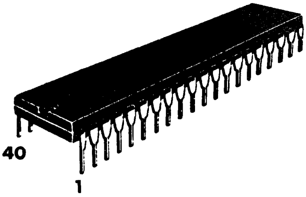
MK38P73 FEATURES

- EPROM version of MK3873
- Piggyback PROM (P-PROM)™ package
- Accepts 24 pin or 28 pin EPROM memories
- Identical pinout as MK3873
- In-Socket emulation of MK3873

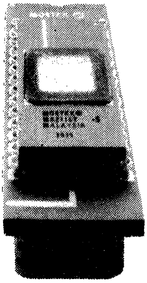
NMOS

CASES

MK3873



MK38P73



FEBRUARY 1987 1/26

MK3873 BLOCK DIAGRAM

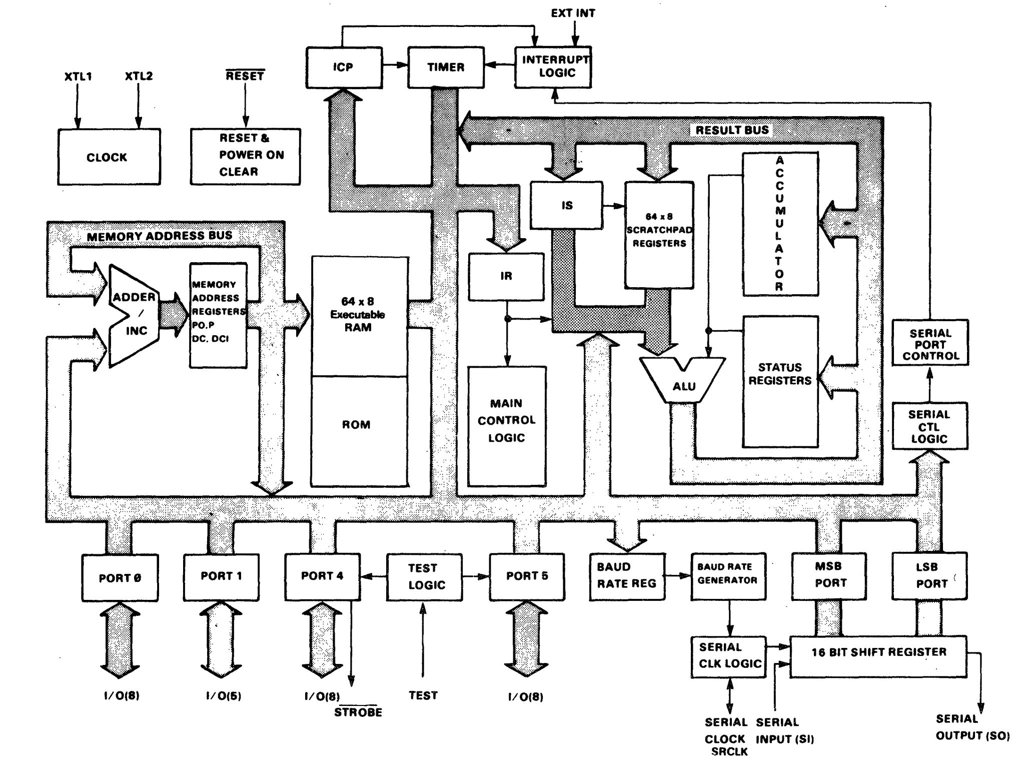
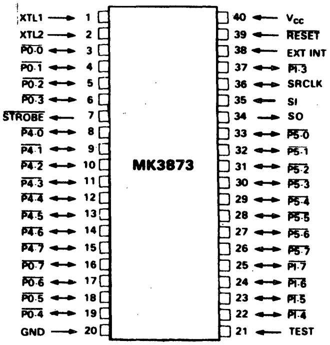
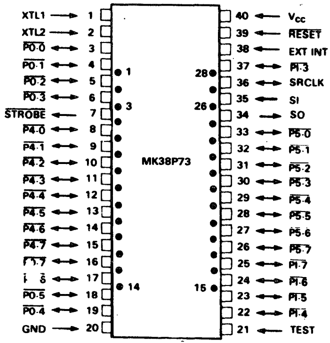


Figure 1

PIN ASSIGNMENTS



PIN NAME	DESCRIPTION	TYPE
P0-0, P0-7	I/O Port 0	Bidirectional
P1-3 - P1-7	I/O Port 1	Bidirectional
P4-0 - P4-7	I/O Port 4	Bidirectional
P5-0 - P5-7	I/O Port 5	Bidirectional
STROBE	Ready Strobe	Output
EXT INT	External Interrupt	Input
RESET	External Reset	Input
SI	Serial Input	Input
SO	Serial Output	Output
SRCLK	Serial Clock	Bidirectional
TEST	Test Line	Input
XTL 1, XTL 2	Time Base	Input
V_{CC}, GND	Power Supply Lines	Input



GENERAL DESCRIPTION

The MK3873 single chip microcomputer introduces a major addition to the 3870 microcomputer family, a serial input/output port. This serial port is capable of either synchronous or asynchronous serial data transfers. The heart of the serial port is a 16-bit Shift Register that is double-buffered on transmit and receive. The Shift Register clock source can be either the internal baud rate generator or an external clock. An end-of-word vectored interrupt is generated in either transmit or receive mode so that the CPU overhead is only at the word rate and not at the serial bit rate. This serial channel can be used to provide a low-cost data channel for communicating between 3873 microcomputers or between a 3873 and another host computer. The serial port is also very flexible so that it could be used for other purposes such as an interface to external serial logic or serial memory devices.

The MK3873 retains commonality with the 3870 family of single chip microcomputers. It has 2048 bytes of mask ROM for program storage, and 64 bytes of scratchpad random-access memory. Certain versions also include up to 64 bytes of Executable RAM. Also, the 3870's sophisticated programmable binary timer is included and provides for system flexibility by operating in 3 different modes. The MK3873 has a large number of parallel I/O lines available to the user. Twenty nine pins of the MK3873 are dedicated to parallel I/O. In addition, three pins are dedicated to the serial I/O port. These pins provide input, output, and clock for the serial port. The serial clock pin can be driven externally or programmed to provide a 50% duty cycle TTL compatible serial clock. No additional CPU instructions are necessary for use with the serial port. Thus, the MK3873 is instruction set compatible with the rest of the 3870 family.

The MK38P73 microcomputer is the PROM based version of the MK3873 single-chip microcomputer. The MK38P73 is called the Piggyback PROM (P-PROM)™ microcomputer because of a new packaging concept. This concept allows a 24 or 28 pin EPROM to be mounted directly on top of the microcomputer itself. The EPROM can then be removed and reprogrammed as required with a standard PROM programmer. The MK38P73 retains exactly the same pinout and architectural features as other members of the MK3873 Family. The MK38P73 is discussed in more detail in a later section of this document.

FUNCTIONAL DESCRIPTION

$\overline{\text{P0-0}}$ - $\overline{\text{P07}}$, $\overline{\text{P1-3}}$ - $\overline{\text{P1-7}}$, $\overline{\text{P4-0}}$ - $\overline{\text{P4-7}}$, $\overline{\text{P5-0}}$ - $\overline{\text{P5-7}}$ are 29 bidirectional I/O lines which can either be used as TTL compatible inputs or latch outputs.

SI - SERIAL IN is a TTL compatible Schmitt Trigger input pin for either serial synchronous or asynchronous data.

SO - SERIAL OUT is an output line for either serial synchronous or asynchronous data.

SRCLK is the clock for the serial port operations. It can be configured by software to be an input or output depending upon whether an internal baud rate or external clock is desired. It has a Schmitt trigger input and can be used to drive up to 3 TTL loads.

$\overline{\text{STROBE}}$ is a ready strobe associated with I/O Port 4. This pin which is normally high provides a single low pulse after valid data is present on the $\overline{\text{P4-0}}$ - $\overline{\text{P4-7}}$ pins during an output instruction. $\overline{\text{STROBE}}$ can be used to drive up to 3 TTL loads.

$\overline{\text{RESET}}$ may be used to externally reset the MK3873. When pulled low the MK3873 will reset. When allowed to go high the MK3873 will begin program execution at program location H'000'.

EXT INT is the external interrupt input. Its active state is software programmable as described in the 3870 Family Technical Manual. This input is also used in conjunction with the timer for pulse width measurement and event counting.

XTL 1 and XTL 2 are the time base inputs (2 MHz to 4 MHz) to which a crystal, LC network, RC network, or an external single-phase clock may be connected. The time base mode must be specified when submitting an order for a mask ROM MK3873. The MK38P73 will operate with any of the four configurations.

MK3873 ARCHITECTURE

The architecture of the MK3873 is identical to that of the rest of the devices in the 3870 family, with the exception of the serial port logic. The serial port logic is shown in the block diagram of the MK3873 (Figure 1). Addressing of the serial port logic is accomplished through I/O instructions. Operation and programming of the serial port is thoroughly discusssed below. A programming-model of the MK3873 is shown in Figure 2. For a more complete discussion of the 3870 family architecture, the user is referred to the 3870 Family Technical Manual.

MAIN MEMORY

The main memory section on the MK3873 consists of a combination of ROM and executable RAM.

There are four registers associated with the main memory section. These are the Program Counter (PO), the Stack Register (P), the Data Counter (DC), and the Auxiliary Data Counter (DC1). The Program Counter is used to address instructions during program execution. P is used to save the contents of PO during an interrupt or subroutine call. Thus, P contains the return address at which processing is to resume upon completion of the subroutine or the interrupt routine. The Data Counter (DC) is used to address data tables. This register is auto-incrementing. Of the two data counters, only DC can access memory directly. However, the XDC instruction allows DC and DC1 to be exchanged. The length of the P0, P, DC, and DC1 registers for the MK3873/22 device is listed in the table shown in Figure 3. The graph and table in Figure 3 also shows the amounts of ROM and executable RAM.

MK3873 PROGRAMMABLE REGISTERS, PORTS, AND MEMORY MAP

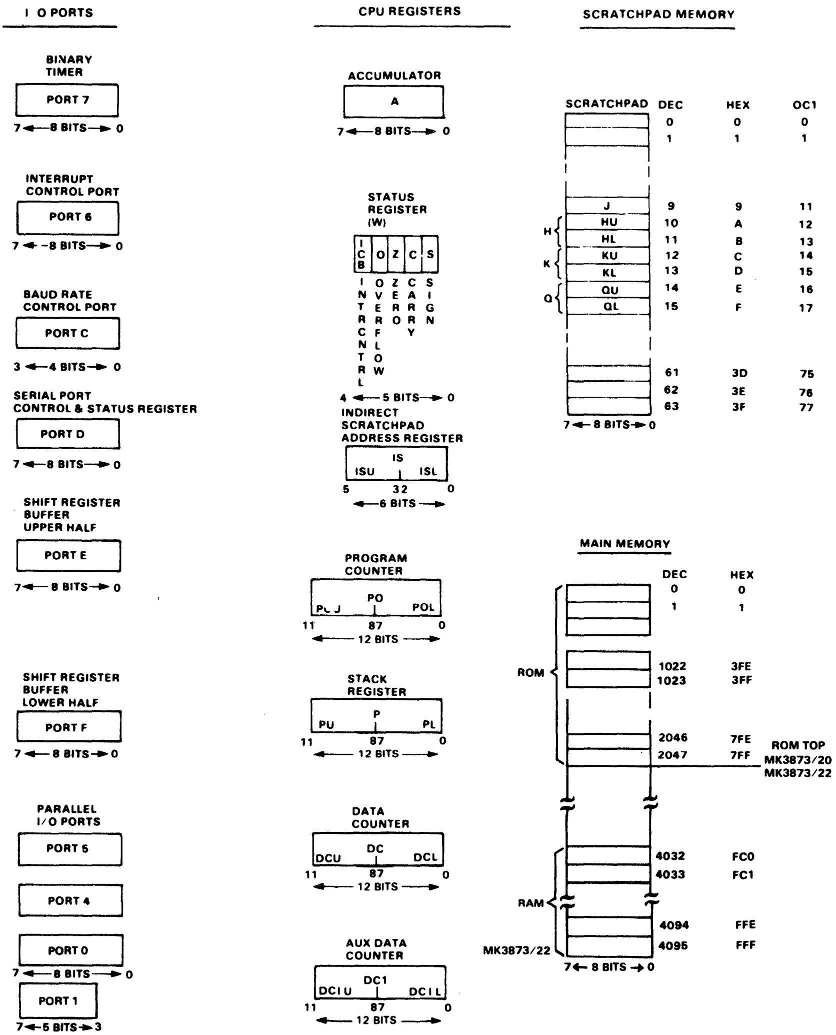
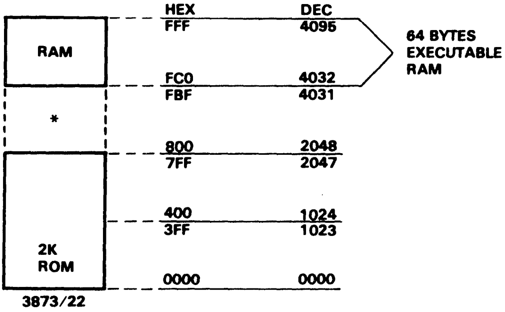


Figure 2

MK3873 MAIN MEMORY SIZE AND TYPE



This device contains 64 bytes of Scratchpad RAM.

*Data derived from addressing these intermediate locations is not tested nor is it guaranteed. Users should refrain from entering this area of the memory map.

Device	Scratchpad RAM Size (Decimal)	Address Register Size PØ, P, DC, DC1	ROM Size (Decimal)	Executable RAM Size
MK3873/22	64 bytes	12 bits	2048 bytes	64 bytes

Figure 3

EXECUTABLE RAM

The upper bytes of the total address space in certain MK3873 devices is RAM memory. As with the ROM memory the RAM may be addressed by the PO and DC address registers. The executable RAM may be accessed by all 3870 instructions which address main memory indirectly through the Data Counter (DC) register. Additionally, the MK3873 may execute an instruction sequence which resides in the Executable RAM. Note that this cannot be done with the scratchpad RAM memory, which is the reason the term "Executable RAM" is given to this additional memory.

I/O PORTS

On the MK3873, 29 lines are provided for bidirectional, parallel I/O. These lines are addressable as four parallel I/O ports at locations 0, 1, 4, and 5. Note that Ports 0, 4, and 5 are 8 bits wide, while Port 1 contains only 5 bits of I/O in bit positions 3, 4, 5, 6, and 7. Bits 0-2 on Port 1 are not available for use as I/O port pins or as storage elements. The remaining three pins are used to provide the serial I/O function. A conceptual schematic of a bidirectional I/O port pin and available output drive options are shown in Figure 4.

As in all other 3870 family devices, an output ready strobe is associated with Port 4. This flag may be used to signal a peripheral device that the MK3873 has just completed an output of new data to port 4. The strobe provides a single low pulse shortly after the output operation is completely finished, so either edge may be used to signal the peripheral. STROBE may also be used as an input strobe by doing a dummy output of H '00' to port 4 after completing the input operation.

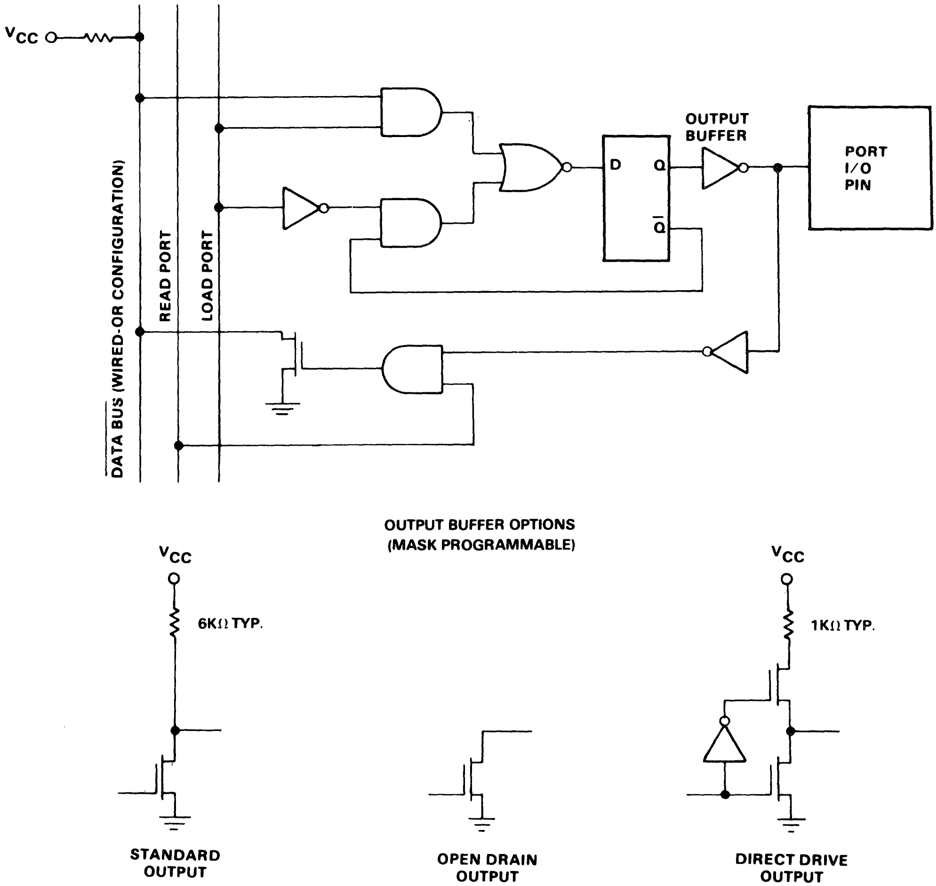
SERIAL I/O OPERATION

The Serial Input/Output Port consists of a serial Shift Register, baud rate generator and control logic as shown in Figure 1. Together these elements provide the MK3873 with a half duplex asynchronous, or a full duplex synchronous, variable bit length serial port. Data is shifted into or out of the shift register at a rate determined by the internal baud rate generator or external clock. An end-of-word interrupt is generated in transmit or receive mode so that the CPU overhead is only at the word rate and not the serial bit rate.

SHIFT CLOCK

The internal clock is used to clock data transfers into and out of the 16 bit Shift Register. It is also used as an input to an

I/O PIN CONCEPTUAL DIAGRAM WITH OUTPUT BUFFER OPTIONS



Ports 0 and 1 are Standard Output type only.

Ports 4 and 5 may both be any of the three output options (mask programmable bit by bit)

The $\overline{\text{STROBE}}$ output is always configured similar to a Direct Drive Output except that it is capable of driving 3 TTL loads.

$\overline{\text{RESET}}$ and EXT INT may have standard 6KΩ (typical) pull-up or may have no pull-up (mask programmable). These two inputs have Schmitt trigger inputs with a minimum of 0.2 volts of hysteresis.

Serial In is a Schmitt trigger input with a minimum of 0.2V hysterisis.

Serial Out (SO) is the Standard Output type.

SRCLK output is capable of driving 3 TTL loads.

Figure 4

internal counter which keeps track of the number of bits which have been shifted into or out of the Shift Register. Input data is sampled on the SERIAL INPUT, (SI), line on the rising edge of the SHIFT clock and is clocked into the most significant bit of the shift register. Output data is gated to the SERIAL OUTPUT line on the falling edge of the internal SHIFT clock.

The clock is derived from the SRCLK pulse. The SRCLK pulse may be generated from the internal baud rate generator or it may be programmed as an input. The internal SHIFT clock operates at the same frequency as the SRCLK pulse when the Sync mode is selected, and at a rate which is divided by 16 (÷16) from the SRCLK pulse when the Async mode is selected.

SHIFT REGISTER

The Serial Port Shift Register is a 16-bit serial to parallel, parallel to serial shift register. This register is addressed and double-buffered by ports E and F as shown in Figure 5A.

PORT D SERIAL PORT CONTROL REGISTER

The Serial Port Control register is write only and is addressed as Port D. The bit assignment is pictured in Figure 5C. The function of each bit is described below.

SERIAL PORT REGISTERS

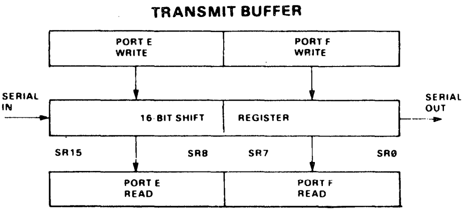


Figure 5A

SERIAL PORT STATUS REGISTER

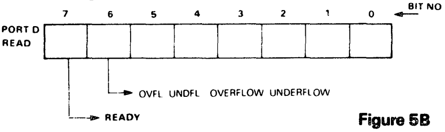
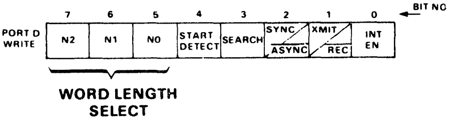


Figure 5B

SERIAL PORT CONTROL REGISTER



N2	N1	N0	WORD LENGTH
0	0	0	4 Bits
0	0	1	7 Bits
0	1	0	8 Bits
0	1	1	9 Bits
1	0	0	10 Bits
1	0	1	11 Bits
1	1	0	12 Bits
1	1	1	16 Bits

Figure 5C

N2, N1, N0 - WORD LENGTH SELECT

These bits select one of the eight possible word lengths which are available with the MK3873 serial port. The serial port will shift the programmed number of bits through the Shift Register. If the Transmit mode is selected, data will be shifted out of the least significant bit (SR0) of the Shift Register to the Serial Out line (SO) while data is simultaneosly sampled at the Serial Input (SI) line and shifted into the most significant bit (SR15) of the Shift Register. When the Receive mode is selected, data will be sampled at SI and shifted in, but the SO line will be disabled such that it remains in a marking condition (Logic "1"). After the programmed number of bits have been shifted, the serial port logic will generate an end-of-word condition. This end-of-word condition will cause an interrupt if the serial port INTERRUPT ENABLE bit has been set.

It should be noted that the word values have been chosen so that the MK3873 can be programmed to send and receive a wide variety of asynchronous serial codes with various combinations of start and stop bits. Shown in Figure 6 is a table which gives the word length.

Values which would be programmed into the MK3873 Serial Port Register for Baudot, ASCII and 8 bit binary codes in an asynchronous word format are shown in the table of Figure 6. Shown in the table are word length values for various combinations of data bits, start and stop bits, and parity. It can be seen that the MK3873 serial port can accommodate many different word lengths of asynchronous or synchronous data.

START DETECT

When the START DETECT bit is enabled the serial port will not shift data through the Shift Register until a valid start bit is detected at the SI input pin. The Start Detect mode is operative only when the Async mode has been selected by programming bit 2 of the Serial Port Control Register to a logic "0". By selecting the Async mode, the internal SHIFT clock frequency is divided by 16 from the clock frequency at the SRCLK pin. (Recall that SRCLK can be an input or an output depending on whether the internal baud rate generator or the external clock is selected). When the START DETECT bit is set, the serial port logic looks for a high to low transition on the SI input. Until this transition occurs, the internal SHIFT clock is held low and no data is shifted in through the shift register. Once the transition is sensed, the SI input will be sampled on every SRCLK pulse for seven clock periods. If the logic level remains at zero on the SI input for each of the seven clock periods, the serial port logic will begin shifting data into the Shift Register on the eighth SRCLK pulse. Data will be shifted in at the ÷16 or SHIFT clock rate until the number of bits which have been programmed into the word length select have been shifted in. Once the programmed number of bits have been shifted in, the start detect circuitry will be rearmed and will begin searching for the next high-to-low transition on SI. This operation is pictured in the example shown in Figure 7.

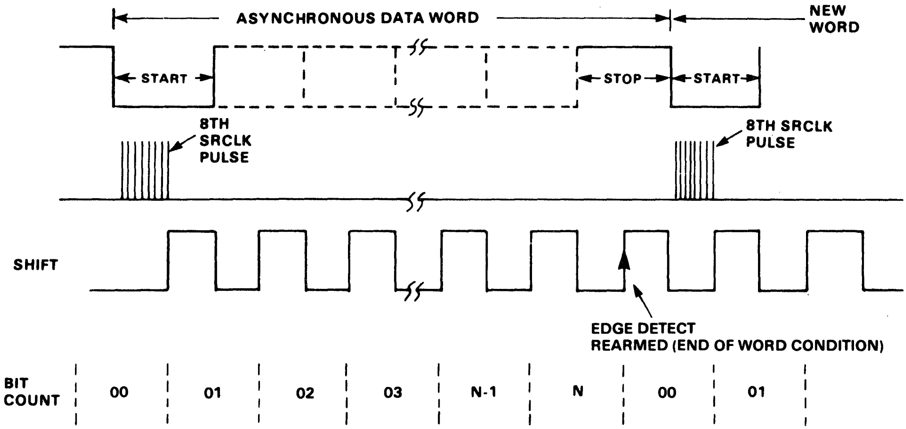
When the START DETECT bit is disabled, data is continuously shifted through the Shift Register. An end-of-

ASYNCHRONOUS WORD LENGTHS

DATA WORD	# OF BITS	START BITS	STOP BITS	PARITY	WORD LENGTH (BITS)
BAUDOT	5	1	1	No	7
	5	1	2	No	8
	5	1	1	Yes	8
	5	1	2	Yes	9
ASCI I	7	1	1	No	9
	7	1	2	No	10
	7	1	1	Yes	10
	7	1	2	Yes	11
8 Bit Binary	8	1	1	No	10
	8	1	2	No	11
	8	1	1	Yes	11
	8	1	2	Yes	12

Figure 6

MK3873 SERIAL PORT START BIT DETECTION



where N is the word length value selected by programming bits N2-N0 in the serial port control register

Figure 7

word condition will be generated each time the programmed number of bits has been shifted into or out of the Shift Register. A serial port interrupt will be generated when the end of word condition occurs if it has been enabled.

SEARCH

The SEARCH bit is enabled by programming it to a logic "1". When enabled, the SEARCH bit causes the serial port logic to generate an interrupt at every bit time if the serial port interrupt has been enabled. This interrupt will occur regardless of whether the Transmit or Receive mode has been selected and whether the Synchronous or Asynchronous mode has been selected. The Search mode is usually used for recognition of a sync character in synchronous serial data transmission. The MK3873 serial port does not automatically detect sync characters.

SYNC/ASYNC

The SYNC/$\overline{\text{ASYNC}}$ bit is used to select either the Synchronous mode of operation or the Asynchronous mode of operation. In the Synchronous mode of operation data is shifted through the Shift Register at a rate which is ÷1 the rate of SRCLK. When the Synchronous mode is selected, the start bit detect circuitry cannot be enabled, even if the START DETECT bit is programmed to a "1". In the Asynchronous mode (SYNC/$\overline{\text{ASYNC}}$ = 0) the internal SHIFT clock operates at a rate which is ÷16 the rate of SRCLK.

XMIT/$\overline{\text{REC}}$

The XMIT/$\overline{\text{REC}}$ bit is used to select either the Transmit or Receive modes of operation. When programmed to a "1" XMIT is selected and the serial port will shift data on the SO line as well as shift data into the SI input. Transmitted data will be enabled on the SO output on the falling edge of the internal SHIFT clock. When the Receive mode is selected (by programming XMIT/$\overline{\text{REC}}$ = 0), data will be clocked into the Shift Register on the rising edge of SHIFT, as it is when the Transmit mode is enabled, but data will be disabled from being shifted out on Serial Out. Serial Out will be held at a marking, or logic "1", condition.

SERIAL PORT INTERRUPT ENABLE

By programming this bit to a "1", the serial port interrupt will be enabled. A serial port interrupt may then occur when an end-of-word condition is generated. Program control will be vectored to one of two locations upon a serial port interrupt, depending on the way the XMIT/$\overline{\text{REC}}$ bit has been programmed. If the Transmit mode has been selected by programming XMIT/$\overline{\text{REC}}$ bit to a "1", then program control will be vectored to location E0 (Hex). For the Receive mode (XMIT/$\overline{\text{REC}}$ = 0) program control will be vectored to 60 (Hex) when the serial port interrupt occurs. With the addition of the Serial Port Interrupt, the MK3873 has three sources of interrupt. If these three interrupts were to occur simultaneously, priority between them would be such that they would be serviced in the following order:

1) Serial Port
2) Timer
3) External Interrupt

STATUS REGISTER

Reading port D of the MK3873 by performing an Input cr Input Short (IN or INS) instruction will load the contents of the Serial Port Status Register into the Accumulator. The two bits which make up the Status Register are shown in Figure 5B. The operation of these two bits is described below:

READY - The meaning of the READY flag depends on whether the Transmit or Receive mode is selected. When the Transmit mode has been selected, the READY flag is set when a Transmit Buffer empty condition occurs. This means that any previous data which may have been loaded into the Transmit Buffer register pair has been transferred into the Shift Register. Loading either byte of the Transmit Buffer will clear the READY flag until the time that the Transmit Buffer register pair is loaded into the Shift Register during an end-of-word condition

In the Receive mode (XMIT/$\overline{\text{REC}}$ = 0), the READY flag is used to indicate a Receive Buffer full condition. This means that a word of the programmed length has been shifted in and has been loaded into the receive buffer register pair. Reading one of the ports E or F which make up the receive buffer register pair will clear the READY flag. The READY flag will remain a 0 until the next word is completely shifted in and loaded into the receive buffer.

OVFL/UNDFL is like the READY flag; the meaning of OVFL/UNDFL depends on the programming of the XMIT/$\overline{\text{REC}}$ bit in the Serial Port Control Register. When the Transmit mode has been selected OVFL/UNDFL is used to indicate a transmitter underflow condition.

A transmitter underflow condition can occur as follows: Assume that the Transmit mode is selected. Suppose that a word is loaded into the Transmit Buffer register. The serial port logic will load the contents of the Transmit Buffer into the Shift Register and will begin to shift the word out on the SO pin. When the contents of the Transmit Buffer are loaded into the Shift Register, the serial port logic will signal the Transmit Buffer empty condition by setting the READY flag to a "1". When the word in the Shift Register is completely shifted out, an end-of-word condition will be generated. The serial port logic will then check to see if new data has been loaded into the Transmit Buffer. If it has not, the OVFL/UNDFL flag will be set, indicating that the serial port logic has run out of data to send. The OVFL/UNDFL flag can be used to signal an error condition to the firmware, or it can be used to signal that all data has been cleared out of the Shift Register for the purposes of line turnaround.

The OVFL/UNDFL flag which, in this case, represents a transmitter underflow condition, is reset by reading the Status Register.

When the Receive mode is programmed, OVFL/UNDFL is used to signal that the Receive Buffer has overflowed. This overflow condition can occur as follows: Suppose that a serial word is shifted in, generating an end-of-word condition. The serial port logic will load the contents of the Shift Register into the Receive Buffer, and will set the READY flag to a "1" to indicate that the Receive Buffer is full. When the next word being received is completely shifted in, generating the next end-of-word condition, the serial port logic will check to see if the Receive Buffer has been read by examining the state of the READY flag. If the READY flag = 0, then the previous word has already been read from the Receive Buffer by the software and the serial port logic will load the current word into the Receive Buffer and will again set the READY flag. If the READY flag = 1, then the previous word has not been read from the Receive Buffer. The serial port logic will load the new word into the

Receive Buffer, destroying the previous word. This action is signalled by the serial port logic setting the OVFL/UNDFL to a "1" signalling a receive buffer overflow condition. In this case reading the status register also 'clears the OVFL/UNDFL flag.

BAUD RATE CONTROL REGISTER

Port C is designated as the Baud Rate Control register. Four bits, 0-3, are used to select nine different internal baud rates or an external clock. When an internal baud rate is programmed, the SRCLK output is generated at a frequency which is divided from the MK3873's time base frequency. The SRCLK frequency can be calculated by dividing the time base frequency by the divide factor shown in Figure 8 for the bit pattern which is programmed into bits C3-C0. Also shown in Figure 7 is the programming of bits C3-C0 to obtain a set of standard baud rates when a 3.6864MHz crystal is used as a time base.

BAUD RATE CONTROL PORT
PORT C WRITE ONLY

PORT C WRITE

Shift Clock Rate @ 3.6864 MHz time base

7	6	5	4	3	2	1	0 Bit No.	SRCLK Divide Factor	SYNC	ASYNC
☒	☒	☒	☒	C3	C2	C1	C0			
				1	0	1	1	÷24	153.6 kbs	9600 bps
				1	0	1	0	÷48	76.8 kbs	4800 bps
				1	0	0	1	÷96	38.4 kbs	2400 bps
				1	0	0	0	÷192	19.2 kbs	1200 bps
				0	1	1	1	÷384	9600 bps	600 bps
				0	1	1	0	÷768	4800 bps	300 bps
				0	1	0	1	÷1536	2400 bps	150 bps
				0	1	0	0	÷2096	1758.8 bps	110 bps
				0	0	1	1	÷3072	1200 bps	75 bps
				0	0	0	0	External Clock Mode		

Figure 8

When any of the internal baud rates are selected, pin 36 becomes an output port pin. This pin is capable of driving three standard TTL inputs and provides a square wave output from the frequency selected in port C. The SYNC/$\overline{\text{ASYNC}}$ bit in the Serial I/O Control register has no effect on the output clock rate. The output will always be ÷1 directly from the baud rate generator.

If all zeros are loaded into this port, the External Clock mode is selected. Pin 36 becomes an input. Any TTL compatible square wave input can be used to generate the clock for the serial port.

TRANSMIT AND RECEIVE BUFFERS

The Receive Buffer registers are two eight bit registers which are addressed as ports E and F (Hex) and are read only. The Receive Buffer registers may be read at any time. The Transmit Buffer registers are also two 8-bit registers which are write only and addressed as ports E and F (Hex).

In the Receive mode, the contents of the 16 bit Shift Register are transferred to the Receive Buffer Register pair when a complete word has been shifted in. Bits SR15-SR8 of the Shift Register are loaded into bits 7-0 of port E while bits SR7-SR0 are loaded into bits 7-0 of Port F.

When entering the Transmit mode, the first data transfer from the Transmit Buffer to the 16 bit Shift Register won't occur until a 1 word time delay after entering Transmit Mode.

In the Receive mode, no transfers between the Transmit Buffer and the 16 bit Shift Register can occur.

The serial port does not automatically right justify incoming data, nor does it insert or strip start and stop bits from an asynchronous data word. Therefore, it is usually necessary to right justify incoming data read from the Receive Buffer registers in software through shift instructions, as well as strip start and stop bits if an asynchronous data format is being used. Likewise, in transmitting an asynchronous data word, it is usually necessary to insert start and stop bits in software into the 16 bit word which is to be loaded in two halves into the Transmit Buffer register.

RESET

The reset circuit on the MK3873 is used to initialize the device to a known condition either during the course of program execution or on a power on condition. This section discusses the effect of $\overline{\text{RESET}}$ on the serial port logic. A more complete description of $\overline{\text{RESET}}$ may be found in the 3870 Family Technical Manual.

Upon reset, both the serial port control register (port D) and the Baud Rate Control register (port C) are loaded with zeroes. This action sets the serial port control logic in the following state:

N2, N1, N0 (word length) = 4 bits
START DETECT disabled
SEARCH disabled
Asynchronous Receive mode
Serial port interrupt disabled
External Clock mode (SRCLK = 1).
Ports E and F are undefined

After the first control word is written to the Serial Port Control Register which selects an internal clock mode, the SRCLK will become an output and will remain high for one-half of a clock period as programmed into port C. It will then go low and produce a clock output waveform with the selected frequency.

ASYNCHRONOUS RECEIVE OPERATION

Figure 7 illustrates the timing for an example using the serial port in the Asynchronous mode. When operating in this mode, the Seria' Port Control Register should be programmed for receive (XMIT/$\overline{\text{REC}}$ = 0) and the START DETECT bit should be enabled. Also, the Async mode should be selected, which allows the start detect circuitry to operate and sets the internal SHIFT clock at a rate which is divided by 16 (÷16) from the SRCLK rate. Upon selecting the

Async mode and the START DETECT bit, the internal SHIFT clock is held low until a negative transition occurs on the SI pin. After a valid edge has been detected (see the START DETECT bit operation section) the SHIFT clock will go high and data will be shifted in at the middle of each bit time. When the programmed number of bits have been shifted in, an end-of-word condition is generated and a serial port receive interrupt will occur if it has been enabled.

After the falling edge of SHIFT following the end-of-word interrupt, the start detect circuitry will be enabled in preparation for the next word. Thus, if a start bit is present immediately following the time when the start detect circuitry is enabled, SHIFT Clock will again go high approximately one bit-time after the rising edge of SHIFT which clocked in the last bit of the preceeding word and caused the end-of-word interrupt. In other words. SHIFT can go high again on the eighth SRCLK pulse as soon as the start detect circuitry is rearmed.

The Shift Register may be read before the next end-of-word condition; otherwise, a receiver overrun error will occur. For a 9600 bps data rate, this would require reading the Receive Buffer within N x 104 μs from the time that the end-of-word condition is generated, where N is the number of bits in the data word.

The example in Figure 7 shows the timing required for asynchronous data reception from a device such as a teletype. Within this data stream are start, data and stop bits. A typical format requires 1 start bit, 8 data bits and 2 stop bits for a total of 11 bits. All of these bits will be residing in the 16 bit Shift Register when the end-of-word interrupt is generated. It is, therefore, necessary to strip the start and stop bits from the data.

SYNCHRONOUS RECEIVE OPERATION

For synchronous operation, the START DETECT bit should not be enabled and the XMIT/$\overline{\text{REC}}$ bit should be programmed to a zero. Also the Sync mode should be enabled so that the internal SHIFT clock is divided by 1, or is equivalent to, SRCLK. Once a control word is written to port D specifying START DETECT = 0, Receive mode, and Sync mode, then the Serial Port will continuously shift data into the MSB of the upper half of the Shift Register at the SRCLK rate and will generate an end-of-word condition when the programmed number of bits have been shifted in.

An illustration of synchronous receive timing is shown in Figure 9. This diagram is a synchronous receive sequence for a word which is N bits in length, where N corresponds to the number of bits which have been programmed into the Serial Port Control Register. Note the relationship of SHIFT clock, the synchronous data stream, and the bit count. Since the START DETECT bit is not enabled, the serial port logic will continuously shift data in and generate end-of-word conditions at regular intervals. When the end-of-word condition occurs, a serial port receive interrupt occurs if it has been enabled, and the contents of the Shift Register will be loaded into the Receive Buffer. The serial port logic will set the READY flag in the Serial Port Status Register, indicating that the receive buffer is full. Since the serial port is double-buffered on receive, the program has entire word time to read the Receive Buffer. At 9600 bps this corresponds to a word time of N x 104μs, where N is the number of bits in a word.

Note that if a new control word is written to port D during the time that a serial word is shifted in, the bit count will be reset.

SYNCHRONOUS TRANSMIT OR RECEIVE TIMING

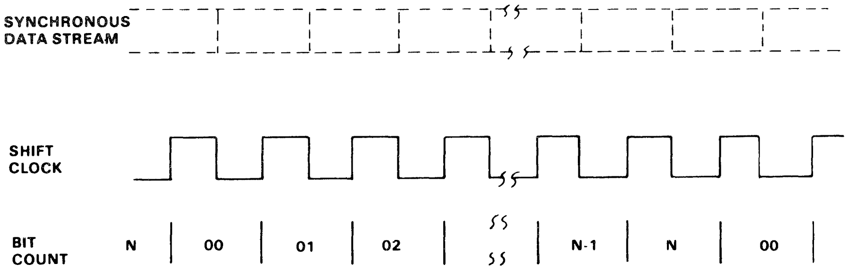


Figure 9

When using the Synchronous Receive mode on the MK3873, it is usually necessary to establish word synchronization in the data stream. The SEARCH bit, when enabled, causes the serial port logic to interrupt on each rising edge of SHIFT so that the data stream can be examined on a bit by bit basis. When the last bit of a sync word is found, the Search mode can be disabled and the serial port logic will shift in data and interrupt at the word rate.

ASYNCHRONOUS TRANSMIT OPERATION

The Asynchronous Transmit mode of operation is initiated by setting the XMIT/$\overline{\text{REC}}$ bit to a "1", and by programming the SYNC/$\overline{\text{ASYNC}}$ bit to a "0". Also, there must be an SRCLK pulse by selecting an internal or external source for SRCLK by programming port C. Upon setting XMIT/$\overline{\text{REC}}$ to a "1", there will be a 1 word length delay prior to the actual transfer of the first word from the Transmit Buffer to the 16 bit Shift Register. Serial data will then be shifted to the right on each rising edge of the internal SHIFT clock, and each new bit in the data stream will be enabled onto the SERIAL OUTPUT pin (SO) at the time of the falling edge of the internal SHIFT clock.

As mentioned, one word time delay is generated between the time that the Transmit mode is initiated by programming XMIT/$\overline{\text{REC}}$ = 1 and the time that the contents of the Transmit Buffer are transferred into the Shift Register. This word time delay is generated internally to the MK3873 by counting the number of SHIFT clock pulses which correspond to the number of bits programmed into the word length select section of the Serial Port Control Register (N2, N1, NO). Therefore, the word time delay is equivalent to the time it takes to shift a complete serial data word out of the Shift Register. The same word time delay will result if data had been loaded prior to programming the XMIT/$\overline{\text{REC}}$ bit to a "1". As mentioned in the "START DETECT" bit description, the internal SHIFT clock is disabled when this bit is programmed to a "1". Since the serial port logic counts SHIFT clock pulses to generate the word time delay, the Transmit Buffer contents will never be transferred to the Shift Register and shifted out when the START DETECT bit is enabled. Also, the Transmit Buffer contents cannot be loaded into the Shift Register when XMIT/$\overline{\text{REC}}$ bit = 0.

When the initial serial data word has been transferred into the Shift Register, the READY flag is set in the Serial Port Status Register which is used to indicate the Transmit Buffer is empty. A transmit interrupt will be generated if the INTERRUPT ENABLE bit has been set in the Serial Port Control Register, and program control will be vectored to location E0 (hex). When operating the serial port in a polled environment with the serial port interrupt disabled, the READY bit can be used as a flag which indicates that new data may be loaded into the Transmit Buffer. In an interrupt driven software configuration, new data may be loaded into the Transmit Buffer at the beginning of the serial port interrupt service routine.

During the operation of the Transmit Mode the SERIAL INPUT pin (SI) is sampled and shifted into the Shift Register. However, since the START DETECT bit must be disabled during a transmit sequence, there is no way of establishing bit synchronization on any incoming serial data. Therefore, in the Asynchronous mode, the serial port can only be used in a half-duplex configuration.

After a block of data has been sent, it is sometimes useful for the program to know when the last serial word has been shifted out of the shift register. This is especially useful when operating the MK3873 with a bidirectional half-duplex transmission line. Once the block of serial data has been completely shifted out of the port, then it is usually desirable to reverse the direction of the line so that data may be received.

One way of determining when the last word has been shifted out of the Shift Register is through the use of the OVFL/UNDFL status bit in the Serial Port Status Register. The sequence would take place as follows: The program loads the Transmit Buffer with the last serial data word which is to be sent out either when the "READY" bit is set or during a transmit interrupt service routine. Loading the Transmit Buffer clears the READY flag. At the next end-of-word condition, the last serial data word is transferred from the Transmit Buffer into the Shift Register, which sets the READY flag once again. At this point the program would not load any more data into the Transmit Buffer and the READY flag will remain set. When the last word is completely shifted out of the Shift Register, the serial port logic will check to see if any new data has been loaded into the Transmit Buffer register pair. When it determines that there is no new data in the Transmit Buffer, the serial port logic will set the OVFL/UNDFL bit in the serial port status register and will return the SERIAL OUTPUT pin (SO) to a marking condition (logic "1"). The SERIAL OUTPUT pin (SO) is always returned to a marking condition on transmitter underflow when the ASYNC mode is selected. Since the OVFL/UNDFL bit is set when the last serial data word has completely been sent out, it can be used as a signal to indicate the end of transmission and that the direction of the transmission line may be set for receive.

SYNCHRONOUS TRANSMIT OPERATION

The Synchronous Transmit mode of operation is selected by programming bit 2 (XMIT/$\overline{\text{REC}}$) of the Serial Port Control register to a "1" and setting the SYNC/$\overline{\text{ASYNC}}$ bit to a "1"

Figure 9 illustrates serial output timing relationships in the Synchronous mode. Data is shifted to the right on each rising edge of the internal SHIFT clock. Output data is not enabled to the SERIAL OUTPUT pin (SO) until the falling edge of the SHIFT clock. In a 16 bit data word, SR0, the least significant bit of the Shift Register is shifted out first, and SR15, the most signifiรant bit of the Shift Register, is shifted out last. While the Shi.. Register contents are being output on a bit by bit basis, data is simultaneously shifted in to the Shift Register through the SI pin.

As discussed in the "ASYNCHRONOUS TRANSMIT

OPERATION" section, a word time delay is generated between the time that data is written to the Transmit Buffer and the time that the contents of the Transmit Buffer are loaded into the Shift Register once the XMIT/$\overline{\text{REC}}$ bit has been programmed to a one (1).

Another way of loading the initial data word into the Transmit Buffer requires the word synchronization having been achieved through recognition of a received sync character. Recall that in the Transmit mode, data is sampled at SI and shifted into the Shift Register at the same time that data is shifted out through SO. Upon power up or reset, a control word may be written to Port D which specifies Transmit and Synchronous modes. Word synchronization can then be achieved through the use of the SEARCH bit as described in the section which covers Synchronous Receive mode. Once word synchronization is achieved, the SEARCH bit is disabled and the serial port shifts in data and generates an end-of-word condition at the word rate.

Each time the end of word condition is reached, receive data is transferred from the shift register into the Receive Buffer. At the same time, data is transferred from the Transmit Buffer into the Shift Register.

Therefore, in the Synchronous Transmit mode, the serial port may be used in a full duplex mode if word synchronization is established. At each end of word condition, output data is transferred to the Shift Register from the Transmit Buffer. At the same time, an incoming data word is transferred from the Shift register to the Receive Buffer register pair. In this case, the End-of-Word transmit routine would be used for sending data by loading the Transmit Buffer register, and for receiving data by reading the Receive Buffer register. Note that once word synchronization is established, an amount of time which is equal to one word time is available following the end-of-word interrupt for loading data into the Transmit Buffer.

The serial port operates differently in the Transmit mode for Synchronous operation than it does for Asynchronous operation. In the Asynchronous mode, after a word has been shifted out, the SO line is returned to a marking condition if no new data has been loaded into the Transmit Buffer.

In the Synchronous mode, after a word has been shifted out, the contents of the Transmit Buffer are loaded into the Shift Register regardless of whether or not new data was loaded into the Transmit Buffer. If new data was not loaded since the last time the transmit buffer was read, the OVFL/UNDFL flag is set which signals a transmitter underflow condition. This feature of always reloading the Shift Register with the contents of the Transmit Buffer when an end-of-word condition occurs allows a sync word to be continuously generated without CPU intervention when the transmitter is idle. This feature also allows variable duty cycle, variable frequency waveforms to be generated on the Serial Output line.

MK3873 CLOCKS

The time base network used with the MK3873 may be one of the four different types listed below.

Crystal
LC
RC
External Clock

The type of network which is to be used with the MK3873 is to be specified at the time when mask ROM MK3873 devices are ordered. The time base specification for each of the four modes are covered in the 3870 Family Technical Manual.

MK38P73 GENERAL DESCRIPTION

The MK38P73 is the EPROM version of the MK3873. It retains an identical pinout with the MK3873. The MK38P73 is housed in the "R" package which incorporates a 28 pin socket located directly on top of the package.

The MK38P73 can act as an emulator for the purpose of verification of user code prior to the ordering of mask ROM MK3870 devices. Thus, the MK38P73 eliminates the need for emulator board products. In addition, several MK38P73s can be used in prototype systems in order to test design concepts in field service before committing to high-volume production with mask ROM MK3873s. The compact size of the MK38P73/EPROM combination allows the packaging of such prototype systems to be the same as that used in production.

Finally, in low-volume applications the MK38P73 can be used as the actual production device.

Most of the material which has been presented for the MK3873 applies to the MK38P73. The MK38P73 has the same architecture and pinout as the MK3873. Additional information is presented in the following sections.

MK38P73 MAIN MEMORY

As can be seen from the block diagram in Figure 10, the MK38P73 contains no on-chip ROM. The memory address and data lines are brought out to the 28 pin socket located directly on top of the 40 pin package. The MK38P73 will address up to 4096 bytes of external EPROM memory.

There is one memory version of the MK38P73, and it is designated as the MK38P73/02. The MK38P73/02 contains 64 bytes of on-chip executable RAM. The MK38P73/02 can emulate the mask ROM MK3873/22 device.

Addressing of main memory on the MK38P73 is accomplished in the same way as it is for the MK3873. See Figure 12 for Main Memory addresses and for address register size in the MK38P73.

MK38P73 BLOCK DIAGRAM

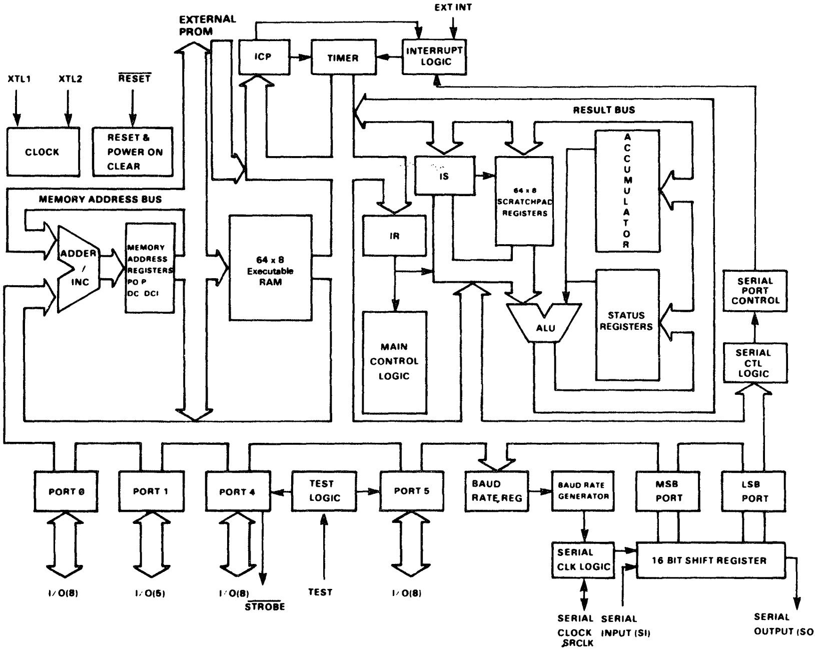


Figure 10

MK38P73 EPROM SOCKET

A 28 pin EPROM socket is located on top of the MK38P73 "R" package. The socket and compatible EPROM memories is shown in Figure 11. When 24 pin memories are used in the 28 pin socket, they should be inserted so that pin 1 of the memory device is plugged into pin 3 of the socket. (The memory should be lower justified in the 28 pin socket.)

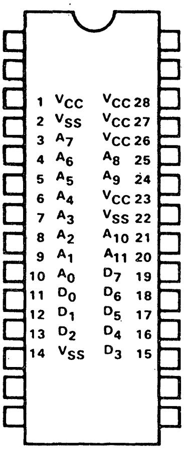
The 28-pin socket has been provided to allow use of both 24-pin and 28-pin memory devices. Minor pin-out differences in the memory devices must be accommodated by providing different versions of the MK38P73.

Initially, the MK38P73 that is compatible with the MK2716 is available. The MK38P73 designed to accommodate the 28-pin memory devices will be available at a later date.

MK38P73 I/O PORTS

The MK38P73 is offered with open drain type output buffers on Ports 4 and 5. This open drain version is provided so that user-selected open drain port pins on the mask ROM MK38P73 can be emulated prior to ordering those mask ROM parts. Figure 11 lists the part ordering number for an MK38P73/02.

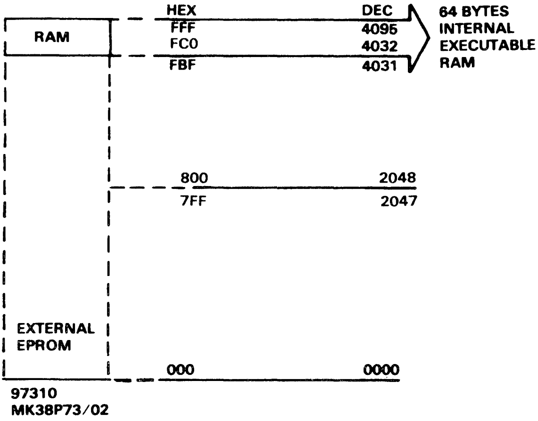
MK38P73 "R" PACKAGE PINOUT



MK97310 (Open Drain)
Compatible Memories
2758
MK2716
2516 2532

Figure 11

MK38P73 MAIN MEMORY MAP



Device	Scratchpad RAM Size (Decimal)	Address Register Size PØ, P, DC, DC1)	ROM Size (Decimal)	Executable RAM Size
MK38P73/02 97300, 97310	64 bytes	12 bits	Ø bytes	64 bytes

Figure 12

MEMORY ACCESS TIMING

A timing diagram depicting the memory access timing of the MK38P73 is shown in Figure 13. The Φ clock signal is derived internally in the MK38P73 by dividing the time base frequency by two and is used to establish all timing frequencies. The WRITE signal is another internal signal to the MK38P73 which corresponds to a machine cycle during which time a memory access may be performed. Each machine cycle is either 4 Φ clock periods or 6 Φ clock periods long. These machine cycles are termed short cycles and long cycles respectively. The worst case memory cycle is the short cycle, during which time an op code fetch is performed. This is edge of the WRITE pulse. The total access time available for the MK38P73 is shown as t_{aas}, or the time when address is stable until data must be valid on the data bus lines.

An equation for calculating available memory access time along with some calculated access times based on the listed time base frequencies is also shown in Figure 13.

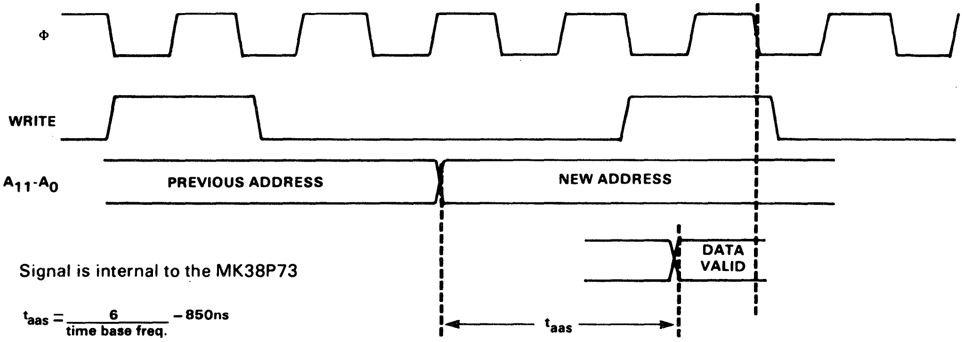
MK38P73 CLOCKS

The MK38P73 has the ability to operate with any one of the following time base configurations.

Crystal
LC
RC
External Clock

This capability has been provided in the MK38P73 so that it can emulate a mask ROM MK3873 operating in any of the possible clock configurations.

MEMORY ACCESS SHORT CYCLE OP CODE FETCH MK38P73



Signal is internal to the MK38P73

$t_{aas} = \frac{6}{\text{time base freq.}} - 850\text{ns}$

(FROM ADDRESS STABLE)

	4MHz	3.58MHz	3MHz	2.5MHz	2MHz
ACCESS TIME	650ns	825ns	1.15μs	1.55μs	2.15μs

Figure 13

ELECTRICAL SPECIFICATIONS
MK3873/MK38P73

OPERATING VOLTAGES AND TEMPERATURES

Dash Number Suffix	Operating Voltage V_{CC}	Operating Temperature T_A
—00	+5V ± 10%	0° - 70°C
—05	+5V ± 5%	0°C - 70°C
—10	+5V ± 10%	-40°C - +85°C
—15	+5V ± 5%	-40°C - +85°C

MAXIMUM RATINGS*

	-00,-05	-10,-15
Temperature Under Bias	-20°C to +85°C	-50°C to +100°C
Storage Temperature	-65°C to +150°C	-65°C to +150°C
Voltage on any Pin With Respect to Ground (Except open drain pins and TEST)	-1.0V to +7V	-1.0V to +7V
Voltage on TEST with Respect to Ground	-1.0V to +9V	-1.0V to +9V
Voltage on Open Drain Pins With Respect to Ground	-1.0V to +13.5V	-1.0V to + 13.5V
Power Dissipation	1.5W	1.5W
Power Dissipation by any one I/O pin[2]	60mW	60mW
Power Dissipation by all I/O pins[2]	600mW	600mW

*Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other condition above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

AC CHARACTERISTICS

T_A, V_{CC} within specified operating range.
I/O Power Dissipation ≤ 100mW (Note 2)

SIGNAL	SYM	PARAMETER	-00,-05		-10,-15		UNIT	NOTES
			MIN	MAX	MIN	MAX		
XTL1 XTL2	t_0	Time Base Period, all clock modes	250	500	250	500	ns	4MHz-2MHz
	$t_{ex(H)}$	External clock pulse width high	90	400	100	390	ns	
	$t_{ex(L)}$	External clock pulse width low	100	400	110	390	ns	
Φ	t_Φ	Internal Φ clock	$2t_0$		$2t_0$			
WRITE	t_W	Internal WRITE Clock period	4tΦ 6tΦ		4tΦ 6tΦ			Short Cycle Long Cycle
I/O	$t_{dI/O}$	Output delay from internal WRITE clock	0	1000	0	1200	ns	50pF plus one TTL load
	$t_{sI/O}$	Input setup time to internal WRITE clock	1000		1200		ns	
$\overline{\text{STROBE}}$	$t_{I/O-s}$	Output valid to $\overline{\text{STROBE}}$ delay	3tΦ -1000	3tΦ +250	3tΦ -1200	3tΦ +300	ns	I/O load = 50pF + 1 TTL load
	t_{sL}	$\overline{\text{STROBE}}$ low time	8tΦ -250	12tΦ +250	8tΦ -300	12tΦ +300	ns	$\overline{\text{STROBE}}$ load = 50pF+3TTL loads
$\overline{\text{RESET}}$	t_{RH}	$\overline{\text{RESET}}$ hold time, low	6tΦ +750		6tΦ +1000		ns	
	t_{RPOC}	$\overline{\text{RESET}}$ hold time, low for power clear	power supply rise time +0.1		power supply rise time 0.15		ms	
EXT INT	t_{EH}	EXT INT hold time in active and inactive state	6tΦ +750		6tΦ +1000		ns	To trigger interrupt
			2tΦ		2tΦ		ns	To trigger timer

CAPACITANCE

T_A = 25°C
All Part Numbers

SYM	PARAMETER	MIN	MAX	UNIT	NOTES
C_{IN}	Input capacitance; I/O, $\overline{\text{RESET}}$, EXT INT, TEST		10	pF	unmeasured pins grounded
C_{XTL}	Input capacitance; XTL1, XTL2	23.5	29.5	pF	

AC CHARACTERISTICS FOR SERIAL I/O PINS

T_A, V_{CC} within specified operating range.
I/O Power Dissipation ≤ 100mW (Note 2)

SIGNAL	SYM	PARAMETER	-00, -05		-10, -15		UNIT	CONDITIONS
			MIN	MAX	MIN	MAX		
SRCLK	$t_{C(SRCLK)}$	Serial Clock Period in External Clock Mode	3.3	∞	∞		µs	
	$t_{W(SRCLKH)}$	Serial Clock Pulse Width, High. External Clock Mode	1.3	∞		∞	µs	
	$t_{W(SRCLKL)}$	Serial Clock Pulse Width, Low. External Clock Mode	1.3	∞		∞	µs	
	$t_{r(SRCLK)}$	Serial Clock Rise Time Internal Clock Mode	60		60		ns	0.8V -2.0V C_L = 100pf
	$t_{f(SRCLK)}$	Serial Clock Fall Time Internal Clock Mode	30		30		ns	2.4V -0.4V C_L = 100pf
SI	$t_{S(SI)}$	Setup Time To Rising Edge of SRCLK (SYNC Mode)	0		0		ns	
	$t_{H(SI)}$	Hold Time From Rising Edge of SRCLK (SYNC Mode)	1500		1500		ns	
SO	$t_{D(SO)}$	Data Output Delay From Falling Edge of SRCLK (SYNC Mode)	1190		1190		ns	

AC CHARACTERISTICS FOR MK38P73

(Signals brought out at socket)
T_A, V_{CC} within specified operating range.
I/O Power Dissipation ≤ 100mW (Note 2)

SYMBOL	PARAMETER	-00, -05		-10, -15		UNIT	CONDITION
		MIN	MAX	MIN	MAX		
t_{aas}*	Access time from Address A_{11}-A_{0}1 stable until data must be valid at D_7-D_0	650				ns	Φ = 2.0MHz

*See Table in Figure 13.

DC CHARACTERISTICS

T_A, V_{CC} within specified operating range
I/O power dissipation ≤ 100mW

SYMBOL	PARAMETER	MIN	MAX	MIN	MAX	UNIT	DEVICE
I_{CC}	Average Power Supply Current		103		138	mA	MK3873/12 Outputs Open
			138		165	mA	MK38P73/02 No EPROM, Outputs Open
P_D	Power Dissipation		485		645	mW	MK3873/22 Outputs Open
			646		775	mW	MK38P73/02 No EPROM, Outputs Open

DC CHARACTERISTICS

T_A, V_{CC} within specified operating range
I/O Power Dissipation ≤ 100mW (Note 2)

		-00,-05		-10,-15			
SYM	PARAMETER	MIN	MAX	MIN	MAX	UNIT	CONDITIONS
V_{IHEX}	External Clock input high level	2.4	5.8	2.4	5.8	V	
V_{ILEX}	External Clock input low level	-.3	.6	-.3	.6	V	
I_{IHEX}	External Clock input high current		100		130	μA	$V_{IHEX}=V_{CC}$
I_{ILEX}	External Clock input low current		-100		-130	μA	$V_{ILEX}=V_{SS}$
$V_{IHI/O}$	I/O input high level	2.0	5.8	2.0	5.8	V	standard pull-up (1)
		2.0	13.2	2.0	13.2	V	open drain (1)
V_{IHR}	Input high level, $\overline{\text{RESET}}$	2.0	5.8	2.2	5.8	V	standard pull-up (1)
		2.0	13.2	2.2	13.2	V	No pull-up
V_{IHEI}	Input high level, EXT INT	2.0	5.8	2.2	5.8	V	standard pull-up (1)
		2.0	13.2	2.2	13.2	V	No pull-up
V_{IL}	I/O ports, $\overline{\text{RESET}}$[1], EXT INT[1] input low level	-.3	.8	-.3	0.7	V	(1)
I_{IL}	Input low current, standard pull-up pins		-1.6		-1.9	mA	$V_{IN}=0.4V$
I_L	Input leakage current, open drain pins $\overline{\text{RESET}}$ and EXT INT inputs With no pull-up resistor		+10 -5		+18 -8	μA μA	$V_{IN}=13.2V$ $V_{IN}=0.0V$
I_{OH}	Output high current, standard	-100		-89		μA	$V_{OH}=2.4V$
	pull-up pins	-30		-25		μA	$V_{OH}=3.9V$
I_{OHDD}	Output high current,	-100		-80		μA	$V_{OH}=2.4V$
	direct drive pins	-1.5	-8.5	-1.3	-11	mA mA	$V_{OH}=1.5V$ $V_{OH}=0.7V$
I_{OL}	Output low current, I/O ports	1.8		1.65		mA	$V_{OL}=0.4V$
I_{OHS}	$\overline{\text{STROBE}}$ Output High current	-300		-270		μA	$V_{OL}=2.4V$
I_{OLS}	$\overline{\text{STROBE}}$ output low current	5.0		4.5		mA	$V_{OL}=0.4V$

DC CHARACTERISTICS FOR MK38P73

(Signals brought out at socket)

T_A, V_{CC} within specifiec operating range, I/O Power Dissipation ≤ 100mW. (Note 2)

		-00, -05		-10, -15			
SYM	PARAMETER	MIN	MAX	MIN	MAX	UNIT	CONDITION
I_{CCE}	Power Supply Current for EPROM		−185		−185	mA	
V_{IL}	Input Low Level Data bus in	-0.3	0.8	-0.3	0.7	V	
V_{IH}	Input High Level Data bus in	2.2	5.8	2.2	5.8	V	
I_{OH}	Output High Current	-100 -30		-90 -25		μA μA	V_{OH}=2.4V V_{OH}=3.9V
I_{OL}	Output Low Current	1.8		1.65		mA	V_{OL}=0.4V
I_{IL}	Input Leakage Current		10		10	μA	Data Bus in Float

DC CHARACTERISTICS FOR SERIAL PORT I/O PINS

T_A, V_{CC} within specified operating range

I/O Power Dissipation ≤ 100mW (Note 2)

		-00, -05		-10, -15			
SYM	PARAMETER	MIN	MAX	MIN	MAX	UNIT	TEST CONDITIONS
V_{IHS}	Input High for SI, SRCLK	2.0	5.8	2.0	5.8	V	
V_{ILS}	Input Low level for SI, SRCLK	-.3	.8	-.3	0.7	V	
I_{ILS}	Input low current for SI, SRCLK		-1.6		-1.9	mA	V_{IL} = 0.4V
I_{OHSO}	Output High Current SO	-100 -30		-90 -25		μA μA	V_{OH} = 2.4V V_{OL} = 3.9V
I_{OLSO}	Output Low Current SO	1.8		1.65		mA	V_{OL} = 0.4V
I_{OHSRC}	Output High Current SRCLK	-300		-270		μA	V_{OH} = 2.4V
I_{OLSRC}	Output Low Current	5.0		4.5		mA	V_{OL} = 0.4V

1. $\overline{\text{RESET}}$ and EXT INT have internal Schmit triggers giving minimum .2V hysteresis.
2. Power dissipation for I/O pins is calculated by $\Sigma(V_{CC} - V_{IL})(|I_{IL}|) + \Sigma(V_{CC} - V_{OH})(|I_{OH}|) + \Sigma(V_{OL})(I_{OL})$

TIMER AC CHARACTERISTICS

Definitions:

Error = Indicated time value - actual time value

tpsc = t Φ x Prescale Value

Interval Timer Mode:

Single interval error, free running (Note 3)	±6tΦ
Cumulative interval error, free running (Note 3)	0
Error between two Timer reads (Note 2)	± (tpsc + tΦ
Start Timer to stop Timer error (Notes 1,4)	+tΦ to —(tpsc +tΦ)
Start Timer to read Timer error (Notes 1,2)	—5tΦ to —(tpsc + 7tΦ)
Start Timer to interrupt request error (Notes 1,3)	—2tΦ to —8tΦ)
Load Timer to stop Timer error (Note 1)	+tΦ to —(tpsc + 2tΦ)
Load Timer to read Timer error (Notes 1,2)	—5tΦ to —(tpsc + 8tΦ)
Load Timer to interrupt request error (Notes 1,3)	—2tΦ to —9tΦ)

Pulse Width Measurement Mode:

Measurement accuracy (Note 4) .... +t Φ to —(tpsc +2t Φ)
Minimum pulse width of EXT INT pin 2tΦ

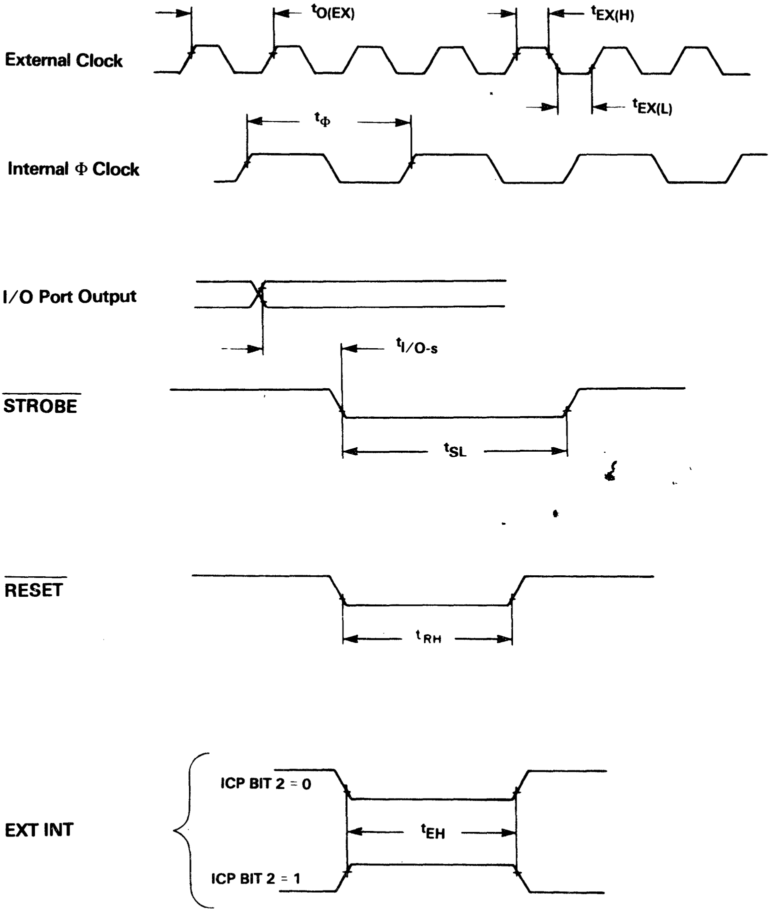
Event Counter Mode:

Minimum active time of EXT INT pin 2tΦ
Minimum inactive time of EXT INT pin 2tΦ

Notes:

1. All times which entail loading, starting, or stopping the Timer are referenced from the end of the last machine cycle of the OUT or OUTS instruction.
2. All times which entail reading the Timer are referenced from the end of the last machine cycle of the IN or INS instruction.
3. All times which entail the generation of an interrupt request are referenced from the start of the machine cycle in which the appropriate interrupt request latch is set. Additional time may elapse if the interrupt request occurs during a privileged or multicycle instruction.
4. Error may be cumulative if operation is repetitively performed.

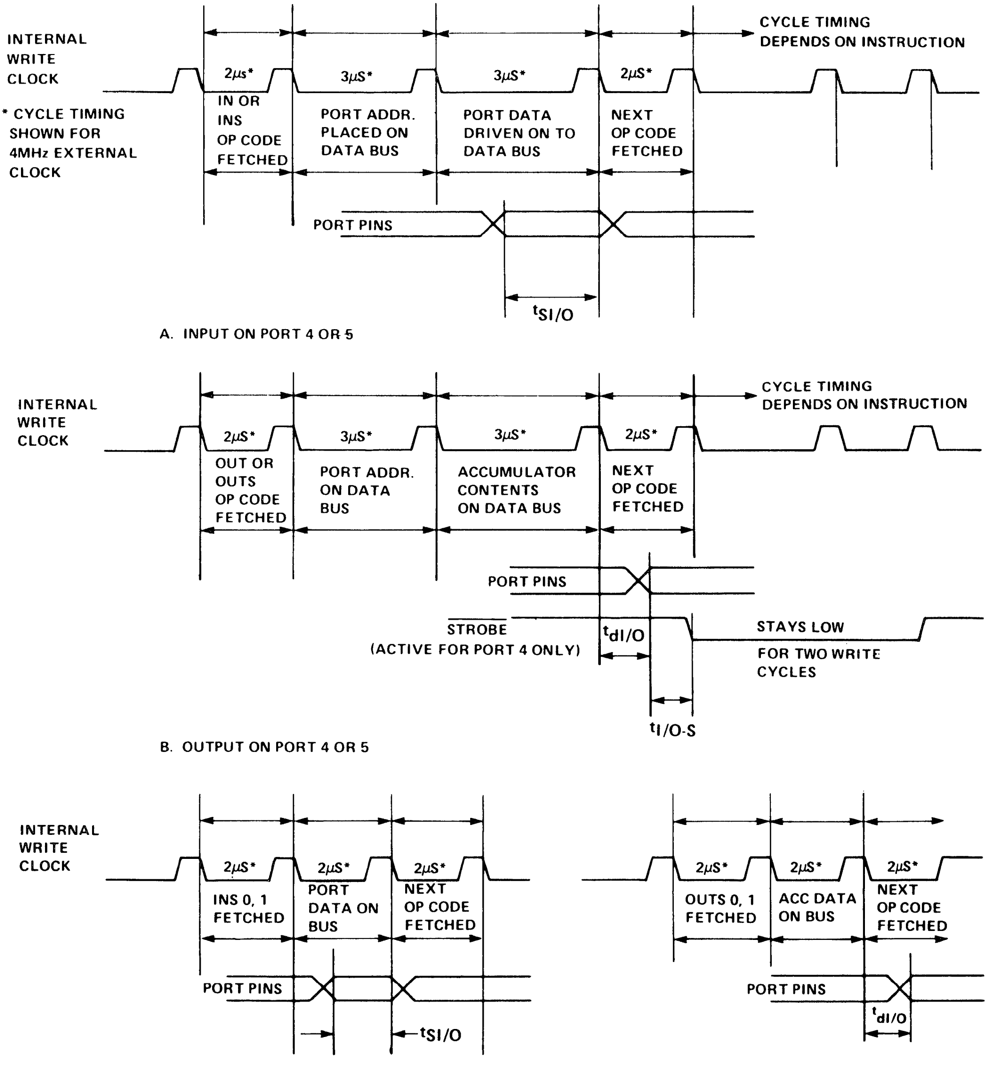
AC TIMING DIAGRAM



Note: All measurements are referenced to V_{IL} max., V_{IH} min., V_{OL} max., or V_{OH} min.

Figure 14

INPUT/OUTPUT AC TIMING



A. INPUT ON PORT 4 OR 5

B. OUTPUT ON PORT 4 OR 5

C. INPUT ON PORT 0 OR 1

D. OUTPUT ON PORT 0, 1

Figure 15

AC TIMING DIAGRAM FOR SERIAL I/O PINS.

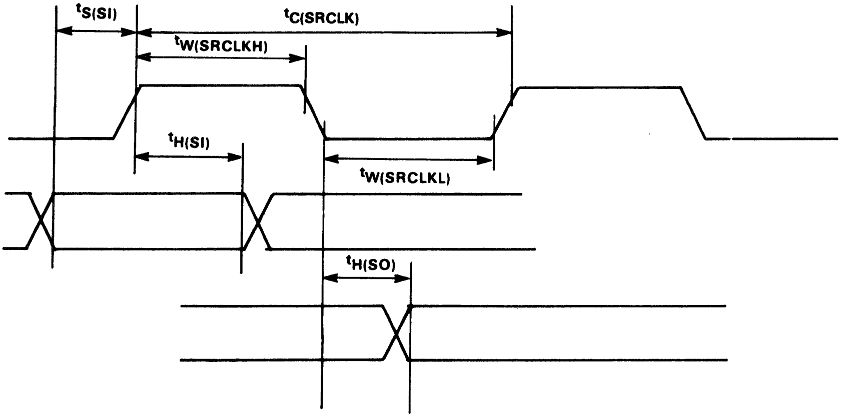


Figure 16

STROBE SOURCE CAPABILITY (TYPICAL AT V_{CC} = 5V, T_A = 25°C)

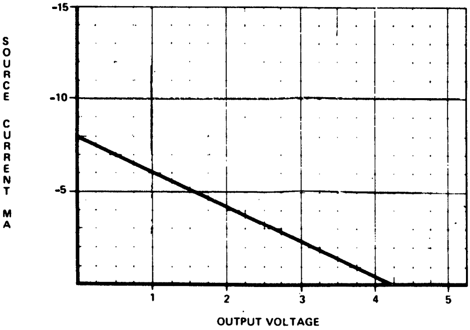


Figure 17

STROBE SINK CAPABILITY (TYPICAL AT V_{CC} = 5V, T_A = 25°C)

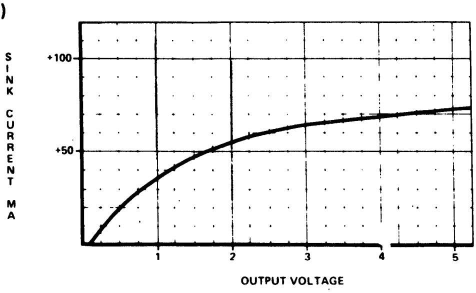
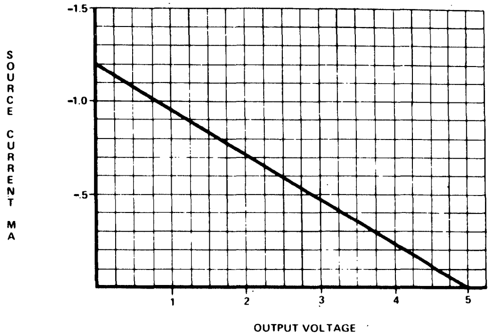


Figure 18

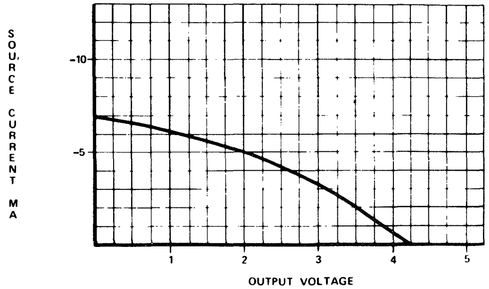
STANDARD I/O PORT SOURCE CAPABILITY (TYPICAL AT V_{CC} = 5V, T_A = 25°C)

Figure 19



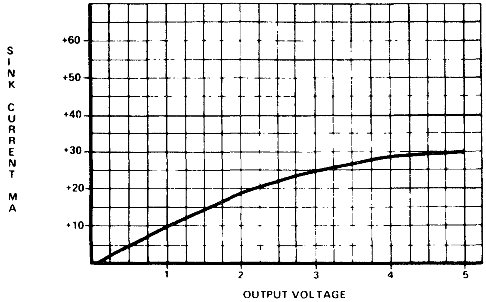
DIRECT DRIVE I/O PORT SOURCE CAPABILITY (TYPICAL AT V_{CC} = 5V, T_A = 25°C)

Figure 20



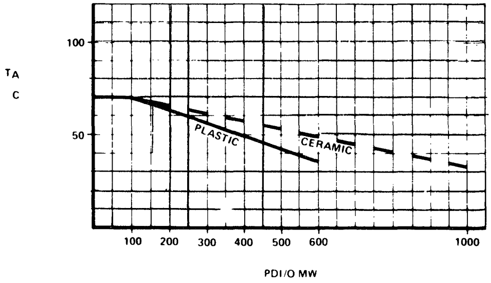
I/O PORT SINK CAPABILITY (TYPICAL AT V_{CC} = 5V, T_A = 25°C)

Figure 21



MAXIMUM OPERATING TEMPERATURE VS. I/O POWER DISSIPATION

Figure 22

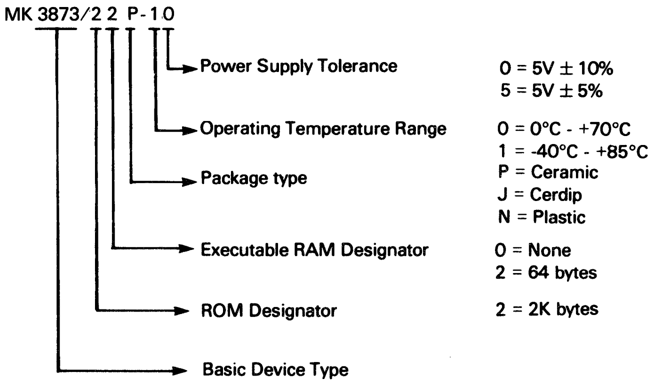


ORDERING INFORMATION

There are two types of part numbers for the 3870 family of devices. The generic part number describes the basic device type, the amount of ROM and Executable RAM, the desired package type, temperature range, and power supply tolerance. For each customer specific code, additional information defining I/O options and oscillator options will be combined with the information described in the generic part number to define a customer/code specific device order number.

GENERIC PART NUMBER

An example of the generic part number is shown below.



MK 3873/2 2 P-1 0

Power Supply Tolerance — 0 = 5V ± 10%; 5 = 5V ± 5%

Operating Temperature Range — 0 = 0°C - +70°C; 1 = -40°C - +85°C

Package type — P = Ceramic; J = Cerdip; N = Plastic

Executable RAM Designator — 0 = None; 2 = 64 bytes

ROM Designator — 2 = 2K bytes

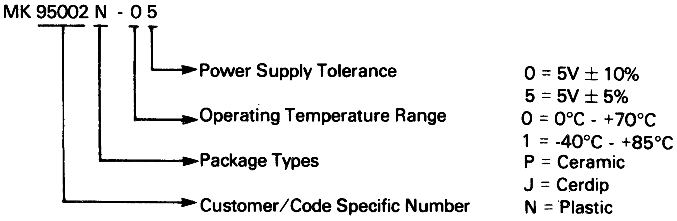
Basic Device Type

An example of the generic part number for the PPROM device is shown below.

MK38P73/02 R-05

DEVICE ORDER NUMBER

An example of the device order number is shown below.



MK 95002 N - 0 5

Power Supply Tolerance — 0 = 5V ± 10%; 5 = 5V ± 5%

Operating Temperature Range — 0 = 0°C - +70°C; 1 = -40°C - +85°C

Package Types — P = Ceramic; J = Cerdip; N = Plastic

Customer/Code Specific Number

The Customer/Code specific number defines the ROM bit pattern, I/O configuration, oscillator type, and generic part type to be used to satisfy the requirements of a particular customer purchase order. For further information on the ordering of mask ROM devices, the customer should refer to the 3870 Family Technical Manual.

These specifications are subjet to change without notice.
Please inquire with our sales offices about the availability of the different products.

HOMSON
COMPONENTS
EMICONDUCTEURS

MK3875 • MK38P75

3870 SINGLE CHIP MICRO FAMILY

MK3875 FEATURES

- Available with 2K or 4K bytes of mask programmable ROM memory
- 64 bytes scratchpad RAM
- 64 bytes of Executable RAM
- Standby feature for low power data retention of executable RAM including:
  Low standby power
  Low standby supply voltage
  No external components required to trickle charge battery.
- Software compatible with 3870 family
- 30 bits (4 ports) TTL compatible I/O
- Programmable binary Timer
  Interval Timer Mode
  Pulse Width Measurement Mode
  Event Counter Mode
- External Interrupt Input
- Crystal, LC, RC, or external time base options available
- Low power under normal operation (285 mW typ.)
- +5 volt main power supply
- Pinout compatible with 3870 family

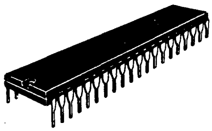
MK38P75 FEATURES

- EPROM version of MK3875
- Piggyback RPOM (P-PROM)™ package
- Accepts 24 pin or 28 pin EPROM memories
- Identical pinout as MK3875
- In-socket emulation of MK3875

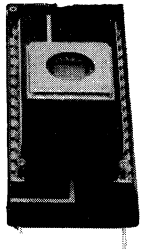
NMOS

CASES

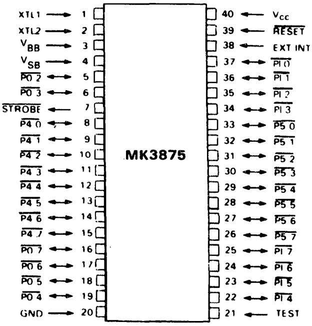
MK3875



MK38P75



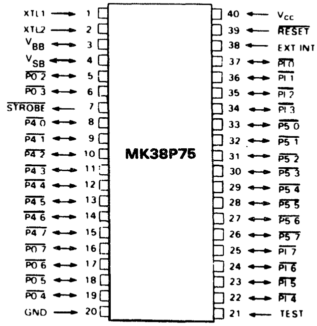
MK3875


XTL1 1
XTL2 2
V_{BB} 3
V_{SB} 4
$\overline{\text{P0 2}}$ 5
$\overline{\text{P0 3}}$ 6
$\overline{\text{STROBE}}$ 7
$\overline{\text{P4 0}}$ 8
$\overline{\text{P4 1}}$ 9
$\overline{\text{P4 2}}$ 10
$\overline{\text{P4 3}}$ 11
$\overline{\text{P4 4}}$ 12
$\overline{\text{P4 5}}$ 13
$\overline{\text{P4 6}}$ 14
$\overline{\text{P4 7}}$ 15
$\overline{\text{P0 7}}$ 16
$\overline{\text{P0 6}}$ 17
$\overline{\text{P0 5}}$ 18
$\overline{\text{P0 4}}$ 19
GND 20
MK3875
40 V_{cc}
39 $\overline{\text{RESET}}$
38 EXT INT
37 $\overline{\text{PI 0}}$
36 $\overline{\text{PI 1}}$
35 $\overline{\text{PI 2}}$
34 $\overline{\text{PI 3}}$
33 $\overline{\text{P5 0}}$
32 $\overline{\text{P5 1}}$
31 $\overline{\text{P5 2}}$
30 $\overline{\text{P5 3}}$
29 $\overline{\text{P5 4}}$
28 $\overline{\text{P5 5}}$
27 $\overline{\text{P5 6}}$
26 $\overline{\text{P5 7}}$
25 $\overline{\text{PI 7}}$
24 $\overline{\text{PI 6}}$
23 $\overline{\text{PI 5}}$
22 $\overline{\text{PI 4}}$
21 TEST

PIN ASSIGNMENTS

PIN NAME	DESCRIPTION	TYPE
$\overline{\text{P0-2}}$- $\overline{\text{P0-7}}$	I/O Port 0	Bidirectional
$\overline{\text{P1-0}}$ - $\overline{\text{P1-7}}$	I/O Port 1	Bidirectional
$\overline{\text{P4-0}}$ - $\overline{\text{P4-7}}$	I/O Port 4	Bidirectional
$\overline{\text{P5-0}}$ - $\overline{\text{P5-7}}$	I/O Port 5	Bidirectional
$\overline{\text{STROBE}}$	Ready Strobe	Output
EXT INT	External Interrupt	Input
$\overline{\text{RESET}}$	External Reset	Input
TEST	Test Line	Input
XTL 1, XTL 2	Time Base	Input
V_{CC}, GND	Power Supply Lines	Input
V_{SB}	Standby Power	Input
V_{BB}	Substrate Decoupling	Input

MK38P75


XTL1 1
XTL2 2
V_{BB} 3
V_{SB} 4
$\overline{\text{P0 2}}$ 5
$\overline{\text{P0 3}}$ 6
$\overline{\text{STROBE}}$ 7
$\overline{\text{P4 0}}$ 8
$\overline{\text{P4 1}}$ 9
$\overline{\text{P4 2}}$ 10
$\overline{\text{P4 3}}$ 11
$\overline{\text{P4 4}}$ 12
$\overline{\text{P4 5}}$ 13
$\overline{\text{P4 6}}$ 14
$\overline{\text{P4 7}}$ 15
$\overline{\text{P0 7}}$ 16
$\overline{\text{P0 6}}$ 17
$\overline{\text{P0 5}}$ 18
$\overline{\text{P0 4}}$ 19
GND 20
MK38P75
40 V_{cc}
39 $\overline{\text{RESET}}$
38 EXT INT
37 $\overline{\text{PI 0}}$
36 $\overline{\text{PI 1}}$
35 $\overline{\text{PI 2}}$
34 $\overline{\text{PI 3}}$
33 $\overline{\text{P5 0}}$
32 $\overline{\text{P5 1}}$
31 $\overline{\text{P5 2}}$
30 $\overline{\text{P5 3}}$
29 $\overline{\text{P5 4}}$
28 $\overline{\text{P5 5}}$
27 $\overline{\text{P5 6}}$
26 $\overline{\text{P5 7}}$
25 $\overline{\text{PI 7}}$
24 $\overline{\text{PI 6}}$
23 $\overline{\text{PI 5}}$
22 $\overline{\text{PI 4}}$
21 TEST

MK3875 BLOCK DIAGRAM

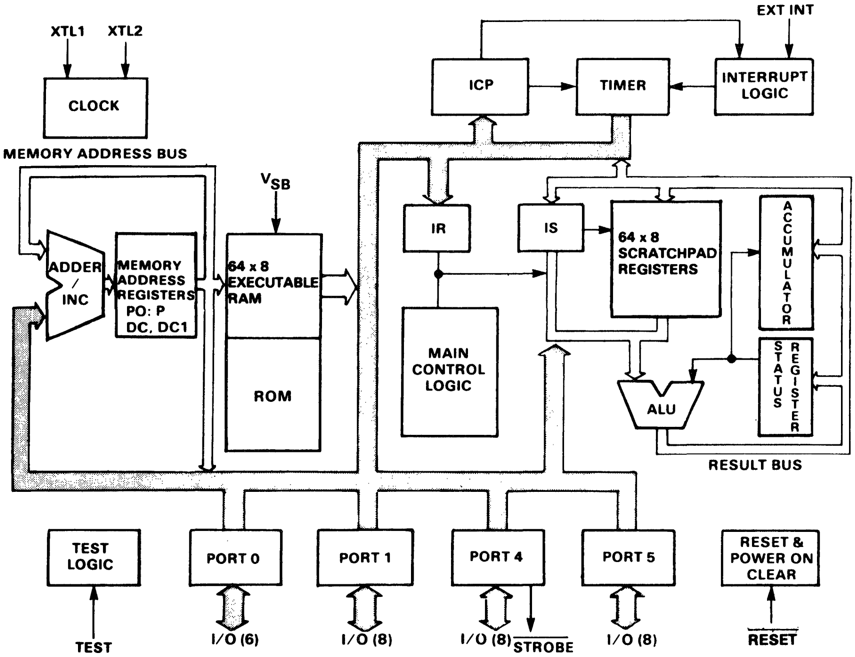


Figure 1

GENERAL DESCRIPTION

The MK3875 Single Chip Microcomputer offers a Low Power Standby mode of operation as an addition to the 3870 Family. The Low Power Standby feature provides a means of retaining data in the executable RAM on the MK3875 while the main power supply line (V_{CC}) is at 0 volts and the rest of the MK3875 microcomputer is shut down. The executable RAM is powered from an auxiliary power supply input (V_{SB}) while operating in the Lower Power Standby mode. When V_{SB} is maintained at or above its minimum level, data is retained in the executable RAM memory with a very low power dissipation.

The MK3875 retains commonality with the rest of the industry standard 3870 family of single chip microcomputers. It has the same central processing unit, oscillator and clock circuits, and 64 byte scratchpad memory array. Also, the 3870's sophisticated programmable binary timer is included which provides three different operating modes. Two pins on the MK3875 are dedicated to the Low Power Standby mode and are designated as V_{SB} and V_{BB}. The $\overline{\text{RESET}}$ line serves to reset the MK3875 and place it in a protected state so that the contents of the Executable RAM will remain unchanged when V_{CC} is being powered down to 0 volts. All other pins on the MK3875 are identical in function to corresponding pins on the MK3870, so that pin compatibility is maintained. The MK3875 executes the entire 3870 instruction set.

The MK38P75 microcomputer is the PROM based version of the MK3875. It is called the piggyback PROM (P-PROM)™ because of its packaging concept. This concept allows a standard 24-pin or 28-pin EPROM to be mounted directly on top of the microcomputer itself. The EPROM can be removed and reprogrammed as required with a standard PROM programmer. The MK38P75 retains the pinout and architectural features as other members of the 3870 family. The MK38P75 is discussed in more detail in a later section.

FUNCTIONAL PIN DESCRIPTION

$\overline{\text{P0-2}}$ - $\overline{\text{P0-7}}$, $\overline{\text{P1-0}}$ - $\overline{\text{P1-7}}$, $\overline{\text{P4-0}}$ - $\overline{\text{P4-7}}$, and $\overline{\text{P5-0}}$ - $\overline{\text{P5-7}}$ are 30 lines which can be individually used as either TTL compatible inputs or as latched outputs.

$\overline{\text{STROBE}}$ is a ready strobe associated with I/O Port 4. This pin, which is normally high, provides a single low pulse after valid data are present on the $\overline{\text{P4-0}}$ - $\overline{\text{P4-7}}$ pins during an output instruction.

$\overline{\text{RESET}}$ - may be used to externally reset the MK3875. When pulled low, the Mk3875 will reset. When allowed to go high the MK3875 will begin program execution at program location H '000'. Additionally, when $\overline{\text{RESET}}$ is brought low all accesses of the executable RAM are prevented and the RAM is placed in a protected state for powering down V_{CC} without loss of data.

EXT INT is the external interrupt input. Its active state is software programmable. This input is also used in conjunction with the timer for pulse width measurement and event counting.

XTL 1 and XTL 2 are the time base inputs to which a crystal (2 to 4 MHz), LC network, RC network, or an external single-phase clock may be connected. The time base network must be specified when ordering an MK3875.

TEST is an input used only in testing the MK3875. For normal circuit function this pin may be left unconnected but it is recommended that TEST be grounded.

V_{CC} is the power supply input +5 V.

V_{SB} is the RAM standby power supply input.

V_{BB} is the substrate decoupling pin. A .01 micro-Farad capacitor is required which is tied between V_{BB} and GND.

MK3875 ARCHITECTURE

The basic functional elements of the mask ROM MK3875 single chip microcomputer are shown in the block diagram in Figure 1. A programming model is shown in Figure 2 Much of the Mk3875 architecture is identical with the rest of the devices in the 3870 family. The significant features of the MK3875 are discussed in the following sections. The user is referred to the 3870 Family Technical Manual for a thorough discussion of the architecture, instruction set, and other features which are common to the 3870 family.

MAIN MEMORY

The main memory section on the MK3875 consists of a combination of ROM and executable RAM.

There are four registers associated with the main memory section. These are the Program Counter (PO), the Stack Register (P), the Data Counter (DC) and the Auxiliary Data Counter (DC1). The Program Counter is used to address instructions during program execution. P is used to save the contents of PO during an interrupt or subroutine call. Thus, P contains the return address at which processing is to resume upon completion of the subroutine or the interrupt routine.

The Data Counter (DC) is used to address data tables. This register is auto-incrementing. Of the two data counters only DC can access the memory. However, the XDC instruction allows DC and DC1 to be exchanged.

The length of the PC P, DC, and DC1 registers for all MK3875 devices is 12 bits. Figure 3 shows the amounts of ROM and Executable RAM for each device in the MK3875 family.

3875 PROGRAMMABLE REGISTERS, PORTS AND MEMORY MAP

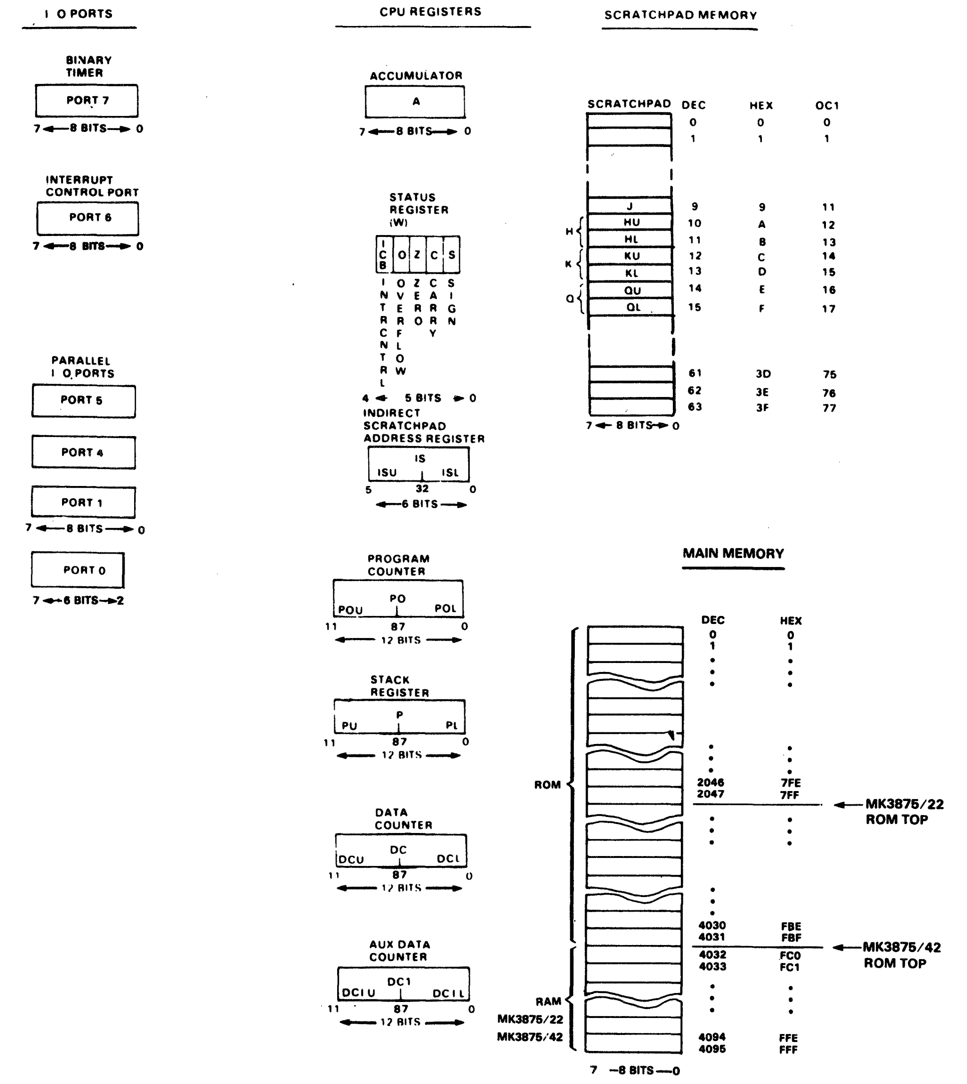
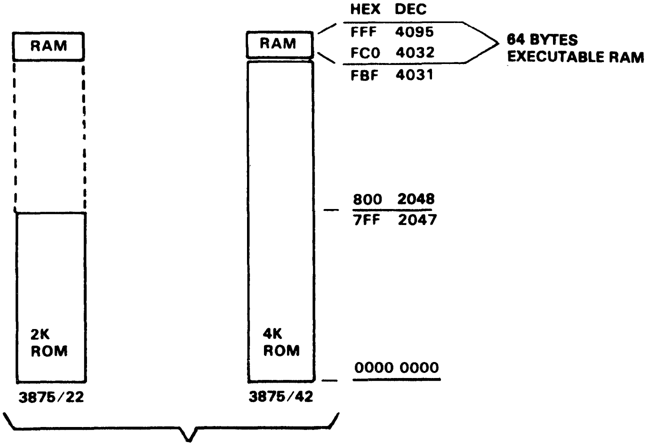


Figure 2

EXECUTABLE RAM

The upper bytes of the total address space in all MK3875 devices are RAM memory. As with the ROM memory, the RAM may be addressed by the PO and DC address registers. The executable RAM may be accessed by all 3870 instructions which address main memory indirectly through the Data Counter (DC) register. Additionally, the MK3875 may execute an instruction sequence which resides in the executable RAM. Note that this sequence cannot be done with the scratchpad RAM memory, which is the reason the term "executable RAM" is given to this additional memory. The contents of the executable RAM memory are preserved when the Low Power Standby mode is in operation.

MK3875 MAIN MEMORY SIZES AND TYPES BY SLASH NUMBER



All devices contain 64 bytes of scratchpad RAM

Data derived from addressing any locations other than within the specified ROM or RAM space is not tested nor is it guaranteed. Users should refrain from entering this area of the memory map.

Device	Scratchpad RAM Size (Decimal)	Address Register Size (P0,P,DC,DC1)	ROM Size (Decimal)	Executable RAM Size
MK3875/22	64 bytes	12 bits	2048 bytes	64 bytes
MK3875/42	64 bytes	12 bits	4032 bytes	64 bytes

Figure 3

I/O PORTS

The MK3875 provides 30 bits of bidirectional parallel I/O. These lines are addressed as Ports 0, 1, 4 and 5. In addition, the Interrupt Control Port is addressed as Port 6 and the binary timer is addressed as Port 7. The programming of Ports 6 and 7 and the bidirectional I/O pins are covered in the 3870 Family Technical Manual.

Since two pins are dedicated to serve the Standby Power mode (V_{SB}), port 0 has only the upper 6 bits, $\overline{P0\text{-}2}$ - $\overline{P0\text{-}7}$, available for use as general purpose I/O pins. Ports 1, 4, and 5 are all a full 8 bits wide.

The schematic of an I/O pin and available output drive options are shown in Figure 4.

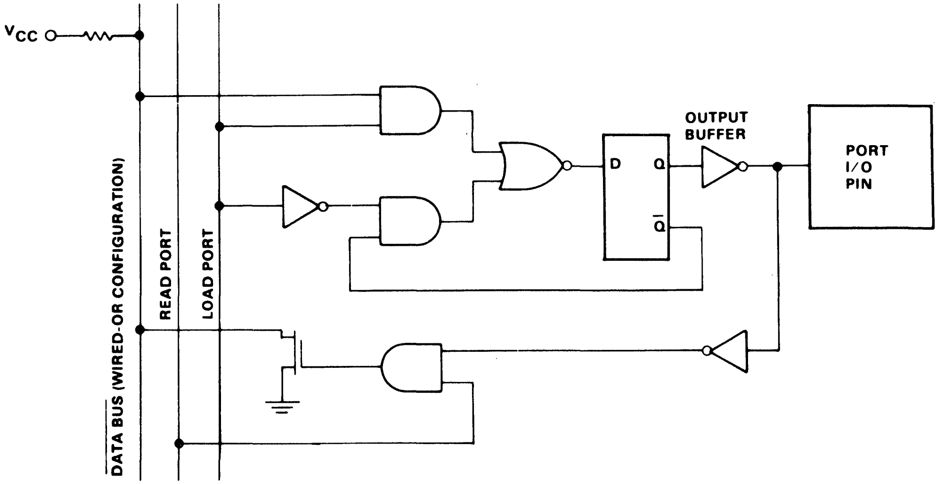
An output ready strobe is associated with Port 4. This flag may be used to signal a peripheral device that the MK3875 has just completed an output of new data to Port 4. The strobe provides a single low pulse shortly after the output operation is completely finished, so either edge may be used to signal the peripheral. $\overline{\text{STROBE}}$ may be used as an input strobe simply by doing a dummy output of H '00' to Port 4 after completing the input operation.

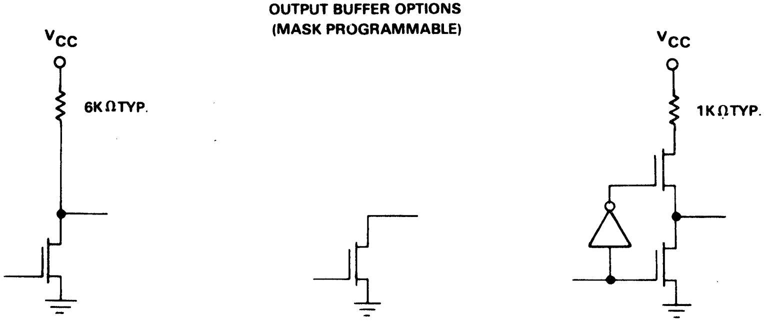
STANDBY POWER MODE

On the MK3875, the contents of the on-chip executable RAM can be saved when the Standby Power mode is operative. The Standby Power mode allows the MK3875's main power supply to drop all way down to 0 volts while the on-chip executable RAM is powered from the auxiliary low power supply input, V_{SB}. Thus, key variables may be maintained within the MK3875 executable RAM during the time that the rest of the microcomputer is powered down.

On the MK3875, two of the pins which are used as bidirectional port pins on the MK3870 are used for the Standby Power feature. Port 0, Bit 0 (P0-0), remains readable and writeable although it is not connected to a package pin. The logic level being applied to the auxiliary power supply input (V_{SB}) can be read at Port 0, Bit 1 ($\overline{P0\text{-}1}$). Writing to $\overline{P0\text{-}1}$ has no effect.

I/O PIN CONCEPTUAL DIAGRAM WITH OUTPUT BUFFER OPTIONS





STANDARD OUTPUT

OPEN DRAIN OUTPUT

DIRECT DRIVE OUTPUT

Ports 0 and 1 are Standard Output type only.

Ports 4 and 5 may both be any of the three output options (mask programmable bit by bit)

The $\overline{\text{STROBE}}$ output is always configured similar to a Direct Drive Output except that it is capable of driving 3 TTL loads.

$\overline{\text{RESET}}$ and EXT INT may have standard 6KΩ (typical) pull-up or may have no pull-up (mask programmable). These two inputs have Schmitt trigger inputs with a minimum of 0.2 volts of hysteresis.

$\overline{\text{RESET}}$ and EXT INT do not have internal pull up on the MK38P75.

Figure 4

A capacitor (.01 microfarads) must be connected between pin 3 (V_{BB}) and ground. V_{BB} is bonded directly to the substrate of the MK3875. The purpose of the capacitor is to decouple noise on the substrate of the circuit when V_{CC} is switched on and off.

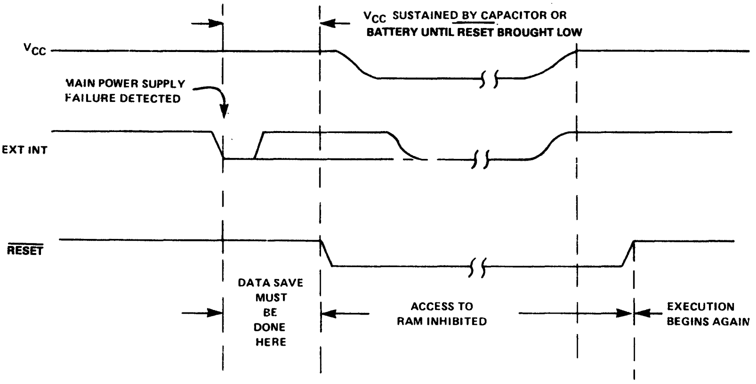
It is recommended that Nickel Cadmium batteries (typical voltage of 3 series cells = 3.6V) be used for standby power, since the MK3875 can automatically trickle charge the three NiCads. If more than three cells in series are used, the charging circuit must be provided outside the MK3875.

Whenever $\overline{\text{RESET}}$ is brought low, the executable RAM is placed in a protected state. Also the RAM is switched from V_{CC} power to the V_{SB} power. When powering down, it may be desirable to interrupt the MK3875 when an impending power down condition is detected, so that the necessary data can be saved before V_{CC} falls below the minimum level. After the save is completed, $\overline{\text{RESET}}$ can fall, which prevents any further access of the RAM. The timing for this power down sequence is illustrated in Figure 5A.

A second power down sequence is illustrated in Figure 5B, and may be used if a special save data routine is not needed. The EXT INT line need not be used. Note that for both cases shown in Figures 5A and 5B, $\overline{\text{RESET}}$ must be low before V_{CC} drops below the minimum specified operating voltage for the MK3875. This is to ensure that the contents of the executable RAM are not altered during the power down sequence.

There may be a set of variables stored in the RAM memory which is continually updated during the tme when the MK3875 is in its normal operating mode. If a particular variable occupies more than one byte of RAM, there can be a problem if a reset occurs in response to an impending power down condition during the time that the multi-byte variable was being modified. If such a reset occurs, then only part of the variable may contain the updated value, while the rest contains the old value. An example of this case would be when a double precision (2 byte) binary number is being saved in the executable RAM. Suppose that a new value of the number has been calculated in the

SAVE ROUTINE REQUIRED, $V_{SB} > 3.2$ VOLTS



V_{CC} SUSTAINED BY CAPACITOR OR BATTERY UNTIL RESET BROUGHT LOW
V_{CC}
MAIN POWER SUPPLY FAILURE DETECTED
EXT INT
$\overline{\text{RESET}}$
DATA SAVE MUST BE DONE HERE
ACCESS TO RAM INHIBITED
EXECUTION BEGINS AGAIN

Figure 5a

NO SAVE ROUTINE REQUIRED, $V_{SB} > 3.2$ VOLTS

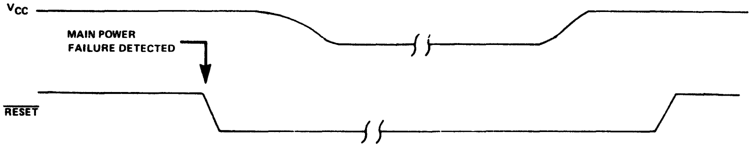


Figure 5b

program, and that this new value is to replace the old value contained in the executable RAM; note that a reset could occur just after the program wrote one byte of the new value into the RAM. When power is restored following the Standby Power mode, the double precision variable would contain an erroneous value.

This problem can be avoided if the external interrupt is used to signal the MK3875 of an impending power down condition. The user's system should be designed so that the MK3875 can properly save all variables between the time that the external interrupt occurs and $\overline{\text{RESET}}$ falls. If multi-byte variables must be saved during the Standby Power mode and it is not desirable to use the external interrupt in the manner described above, then each byte of a multi-byte variable may be kept with an associated flag. The method of updating a two byte variable would be as follows:

- Clear Flag Word 1
- Update Byte 1
- Set Flag Word 1
- Clear Flag Word 2
- Update Byte 2
- Set Flag Word 2

Now if $\overline{\text{RESET}}$ goes low during the update of a byte of a variable, the flag word associated with that byte of data will be reset. Any byte of the variable where the flag word is "set" is a good byte of data. While this method significantly encumbers the data storage process, it eliminates the need for a power fail interrupt which both reduces external circuitry and leaves the external interrupt pin completely free for other use.

Often it is necessary to distinguish between an initial power-on condition wherein there is no valid data stored in the RAM (or where V_{SB} has dropped below the minimum required stand-by level) and a re-application of power wherein valid RAM data has been maintained during the power outage. One method of distinguishing between these two conditions is to reserve several memory locations for key words and checksums. When V_{CC} is applied and processor operation begins, these locations can be checked for proper contents. However, this method may not be perfectly accurate as those locations holding key codes may be maintained even though V_{SB} drops below its minimum required level while other RAM locations may lose data, or they could power up with the exact data required to match the key codes. Also a checksum may be matched on occasion even though RAM data has been corrupted. The accuracy of this method is improved by increasing the number of memory locations used and the variety of key codes and or checksums used.

A more reliable method is the external V_{SB} flip-flop. The flip-flop is designed to power up in a known first state and hold that first state until forced into a second state. As long as V_{SB} is above the minimum operating level, the flip-flop can hold the second state, but, if V_{SB} drops below the minimum level, the flip-flop will flip back to the first state. Thus when power is initially applied or if V_{SB} drops below the minimum level during a V_{CC} outage, the flip-flop will be in the first state. The flip-flop output can be read through a port pin by the processor when processor operation begins to determine whether the RAM data is valid (second state) or invalid (first state). If the flip-flop is found to be in the first state it can be forced to the second state by the processor. If it holds the second state, V_{SB} is above the minimum level (batteries are charged).

A conceptual diagram is shown in Figure 6.

CONCEPTUAL DIAGRAM

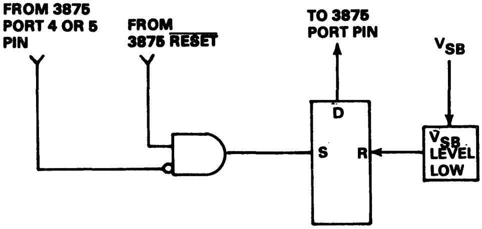

FROM 3875 PORT 4 OR 5 PIN
FROM 3875 $\overline{\text{RESET}}$
TO 3875 PORT PIN
V_{SB}
D
S
R
V_{SB} LEVEL LOW

Figure 6

MK38P75 GENERAL DESCRIPTION

The MK38P75 is the EPROM version of the MK3875. It retains an identical pinout with the MK3875, which is documented in the section of this data sheet entitled "FUNCTIONAL PIN DESCRIPTION". The MK38P75 is housed in the "R" package which incorporates a 28-pin socket located directly on top of the package. A number of standard EPROMs may be plugged into this socket.

The MK38P75 can act as an emulator for the purpose of verification of user code prior to the ordering of mask ROM MK3875 devices. Thus, the MK38P75 eliminates the need for emulator board products. In addition, several MK38P75s can be used in prototype systems in order to test design concepts in field service before commiting to high-volume production with mask ROM MK3875s. The compact size of the MK38P75/EPROM combination allows the packaging of such prototype systems to be the same as that used in production. Finally, in low-volume applications, the MK38P75 can be used as the actual production device.

Most of the material which has been presented for the MK3875 in this document applies to the MK38P75. This includes the description of the pin configuration, architecture, and programming mode. Additional information is presented in the following sections.

MK38P75 I/O PORTS

The MK38P75 is offered with two types of output buffer options on Ports 4 and 5. These are the open drain output buffer and the standard output buffer which are pictured in Figure 4. The open drain version of the MK38P75 is provided so that user-selected open drain port pins on the MK3875 can be emulated prior to ordering those mask ROM devices. Figure 9 lists which version(s) of the MK38P75 has open drain output buffers and which has standard output buffers in parentheses following the specified MK38P75 part ordering number (MK9XXXX).

MK38P75 MAIN MEMORY

As can be seen from the block diagram in Figure 7, the MK38P75 contains executable RAM in the main memory map. The MK38P75 contains no on-chip ROM. Instead, the memory address lines are brought out to the 28-pin socket located directly on top of the 40-pin package, so the external ERPOM memory is addressed as main memory.

There is one memory version of the MK38P75 and it is designated as the MK38P75/02. The MK38P75/02 contains 64 bytes of on-chip executable RAM. The MK38P75/02 can emulate the following devices.

MK3875/22
MK3875/42

The MK38P75/02 cannot exactly emulate the MK3875/40 because of the 64 bytes of executable RAM in the upper ROM space of the MK3875/40.

Addressing of main memory on the MK38P75 is accomplished in the same way as it is for the MK3875. See Figure 8 for main memory addresses and for address register size in the MK38P75.

MK38P75 BLOCK DIAGRAM

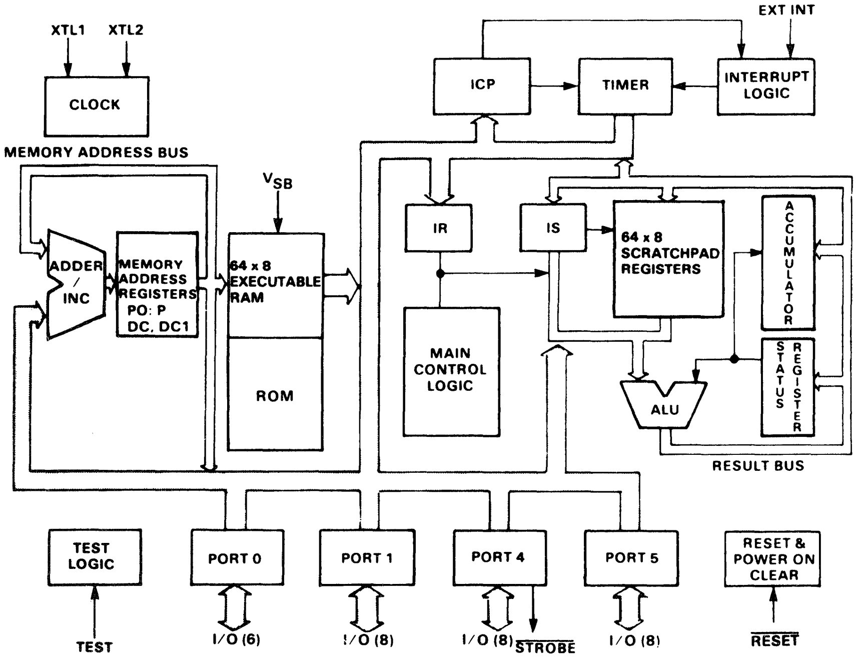
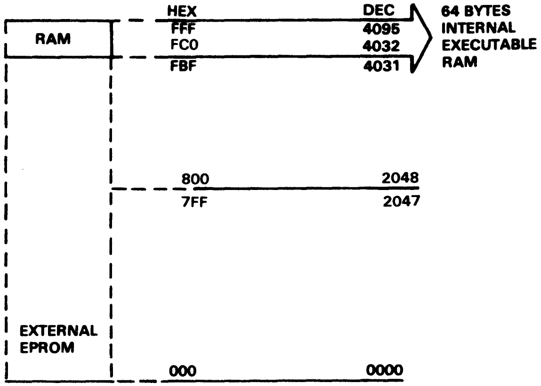


Figure 7

MK38P75 MAIN MEMORY MAP



MK38P75/02

Device	Scratchpad RAM Size (Decimal)	Address Register Size P0, P, DC, DC1)	ROM Size (Decimal)	Executable RAM Size
MK38P75/02 97403	64 bytes	12 bits	0 bytes	64 bytes

Figure 8

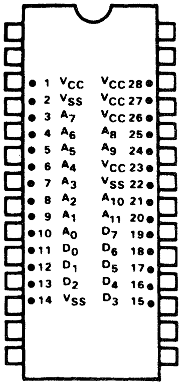
MK38P75 EPROM SOCKET

A 28-pin ERPOM socket is located on top of the MK38P75 "R" package. The socket and compatible ERPOM memories are shown in Figure 9. When 24-pin memories are used in the 28-pin socket, they should be inserted so that pin 1 of the memory device is plugged into pin 3 of the socket (the 24-pin memory should be lower justified in the 28-pin socket).

The 28-pin socket has been provided to allow use of both 24-pin and 28-pin memory devices. Minor pin-out differences in the memory devices must be accommodated by providing different versions of the MK38P75.

Initially, the MK38P75 that is compatible with the MK2716 is available. The MK38P75 designed to accommodate the 28-pin memory devices will be available at a later date.

MK38P75 "R" PACKAGE SOCKET PINOUT



MK97413 (Open Drain Outputs) Compatible Memories	MK97403 (Standard Outputs) Compatible Memories
2758	2758
MK2716	MK2716
2516	2516
2532	2532

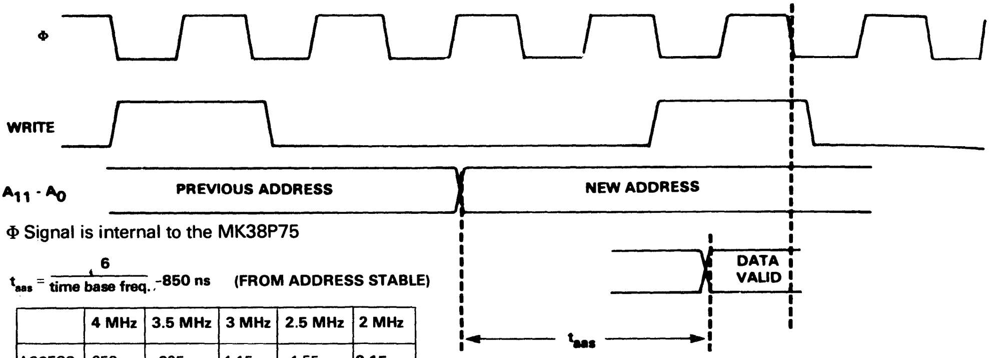
Figure 9

10/24

MEMORY ACCESS TIMING

A timing diagram depicting the memory access timing of the MK38P75 is shown in the next table. The Φ clock signal is derived internally in the MK38P75 by dividing the time base frequency by two and is used to establish all timing frequencies. The WRITE signal is another internal signal to the MK38P75 which corresponds to a machine cycle, during which time a memory access may be performed. Each machine cycle is either 4 Φ clock periods or 6 Φ clock periods long. These machine cycles are termed short cycles and long cycles, respectively. The worst case memory cycle is the short cycle, during which time an op code fetch is performed. This is the cycle which is pictured in the timing diagram. After a delay from the falling edge of the WRITE clock, the address lines become stable. Data must be valid at the data out lines of the PROM for a setup time prior to the next falling edge of the WRITE pulse. The total access time available for the MK38P75 version is shown as t_{aas} or the time when address is stable until data must be valid on the data bus lines. The equation for calculating available memory access time along with some calculated access times based on the listed time base frequencies is shown in the following table.

MEMORY ACCESS SHORT CYCLE OP CODE FETCH MK38P75



Φ Signal is internal to the MK38P75

$t_{aas} = \frac{6}{\text{time base freq.}} - 850$ ns (FROM ADDRESS STABLE)

	4 MHz	3.5 MHz	3 MHz	2.5 MHz	2 MHz
ACCESS TIME	650 ns	825 ns	1.15 μs	1.55 μs	2.15 μs

Figure 10

3875 TIME BASE OPTIONS

The 3875 contains an on-chip oscillator circuit which provides an internal clock. The frequency of the oscillator circuit is set from the external time base network. The time base for the 3875 may originate from one of four sources:

1) Crystal
2) LC Network
3) RC Network
4) External Clock

The type of network which is to be used with the mask ROM MK3875 must be specified at the time when mask ROM devices are ordered. However, the MK38P75 may operate with any of the four configurations so that it may emulate any configuration used with a mask ROM device.

The specifications for the four configurations are given in the following text. There is an internal 26 pF capacitor between XTL 1 and GND and an internal 26 pF capacitor between XTL 2 and GND. Thus, external capacitors are not necessarily required. In all external clock modes the external time base frequently is divided by two to form the internal PHI clock.

CRYSTAL SELECTION

The use of a crystal as the time base is highly recommended as the frequency stability and reproducability from system to system is unsurpassed. The 3875 has an internal divide by two to allow the use of inexpensive and widely available TV Color Burst Crystals (3.58 MHz). Figure 12 lists the required crystal parameters for use with the 3875. The Crystal Mode time base configuration is shown in Figure 11.

Through careful buffering of the XTL1 pin it may be possible to amplify this waveform and distribute it to other devices. However, Mostek recommends that a separate active device (such as a 7400 series TTL gate) be used to oscillate the crystal and that the waveform from that oscillator be buffered and supplied to all devices, including the 3875, in the event that a single crystal is to provide the time base for more than just a single 3875.

While a ceramic resonator may work with the 3875 crystal oscillator, it was not designed specifically to support the use of this component. Thus, Mostek does not support the use of a ceramic resonator either through proper testing, parametric specification, or applications support.

CRYSTAL MODE CONNECTION

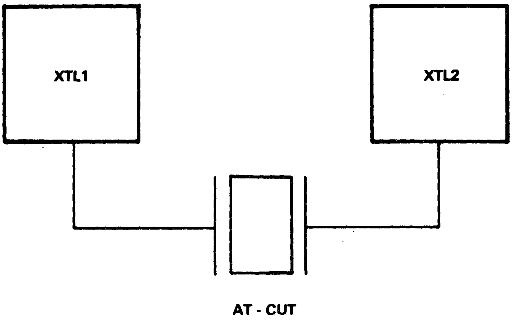


Figure 11

CRYSTAL PARAMETERS

a) Parallel resonance, fundamental mode AT-Cut
b) Shunt capacitance (C_0) = 7 pf max.
c) Series resistance (R_S) = See table
d) Holder = See table below.

Frequency	Series Resistance	Holder
f = 2-2.7 MHz	Rs = 300 ohms max	HC-6 HC-33
f = 2.8-4 MHz	Rs = 150 ohms max	HC-6 HC-18* HC-25* HC-33

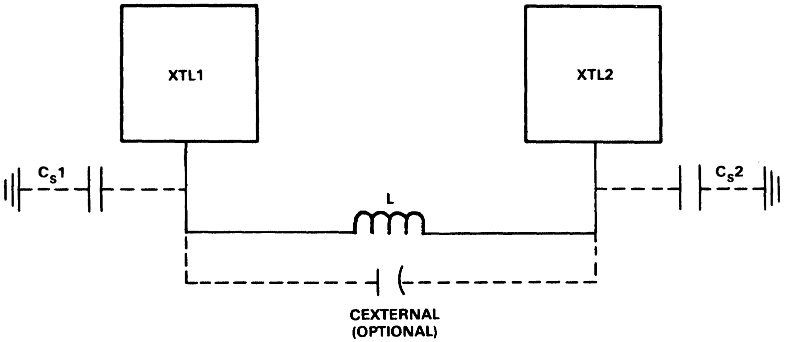
*This holder may not be available at frequencies near the lower end of this range.

Figure 12

LC NETWORK

The LC time base configuration can be used to provide a less expensive time base for the 3875 than can be provided with a crystal. However, the LC configuration is much less accurate than is the crystal configuration. The LC time base configuration is shown in Figure 13. Also shown in the figure are the specified parameters for the LC components, along with the formula for calculating the resulting time base frequency. The minimum value of the inductor which is required for proper operation of the LC time base network is 0.1 millihenries. The inductor must have a Q factor which is no less than 40. The value of C is derived from C external, the internal capacitance of the 3875, C_{XTL}, and the stray capacitances, C_{S1} and C_{S2}. C_{XTL} is the at XTL1 and capacitance looking into the internal two port network XTL2. C_{XTL} is listed under the "Capacitance" section of the Electrical Specifications. C_{S1} and C_{S2} are stray capacitances from XTL1 to ground and from XTL2 to ground, respectively. C external should also include the stray shunt capacitance across the inductor. This is typically in the 3 to 5 pf range and significant error can result if it is not included in the frequency calculation.

LC MODE CONNECTION



$$f = \frac{1}{2\pi\sqrt{LC}}$$

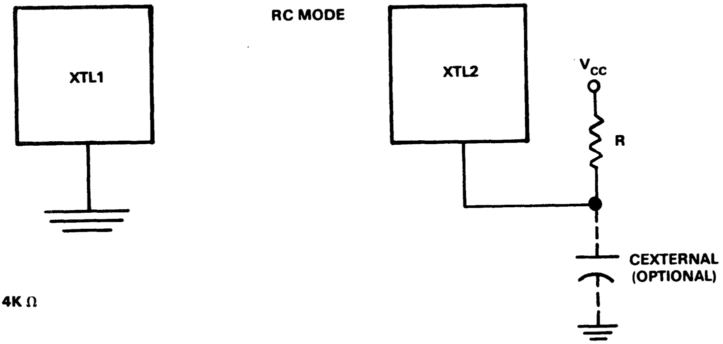
Figure 13

Variation in time base frequency with the LC network can arise from one of four sources: 1) Variation in the value of the inductor. 2) Variation in the value of the external capacitor. 3) Variation in the value of the internal capacitance of the 3875 at XTL1 and XTL2, and 4) Variation in the amount of stray capacitance which exists in the circuit. Therefore, the actual frequency which is generated by the LC circuit is within a range of possible frequencies, where the range of frequencies is determined by the worst case variation in circuit parameters. The designer must select component values such that the range of possible frequencies with the LC mode does not go outside of the specified operating frequency range for the 3875.

RC CLOCK CONFIGURATION

The time base for the 3875 may be provided from an RC network tied to the XTL2 pin, when XTL1 is grounded. A schematic picturing the RC clock configuration is shown in Figure 14. The RC time base configuration is intended to provide an inexpensive time base source for applications in which timing is not critical. Some users have elected to tune each unit using a variable resistor or external capacitor thus reducing the variation in frequency. However, for increased time base accuracy Mostek recommends the use of the Crystal or LC time base configuration. Figure 15 illustrates a curve which gives the resulting operating frequency for a particular RC value. The x-axis represents the product of the value of the resistor times the value of the capacitor. Note that three curves are actually shown. The curve in the middle represents the nominal frequency obtained for a given value of RC. A maximum curve and a minimum curve for different types of 3875 devices are also shown in the diagram.

RC MODE CONNECTION



MINIMUM R = 4K Ω

C = 26.5 pF ± 2.6pF + Cexternal

Figure 14

FREQUENCY VS. RC

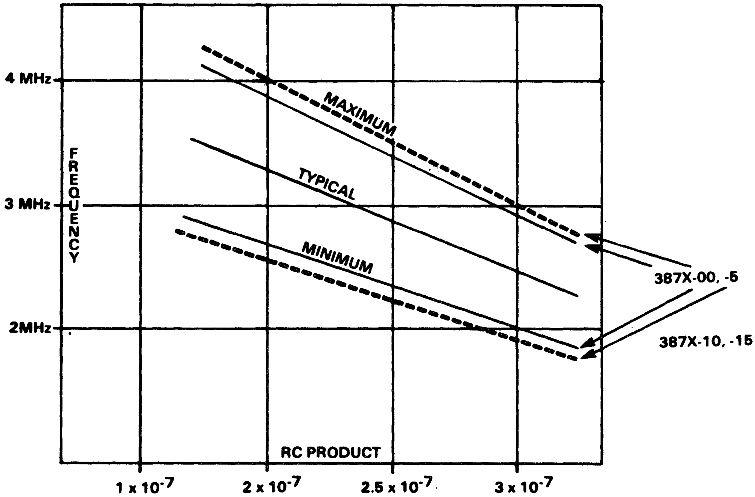


Figure 15

The designer must select the RC product such that a frequency of less than 2 MHz is not possible taking into account the maximum possible RC product and using the minimum curve shown in Figure 15. Also, the RC product must not allow a frequency of more than 4 MHz taking into account the minimum possible R and C and using the Maximum curve shown. Temperature induced variations in the external components should be considered in calculating the RC product.

Frequency variation from unit to unit due to switching speed and level at constant temperature and V_{CC} = + or − 5 percent.

Frequency variation due to V_{CC} with all other parameters constant with respect to +5V = +7 percent to −4 percent on all devices.

Frequency variation due to temperature with respect to 25 C (all other parameters constant) is as follows:

PART #	VARIATION
387X-00, -05	+6 percent to − 9 percent
387X-10, −15	+9 percent to −12 percent

Variations in frequency due to variations in RC components may be calculated as follows:

Maximum RC = (R max) (C external max + C_{XTL} max)

Minimum RC = (R min) (C external min + C_{XTL} min)

$$\text{Typical RC} = (\text{R typ})\left(\text{C external typ} + \frac{\{C_{XTL}\ \text{max} + C_{XTL}\ \text{min}\}}{2}\right)$$

$$\text{Positive Freq. Variation} = \frac{\text{RC typical - RC minimum}}{\text{RC typical}}$$

$$\text{Negative Freq. Variation due to RC Components} = \frac{\text{RC maximum - RC typical}}{\text{RC typical}}$$

Total frequency variation due to all factors:

387X-00, -05	387X-10, −15
= +18 percent plus positive frequency variation due to RC components	= +21 percent plus positive frequency variation due to RC components
= −18 percent minus negative frequency variation due to RC components	= −21 percent minus negative frequency variation due to RC components

Total frequency variation due to V_{CC} and temperature of a unit tuned to frequency at +5V V_{CC}, 25 C

387X-00, -05	387X-10, −15
= + 13 percent	= + 16 percent

EXTERNAL CLOCK CONFIGURATION

The connection for the external clock time base configuration is shown in Figure 16. Refer to the DC Characteristics section for proper input levels and current requirements.

Refer to the Capacitai.ce section of the appropriate 3875 Family device data sheet for input capacitance.

EXTERNAL MODE CONNECTION

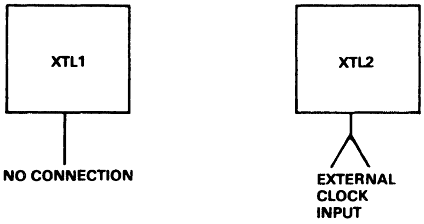


Figure 16

MK3875, MK38P75
ELECTRICAL SPECIFICATIONS

OPERATING VOLTAGES AND TEMPERATURES

Dash Number Suffix	Operating Voltage V_{CC}	Operating Temperature T_A
-00	¯V ± 10%	0°C - 70°C
-05	+5V ± 5%	0°C - 70°C
-10	+5V ± 10%	-40°C - +85°C
-15	+5V ± 5%	-40°C - +85°C

See order information for explanation of part numbers.

MAXIMUM RATINGS*

	-00, -05	-10, -15
Temperature Under Bias	-20°C +85°C	-50°C to 100°C
Storage Temperature	-65°C +150°C	-65°C to +150°C
Voltage on any Pin With Respect to Ground (Except open drain pins and TEST)	-1.0V to +7V	-1.0V to +7V
Voltage on TEST with Respect to Ground	-1.0V to +9V	-1.0V to +9V
Voltage on Open Drain Pins with Respect to Ground	-1.0V to +13.5V	-1.0V to 13.5V
Power Dissipation	1.5W	1.5W
Power Dissipation by any one I/O pin	60mW	60mW
Power Dissipation by all I/O pins	600mW	600mW

*Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other condition above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating and conditions for extended periods may affect device reliability.

AC CHARACTERISTICS

T_A, V_{CC} within specified operating range
I/O Power Dissipation < 100mW (Note 4)

SIGNAL	SYM	PARAMETER	-00,-05		-10,-15		UNIT	NOTES
			MIN	MAX	MIN	MAX		
XTL1 XTL2	t_o	Time Base Period, all clock modes	250	500	250	500	ns	4MHz-2MHz
	$t_{ex(H)}$	External clock pulse width high	90	400	100	390	ns	
	$t_{ex(L)}$	External clock pulse width low	100	400	110	390	ns	
Φ	t_Φ	Internal Φ clock	$2t_o$		$2t_o$			
WRITE	t_W	Internal WRITE Clock period	4 t Φ 6 t Φ		4tΦ 6tΦ			Short Cycle Long Cycle
I/O	$t_{dI/O}$	Output delay from internal WRITE clock	0	1000	0	1200	ns	50pF plus one TTL load
	$t_{sI/O}$	Input setup time to internal WRITE clock	1000		1200		ns	
$\overline{\text{STROBE}}$	$t_{I/O\text{-}s}$	Output valid to $\overline{\text{STROBE}}$ delay	3tΦ -1000	3tΦ +250	3tΦ -1200	3tΦ +300	ns	I/O load = 50fF + 1 TTL load
	t_{sL}	$\overline{\text{STROBE}}$ low time	8tΦ -250	12tΦ +250	8tΦ -300	12tΦ +300	ns	$\overline{\text{STROBE}}$ load = 50pF + 3TTL loads
$\overline{\text{RESET}}$	t_{RH}	$\overline{\text{RESET}}$ hold time, low	6tΦ +750		6tΦ +1000		ns	
	t_{RPOC}	$\overline{\text{RESET}}$ hold time, low for power clear	power supply rise time +5.0		power supply rise time +5.5		ms	
EXT INT	t_{EH}	EXT INT hold time in active and inactive state	6tΦ +750		6tΦ +1000		ns	To trigger interrupt
			2tΦ		2tΦ		n[illegible]	To tri[illegible]r

AC CHARACTERISTICS FOR MK38P75

(Signals brought out at socket)
T_A, V_{CC} within specified operating range.
I/O Power Dissipation ≤ 100 mW. (Note 2)

SYMBOL	PARAMETER	-00, -05		-10, -15		UNIT	CONDITION
		MIN	MAX	MIN	MAX		
t_{aas}*	Access time from Address A_{11}-A_0 stable until data must be valid at D_7-D_0	650		650		ns	Φ = 2.0 MHz

*See Table in Figure 10

CAPACITANCE

T_A = 25°C All Part Numbers

SYM	PARAMETER	MIN	MAX	UNIT	NOTES
C_{IN}	Input capacitance; I/O $\overline{\text{RESET}}$, EXT INT, TEST		10	pF	unmeasured pins grounded
C_{XTL}	Input capacitance; XTL1, XTL2	23.5	29.5	pF	

DC CHARACTERISTICS

T_A, V_{CC} within specified operating range
I/O Power Dissipation ≤ 100mW (Note 4)

		-00,-05		-10,-15			
SYM	**PARAMETER**	**MIN**	**MAX**	**MIN**	**MAX**	**UNIT**	**NOTES**
I_{CC}	Average Power Supply Current		94		125	mA	Outputs Open (5)
P_D	Average Power Dissipation		440		575	mW	Outputs Open (6)
V_{IHEX}	External Clock input high level	2.4	5.8	2.4	5.8	V	
V_{ILEX}	External Clock input low level	-.3	.6	-.3	.6	V	
I_{IHEX}	External Clock input high current		100		130	μA	$V_{IHEX}=V_{CC}$
I_{ILEX}	External Clock input low current		-100		-130	μA	$V_{ILEX}=V_{SS}$
$V_{IHI/O}$	Input high level, I/O pins	2.0	5.8	2.0	5.8	V	Standard Pull-Up (1,2)
		2.0	13.2	2.0	13.2	V	Open Drain (1,3)
V_{IHR}	Input high level, $\overline{\text{RESET}}$	2.0	5.8	2.2	5.8	V	Standard Pull-Up(1, 2)
		2.0	13.2	2.2	13.2	V	No Pull-Up (1,3)
V_{IHEI}	Input high level, EXT INT	2.0	5.8	2.2	3.8	V	Standard Pull-Up(1, 2)
		2.0	13.2	2.2	13.2	V	No Pull-Up (1,3)
V_{IL}	I/O ports, $\overline{\text{RESET}}$, EXT INT input low level	-.3	.8	-.3	.7	V	
V_{ILRPT}	$\overline{\text{RESET}}$ input low level to protect RAM during loss at V_{CC}	-.3	.4	-.3	.4	V	
I_{IL}	Input low current, standard pull-up pins		-1.6		-1.9	mA	V_{IN}=0.4V (2)

DC CHARACTERISTICS (Continued)

T_A, V_{CC} within specified operating range
I/O Power Dissipation ≤ 100 mW (Note 4)

		-00, -05		-10, -15			
SYM	**PARAMETER**	**MIN**	**MAX**	**MIN**	**MAX**	**UNIT**	**NOTES**
I_L	Input leakage current, open drain pins		+10		+18	μA	V_{IN}=13.2V
	Reset and EXT INT inputs With no pull-up resistor		-5		-8	μA	V_{IN}=0.0V (3)
I_{OH}	Output high current, standard	-100		-89		μA	V_{OH}=2.4V
	Pull-Up pins	-30		-25		μA	V_{OH}=3.9V
I_{OHDD}	Output high current	-100		-80		μA	V_{OH}=2.4V
	Direct Drive pins	-1.5		-1.3		mA	V_{OH}=1.5V
			-8.5		-11	mA	V_{OH}=0.7V
I_{OL}	Output low current, I/O ports	1.8		1.65		mA	V_{OL}=0.4V
I_{OHS}	STROBE Output High current	-300		-270		μA	V_{OL}=2.4V
I_{OLS}	STROBE output low current	5.0		4.5		mA	V_{OL}=0.4V

DC CHARACTERISTICS FOR STANDBY POWER PINS

V_{CC}, T_A within operating range I/O Power Dissipation ≤ 100 mW (Note 4)

		-00, -05		-10,-15			
SYMBOL	PARAMETER	MIN	MAX	MIN	MAX	UNIT	NOTES
V_{SB}	Standby V_{CC} for RAM	3.2	V_{CC} MAX	3.2	V_{CC} MAX	V	
I_{SB}	Standby Current		6		7.5	mA	$V_{SB} = V_{SB}$ MAX
			3.7		5.0	mA	$V_{SB} = V_{SB}$ MIN
I_{CHARGE}	Trickle charge available on $V_{SB}$with V_{CC} in operating range.	-.8		-.7		mA	V_{SB} = 3.8V
			-15		-19	mA	V_{SB} = 3.2V

DC CHARACTERISTICS FOR MK38P75

(Signals brought out at socket)

T_A, V_{CC} within specified operating range, I/O power dissipation ≤ 100 mW (Note 2)

		-00, -05		-10, -15			
SYM	PARAMETER	MIN	MAX	MIN	MAX	UNIT	CONDITION
I_{CCE}	Power Supply Current for EPROM		–185		–185	mA	
V_{IL}	Input Low Level Data bus in	-0.3	0.8	-0.3	0.8	V	
V_{IH}	Input'High Level Data bus in	2.2	5.8	2.2	5.8	V	
I_{OH}	Output High Current	-100		-90		μA	V_{OH}=2.4 V
		-30		-25		μA	V_{OH}=3.9 V
I_{OL}	Output Low Current	1.8		1.65		mA	V_{OL}=0.4 V
I_{IL}	Input Leakage Current		10		10	μA	Data Bus in Float

1. $\overline{\text{RESET}}$ and ET INT have internal Schmit triggers giving minimum .2V hysteresis.
2. $\overline{\text{RESET}}$ and EXT INT prgrammed with standard pull-up
3. $\overline{\text{RESET}}$ or EXT INT programmed without standard pull-up
4. Power dissipation for I/O pins is calculated by $\Sigma (V_{CC} - V_{IL}) (|I_{IL}|) + \Sigma (V_{CC} - V_{OH}) (|I_{OH}|) + \Sigma (V_{OL}) (I_{OL})$
5. I_{CC} exclusive of I_{charge}.
6. P_D exclusive of battery charging power. Battery charging power dissipated inside the MK3875 $(V_{CC} - V_{SB}) (I_{charge})$.

TIMER AC CHARACTERISTICS

Definitions:

Error = Indicated time value - actual time value

tpsc = tΦ x Prescale Value

Interval Timer Mode

Single interval error, free running (Note 3)	±6tΦ
Cumulative interval error, free running (Note 3)	0
Error between two Timer reads (Note 2)	±(tpsc + tΦ)
Start timer to stop Timer error (Notes 1, 4)	+tΦ to - (tpsc + tΦ)
Start Timer to read Timer error (Notes 1, 2)	-5tΦ to -(tpsc + 7tΦ)
Start Timer to interrupt request error (Notes 1, 3)	-2tΦ to -8tΦ
Load Timer to stop Timer error (Note 1)	+tΦ to -(tpsc + 2tΦ)
Load Timer to read Timer error (Notes 1, 2)	-5tΦ to -(tpsc + 8tΦ)
Load Timer to interrupt request error (Notes 1, 3)	-2tΦ to -9tΦ

Pulse Width Measurement Mode

Measurement accuracy (Note 4) .... +t Φ to -(tpsc +2t Φ
Minimum pulse width of EXT INT pin .... 2t Φ

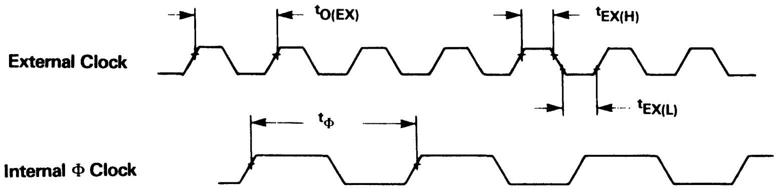
Event Counter Mode

Minimum active time of EXT INT pin .... 2t Φ
Minimum inactive time of EXT INT pin .... 2t Φ

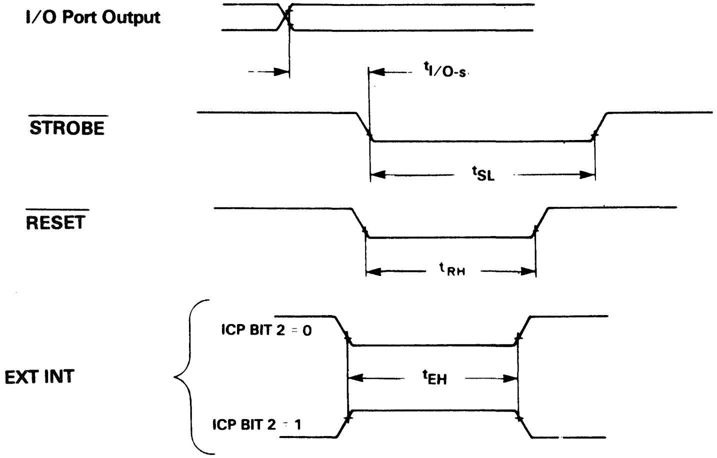
Notes:

1. All times which entail loading, starting, or stopping the Timer are referenced from the end of the last machine cycle of the OUT or OUTS instruction.
2. All times which entail reading the Timer are referenced from the end of the last machine cycle of the IN or INS instruction.
3. All times which entail the generation of an interrupt request are referenced from the start of the machine cycle in which the appropriate interrupt request latch is set. Additional time may elapse if the interrupt request occurs during a privileged or multicycle instruction.
4. Error may be cumulative if operation is repetitively performed.

AC TIMING DIAGRAM



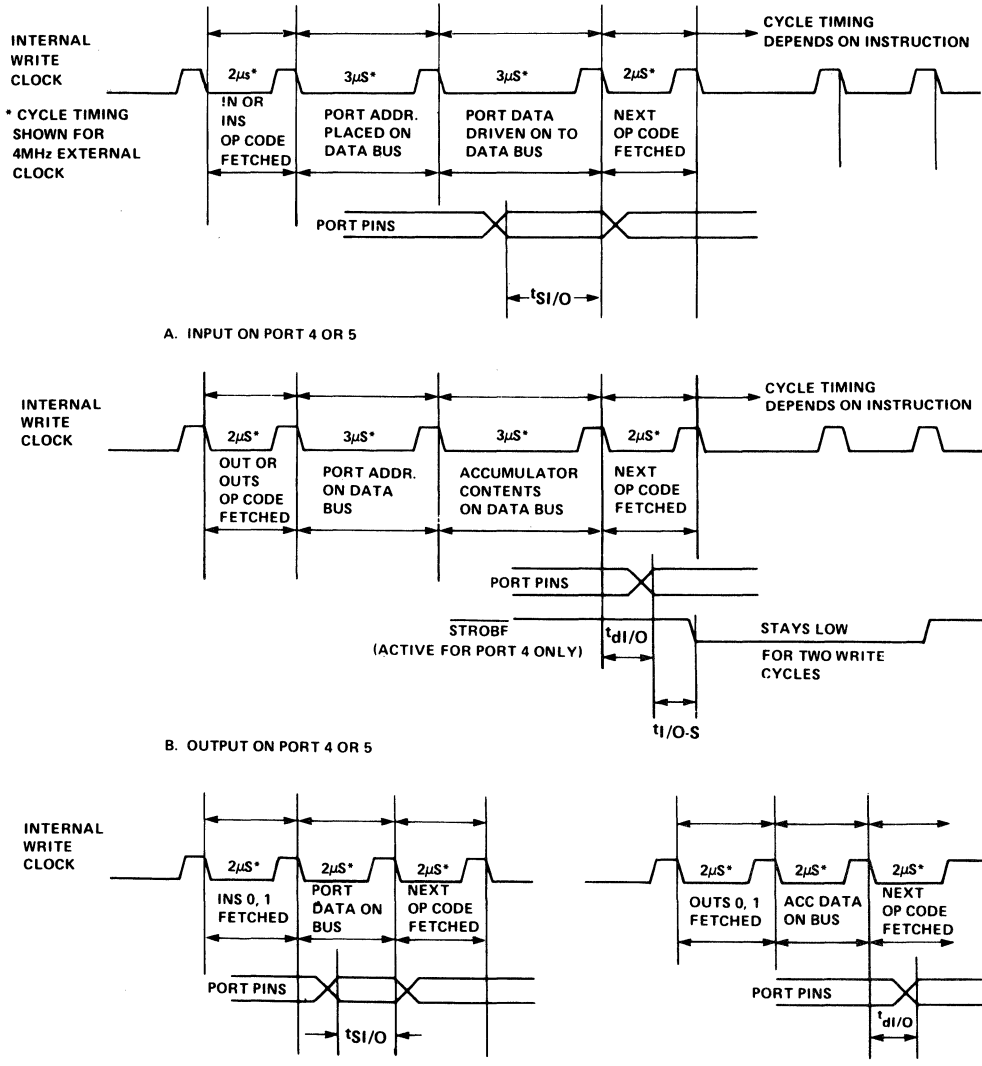
Input capacitance; I/O, RESET, EXT INT,



Note: All AC measurements are referenced to V_{IL} max., V_{IH} min., V_{OL} (.8v), or V_{OH} (2.0v).

Figure 17

INPUT/OUTPUT AC TIMING



A. INPUT ON PORT 4 OR 5

B. OUTPUT ON PORT 4 OR 5

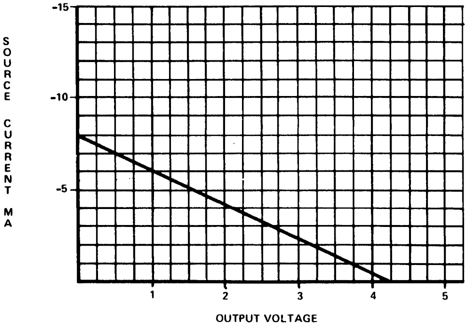
C. INPUT ON PORT 0 OR 1

D. OUTPUT ON PORT 0, 1

Figure 18

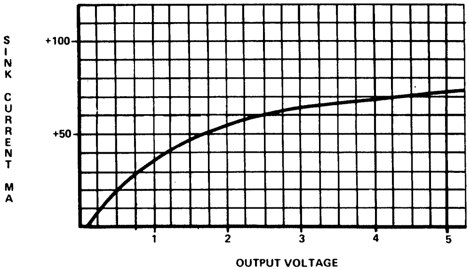
STROBE SOURCE CAPABILITY (TYPICAL AT V_{CC} = 5V, T_A = 25°C)

Figure 19



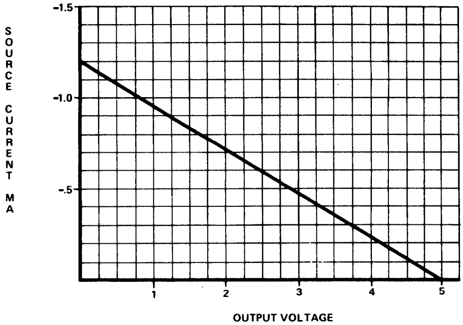
STROBE SINK CAPABILITY (TYPICAL AT V_{CC} = 5V, T_A = 25°C)

Figure 20



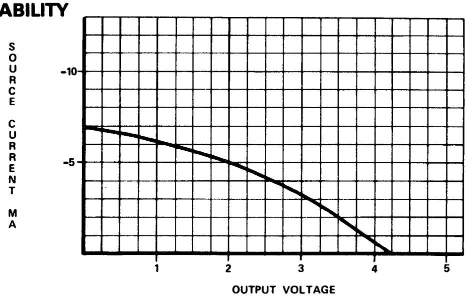
STANDARD I/O PORT SOURCE CAPABILITY (TYPICAL AT V_{CC} = 5V, T_A = 25°C)

Figure 21



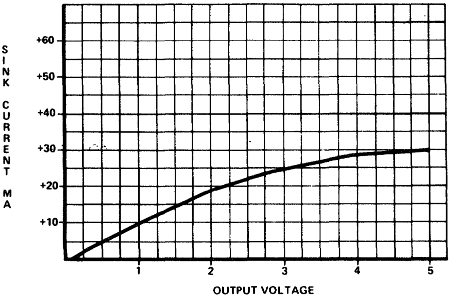
DIRECT DRIVE I/O PORT SOURCE CAPABILITY (TYPICAL AT V_{CC} = 5V, T_A = 25°C)

Figure 22



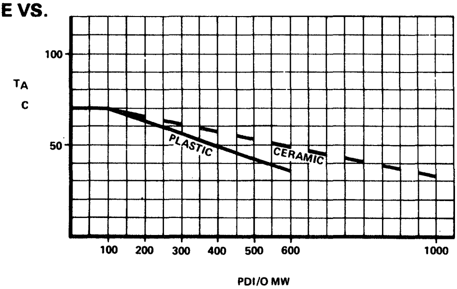
I/O PORT SINK CAPABILITY (TYPICAL AT V_{CC} = 5V, T_A = 25°C)

Figure 23

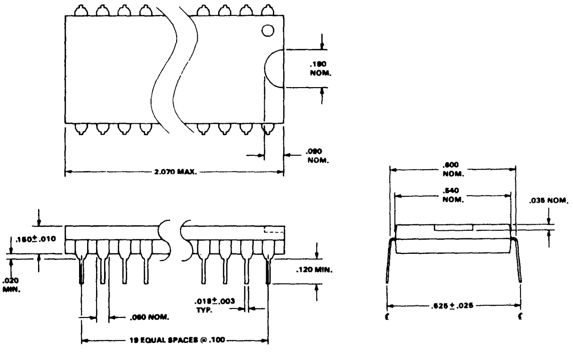


MAXIMUM OPERATING TEMPERATURE VS. I/O POWER DISSIPATION

Figure 24

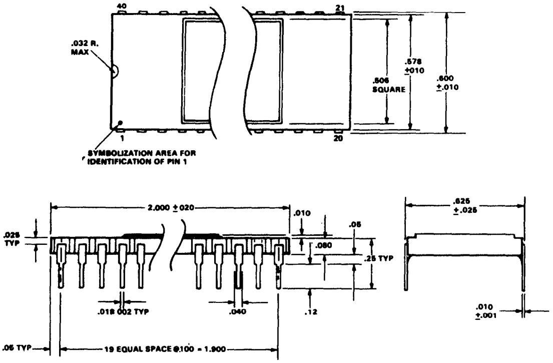


PHYSICAL DIMENSIONS

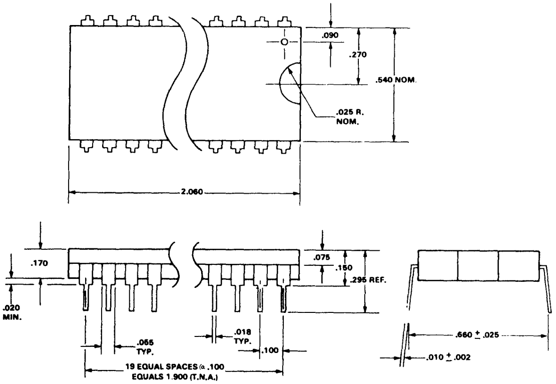


NOTE: Overall length includes .005 flash on either end of package

40-Pin Dual-in-Line Plastic Package (N)



40-Pin Dual-in-Line Ceramic Package (P)



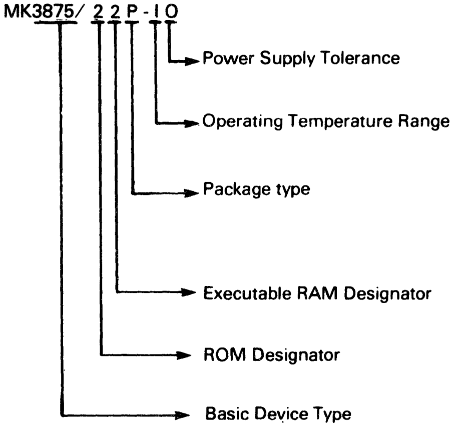
40-Pin Dual-in-Line Cerdip Package (J)

ORDERING INFORMATION

There are two types of part numbers for the 3870 family of devices. Th϶ generic part number describes the basic device type, th϶ amount of ROM and executable RAM, the desired package type, temperature range and power supply tolerance. For each customer specific code, additional information defining I/O options and oscillator options will be combined with the information described in the generic part number to define a customer/code specific device order number.

GENERIC PART NUMBER

An example of the generic part number is shown below.



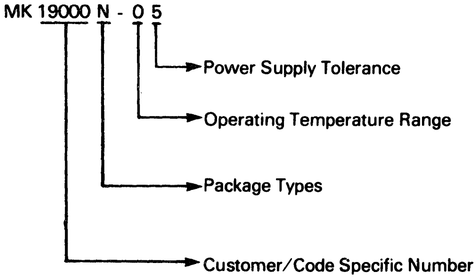
Power Supply Tolerance	0 = 5V ± 10% 5 = 5V ± 5%
Operating Temperature Range	0 = 0°C - +70°C 1 = -40°C - +85°C

Package type	P = Ceramic
	N = Plastic

Executable RAM Designator	2 = 64 Bytes
ROM Designator	2 = 2K Bytes 4 = 4K Bytes
Basic Device Type	

DEVICE ORDER NUMBER

An example of the device order number is shown below.



Power Supply Tolerance	0 = +5V ± 10% 5 = +5V ± 5%
Operating Temperature Range	0 = 0°C - +70°C 1 = -40°C - +85°C

Package Types	P = Ceramic
	N = Plastic

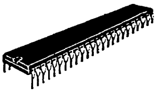
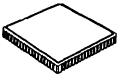
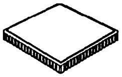
Customer/Code Specific Number	

The customer/code specific number defines the ROM bit pattern, I/O configuration, oscillator type, and generic part type to be used to satisfy the requirement of a particular customer purchase order. For further information on the ordering of mask ROM devices, the customer should refer to the 3870 Family Technical Manual.

Printed in France

Ref DSMK3875P75

16-bit microcomputers

Function	Part number	Characteristic		Package	Page
HIGH PERFORMANCE MCUs	MK68200	MCU designed to serve the needs of a wide variety of control applications, which require high performance operation with a minimal parts count implementation as industrial controls, instrumentation, intelligent computer peripheral control. • 40 parallel I/O • Serial channel, double-buffered receive and transmit • Advanced 16-bit instruction set • 6 MHz instruction clock • Three 16-bit timers • 16 independant vectored interrupt • 8 external interrupts • Crystal or external TTL time base	• RAM: 256 bytes • ROM: 0 or 4 Kbytes • Emulator version available with added private bus	 DIL48  LCCC84	3-5
HIGH PERFORMANCE MCUs WITH MODULAL ARCHITECTURE	MK68HC200	Series of new high-performance, 16-bit, single-chip microcomputers implemented in 1.5 micron HCMOS technology. The MK68HC200 can be used to design a true application specific microcontroller. • 40 parallel I/O on first version • Serial channel, double-buffered receive and transmit • Advanced 16-bit instruction set • 10 MHz instruction clock • Three 16-bit timers • 16 independant vectored interrupt on first version • 8 external interrupts • Crystal or external TTL time base • Power saving stop and idle modes	• RAM: 256 bytes • ROM: 0 or 4 Kbytes • Emulator version available with added private bus	 PLCC52  LCCC84	3-77

THOMSON COMPONENTS SEMICONDUCTEURS

MK68200 FAMILY

16-BIT SINGLE CHIP MICROCOMPUTERS

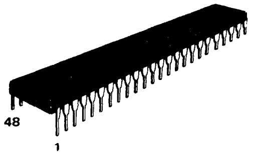
MK68201/68E201/68211/68E211/68E221

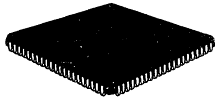
16-bit, high performance, single-chip microcomputer

- 14 address and data registers
  - — Eight 16-bit or sixteen 8-bit data registers
  - — Six 16-bit address registers
- Advanced 16-bit instruction set
  - — Bit, byte, and word operands
  - — Nine addressing modes
  - — Byte and word BCD arithmetic
- High performance (6 MHz instruction clock)
  - — 500 ns register-to-register move or add
  - — 3.5 μs 16 × 16 multiply
  - — 4.0 μs 32/16 divide
- Available with 0, 4K (2K × 16) of ROM
- 256 (128 × 16) or 512 (256 × 16) byte RAM
- Up to 1K byte RAM on MK68E221
- Three 16-bit timers
  - — Interval modes
  - — Event modes
  - — One-shot modes
  - — Pulse and period measurement modes
  - — Two input and two output pins
- Serial channel
  - — Double-buffered receive and transmit
  - — Asynchronous to 375 Kbps
  - — Synchronous to 1.5 Mbps
  - — Address wake-up recognition and generation
  - — Internal/external baud rate generation
- Parallel I/O
  - — Up to 40 pins
  - — Direction programmable by bit
  - — One 16-bit or two 8-bit port(s) with handshaking
- Interrupt controller
  - — 16 independent vectors
  - — Eight external interrupt sources
  - — One non-maskable interrupt
  - — Individual interrupt masking
- Optional external bus
  - — 16-bit, multiplexed address/data bus
  - — Automatic bus request/grant arbitration
  - — Two control bus versions:
    - 68000-compatible bus (UPC)
    - General Purpose Bus (GP)
- 8 and 12 MHz time base versions produce 4 and 6 MHz instruction clock rates, respectively.
  - — Crystal or external TTL clock
- Single +5 volt power supply
- DIP, chip carrier or pin-grid packaging

NMOS

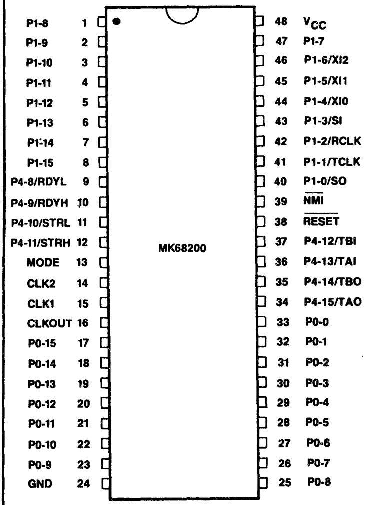
CASES





LCC 84

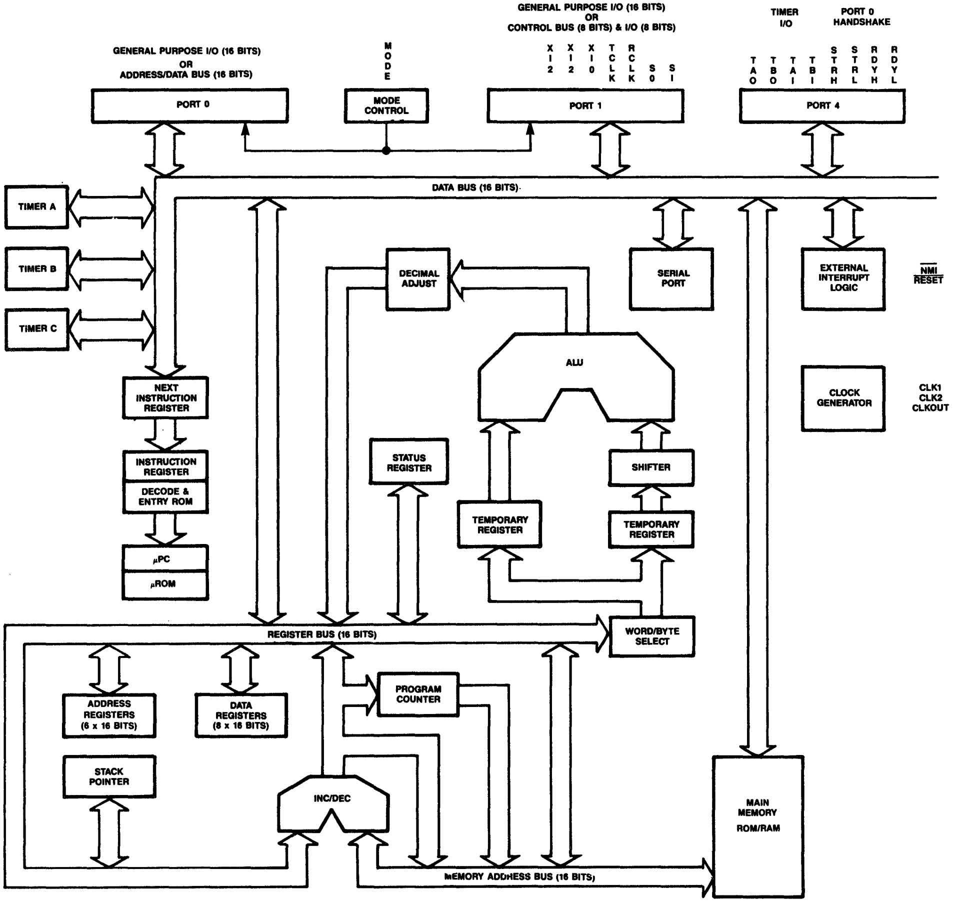
PIN ASSIGNMENT



P1-8	1	48	V_{CC}
P1-9	2	47	P1-7
P1-10	3	46	P1-6/XI2
P1-11	4	45	P1-5/XI1
P1-12	5	44	P1-4/XI0
P1-13	6	43	P1-3/SI
P1-14	7	42	P1-2/RCLK
P1-15	8	41	P1-1/TCLK
P4-8/RDYL	9	40	P1-0/SO
P4-9/RDYH	10	39	$\overline{NMI}$
P4-10/STRL	11	38	$\overline{RESET}$
P4-11/STRH	12	37	P4-12/TBI
MODE	13	36	P4-13/TAI
CLK2	14	35	P4-14/TBO
CLK1	15	34	P4-15/TAO
CLKOUT	16	33	P0-0
P0-15	17	32	P0-1
P0-14	18	31	P0-2
P0-13	19	30	P0-3
P0-12	20	29	P0-4
P0-11	21	28	P0-5
P0-10	22	27	P0-6
P0-9	23	26	P0-7
GND	24	25	P0-8

JANUARY 1987 1/72

BLOCK DIAGRAM



FUNCTIONAL DESCRIPTION

MK68200 designates a series of new, high-performance, 16-bit, single-chip microcomputers from Thomson - Mostek. Implemented in Scaled Poly-5 NMOS, they incorporate an architecture designed for superior performance in computation-intensive control applications. A modern, comprehensive instruction set (which features both high speed execution and code space efficiency) is combined on-chip with extensive, flexible I/O capabilities. On-chip RAM and optional on-chip ROM are provided within a full 64K byte addressing space.

The MK68200 is designed to serve the needs of a wide variety of control applications, which require high performance operation with a minimal parts count implementation. Industrial controls, instrumentation, and intelligent computer peripheral controls are all examples of applications served by the MK68200. High speed mathematical ability, rapid I/O addressing and interrupt response, and powerful bit manipulation instructions provide the necessary tools for these applications. In addition to its single-chip microcomputer configuration, both distributed intelligence and parallel multiprocessing system configurations are supported by the MK68200, as illustrated in Figures 10 and 11.

In applications requiring loosely-coupled distributed intelligence, several MK68200's may be interconnected on a common serial network. The on-chip USART supports a wake-up mode in which an additional bit is appended to the data stream to distinguish a serial data word as address or data. The wake-up logic prevents the serial channel from generating interrupts unless certain criteria have been met. The wake-up options available are: Wake-up on any address or data character, wake-up on any address, or wake-up on address match.

Alternately, the MK68200 may be configured as an expandable CPU device which can access external memory and I/O resources. In this operating mode, parallel I/O pins are replaced by multiplexed address/data and control lines. Bus arbitration logic is incorporated on the chip to support a direct interface in parallel shared bus multiprocessor system configurations. Two versions exist which support two types of control signals present on the expanded bus configuration. The General Purpose (GP) bus option allows the MK68200 to operate either as an executive or a peripheral processor. As an executive procesor, the MK68200 can control an external system bus and grant the use of it to requesting devices, such as DMA controllers and/or peripheral processors. As a peripheral control processor, the MK68200 can provide intelligent local control of an I/O device in a computer system and, thereby, relieve the executive processor of these tasks. In this configuration, the MK68200 has the capability of effectively performing DMA transfers between system memory and the I/O device. The on-chip resources of ROM, RAM, and I/O are accessed within the MK68200 without affecting utilization of the shared system bus. Therefore, only external communications compete for bus bandwidth.

The Universal Peripheral Controller (UPC) bus option supports a direct interface to a 68000 executive processor. Thus, the MK68200 can be used as a cost-effective, intelligent peripheral controller in 68000 systems. The UPC version's direct bus interface to the 68000 makes the MK68200 particularly well-suited for performing many intelligent I/O functions in a 68000 system. For example, since the MK68200 includes both a serial channel and an external bus capable of performing DMA transfers, it can be programmed to act as serial protocol controller with DMA capability, as shown in Figure 1.

Table 1 summarizes the specific MK68200 device types that are discussed in this data sheet. A complete guide to the part numbering scheme used throughout this document may be found in the Ordering Information section. All MK68200 devices retain most of the I/O features when they are used in the expanded bus mode; however, 24 pins of parallel I/O are sacrificed when this mode is used. When the expanded bus mode is selected, the MK68201/XX generates UPC (68000-compatible) control signals, while the MK68211/XX generates GP control signals. Also available are 84-pin emulator versions of these devices that do not have on-chip ROM, but instead have additional pins to support a second complete address/data bus to access off-chip ROM, RAM, EPROM, or I/O devices. This bus is referred to as the private bus and is not bonded out on 48-pin versions.

For additional information on the MK68200, refer to the MK68200 Principles of Operation Manual, publication number 4420399.

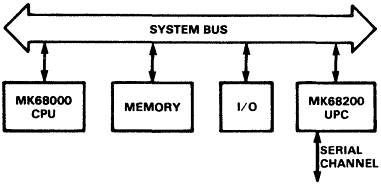


Figure 1. Serial DMA Controller

Device Type	Expanded Bus Version	ROM (Bytes)	RAM (Bytes)	PKG.
MK68201/04	UPC	0	256	48-pin DIP
MK68201/44	UPC	4K	256	48-pin DIP
MK68E201/04	UPC	0	256	84-pin LCC
MK68211/04	GP	0	256	48-pin DIP, 52-pin PLCC
MK68211/44	GP	4K	256	48-pin DIP, 52-pin PLCC
MK68E211/04	GP	0	256	84-pin LCC
MK68E221/0C	UPC/GP	0	1024	84-pin LCC, PGA

Table 1. Device Type Summary

SINGLE-CHIP DESCRIPTION

Figure 2 illustrates the functions of specific pins for an MK68201 or MK68211, operating in a single-chip mode. When the device is operating in one of the expanded bus modes, the pins on Port 0 become the multiplexed address/data bus, and the upper half of Port 1 becomes the control signals (GP or UPC) for the bus. The following description applies to the pins only when the device is used in the non-expanded or single-chip mode. Descriptions of the pin functions for the expanded bus modes are in the Expanded Bus Operation section of this data sheet.

V_{CC}, GND
(Power, Ground)
Power Supply pins.
(single +5 V)

$\overline{\text{RESET}}$
Input, active low. $\overline{\text{RESET}}$ input overrides ongoing execution (including interrupts) and resets the chip to its initial power-up condition. $\overline{\text{RESET}}$ cannot be masked.

CLKOUT
(Clock Output)
Output. CLKOUT will output the instruction clock rate, which is one-half of the frequency provided on CLK1 and CLK2.

CLK1, CLK2
(Time base Inputs)
Inputs. CLK1 and CLK2 may be connected to a crystal, or CLK1 may be connected to an external TTL-compatible oscillator while CLK2 is left floating. The instruction clock rate is one-half of the frequency provided on CLK1 and CLK2.

$\overline{\text{NMI}}$
(Non-Maskable Interrupt)
Input, active low, negative edge triggered. The NMI request line has a higher priority than all of the maskable interrupts. NMI is always enabled regardless of the state of the L1E (Level 1 Interrupt Enable) bit in the Status Register.

MODE
Input. The MODE pin is used to configure the MK68200 on power-up and reset to one of the following states:

Mode Pin	
V_{CC}	- No expansion (single chip mode)
GND	- Partial Expansion
CLKOUT	- Full Expansion

P0-0 - P0-15
(Port 0)
Input/Output. Each bit in Port 0 may be individually programmed for general purpose input or output. Port 0 also has several handshaking modes to allow parallel, asynchronous communication with other devices. The high and low bytes may be programmed individually or jointly to be inputs, outputs, or bidirectional.

P1-0 - P1-15
(Port 1)
Input/Output. Each of the 16 bits in Port 1 may be individually programmed for input or output. Additionally, the lowest seven bits of Port 1 may be programmed to serve specific alternate functions, as listed below.

P1-6/XI2
(External Interrupt 2)
Input, rising or falling edge triggered. The programmer may select the rising or falling edge as active for XI2.

P1-5/XI1
(External Interrupt 1)
Input, fixed falling edge triggered. The XI1 interrupt may be used to interrupt the MK68200 on the falling edge of an input pulse.

P1-4/XI0
(External Interrupt 0)
Input, low level triggered. The XI0 interrupt input is level-triggered (unlike XI1, XI2). It may be used to produce an internally vectored interrupt or to cause an external fetch of an interrupt vector number when the MK68200 is used in an expanded mode with the GP bus.

P1-3/SI
(Serial Input)
Input, active high. SI is used to input receive serial data when the receiver is enabled.

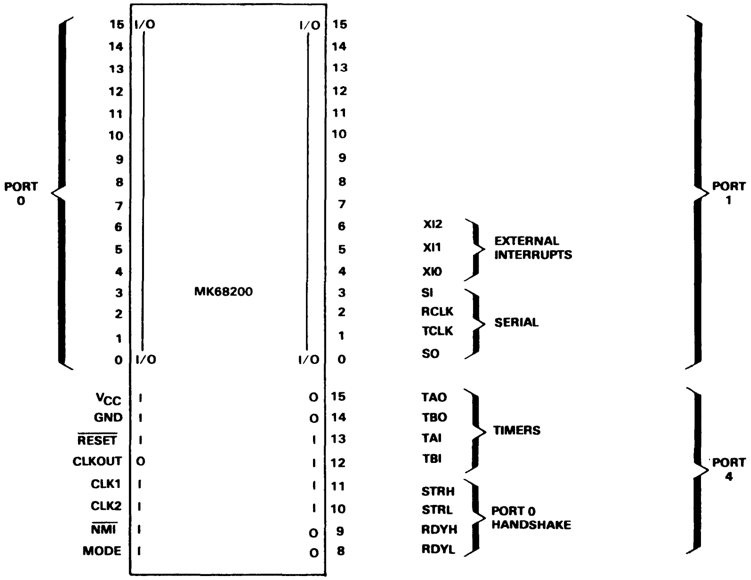


Figure 2. MK68200 Logical Pinout, Single-Chip Mode

P1-2/RCLK
(Receive Clock)
Input/Output, active high. Depending on the mode programmed, RCLK can be used by the serial port as either an input or an output pin. When used as an input pin, RCLK provides the receive clock and/or the transmit clock. When RCLK is not providing the transmit or receive clock, it can be used as an output for Timer C. In this mode, the receive clock is being provided by Timer C.

P1-1/TCLK
(Transmit-Clock)
Input/Output, active high. Depending on the mode programmed, TCLK can be used by the serial port as either an input or an output pin. When used as an input pin, TCLK provides the transmit clock. When TCLK is not providing the transmit clock, it can be used as an output for Timer C. In this mode, the transmit clock is being provided by either Timer C or RCLK.

P1-1/SO
(Serial Output)
Output, active high. SO is used to output transmit serial data when the transmitter is enabled.

P4-8 - P4-15
(Port 4)
Inputs and Outputs. P4-8, P4-9, P4-14, and P4-15 may be used as general purpose outputs, and P4-10, P4-11, P4-12, and P4-13 may be used as general purpose inputs. Interrupts may be generated on the positive transitions on P4-10 and P4-11. Depending on the mode selected, interrupts may be generated on the positive or negative transitions on P4-12, or they may be generated on the positive, negative, or combined transitions on P4-13. Additionally, these bits may be programmed to serve specific alternate functions, as listed below.

P4-15/TAO
(Timer A Output)
Output. TAO may be programmed for special functions in the interval, event, and pulse modes for Timer A. In the interval mode, TAO's state is determined by the Timer A latch (high or low) that is currently active. That is, if the counter is using the high latch for comparison, TAO is high. If the counter is using the low latch for comparison, TAO is low. In the event mode, TAO is initialized to a "1" state and toggles each time the counter matches the Timer A high latch. In the pulse/period modes, TAO is initiated to a "1" state and toggles on positive transitions on TAI.

P4-14/TBO
(Timer B Ouput)
Output. TBO may be programmed for special functions in the interval and one-shot modes for Timer B. In the interval mode, TBO is initialized to a "1" state and toggles each time the counter matches the Timer B latch value. In the one-shot modes, TBO is initialized to a "1" state, and the counter begins counting in response to the occurrence of an active edge on TBI. TBO will not go low until the counter matches the value loaded into the Timer B latch.

P4-13/TAI
(Timer A Input)
Input, positive and/or negative edge triggered. TAI may be programmed for special functions in the event mode or pulse/period modes for Timer A. In the event mode, the counter is incremented on each active transition (positive or negative edge programmable) on TAI. In the pulse/period modes, the counter measures the time during which the signal on TAI remains high and low.

P4-12/TBI
(Timer B Input)
Input, positive or negative edge triggered. TBI may be programmed for special functions for the Timer B one-shot modes. In the one-shot modes, TBI acts as a trigger input.

P4-11/STRH, P4-10/STRL
(Strobe High Byte, Strobe Low Byte)
Input, active high. STRH and STRL are both used for input, output, and bidirectional handshaking on Port 0.

1) Output mode: The positive edge of this strobe is issued by the peripheral to acknowledge the receipt of data made available by the MK68200.

2) Input mode: The strobe is issued by the peripheral to load data from the peripheral into the Port 0 input register. Data is latched into the MK68200 on the negative edge of this signal.

3) Bidirectional mode: When the STRH signal is active, data from the Port 0 output register is gated onto the Port 0 bidirectional data bus.

The negative edge of STRH acknowledges the receipt of the output data. The negative edge of the signal applied to the STRL signal is used to latch input data into Port 0.

P4-9/RDYH, P4-8/RDYL
(Ready High Byte, Ready Low Byte)
Output, active high. RDYH and RHYL are used for input, output, and bidirectional handshaking on Port 0.

1) Output mode: The ready signal goes active to indicate that the Port 0 output register has been loaded, and the peripheral data is stable and ready for transfer to the peripheral device.

2) Input mode: The ready signal is active when the Port 0 input register is empty and is ready to accept data from the peripheral device.

3) Bidirectional mode: The RDYH signal is active when data is available in Port 0 output register for transfer to the peripheral device. In this mode, data is not placed on the Port 0 data bus unless STRH is active. The RDYL signal is active when the Port 0 input register is empty and is ready to accept data from the peripheral device.

PROCESSOR ARCHITECTURE

The MK68200 microcomputer contains an advanced processor architecture, combining the best properties of both 8- and 16-bit processors. A large majority of instructions operate on either byte or word operands.

REGISTERS

The MK68200 register set includes three system registers, six address registers, and eight data registers. The three 16-bit system registers (Figure 3) include a Program Counter, a Status Register, and a Stack Pointer. The six address registers may be used either for 16-bit data or for memory addressing. The eight 16-bit data registers are used for data and may be referenced as sixteen 8-bit registers, providing great flexibility in register allocation.

ADDRESSING

The MK68200 directly addresses a 64K byte memory space, which is organized as 32K 16-bit words. The memory is byte-addressable, but most transfers occur 16 bits at a time, for increased performance over 8-bit microcomputers. All input/output is memory-mapped, and the on-chip I/O is situated in the top 1K bytes of the address space. In the single-chip mode, all resources including ROM, RAM, and I/O, are accessed via an internal or private bus. The memory map, which is accessed by this bus in the single-chip mode, is depicted in Figure 4. Note on-chip RAM always begins at $FBFF and extends downward. ROM always begins at zero and extends upward.

Nine addressing modes provide ease of access to data in the MK68200, as depicted in Table 2. The four register indirect forms utilize the address registers and the Stack Pointer and support many common data structures such as arrays, stacks, queues, and linked lists. I/O Port addressing is a short form addressing mode for the first 16 words of the I/O port space and allows most instructions to access the most often referenced I/O ports in just one word. Many microcomputer applications are I/O intensive and short, fast addressing of I/O has a significant impact on performance.

Register
Register Indirect
Register Indirect with Post-increment
Register Indirect with Pre-decrement
Register Indirect with Displacement
Program Counter Relative
Memory Abso' ite
Immediate
I/O Port

Table 2. Addressing Modes

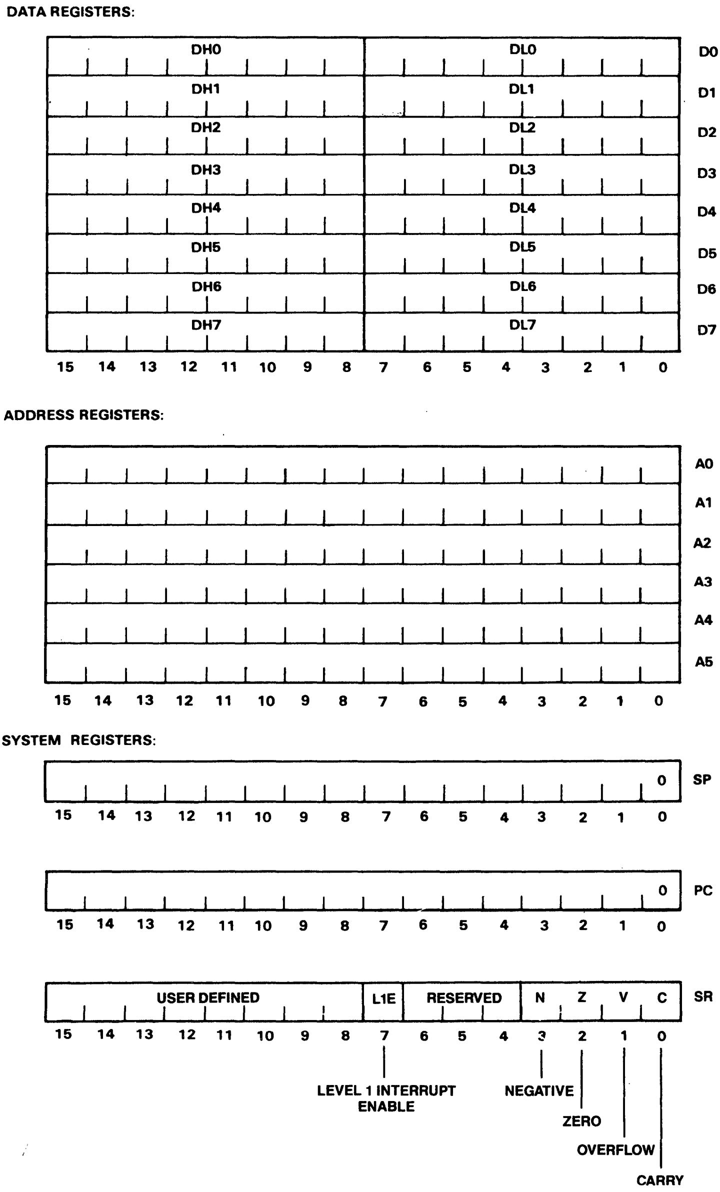


Figure 3. Register Set

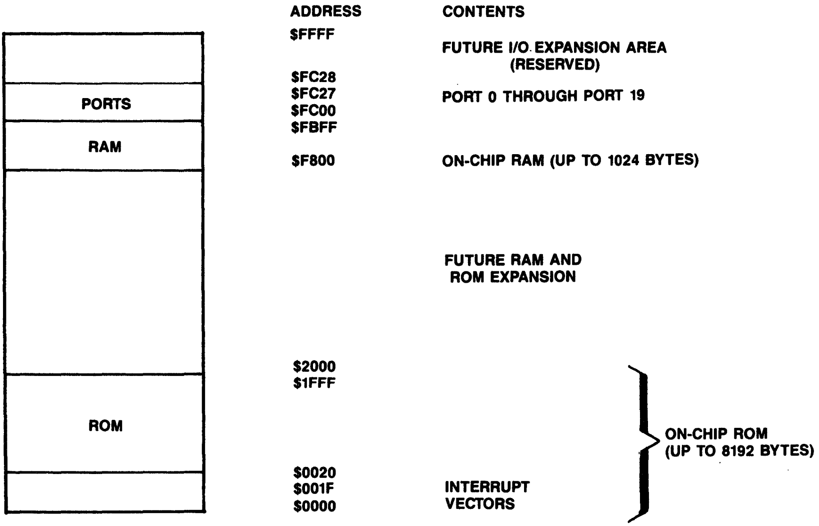

ADDRESS
CONTENTS
$FFFF
FUTURE I/O EXPANSION AREA (RESERVED)
$FC28
$FC27
PORTS
PORT 0 THROUGH PORT 19
$FC00
$FBFF
RAM
$F800
ON-CHIP RAM (UP TO 1024 BYTES)
FUTURE RAM AND ROM EXPANSION
$2000
$1FFF
ROM
ON-CHIP ROM (UP TO 8192 BYTES)
$0020
$001F
INTERRUPT VECTORS
$0000

Figure 4. Addressing Space For Single-Chip Configuration

INSTRUCTION SET

The MK68200 instruction set has been designed with regularity and ease of programming in mind. In addition, instructions have been encoded to minimize code space, a feature which is especially important in single-chip microcomputers. Small code space is related to execution speed, and most instructions execute in either three or six instruction clock periods. (An instruction clock period is equal to 167 ns with a 6 MHz instruction clock). See Table 3.

In addition to operations on bytes and words, the MK68200 has rapid bit manipulation instructions that can operate on registers, memory, and ports. The bit to be affected may be an immediate operand of the instruction, or it may be dynamically specified in a register. Operations available include bit set, clear, test, change, and exchange; and all bit operations perform a bit test as well. Since each instruction is indivisible, this provides the necessary test-and-set function for the implementation of semaphores.

Instruction Type	Clock Periods	Execution Time with 6 MHz Clock (μs)
Move Register-to-register	3	0.5
Add Register-to-register (binary or BCD)	3	0.5
Move Memory-to-register	6	1.0
Add Register-to-memory	9	1.5
Multiply (16 × 16)	21	3.5
Divide (32/16)	23	3.84
Move Multiple (save or restore all registers)	55	9.2

Table 3. Instruction Execution Times

The MOVE group of instructions has the most extensive capabilities. A wide variety of addressing mode combinations is supported including memory-to-memory transfers. A special move multiple is included to save and restore a specified portion of the registers rapidly.

In total, the MK68200 instruction set provides a programming environment, similar to the 68000, which has been optimized for the needs of the single-chip microcomputer marketplace. A summary of the instruction set is provided in Table 4.

INSTRUCTION	DESCRIPTION
ADD	ADD
ADD.B	ADD BYTE
ADDC	ADD WITH CARRY
ADDC.B	ADD WITH CARRY BYTE
AND	LOGICAL AND
AND.B	LOGICAL AND BYTE
ASL	ARITHMETIC SHIFT LEFT
ASL.B	ARITHMETIC SHIFT LEFT BYTE
ASR	ARITHMETIC SHIFT RIGHT
ASR.B	ARITHMETIC SHIFT RIGHT BYTE
BCHG	BIT CHANGE
BCLR	BIT CLEAR
BEXG	BIT EXCHANGE
BSET	BIT SET
BTST	BIT TEST
CALLA	CALL ABSOLUTE
CALLR	CALL RELATIVE
CLR	CLEAR
CLR.B	CLEAR BYTE
CMP	COMPARE
CMP.B	COMPARE BYTE
DADD	DECIMAL ADD
DADD.B	DECIMAL ADD BYTE
DADDC	DECIMAL ADD WITH CARRY
DADDC.B	DECIMAL ADD WITH CARRY BYTE
DI	DISABLE INTERRUPTS
DIVU	DIVIDE UNSIGNED
DJNZ	DECREMENT COUNT AND JUMP IF NON-ZERO
DJNZ.B	DECREMENT COUNT BYTE AND JUMP IF NON-ZERO
DNEG	DECIMAL NEGATE
DNEG.B	DECIMAL NEGATE BYTE
DNEGC	DECIMAL NEGATE WITH CARRY
DNEGC.B	DECIMAL NEGATE WITH CARRY BYTE
DSUB	DECIMAL SUBTRACT
DSUB.B	DECIMAL SUBRTRACT BYTE
DSUBC	DECIMAL SUBTRACT WITH CARRY
DSUBC.B	DECIMAL SUBTRACT WITH CARRY BYTE
EI	ENABLE INTERRUPTS
EOR	EXCLUSIVE OR
EOR.B	EXCLUSIVE OR BYTE
EXG	EXCHANGE
EXG.B	EXCHANGE BYTE
EXT	EXTEND SIGN
HALT	HALT
JMPA	JUMP ABSOLUTE
JMPR	JUMP RELATIVE
LIBA	LOAD INDEXED BYTE ADDRESS
LIWA	LOAD INDEXED WORD ADDRESSED
LSR	LOGICAL SHIFT RIGHT
LSR.B	LOGICAL SHIFT RIGHT BYTE
MOVE	MOVE
MOVE.B	MOVE BYTE
MOVEM	MOVE MULTIPLE REGISTERS
MOVEM.B	MOVE MULTIPLE REGISTERS BYTE
MULS	MULTIPLY SIGNED
MULU	MULTIPLY UNSIGNED
NEG	NEGATE
NEG.B	NEGATE BYTE
NEGC	NEGATE WITH CARRY
NEGC.B	NEGATE WITH CARRY BYTE
NOP	NO OPERATION
NOT	ONE'S COMPLEMENT
NOT.B	ONE'S COMPLEMENT BYTE
OR	LOGICAL OR
OR.B	LOGICAL OR BYTE
POP	POP
POPM	POP MULTIPLE REGISTERS
PUSH	PUSH
PUSHM	PUSH MULTIPLE REGISTERS
RET	RETURN FROM SUBROUTINE
RETI	RETURN FROM INTERRUPT
ROL	ROTATE LEFT
ROL.B	ROTATE LEFT BYTE
ROLC	ROTATE LEFT THROUGH CARRY
ROLC.B	ROTATE LEFT THROUGH CARRY BYTE
ROR	ROTATE BYTE
ROR.B	ROTATE RIGHT BYTE
RORC	ROTATE RIGHT THROUGH CARRY
RORC.B	ROTATE RIGHT THROUGH CARRY BYTE
SUB	SUBTRACT
SUB.B	SUBTRACT BYTE
SUBC	SUBTRACT WITH CARRY
SUBC.B	SUBTRACT WITH CARRY BYTE
TEST	TEST
TEST.B	TEST BYTE
TESTN	TEST NOT
TESTN.B	TEST NOT BYTE

Table 4. Instruction Set Summary

INPUT/OUTPUT ARCHITECTURE

The I/O capabilities of the MK68200 are extensive, encompassing timers, a serial channel, parallel I/O, and an interrupt controller. All of these devices are accessible to the programmer as ports within the top 1K bytes of the address space, and the most commonly accessed ports may be accessed with the short port addressing mode. A description of these ports is given in Table 5.

In total, 40 pins of the 48 are used for I/O, and their functions are highly programmable by the user. In particular, many pins can perform multiple functions, and the programmer selects which ones are to be used. For example, TAI may be used as an input for Timer A, an interrupt source, or a general purpose input pin. The interrupt source may be selected simultaneously with either of the other functions.

PORT	ADDRESS	READ/WRITE	BYTE- ADDRESSABLE	FUNCTION
0	$FC00	READ/WRITE	YES	16 EXTERNAL I/O PINS OR ADDRESS/DATA BUS
1	$FC02	READ/WRITE	YES	16 EXTERNAL I/O PINS (INCLUDING INTERRUPT, SERIAL I/O PINS, AND BUS CONTROL)
2	$FC04	—	—	(RESERVED)
3	$FC06	LOW BYTE: READ/WRITE HIGH BYTE: READ	YES	SERIAL TRANSMIT (LOW BYTE) AND RECEIVE (HIGH BYTE) BUFFER
4	$FC08	INPUTS: READ ONLY OUTPUTS: READ/WRITE	NO	8 EXTERNAL I/O PINS (TIMER CONTROL AND PORT 0 HANDSHAKE CONTROL)
5	$FC0A	—	—	(RESERVED)
6	$FC0C	—	—	(RESERVED)
7	$FC0E	READ/WRITE	NO	INTERRUPT LATCH REGISTER
8	$FC10	READ/WRITE	NO	INTERRUPT MASK REGISTER
9	$FC12	STATUS: READ ONLY CONTROL: READ/WRITE	NO	SERIAL I/O RECEIVE CONTROL AND STATUS
10	$FC14	STATUS: READ ONLY CONTROL: READ/WRITE	NO	SERIAL I/O TRANSMIT CONTROL AND STATUS
11	$FC16	READ GETS COUNTER WRITE GOES TO LATCH	NO	TIMER B LATCH
12	$FC18	READ GETS COUNTER OR LATCH WRITE GOES TO LATCH	NO	TIMER A, LOW LATCH
13	$FC1A	READ GETS COUNTER OR LATCH WRITE GOES TO LATCH	NO	TIMER A, HIGH LATCH
14	$FC1C	READ/WRITE	NO	TIMER CONTROL, INTERRUPT EDGE SELECT
15	$FC1E	READ/WRITE	NO	PORT 0 HANDSHAKE MODE BITS, FAST/ STANDARD, BUS LOCK, BUS SEGMENT BITS
16	$FC20	READ/WRITE	NO	PORT 0 DIRECTION CONTROL (DDR0)
17	$FC22	READ/WRITE	NO	PORT 1 DIRECTION CONTROL (DDR1)
18	$FC24	READ/WRITE	NO	SERIAL I/O MODE AND SYNC REGISTER
19	$FC26	READ GETS COUNTER WRITE GOES TO LATCH AND COUNTER	NO	TIMER C LATCH

Table 5. Port Descriptions

TIMERS

The MK68200 includes three on-chip timers, each with unique features. They are denoted Timer A, Timer B, and Timer C. All three timers are a full 16 bits in width, and count at the instruction clock rate of the MK68200 processor. Thus, this rate provides a resolution equal to the instruction clock period (tc) of the MK68200. The maximum count interval is equal to tc $*2^{16}$. For a 6 MHz MK68200, a 167 nanosecond clock is provided with a maximum count interval of 10.945 milliseconds. Each timer has the capability to interrupt the processor when it matches a predetermined value stored in an associated latch.

Timer A is capable of operating in interval, event, or two pulse/period modes. There is one 16-bit counter and two 16-bit latches (high and low) associated with Timer A. Once Timer A is initialized in the interval mode, the counter is reset, then increments at the instruction clock rate until the value loaded into the high latch is reached. The counter is then reset, increments until the low latch value is reached, and the cycle is repeated. In the event mode, the counter is incremented for every active edge on TAI (programmable as positive or negative) until the value in the high latch is reached. The counter is then reset, and the cycle repeats. In the pulse/period modes, the times are measured during which the pulse applied stays high and low. The counter is reset on the occurrence of any transition on TAI, and increments at the instruction clock rate until the occurrence of the next transition. The value in the counter at the end of the high level or low level time is loaded into the appropriate latch. Interrupts may be generated each time the counter reaches the high latch or low latch value in the interval mode or when the counter reaches the high latch in the event mode. Also, an interrupt is generated whenever the counter overflows. See the Pin Description section of this data sheet for TAI and TAO functions in the various Timer A modes.

Timer B is capable of operating in interval and one-shot modes. There is one 16-bit counter and one 16-bit latch associated with Timer B. In the interval mode, the counter is initially reset and incremented at the instruction clock rate until the value in the latch is reached. The counter is then reset, and the cycle repeats. In the one-shot modes, the counter begins incrementing in response to an active transition (programmable as positive or negative) on TBI. The counter is reset when the value in the Timer B latch is reached. In the retriggerable one-shot mode, active transitions on TBI always cause the counter to reset and begin incrementing. In the non-retriggerable one-shot mode, active transitions on TBI have no effect until the counter reaches the latch value. Interrupts may be generated each time the counter reaches the latch value. See the Pin Description section of this data sheet for TBI and TBO functions in the various Timer B modes.

Timer C has a 16-bit down counter and latch associated with it and operates only in the interval mode. The output of Timer C toggles each time the counter value rolls over from 0 to the latch value and may be used to internally supply the baud rate clock for the serial port. Also, an interrupt may be generated each time the counter rolls over to the latch value. Timer C may be output on the TCLK pin (P1-3), depending on the mode programmed.

Timer	Modes
A	Interval
A	Event
A	Pulse Width and Period Measurement
B	Interval
B	Retriggerable One-shot
B	Non-retriggerable One-shot
C	Interval
C	Baud Rate Generation

Table 6. Timer Modes

SERIAL CHANNEL

The serial channel on the MK68200 (Figure 5) is a full-duplex USART with double buffering on both transmit and receive. Word length, parity, stop bits, and modes are fully programmable. The asynchronous mode supports bit rates up to 375 Kbps, and the byte synchronous mode operates up to 1.5 Mbps. Either internal or external clocks may be used.

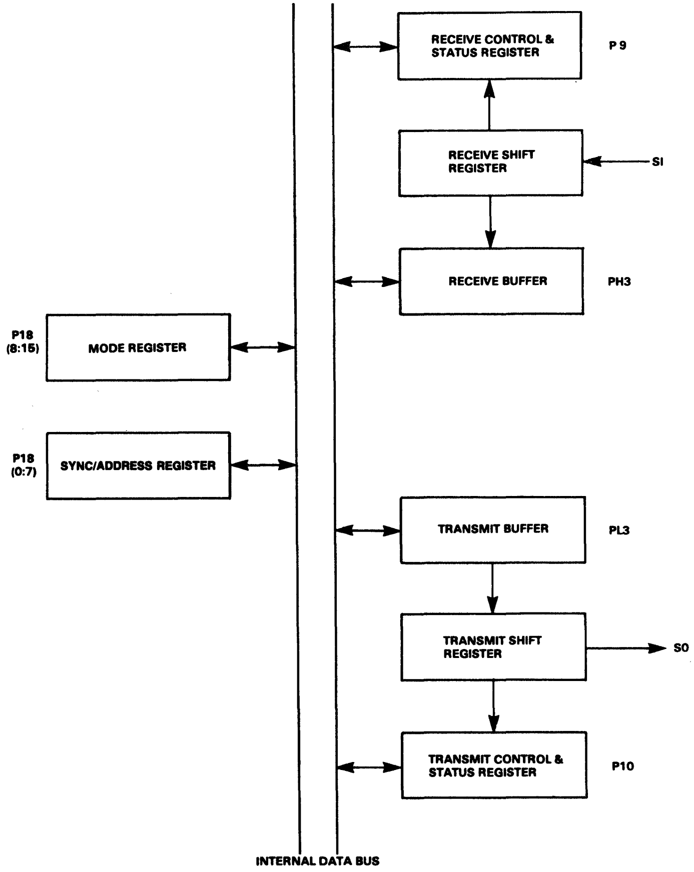


Figure 5. Serial Channel

In addition to the typical USART functions, the serial channel can operate in a special wake-up mode with a wake-up bit appended to each data word, as illustrated in Figure 6. This wake-up bit is used to differentiate normal data words and special address words. The receiver can be programmed to receive only address words or only address words with a specific data value. In this way, the processor can be interrupted only when it receives its particular address and can then change mode to receive the following data words. Wake-up capability is especially useful when several MK68200 microcomputers are interconnected on one serial link.

START†	DATA LSB MSB	PARITY (OPTIONAL)	WAKE-UP (OPTIONAL)	STOP†

†USED IN ASYNCHRONOUS MODE ONLY

Figure 6. Serial Frame Format

PARALLEL I/O

Two 16-bit ports, P0 and P1, may be used for parallel I/O. If individual bits are desired, each of the 32 bits may be separately defined as input or output. Bits may be grouped to provide the exact data widths desired. Port 0 has the additional capability of operating under the control of external handshaking signals. Eight- or sixteen-bit sections of P0 may be individually controlled as input, output, or bidirectional I/O. Two pairs of Ready and Strobe signals, which are available as programmable options on Port 4, provide the necessary control.

INTERRUPT CONTROLLER

The MK68200 interrupt controller provides rapid service of up to 15 interrupt sources, each with a unique internal vector. The lowest 16 words of the address space contain the starting addresses of the service routines of each potential interrupt source and reset, as shown in Figure 7.

Interrupt sources and RESET are prioritized in the order shown in Figure 7, with RESET having the highest priority. NMI is the only non-maskable interrupt. All of

VECTOR NUMBER	NAME	MNEMONIC	VECTOR LOCATION	
0	RESET	RESET	0000	
1	NON-MASKABLE INTERRUPT	NMI	0002	LEVEL 2
2	SPARE	SPARE	0004	LEVEL 1
3	EXTERNAL INTERRUPT 2	XI2	0006	
4	STROBE LOW	STRL	0008	
5	TIMER A OUTPUT	TAO	000A	
6	TIMER A INPUT	TAI	000C	
7	STROBE HIGH	STRH	000E	
8	RECEIVE SPECIAL CONDITION	RSC	0010	
9	RECEIVE NORMAL	RN	0012	
A	EXTERNAL INTERRUPT 1	XI1	0014	
B	TIMER B OUTPUT	TBO	0016	
C	TIMER B INPUT	TBI	0018	
D	EXTERNAL INTERRUPT 0	XI0	001A	
E	TRANSMIT	XMT	001C	
F	TIMER C	TC	001E	

Figure 7. Interrupt and Reset Vectors

the other sources share an interrupt enable bit in the processor Status Register. This bit is automatically cleared whenever an interrupt is acknowledged. Also, each of these sources has a corresponding individual mask bit. This feature allows selective masking of particular interrupts, including the ability to choose (with minimal software overhead) any priority scheme desired. In fact, 15 levels of nested priority may be programmed.

EXPANDED BUS OPERATION

When it is necessary to expand beyond the on-chip complement of RAM, ROM, or I/O, or when operation in a parallel multiprocessing system is desired, the MK68200 may be placed in an external bus mode. The MODE pin is used to select the expansion capability on reset. The MODE pin has three states, which select fully expanded external bus, partially expanded external bus, or no expanded bus (single-chip configuration). The MK68200 may also be reconfigured dynamically through software. In an expansion mode, Port 0 becomes the 16-bit multiplexed, address/data bus, and eight bits from Port 1 become control signals which handle data transfer and bus arbitration. Sixteen lines are still available for I/O functions, including eight lines from Port 1 and all eight lines of Port 4.

As shown in Figure 8 , two different control bus versions are available: a Universal Peripheral Controller (UPC),

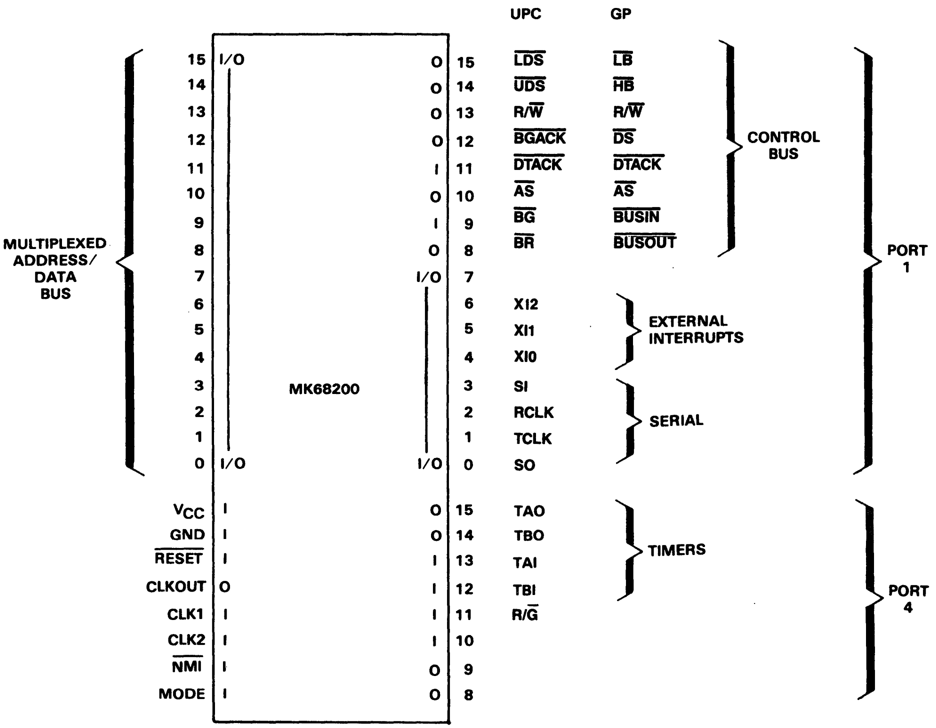

UPC
GP
MULTIPLEXED ADDRESS/ DATA BUS
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
I/O
I/O
MK68200
O 15 $\overline{LDS}$ $\overline{LB}$
O 14 $\overline{UDS}$ $\overline{HB}$
O 13 R/$\overline{W}$ R/$\overline{W}$
O 12 $\overline{BGACK}$ $\overline{DS}$
I 11 $\overline{DTACK}$ $\overline{DTACK}$
O 10 $\overline{AS}$ $\overline{AS}$
I 9 $\overline{BG}$ $\overline{BUSIN}$
O 8 $\overline{BR}$ $\overline{BUSOUT}$
CONTROL BUS
I/O 7
6 XI2
5 XI1
4 XI0
EXTERNAL INTERRUPTS
3 SI
2 RCLK
1 TCLK
I/O 0 SO
SERIAL
PORT 1
V_{CC} I
GND I
$\overline{RESET}$ I
CLKOUT O
CLK1 I
CLK2 I
$\overline{NMI}$ I
MODE I
O 15 TAO
O 14 TBO
I 13 TAI
I 12 TBI
I 11 R/$\overline{G}$
I 10
O 9
O 8
TIMERS
PORT 4

Figure 8. MK68200 Logical Pinout Expanded Bus

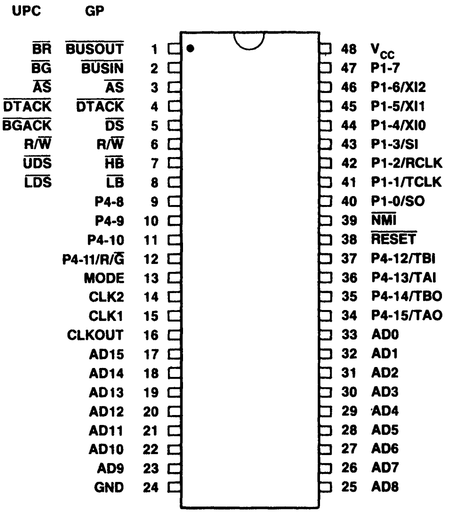


Figure 9:. MK68200 Expanded Bus

which generates 68000-compatible signals, and a General Purpose (GP) bus, which can be used to interface to a wide variety of existing microprocessor buses. With the selection of an expanded bus mode, the MK68200 can act either as a general purpose CPU chip (bus grant device) or as an intelligent peripheral I/O controller to a host CPU (bus request device). These two system configurations are illustrated in Figures 10 and 11.

With the GP bus option, the user may configure the MK68200 in either of the two ways shown in Figures 10 and 11. As a host CPU (Figure 10), the MK68200 bus arbitration logic causes the device to act as the system bus grantor. In other words, the MK68200 would normally have control of the system bus and would grant its use to DMA devices or peripheral CPUs. Alternately, the MK68200 may be configured as a peripheral CPU (Figure 11) that must issue a request to the bus grant device before being allowed to use the system bus. The selection of one of these two configurations is accomplished by the P4-11 pin at reset time. During reset, P4-11 serves as the R/$\overline{G}$ input (0=bus grantor, 1 = bus requestor). Following reset and at all times during program execution, P4-11 may be used as a general purpose input pin.

With the GP bus operating in the host CPU configuration, the MK68200 may be used to interface with external memory and I/O devices in a manner that is analogous to any general purpose microprocessor. Additionally, the MK68200 retains its on-chip RAM and I/O resources, with on-chip ROM as an option.

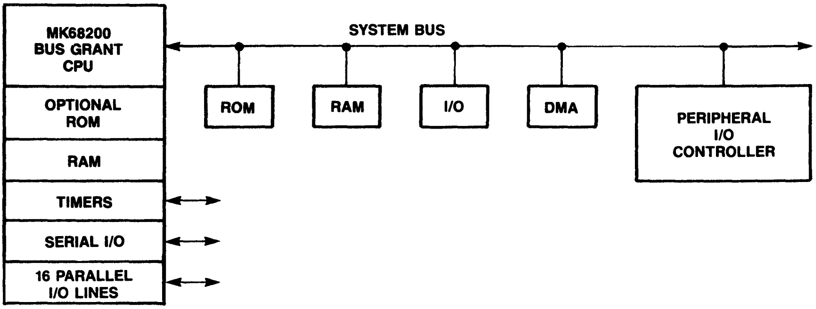


Figure 10. Host CPU Hardware Configuration

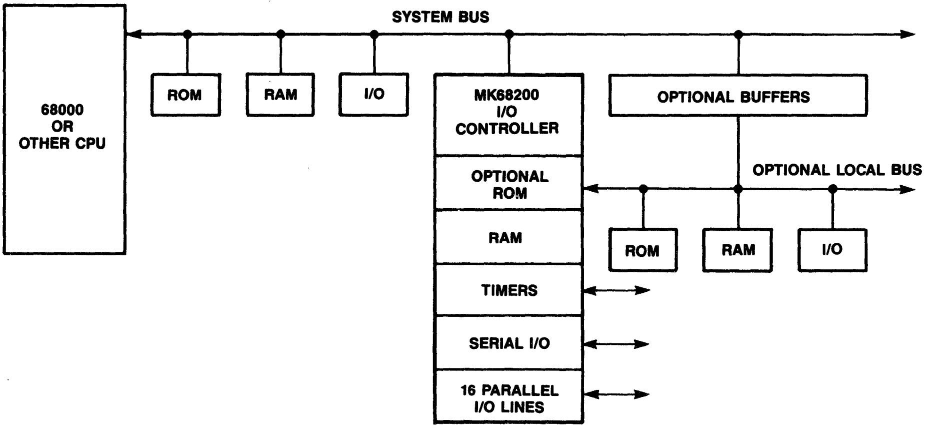


Figure 11. Peripheral I/O Controller Configuration

depending on the expansion configuration selected. $\overline{\text{BUSIN}}$ and $\overline{\text{BUSOUT}}$ are used to perform the bus arbitration handshake function, where $\overline{\text{BUSIN}}$ acts as the bus request input and $\overline{\text{BUSOUT}}$ as the bus grant output.

In the full expansion configuration, any on-chip ROM is disabled, and program memory starting at location $0000 is located off-chip and is addressed via the expanded bus, as shown in Figure 12. In effect, the internal bus from locations $0000-$FAFF is mapped onto the external bus. In the partially expanded configuration (Figure 13), on-chip ROM may be accessed on the internal bus. To gain greater addressability in the partial expansion configuration, a scheme is implemented to allow access of a full 64K-byte address space in four segments on the expanded system bus through the 16K byte "window" on the internal bus. Basically, the most significant two bits of address on the expanded bus are replaced with two user-defined segment bits available to the programmer in an internal I/O control port location.

As a peripheral I/O controller, the MK68200 operates as a bus requestor that gains mastership of the system bus from the bus grant CPU. The GP bus version may be selected to implement this system configuration in cases where an interface to a general purpose CPU is desired. In this case, the $\overline{\text{BUSIN}}$ and $\overline{\text{BUSOUT}}$ lines are again used to perform the bus arbitration handshake function, where $\overline{\text{BUSOUT}}$ now acts as bus request out put, and $\overline{\text{BUSIN}}$ acts as bus grant input. In this configuration, the MK68200 can conceivably act as a complete peripheral I/O control subsystem on a single chip, with 16 lines of I/O and its on-chip ROM, RAM, timers, and serial I/O performing the necessary interface to the I/O device. The UPC bus version provides the peripheral I/O control function with a direct interface to a 68000 bus grant CPU. Note that the UPC bus version can operate only as a bus request device. Once the MK68200 has gained mastership of the system bus via the 68000 bus arbitration handshake lines ($\overline{\text{BR}}$, $\overline{\text{BG}}$, and $\overline{\text{BGACK}}$), it may proceed to perform DMA transfers and communicate with system memory or other I/O devices in the system.

As in the case of the GP bus grant configuration, the portion of the internal (or private) bus address space that is mapped onto the expanded bus when the part is operating as either a GP or a UPC bus request device is determined by the expansion configuration selected. In the partial expansion bus requestor case, the

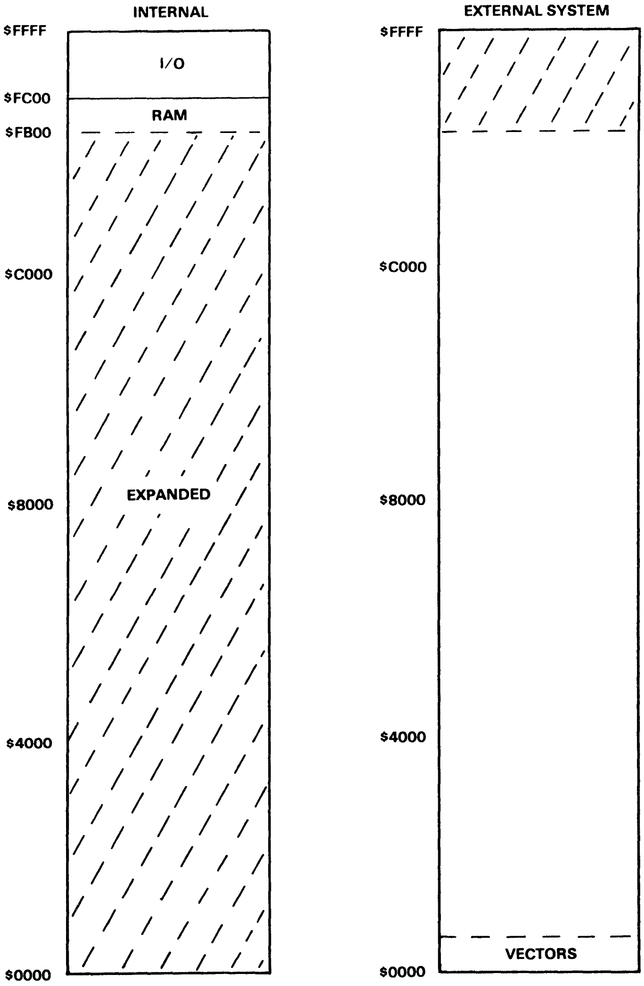


Figure 12. Full Expansion Bus Grantor Memory Map
(256 byte RAM version shown)

resulting memory map is identical to that shown for the GP bus grant configuration in Figure 13. During the time the MK68200 is executing its programs from ROM and accessing internal RAM and I/O resources, the expanded bus is held in a tri-state condition. The bus arbitration logic within the MK68200 monitors each memory reference to detect external bus addresses (referenced in segments via the 16K byte DMA window). Whenever such an external reference occurs, the logic automatically holds the processor in a wait state as it proceeds to obtain mastership of the bus. When use of the system bus is obtained, the processor is allowed to continue the reference. This procedure is transparent to the programmer. In case of successive external references, the expanded bus is retained until an internal reference is encountered.

Finally, if the on-chip resources are insufficient to perform the control task in the bus requestor configuration, the internal bus address range (excluding on-chip RAM, I/O) may be mapped onto an external local bus, which is physically the same as the system bus but logically separated with bus buffers. This is the full expansion bus requestor configuration. The memory map for this configuration is shown in Figure 14 The bus arbitration sequence is performed only when the system bus is referenced through the DMA window. In this manner, the I/O subsystem is isolated from the host CPU.

When operating as a bus request device, it is possible to retain the external bus for an indefinite duration by using a bus lock feature. This will help facilitate the transfer of large blocks of data. Thus, the on-chip bus arbitration logic allows (with a minimum of hardware and software overhead) a maximum of concurrent processing in parallel, multiprocessing configurations. The bus lock feature may be used by the MK68200 in a bus grantor mode to keep any peripheral from gaining mastership of the bus.

In any of the GP expanded bus modes, the MK68200 may respond to peripheral devices on the expanded bus which generate an interrupt request on XI0. The MK68200 will obtain the XI0 interrupt vector number from the requesting peripheral on the bus during an interrupt acknowledge cycle. When responding to an interrupt on XI0, the MK68200 will wait for the bus arbitration logic to gain control of the bus and then asserts neither $\overline{HB}$ nor $\overline{LB}$ while asserting $\overline{AS}$ to signify that an interrupt acknowledge cycle is in progress.

Timing diagrams and design parameters for the read, write, and bus arbitration cycles are given in the AC Electrical Specifications section for both the GP and the UPC bus options. Bus timing for the interrupt acknowledge cycle is given for the GP device in the AC Electrical Specifications section. There is a user-programmable speed selection associated with the read and write cycles for both the UPC and GP mask option parts. A bit in an internal I/O port allows the user to select either the standard or the fast read/write cycle on the expanded bus. The standard bus cycle is four clock periods, while the fast bus cycle is three clock periods.

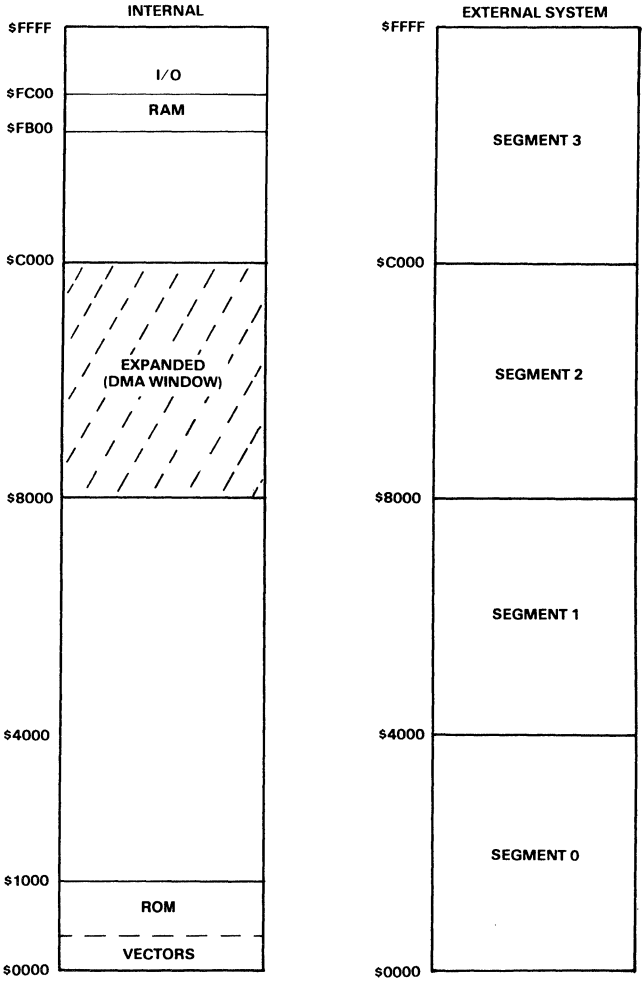


Figure 13. Partial Expansion Memory Map
(256 byte RAM, 4K byte ROM version shown)

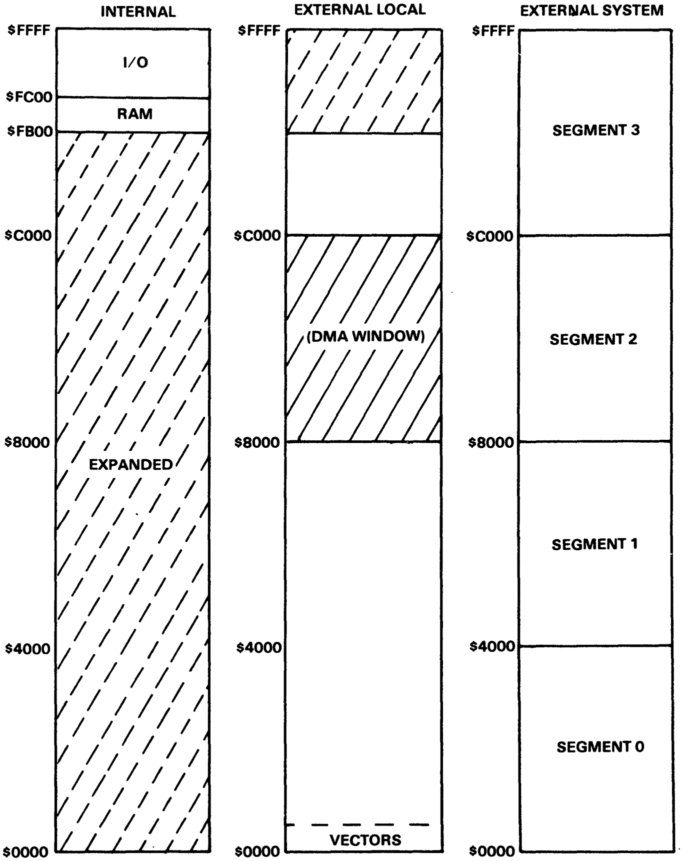


Figure 14. Full Expansion Bus Requestor Memory Map
(256 byte RAM version shown)

EXPANDED BUS SIGNALS (Common for GP and UPC Options)

R/$\overline{\text{W}}$
(Read/Write)
Output, active high and low. R/$\overline{\text{W}}$ determines whether a read or a write is being performed during the current bus cycle. It is stable for the entire bus operation. A high signal denotes a read, and a low signal denotes a write.

$\overline{\text{DTACK}}$
(Data Transfer Acknowledge)
Input, active low. When the addressed device has either placed the requested read data on the bus or taken the write data from the bus, $\overline{\text{DTACK}}$ should be brought low to signify completion. The data portion of the bus cycle will be extended indefinitely until this signal is asserted. For systems using the GP bus, in which no devices need wait states, $\overline{\text{DTACK}}$ may be strapped low.

$\overline{\text{AS}}$
(Address Strobe)
Output, active low. $\overline{\text{AS}}$ is used to signify that the address is stable on the multiplexed bus. $\overline{\text{AS}}$ is high at the beginning of each bus cycle, goes low after the address has stabilized, and returns to the high state near the end of the bus cycle.

UPC BUS SIGNALS

$\overline{\text{UDS}}$
(Upper Data Strobe)
Output, active low. $\overline{\text{UDS}}$ is used to signify the data portion of the bus cycle for the upper byte of the data bus. For read operations, $\overline{\text{UDS}}$ should be used by the external device to gate its most significant byte onto the multiplexed address/data bus. For writes, $\overline{\text{UDS}}$ signifies that the upper byte of the bus contains valid data to be written from the processor.

$\overline{\text{LDS}}$
(Lower Data Strobe)
Output, active low. $\overline{\text{LDS}}$ is used to signify the data portion of the bus cycle for the lower byte of the data bus. For read operations, $\overline{\text{LDS}}$ should be used by the external device to gate its least significant byte onto the multiplexed address/data bus. For writes, $\overline{\text{LDS}}$ signifies that the lower byte of the bus contains valid data to be written from the processor.

$\overline{\text{BR}}$
(Bus Request)
Output, active low, open drain. $\overline{\text{BR}}$ goes low when the MK68200 requires external bus master status.

$\overline{\text{BG}}$
(Bus Grant)
Input, active low. $\overline{\text{BG}}$ notifies that the MK68200 has been granted the external bus master status.

$\overline{\text{BGACK}}$
(Bus Grant Acknowledge)
Output, active low, open drain. The MK68200 will assert $\overline{\text{BGACK}}$ when it assumes mastership of the system bus.

GP BUS SIGNALS

P4-11 / R/$\overline{\text{G}}$
(Request/Grant)
During reset, P4-11 serves as the R/$\overline{\text{G}}$ input (0 = bus grantor, 1 = bus requestor). Following reset, and at all times during program execution, P4-11 may be used as a general purpose input pin.

$\overline{\text{DS}}$
(Data Strobe)
Output, active low. $\overline{\text{DS}}$ is used to signify the data portion of the bus cycle. For read operations, $\overline{\text{DS}}$ should be used by the external device to gate its contents onto the multiplexed address/data bus. For writes, $\overline{\text{DS}}$ signifies that valid data from the processor is on the bus.

$\overline{\text{HB}}$
(High Byte)
Output, active low. $\overline{\text{HB}}$ signifies that the upper byte of the data is to be read or written. $\overline{\text{HB}}$ remains active for the entire bus cycle.

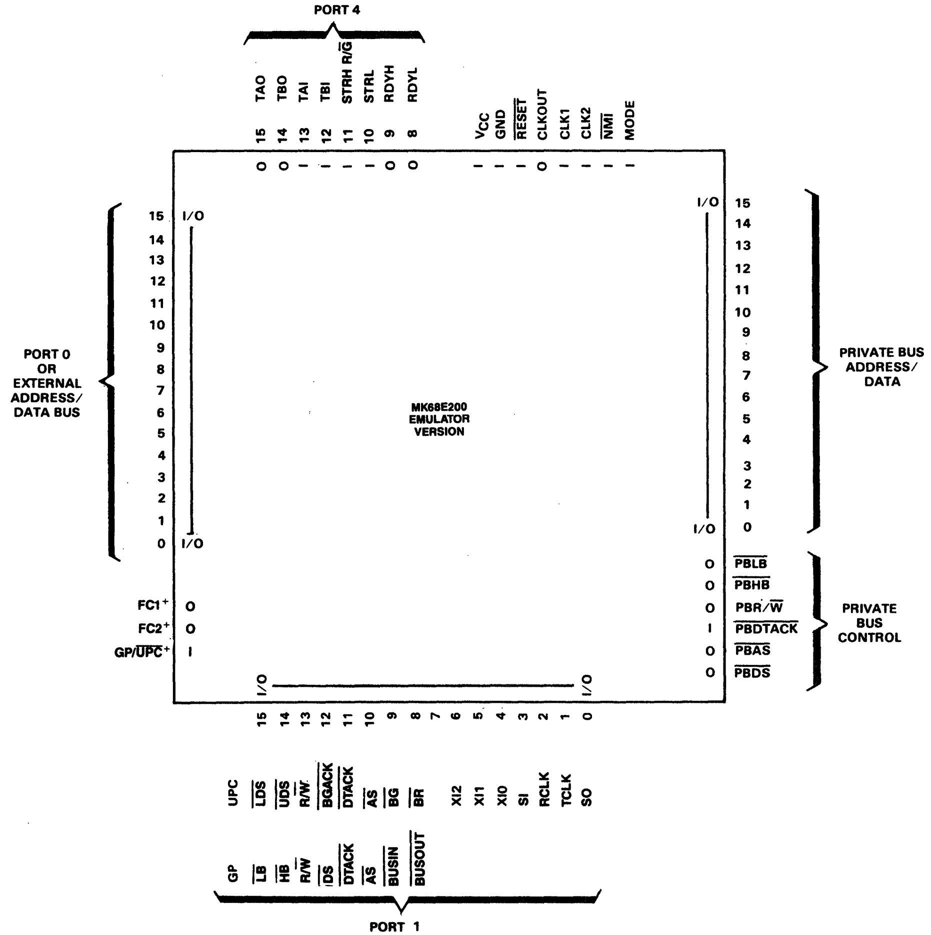
$\overline{\text{LB}}$
(Low Byte)
Output, active low. $\overline{\text{LB}}$ signifies that the lower byte of the data bus is to be read or written. (Both $\overline{\text{HB}}$ and $\overline{\text{LB}}$ active imply that an entire word is to be read or written). $\overline{\text{LB}}$ remains active for the entire bus cycle.

$\overline{\text{BUSIN}}$
(Bus Input)
Input, active low. $\overline{\text{BUSIN}}$ provides either bus request or bus grant. When the MK68200 is the bus grant device, its $\overline{\text{BUSIN}}$ signal is a bus request input from a requesting device on the bus. When the MK68200 is a bus request device, its $\overline{\text{BUSIN}}$ signal is a bus grant from the granting device on the bus.

$\overline{\text{BUSOUT}}$
(Bus Output)
Output, active low. $\overline{\text{BUSOUT}}$ provides the opposite function of $\overline{\text{BUSIN}}$. When $\overline{\text{BUSIN}}$ is a bus request signal, $\overline{\text{BUSOUT}}$ is the corresponding bus grant, and vice versa.

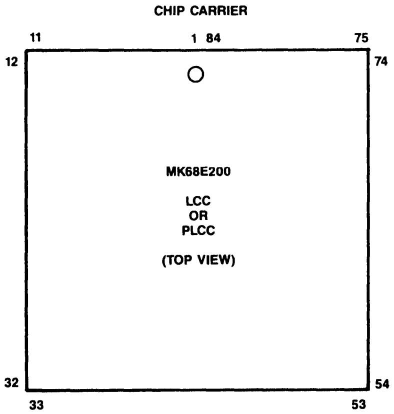
EMULATOR VERSION

The emulator versions of the MK68200 are available in 84-pin, leadless or leaded chip carrier packages or pin-grid array packages. Figure 15 illustrates the logical pinout of the emulator version. Table 1 summarizes the emulator parts described in this data sheet. The emulator versions have no on-chip ROM, but instead include a second complete bus, referred to as the private bus. The private bus includes a multiplexed address/data bus as well as bus control signals. There are 22 pins associated with the private bus. All 40 I/O port pins that exist on the 48-pin versions are available to the user for configuration either as general purpose or special I/O pins, or as expanded bus pins.



+ AVAILABLE ON MK68E221 (1K RAM VERSION)

Figure 15. MK68E200 Logical Pinout



LCC	FUNCTION	LCC	FUNCTION	LCC	FUNCTION	LCC	FUNCTION
1	P1-4/XIO	24	P4-8/RDYL	44	P0-6	65	PB-9
2	P1-5/XI1	25	P4-9/RDYH	45	P0-5	66	PB-10
3	P1-6/XI2	26	P4-10/STRL	46	P0-4	67	PB-11
4	P1-7	27	P4-11/STRH	47	P0-3	68	PB-12
5	P1-8	28	MODE	48	P0-2	69	PB-13
6	P1-9	29	CLK2	49	P0-1	70	PB-14
7	P1-10	30	CLK2	50	P0-0	71	PB-15
8	P1-11	31	CLKOUT	51	FC2 +	72	P4-13/TAI
13	GP/$\overline{\text{UPC}}$ +	32	FC1 +	54	VCC	73	P4-12/TBI
14	P1-12	34	GROUND	55	GROUND	76	P4-15/TAO
15	P1-13	35	P0-15	56	PB-0	77	P4-14/TBO
16	P1-14	36	P0-14	57	PB-1	78	$\overline{\text{RESET}}$
17	P1-15	37	P0-13	58	PB-2	79	$\overline{\text{NMI}}$
18	$\overline{\text{PBLB}}$	38	P0-12	59	PB-3	80	P1-0/SO
19	$\overline{\text{PBHB}}$	39	P0-11	60	PB-4	81	P1-1/TCLK
20	PBR/$\overline{\text{W}}$	40	P0-10	61	PB-5	82	P1-2/RCLK
21	$\overline{\text{PBDTACK}}$	41	P0-9	62	PB-6	83	P1-3/SI
22	$\overline{\text{PBAS}}$	42	P0-8	63	PB-7	84	VCC
23	$\overline{\text{PBDS}}$	43	P0-7	64	PB-8		

+ AVAILABLE ON MK68E221 (1K RAM VERSION)

Figure 16. MK68E200 Pin Assignment, Emulator Version

PRIVATE BUS OPERATION

The address/data lines and control signals that constitute the private bus are functionally equivalent to the internal signals used to access internal resources on the 48-pin versions of the MK68200. Thus, the private bus may be used to interface EPROM memory in emulating mask ROM versions of the MK68200. Alternately, any combination of ROM, RAM, and I/O may reside on the private bus.

The address that is generated on the private bus is identical to that which is internally generated for 48-pin versions. When the part is used in a configuration that supports system bus addressing through the DMA window, any references in this region of the memory map produce an address on the private bus identical to that specified by the programmer. In other words, the segment bits have no effect on the private bus address. Write data appears on the private bus pins for all write operations, regardless of whether the reference is on-chip or off-chip. The MK68200 emulator version reads data from the private bus unless data is read from on-chip RAM, I/O, or the external bus formed by the Port 0 and Port 1 I/O pins.

The I/O port range of the memory map ($FC00-$FFFF) is actually subdivided into space which is exclusively reserved for on-chip I/O ($FC00-$FDFF) and space which is exclusively reserved for in-circuit-emulator, or AIM, use ($FE00-$FFFF). The user should ensure that no external devices reside in the in-circuit-emulator area.

The private bus interface is the same as that for the GP expanded bus. All read/write transfers made exclusively on the private bus are three clock periods, regardless of the state of the Fast/Standard (F/S) bus timing selection bit. The user should ignore all activity on the private bus while accesses are in progress on the expanded bus. Care should also be taken that no external devices reside on the private bus in the memory space intended for expanded bus accesses.

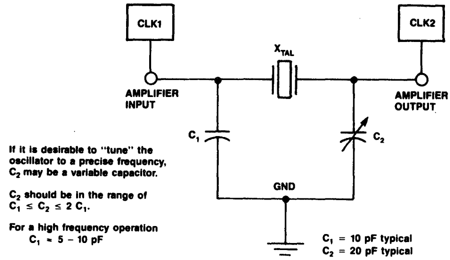
FUNCTION CODE PINS

Function code pins will be available on some versions of the emulator to define the memory cycle currently being executed. They are valid during the time private bus address strobe (PBAS*) is active. The cycle types are interrupt, data fetch, branch, and program fetch. The branch cycle is defined as the first program fetch after a branch occurs. A branch can occur as a result of a jump or call instruction, or an interrupt. For internal interrupts, the interrupt cycles are defined as the two writes to the stack and the read of the vector location which occur during the interrupt acknowledge routine. For external interrupts, the interrupt cycles are defined as the 3 cycles above plus the read of the vector number. The interrupt cycle is a special case of the data fetch cycle. The function code pins are defined below.

TYPE OF CYCLE	FC1	FC2
Interrupt	0	0
Data Fetch	0	1
Branch	1	0
Program Fetch	1	1

CRYSTAL SELECTION

The wide frequency range of crystals that can be chosen for the MK68200 offers the user a large degree of flexibility. To aid in the selection of a suitable crystal, the suggestions shown in Figure 18 should be considered by the user. The MK68200 offers an output pin that will provide a system clock signal at one-half of the crystal frequency.



If it is desirable to "tune" the oscillator to a precise frequency, C_2 may be a variable capacitor.

C_2 should be in the range of $C_1 \leq C_2 \leq 2\,C_1$.

For a high frequency operation $C_1 \approx 5 - 10$ pF

$C_1 = 10$ pF typical
$C_2 = 20$ pF typical

Figure 17. Crystal Connection

FREQUENCY RANGE	SPECIFICATION
1 MHz through 12.0 MHz	PARALLEL RESONANCE FUNDAMENTAL MODE C_L = 20 pF to 40 pF AT CUT

Figure 18. Summary of Crystal Specifications

ASSEMBLER DIRECTIVES

Directive	Function	Assembler Syntax		
DC	Define constant	[label:]	DC[.size]*1	expr {,expr}
DS	Define storage	[label:]	DS[.size]*1	expr
DUP	Duplicate constant block	[label:]	DUP[.size]*1	length, value
END	Program end		END	[start address]
EQU	Equate symbol value	label:	EQU	expr
FAIL	Programmer generated error		FAIL	expr
FORMAT	Format the source listing		FORMAT	
IDNT	Generate module ID	module_name:	IDNT	version, revision
LIST	Enable the assembly listing		LIST	
LLEN	Specify line length		LLEN	length
NOFORMAT	Do not format listing		NOFORMAT	
NOLIST	Disable assembly listing		NOLIST	
NOOBJ	Disable object code generation		NOOBJ	
NOPAGE	Suppress paging		NOPAGE	
OFFSET	Define Offsets		OFFSET	expr
OPT	Assembler output options		OPT	option[2] {, option}
ORG	Define absolute origin		ORG	expr
PAGE	Eject a page in the listing		PAGE	
REG	Define register list	reg_list_name:	REG[.size]	register list
SECTION	Define relocatable program section	[section_name:]	SECTION	number
SET	Set symbol value	label:	SET	expr
SPC	Space between source lines		SPC	number
TTL	Specify heading title string		TTL	title string
XDEF	External symbol definition		XDEF	symbol {, symbol}
XREF	External symbol reference		XREF	[sect no:] symbol - {,[sect no]: symbol}

NOTES:

1. .size = .B or .W (byte or word size)
2. Options for the OPT directive include:

CEX	Print DC expansions
NOCEX	Do not print DC expansions (default)
CL	Print conditional assembly directives (default)
NOCL	Do not print conditional assembly directives
CRE	Print cross-reference table
IMM.L	Forces immediate operands for arithmetic instructions ADD, SUB, DADD, and DSUB to use the long instruction form
IMM.S	Allows the assembler to select automatically the short form of the arithmetic instructions for small immediate values (0-15) (default)
MC	Print macro calls (default)
NOMC	Do not print macro calls
MD	Print macro definitions (default)
NOMD	Do not print macro definitions
MEX	Print macro expansions
NOMEX	Do not print macro expansions (default)
O	Create object module (default)
NOO	Do not create object module
STR	Print code generated by structured statements
NOSTR	Do not print code generated by structured statements (default)

GENERAL SYMBOL DEFINITIONS

SYMBOL	GENERAL SYMBOL DEFINITIONS
Rn	General Purpose Registers - D0-D7, A0-A5, SP, SR, DH0-DH7, DL0-DL7.
RPn	Register Pairs - D0-D1, D2-D3, D4-D5, D6-D7, A0-A1, A2-A3, A4-A5.
An	Address Registers - A0-A5, SP.
Pn	Ports - P0-P15, PH0-PH3, PL0-PL3.
CC	Condition Code - See Table.
d16	16-Bit Address Displacement Field In Words.
d13	13-Bit Address Displacement Field In Bytes.
d9	9-Bit Address Displacement Field In Bytes.
d8	8-Bit Address Displacement Field In Bytes.
#nx	Immediate Data Field - x Number of Bits.
$\boxed{\text{s}}$	Size Bit - '1' = Word, '0' = Byte.
REGn	4-Bit Register Field - See Table.
$\boxed{\text{PORTn}}$	4-Bit Port Field - See Table.
$\boxed{\text{An}}$	3-Bit Address Register Field - See Table.
$\boxed{\text{PRTn}}$	3-Bit Port Field - See Table.
$\boxed{\text{RGn}}$	3-Bit Register Pair Field - See Table.
M	Register Mask Field - See Table.
$\boxed{\text{COND}}$	Condition Code Field - See Table.
$\boxed{\text{c3}}$	3-Bit Class Field - See Table.
$\boxed{\text{c2}}$	2-Bit Class Field - See Table
$\boxed{\text{c1}}$	1-Bit Class Field—See Table
a	Address Field - 16 Bits.
#	Immediate Data Field.
n	3-Bit Shift Field - $2 \leq n \leq 7$.
b#	4-Bit Bit Select Field.
d	Displacement Field.
.B	Byte Attribute.
.W	Word Attribute.
.L	Long Attribute.
.S	Short Attribute.
[ ]	Optional Field.

FIELD DEFINITIONS

REGn 4-Bit Register Map					
Register		Bit Field	Register		Bit Field
D0	DH0	0 0 0 0	A0	DL0	1 0 0 0
D1	DH1	0 0 0 1	A1	DL1	1 0 0 1
D2	DH2	0 0 1 0	A2	DL2	1 0 1 0
D3	DH3	0 0 1 1	A3	DL3	1 0 1 1
D4	DH4	0 1 0 0	A4	DL4	1 1 0 0
D5	DH5	0 1 0 1	A5	DL5	1 1 0 1
D6	DH6	0 1 1 0	SP	DL6	1 1 1 0
D7	DH7	0 1 1 1	SR	DL7	1 1 1 1

PORTn 4-Bit Port Map				
Port		Bit Field	Port	Bit Field
P0	PH0	0 0 0 0	P8	1 0 0 0
P1	PL0	0 0 0 1	P9	1 0 0 1
P2	PH1	0 0 1 0	P10	1 0 1 0
P3	PL1	0 0 1 1	P11	1 0 1 1
P4	PH2	0 1 0 0	P12	1 1 0 0
P5	PL2	0 1 0 1	P13	1 1 0 1
P6	PH3	0 1 1 0	P14	1 1 1 0
P7	PL3	0 1 1 1	P15	1 1 1 1

An 3-Bit Addr Reg Map	
Register	Bit Field
A0	0 0 0
A1	0 0 1
A2	0 1 0
A3	0 1 1
A4	1 0 0
A5	1 0 1
SP	1 1 0

PTRn 3-Bit Port Map	
Port	Bit Field
PH0	0 0 0
PL0	0 0 1
PH1	0 1 0
PL1	0 1 1
PH2	1 0 0
PL2	1 0 1
PH3	1 1 0
PL3	1 1 1

RGn 3-Bit Reg Pair Map	
Register	Bit Field
D0-D1	0 0 0
D2-D3	0 0 1
D4-D5	0 1 0
D6-D7	0 1 1
A0-A1	1 0 0
A2-A3	1 0 1
A4-A5	1 1 0

M—REGISTER MASK MAP FOR MOVEM, PUSHM, AND POPM																
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Inc Word	SR	SP	A5	A4	A3	A2	A1	A0	D7	D6	D5	D4	D3	D2	D1	D0
Dec Word	D0	D1	D2	D3	D4	D5	D6	D7	A0	A1	A2	A3	A4	A5	SP	SR
Inc Byte	DH7	DL7	DH6	DL6	DH5	DL5	DH4	DL4	DH3	DL3	DH2	DL2	DH1	DL1	DH0	DL0
Dec Byte	DL0	DH0	DL1	DH1	DL2	DH2	DL3	DH3	DL4	DH4	DL5	DH5	DL6	DH6	DL7	DH7

$\boxed{\text{COND}}$ CONDITION CODE TABLE			
Condition Code	**Bit Field**	**Description**	**Test**
Z EQ	0 0 0 0	Zero Equal	Z
MI	0 0 0 1	Minus	N
LO[2] CS	0 0 1 0	Lower Carry Set	C
VS	0 0 1 1	Overflow Set	V
GE[2]	0 1 0 0	Greater than or Equal	$\overline{N \text{ .EOR. } V}$
GT[2]	0 1 0 1	Greater than	$\overline{Z}$ .AND. ($\overline{N \text{ .EOR. } V}$)
HI[2]	0 1 1 0	Higher	$\overline{C}$ .AND. $\overline{Z}$
F[1]	0 1 1 1	False	Always False
NE NZ	1 0 0 0	Not Equal Not Zero	$\overline{Z}$
PL	1 0 0 1	Plus	$\overline{N}$
HS[2] CC	1 0 1 0	Higher or Same Carry Clear	$\overline{C}$
VC	1 0 1 1	Overflow Clear	$\overline{V}$
LT[2]	1 1 0 0	Less than	N .EOR. V
LE[2]	1 1 0 1	Less than or Equal	Z .OR. (N .EOR. V)
LS[2]	1 1 1 0	Lower or Same	C .OR. Z
T[1]	1 1 1 1	True	Always True

NOTES:

1. The assembler does not recognize the T and F condition codes.
2. LT, LE, GT, and GE are used for unsigned conditions; LO, LS, HI, and HS are for unsigned conditions.

INSTRUCTION CLASS FIELDS									
$\boxed{c3}$ - 3-Bit Field			$\boxed{c2}$ - 2-Bit Field			$\boxed{c1}$ - 1-Bit Field			
Bit Field	Shift Instr	Bit[1] Instr	Bit Field	Arith[2] Instr	Logical Instr	Bit Field	Arith[2] Instr	Test Instr	Neg[2] Instr
0 0 0	ROR	BSET	0 0	ADDC	OR	0	ADD	TESTN	NEGC
0 0 1	ROL	BCHG	0 1	SUBC	EOR	1	SUB	TEST	NEG
0 1 0	RORC	BCLR	1 0	ADD	AND				
0 1 1	ROLC	BTST	1 1	SUB	—				
1 0 0	ASR	—							
1 0 1	ASL	—							
1 1 0	LSR	—							
1 1 1	—	BEXG							

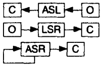
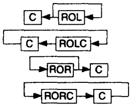
NOTES:
1. The bit fields do not apply to bit instructions using a port operand.
2. These fields also apply to BCD instructions.

INSTRUCTION FORMAT

15	0
PREFIX WORD **(used only in some forms of the decimal and bit instructions)**	
OPERATION WORD **(contains the opcode and possibly the operands)**	
EXTENSION WORD **(optional - specified by the operation word to be immediate operand, mask field, displacement or absolute address)**	

INSTRUCTION MNEMONIC	ATTR SIZE	OPERAND ASSEMBLER SYNTAX Source=Src Destination=Dst	OPERATION WORD									WORDS	EXTENSION	CYCLES	OPERATION	STATUS FLAGS			
		Src , Dst	15 14 13 12	11	10 9 8	7	6	5	4	3	2 1 0					N	Z	V	C
ADD ADDC SUB SUBC	.B [.W]	Ry,Rx	2	REGx		s	c2		0	REGy		1	—	3	ADD: Src + Dst → Dst	*	*	*	*
	.B [.W]	(Ay),Rx	2	REGx		s	c2		1	0	Ay	1	—	6	ADDC: Src + Dst + C → Dst				
	.B [.W]	d16(Ay),Rx	2	REGx		s	c2		1	1	Ay	2	d	9	SUB: Dst − Src → Dst				
	.B [.W]	Addr,Rx	2	REGx		s	c2		1	F		2	a	9	SUBC: Dst − Src − C → Dst				
	.B [.W]	#n16,Rx	2	REGx		s	c2		1	7		2	#	6	Note: For addressing modes #n,Rx and #n,(Ax) with the ADD and SUB instructions, the assembler uses the short version for immediate values ≤4 bits.				
	.B [.W]	Ry,(Ax)	3	0	Ax	s	c2		0	REGy		1	—	9					
	.B [.W]	(Ax),(Ay)	3	0	Ax	s	c2		1	0	Ay	1	—	12					
	.B [.W]	#n16,(Ax)	3	0	Ax	s	c2		1	7		2	#	12					
	.B [.W]	(Ax)+,(Ay)+	3	1	Ax	s	c2		1	0	Ay	1	—	12					
	.B [.W]	#n16,(Ax)+	3	1	Ax	s	c2		1	7		2	#	12					
	.B [.W]	−(Ax),−(Ay)	3	1	Ax	s	c2		1	1	Ay	1	—	12					
	.B [.W]	#n16,−(Ax)	3	1	Ax	s	c2		1	F		2	#	12					
	.B [.W]	Ry,d16(Ax)	3	1	Ax	s	c2		0	REGy		2	d	12					
	.B [.W]	Ry,Addr	3	F		s	c2		0	REGy		2	a	12					

INSTRUCTION MNEMONIC	ATTR SIZE	OPERAND ASSEMBLER SYNTAX Source=Src Destination=Dst	OPERATION WORD									WORDS	EXTENSION	CYCLES	OPERATION	STATUS FLAGS			
		Src , Dst	15 14 13 12	11	10 9 8	7	6	5	4	3	2 1 0					N	Z	V	C
ADD SUB	.B [.W]	#n4,Rx	B	REGx		s	1	c1	0	#		1	—	3	ADD: Src + Dst → Dst	*	*	*	*
	.B [.W]	#n4,(Ax)	B	0	Ax	s	1	c1	1	#		1	—	9					
	.B [.W]	#n4,d16(Ax)	B	1	Ax	s	1	c1	1	#		2	d	12	SUB: Dst − Src → Dst				
	.B [.W]	#n4,Addr	B	F		s	1	c1	1	#		2	a	12					
AND EOR OR	.B [.W]	Ry,Rx	6	REGx		s	c2		0	REGy		1	—	3	AND: Src .AND. Dst → Dst	*	*	0	0
	.B [.W]	Py,Rx	5	REGx		s	c2		0	PORTy		1	—	6					
	.B [.W]	(Ay),Rx	6	REGx		s	c2		1	0	Ay	1	—	6	EOR: Src .EOR. Dst → Dst				
	.B [.W]	d16(Ay),Rx	6	REGx		s	c2		1	1	Ay	2	d	9					
	.B [.W]	Addr,Rx	6	REGx		s	c2		1	F		2	a	9	OR: Src .OR. Dst → Dst				
	.B [.W]	#n16,Rx	6	REGx		s	c2		1	7		2	#	6					
	.B [.W]	Ry,Px	4	PORTx		s	c2		0	REGy		1	—	9					
	[.W]	#n16,Px	5	PORTx		1	c2		1	9		2	#	12					
	.B [.W]	Ry,(Ax)	7	0	Ax	s	c2		0	REGy		1	—	9					
Continued on next page	.B [.W]	(Ax),(Ay)	7	0	Ax	s	c2		1	0	Ay	1	—	12					

INSTRUCTION MNEMONIC	ATTR SIZE	OPERAND ASSEMBLER SYNTAX Source=Src Destination=Dst Src , Dst	OPERATION WORD				WORDS	EXTENSION	CYCLES	OPERATION	STATUS FLAGS			
			15 14 13 12	11 10 9 8	7 6 5 4	3 2 1 0					N	Z	V	C
AND EOR OR (cont.)	.B [.W]	#n16,(Ax)	7	0 Ax	s c2 1	7	2	#	12	AND: Src .AND. Dst → Dst	*	*	0	0
	.B [.W]	(Ax)+,(Ay)+	7	1 Ax	s c2 1	0 Ay	1	—	12					
	.B [.W]	#n16,(Ax)+	7	1 Ax	s c2 1	7	2	#	12	EOR: Src .EOR. Dst → Dst				
	.B [.W]	−(Ax),−(Ay)	7	1 Ax	s c2 1	1 Ay	1	—	12					
	.B [.W]	#n16,−(Ax)	7	1 Ax	s c2 1	F	2	#	12	OR: Src .OR. Dst → Dst				
	.B [.W]	Ry,d16(Ax)	7	1 Ax	s c2 0	REGy	2	d	12					
	.B [.W]	Ry,Addr	7	F	s c2 0	REGy	2	a	12					
ASL ASR LSR ROL ROLC ROR RORC	.B [.W]	[#1],Rx	B	REGx	s 0 c3	0 0 1	1	—	3	Dst → Dst SHIFT:  	*	*	*	*
	.B [.W]	#n3,Rx	B	REGx	s 0 c3	n	1	—	4+n*	ROTATE:  	NOTE: The ASR, LSR, ROR, and RORC instructions will clear the V status bit.			
	.B [.W]	(Ax)	B	0 Ax	s 0 c3	0 0 0	1	—	9					
	.B [.W]	d16(Ax)	B	1 Ax	s 0 c3	0 0 0	2	d	12					
	.B [.W]	Addr	B	F	s 0 c3	0 0 0	2	a	12					
										*NOTE: 2≤ n ≤7				

INSTRUCTION MNEMONIC	ATTR SIZE	OPERAND ASSEMBLER SYNTAX Source=Src Destination=Dst Src , Dst	OPERATION WORD				WORDS	EXTENSION	CYCLES	OPERATION	STATUS FLAGS			
			15 14 13 12	11 10 9 8	7 6 5 4	3 2 1 0					N	Z	V	C
BCHG BCLR BEXG BSET BTST	—	#n4,Rx	4	REGx	c3 1	b#	1	—	3	BCHG: Dst(b#) → C 1 — Dst(b#) → Dst(b#)	*	*	0	*
	—	#n4,(Ax)	5	0 Ax	c3 1	b#	1	—	9*		If Dst(b#) is in the low byte of the SR,then the other SR bits are unchanged.			
	—	#n4,d16(Ax)	5	1 Ax	c3 1	b#	2	d	12*	BCLR: Dst(b#) → C 0 → Dst(b#)				
	—	#n4,Addr	5	F	c3 1	b#	2	a	12*					
	—	Ry,Rx	0	REGy	F	7	2	—	6	BEXG: Dst(b#) ← C				
			4	REGx	c3 1	0								
	—	Ry,(Ax)	0	REGy	F	7	2	—	12*	BSET: Dst(b#) → C 1 → Dst(b#)				
			5	0 Ax	c3 1	0								
	—	Ry,d16(Ax)	0	REGy	F	7	3	d	15*	BTST: Dst(b#) → C				
			5	1 Ax	c3 1	0								
	—	Ry,Addr	0	REGy	F	7	3	a	15*	*NOTE: The BTST instruction executes in 3 less clock cycles.				
			5	F	c3 1	0								

INSTRUCTION MNEMONIC	ATTR SIZE	OPERAND ASSEMBLER SYNTAX Source=Src Destination=Dst Src , Dst	OPERATION WORD				WORDS	EXTENSION	CYCLES	OPERATION	STATUS FLAGS			
			15 14 13 12	11 10 9 8	7 6 5 4	3 2 1 0					N	Z	V	C
BCHG	—	#n4,Px	4	PORTx	B	b#	1	—	9	Dst(b#) → C 1 — Dst(b#) → Dst(b#)	*	*	0	*
	—	Ry,Px	0	REGy	F	7	2	—	12					
			4	PORTx	B	0								
BCLR	—	#n4,Px	4	PORTx	D	b#	1	—	9	Dst(b#) → C 0 → Dst(b#)				
	—	Ry,Px	0	REGy	F	7	2	—	12					
			4	PORTx	D	0								
BEXG	—	#n4,Px	7	PORTx	E	b#	1	—	9	Dst(b#) → C				
	—	Ry,Px	0	REGy	F	7	2	—	12					
			7	PORTx	E	0								
BSET	—	#n4,Px	4	PORTx	9	b#	1	—	9	Dst(b#) → C 1 → Dst(b#)				
	—	Ry,Px	0	REGy	F	7	2	—	12					
			4	PORTx	9	0								
BTST Continued on next page	—	#n4,Px	7	PORTx	6	b#	1	—	6	Dst(b#) → C				

INSTRUCTION MNEMONIC	ATTR SIZE	OPERAND ASSEMBLER SYNTAX Source=Src Destination=Dst Src , Dst	OPERATION WORD				WORDS	EXTENSION	CYCLES	OPERATION	STATUS FLAGS			
			15 14 13 12	11 10 9 8	7 6 5 4	3 2 1 0					N	Z	V	C
BTST (cont.)	—	Ry,Px	0	REGy	F	7	2	—	9	Dst (b#) → C	*	*	0	*
			7	PORTx	6	0								
CALLA	[.L]	(Ax) (unconditional)	5	0 Ax	D	F	1	—	9	PC + 2 → −(SP) (Ax) → PC	—	—	—	—
	[.L]	Addr (unconditional)	5	F	B	F	2	a	9	PC + 4 → −(SP) Addr → PC				
	[.L]	CC, Addr	5	COND	B	F	2	a	F:6 T:12	If COND is true, PC + 4 → −(SP); Addr → PC				
										NOTE: The initial PC value is the location of the CALLA instruction.				
CALLR	[.S]	d13 (unconditional)	F	d			1	—	10	PC + 2 → −(SP) PC + 4 + 2*(d) → PC	—	—	—	—
	.L	d16 (unconditional)	5	F	9	F	2	d	9	PC + 4 → −(SP) PC + 4 + d → PC				
	.L	CC, d16	5	COND	9	F	2	d	F:6 T:12	If COND is true, PC + 4 → −(SP); PC + 4 + d → PC				
										NOTE: The initial PC value is the location of the CALLR instruction. The displacement, d13, is in signed magnitude representation, and the displacement, d16, is in two's complement representation.				

INSTRUCTION MNEMONIC	ATTR SIZE	OPERAND ASSEMBLER SYNTAX Source=Src Destination=Dst Src , Dst	OPERATION WORD				WORDS	EXTENSION	CYCLES	OPERATION	STATUS FLAGS			
			15 14 13 12	11 10 9 8	7 6 5 4	3 2 1 0					N	Z	V	C
CLR	.B [.W]	Rx	1	REGx	s 0 1 1	7	1	—	3	0 → Dst	—	—	—	—
	.B [.W]	Px	1	PORTx	s 0 1 0	7	1	—	6					
	.B [.W]	(Ax)	1	0 Ax	s 0 1 1	F	1	—	6					
	.B [.W]	d16(Ax)	1	1 Ax	s 0 1 1	F	2	d	9					
	.B [.W]	Addr	1	F	s 0 1 1	F	2	a	9					
CMP	.B [.W]	Ry,Rx	6	REGx	s 1 1 0	REGy	1	—	3	Dst — Src	*	*	*	*
	.B [.W]	Py,Rx	5	REGx	s 1 1 0	PORTy	1	—	6					
	.B [.W]	(Ay),Rx	6	REGx	s 1 1 1	0 Ay	1	—	6					
	.B [.W]	(Ay)+,Rx	4	REGx	s 1 1 0	0 Ay	1	—	6					
	.B [.W]	-(Ay),Rx	4	REGx	s 1 1 0	1 Ay	1	—	6					
	.B [.W]	d16(Ay),Rx	6	REGx	s 1 1 1	1 Ay	2	d	9					
	.B [.W]	Addr,Rx	6	REGx	s 1 1 1	F	2	a	9					
	.B	#n8,Rx	C	REGx	#		1	—	3					
Continued on next page	[.W]	#n16,Rx	6	REGx	F	7	2	#	6					

INSTRUCTION MNEMONIC	ATTR SIZE	OPERAND ASSEMBLER SYNTAX Source=Src Destination=Dst Src , Dst	OPERATION WORD				WORDS	EXTENSION	CYCLES	OPERATION	STATUS FLAGS			
			15 14 13 12	11 10 9 8	7 6 5 4	3 2 1 0					N	Z	V	C
CMP (cont.)	.B	#n8,Px	D	0 PRTx	#		1	—	6	Dst — Src	*	*	*	*
	[.W]	#n16,Px	4	PORTx	E	F	2	#	9					
	.B [.W]	(Ax),(Ay)	7	0 Ax	s 1 1 1	0 Ay	1	—	9					
	.B [.W]	#n16,(Ax)	7	0 Ax	s 1 1 1	7	2	#	9					
	.B [.W]	(Ax)+,(Ay)+	7	1 Ax	s 1 1 1	0 Ay	1	—	9					
	.B [.W]	#n16,(Ax)+	7	1 Ax	s 1 1 1	7	2	#	9					
	.B [.W]	-(Ax),-(Ay)	7	1 Ax	s 1 1 1	1 Ay	1	—	9					
	.B [.W]	#n16,-(Ax)	7	1 Ax	s 1 1 1	F	2	#	9					
DADD DADDC DSUB DSUBC	.B [.W]	Ry,Rx	A	REGx	s c2 0	REGy	1	—	3	DADD: BCD: [Src + Dst → Dst] DADDC: BCD: [Src + Dst + C → Dst] DSUB: BCD: [Dst − Src → Dst] DSUBC: BCD: [Dst − Src − C → Dst]	U	*	U	*
	.B [.W]	(Ay),Rx	A	REGx	s c2 1	0 Ay	1	—	6					
	.B [.W]	d16(Ay),Rx	A	REGx	s c2 1	1 Ay	2	d	9					
	.B [.W]	Addr,Rx	A	REGx	s c2 1	F	2	a	9					
Continued on next page	.B [.W]	#n16,Rx	A	REGx	s c2 1	7	2	#	6					

INSTRUCTION MNEMONIC	ATTR SIZE	OPERAND ASSEMBLER SYNTAX Source=Src Destination=Dst Src , Dst	OPERATION WORD 15 14 13 12	11 10 9 8	7 6 5 4	3 2 1 0	WORDS	EXTENSION	CYCLES	OPERATION	STATUS FLAGS N	Z	V	C
DADD DADDC DSUB DSUBC (cont.)	.B [.W]	Ry,(Ax)	0	F	F	F	2	—	12	NOTE: For addressing modes #n,Rx and #n,(Ax) with the DADD and DSUB instructions, the assembler uses the short version for immediate values ≤4 bits.	U	*	U	*
			3	0 Ax	s c2 0	REGy								
	.B [.W]	(Ax),(Ay)	0	F	F	F	2	—	15					
			3	0 Ax	s c2 1	0 Ay								
	.B [.W]	#n16,(Ax)	0	F	F	F	3	#	15					
			3	0 Ax	s c2 1	7								
	.B [.W]	(Ax)+,(Ay)+	0	F	F	F	2	—	15					
			3	1 Ax	s c2 1	0 Ay								
	.B [.W]	#n16,(Ax)+	0	F	F	F	3	#	15					
			3	1 Ax	s c2 1	7								
	.B [.W]	−(Ax),−(Ay)	0	F	F	F	2	—	15					
			3	1 Ax	s c2 1	1 Ay								
	.B [.W]	#n16,−(Ax)	0	F	F	F	3	#	15					
Continued on next page			3	1 Ax	s c2 1	F								

INSTRUCTION MNEMONIC	ATTR SIZE	OPERAND ASSEMBLER SYNTAX Source=Src Destination=Dst Src , Dst	OPERATION WORD 15 14 13 12	11 10 9 8	7 6 5 4	3 2 1 0	WORDS	EXTENSION	CYCLES	OPERATION	STATUS FLAGS N	Z	V	C
DADD DADDC DSUB DSUBC (cont.)	.B [.W]	Ry,d16(Ax)	0	F	F	F	3	d	15		U	*	U	*
			3	1 Ax	s c2 0	REGy								
	.B [.W]	Ry,Addr	0	F	F	F	3	a	15					
			3	F	s c2 0	REGy								
DADD DSUB	.B [.W]	#n4,Rx	0	F	F	F	2	—	6	DADD: BCD: [Src + Dst → Dst]	U	*	U	*
			B	REGx	s 1 c1 0	#								
	.B [.W]	#n4,(Ax)	0	F	F	F	2	—	12	DSUB: BCD: [Dst − Src → Dst]				
			B	0 Ax	s 1 c1 1	#								
	.B [.W]	#n4,d16(Ax)	0	F	F	F	3	d	15					
			B	1 Ax	s 1 c1 1	#								
	.B [.W]	#n4,Addr	0	F	F	F	3	a	15					
			B	F	s 1 c1 1	#								
DI	—	—	4	F	5	7	1	—	3	0 → L1E Disable Interrupts	—	—	—	—

INSTRUCTION MNEMONIC	ATTR SIZE	OPERAND ASSEMBLER SYNTAX Source = Src Destination = Dst Src , Dst	OPERATION WORD				WORDS	EXTENSION	CYCLES	OPERATION	STATUS FLAGS			
			15 14 13 12	11 10 9 8	7 6 5 4	3 2 1 0					N	Z	V	C
DIVU	—	Rx,RPy	5	REGx	D	RGy 0	1	—	F:6 T:23	RPy(even),RPy(odd)/Rx → RPy(even) REM → RPy(odd) MSW – even; LSW – odd	—	—	*	—
DJNZ	.B [.W]	Rx,d8	9	REGx	s	d	1	—	F:6 T:9	Rx – 1 → Rx; If Rx ≠ 0 then PC + 4 – [2•(d)] → PC NOTE: The initial PC value is the location of the DJNZ instruction. The displacement value, d, is a magnitude.	—	—	—	—
DNEG DNEGC	.B [.W]	Ry	0	F	F	F	2	—	6	DNEG: BCD: [0 – Dst → Dst]	U	*	U	*
			3	7	s c1 1 0	REGy								
	.B [.W]	(Ay)	0	F	F	F	2	—	12	DNEGC: BCD: [0 – Dst – C → Dst]				
			3	7	s c1 1 1	0 Ay								
	.B [.W]	d16(Ay)	0	F	F	F	3	d	15					
			3	7	s c1 1 1	1 Ay								
	.B [.W]	Addr	0	F	F	F	3	a	15					
			3	7	s c1 1 1	F								
DSUB DSUBC		see DADD (page 15)												

INSTRUCTION MNEMONIC	ATTR SIZE	OPERAND ASSEMBLER SYNTAX Source = Src Destination = Dst Src , Dst	OPERATION WORD				WORDS	EXTENSION	CYCLES	OPERATION	STATUS FLAGS			
			15 14 13 12	11 10 9 8	7 6 5 4	3 2 1 0					N	Z	V	C
EI	—	—	4	F	1	7	1	—	3	1 → L1E Enable Interrupts	—	—	—	—
EOR		see AND (page 9)												
EXG	.B [.W]	Rx,Ry	0	REGx	s 1 0 0	REGy	1	—	4	Src → Dst	—	—	—	—
	.B [.W]	Rx,(Ay)	0	REGx	s 1 0 1	0 Ay	1	—	9					
	.B [.W]	Rx,(Ay)+	0	REGx	s 1 1 1	0 Ay	1	—	9					
	.B [.W]	Rx,–(Ay)	0	REGx	s 1 1 1	1 Ay	1	—	9					
	.B [.W]	Rx,d16(Ay)	0	REGx	s 1 0 1	1 Ay	2	d	12					
	.B [.W]	Rx,Addr	0	REGx	s 1 0 1	F	2	a	12					
EXT	—	Rx	0	REGx	D	7	1	—	3	SIGN EXTEND: SignBit(RLx) → RHx	—	—	—	—
HALT	—	—	1	7	3	F	1	—	3	PC + 2 → PC Stop	—	—	—	—
JMPA	[.L]	(Ax) (unconditional)	5	1 Ax	D	F	1	—	6	(Ax) → PC	—	—	—	—
	[.L]	Addr (unconditional)	5	F	B	7	2	a	6	Addr → PC				
	[.L]	CC,Addr	5	COND	B	7	2	a	F:6 T:9	If COND is true, Addr → PC				

INSTRUCTION MNEMONIC	ATTR SIZE	OPERAND ASSEMBLER SYNTAX Source=Src Destination=Dst Src , Dst	OPERATION WORD				WORDS	EXTENSION	CYCLES	OPERATION	STATUS FLAGS			
			15 14 13 12	11 10 9 8	7 6 5 4	3 2 1 0					N	Z	V	C
JMPR	[.S]	d9 (unconditional)	8	F	d		1	—	7	PC + 4 + 2•(d) → PC	—	—	—	—
	[.S]	CC,d9	8	COND	d		1	—	F:4 T:7	If COND is true, PC + 4 + 2•(d) → PC				
	.L	d16 (unconditional)	5	F	9	7	2	d	9	PC + 4 + d → PC				
	.L	CC,d16	5	COND	9	7	2	d	F:6 T:9	If COND is true, PC + 4 + d → PC				
										NOTE: The initial PC value is the location of the JMPR instruction. The displacement, d16, is in two's complement representation, and the displacement, d9, is in signed magnitude representation.				
LIBA LIWA	—	#d16(Rx),Az	1	REGx	s 1 Az	1 1 1	2	#	6	Rx[•2] + d → Az	—	—	—	—
	—	#d16(Ay,Rx),Az	1	REGx	s 1 Az	Ay	2	#	6	Ay + (Rx[•2!) + d → Az				
										NOTE: [•2] used for LIWA only. REGx and Az must not refer to same register when using 2nd addressing mode.				
LSR		see ASL (page 10)												
MOVE	.B [.W]	Ry,Rx	0	REGx	s 0 0 0	REGy	1	—	3	Src → Dst	—	—	—	—
	.B [.W]	Py,Rx	0	REGx	s 0 1 1	PORTy	1	—	6					
Continued on next page	.B [.W]	(Ay),Rx	0	REGx	s 0 0 1	0 Ay	1	—	6					

INSTRUCTION MNEMONIC	ATTR SIZE	OPERAND ASSEMBLER SYNTAX Source=Src Destination=Dst Src , Dst	OPERATION WORD				WORDS	EXTENSION	CYCLES	OPERATION	STATUS FLAGS			
			15 14 13 12	11 10 9 8	7 6 5 4	3 2 1 0					N	Z	V	C
MOVE (cont.)	.B [.W]	(Ay)+,Rx	1	REGx	s 0 1 1	0 Ay	1	—	6	Src → Dst	—	—	—	—
	.B [.W]	−(Ay),Rx	1	REGx	s 0 1 1	1 Ay	1	—	6					
	.B [.W]	d16(Ay),Rx	0	REGx	s 0 0 1	1 Ay	2	d	9					
	.B [.W]	Addr,Rx	0	REGx	s 0 0 1	F	2	a	9					
	.B	#n8,Rx	E	REGx	#		1	—	3					
	[.W]	#n16,Rx	0	REGx	9	7	2	#	6					
	.B [.W]	Ry,Px	0	PORTx	s 0 1 0	REGy	1	—	6					
	.B [.W]	(Ay)+,Px	0	PORTx	s 1 1 0	0 Ay	1	—	9					
	.B	#n8,Px	D	1 PRTx	#		1	—	6					
	[.W]	#n16,Px	0	PORTx	E	7	2	#	9					
	.B [.W]	Ry,(Ax)	1	0 Ax	s 0 0 0	REGy	1	—	6					
	.B [.W]	(Ax),(Ay)	1	0 Ax	s 0 0 1	0 Ay	1	—	9					
	.B [.W]	#n16,(Ax)	1	0 Ax	s 0 0 1	7	2	#	9					
Continued on next page	.B [.W]	Rx,(Ay)+	1	REGx	s 0 1 0	0 Ay	1	—	6					

INSTRUCTION MNEMONIC	ATTR SIZE	OPERAND ASSEMBLER SYNTAX Source=Src Destination=Dst	OPERATION WORD				WORDS	EXTENSION	CYCLES	OPERATION	STATUS FLAGS			
		Src , Dst	15 14 13 12	11 10 9 8	7 6 5 4	3 2 1 0					N	Z	V	C
MOVE (cont.)	.B [.W]	(Ax)+,(Ay)+	1	1 Ax	s 0 0 1	0 Ay	1	—	9	Src → Dst	—	—	—	—
	.B [.W]	#n16,(Ax)+	1	1 Ax	s 0 0 1	7	2	#	9					
	.B [.W]	Rx,–(Ay)	1	REGx	s 0 1 0	1 Ay	1	—	6					
	.B [.W]	Px,–(Ay)	0	PORTx	s 1 1 0	1 Ay	1	—	9					
	.B [.W]	–(Ax),–(Ay)	1	1 Ax	s 0 0 1	1 Ay	1	—	9					
	.B [.W]	#n16,–(Ax)	1	1 Ax	s 0 0 1	F	2	#	9					
	.B [.W]	Ry,d16(Ax)	1	1 Ax	s 0 0 0	REGy	2	d	9					
	.B [.W]	Ry,Addr	1	F	s 0 0 0	REGy	2	a	9					
MOVEM	.B [.W]	(Ay)+,REGLIST	1	7	s 0 0 1	0 Ay	2	M	9+3n	(Ay)+ → REGLIST	—	—	—	—
	.B [.W]	–(Ay),REGLIST	1	7	s 0 0 1	1 Ay	2	M	9+3n	–(Ay) → REGLIST				
	.B [.W]	REGLIST,(Ay)+	1	F	s 0 0 1	0 Ay	2	M	7+3n	REGLIST → (Ay)+				
	.B [.W]	REGLIST,–(Ay)	1	F	s 0 0 1	1 Ay	2	M	7+3n	REGLIST → –(Ay)				
										NOTE: A minimum of 2 registers must be specified and may be specified in any order.				

INSTRUCTION MNEMONIC	ATTR SIZE	OPERAND ASSEMBLER SYNTAX Source=Src Destination=Dst	OPERATION WORD				WORDS	EXTENSION	CYCLES	OPERATION	STATUS FLAGS			
		Src , Dst	15 14 13 12	11 10 9 8	7 6 5 4	3 2 1 0					N	Z	V	C
MULS	—	Rx,RPy	5	REGx	B	RGy 0	1	—	21	Rx • RPy(even) → RPy(even), RPy(odd) MSW → even; LSW → odd	—	—	—	—
MULU	—	Rx,RPy	5	REGx	9	RGy 0	1	—	21	Rx • RPy(even) → RPy(even), RPy(odd) MSW → even; LSW → odd	—	—	—	—
NEG NEGC	.B [.W]	Ry	3	7	s c1 1 0	REGy	1	—	3	2's Complement	*	*	*	*
	.B [.W]	(Ay)	3	7	s c1 1 1	0 Ay	1	—	9	NEG: 0 – Dst → Dst				
	.B [.W]	d16(Ay)	3	7	s c1 1 1	1 Ay	2	d	12	NEGC: 0 – Dst – C → Dst				
	.B [.W]	Addr	3	7	s c1 1 1	F	2	a	12					
NOP	—	—	0	0	0	0	1	—	3	PC + 2 → PC	—	—	—	—
NOT	.B [.W]	Ry	3	7	s 0 0 0	REGy	1	—	3	1's Complement	*	*	0	0
	.B [.W]	(Ay)	3	7	s 0 0 1	0 Ay	1	—	9	NOT(Dst) → Dst				
	.B [.W]	d16(Ay)	3	7	s 0 0 1	1 Ay	2	d	12					
	.B [.W]	Addr	3	7	s 0 0 1	F	2	a	12					
OR		see AND (page 9)												

INSTRUCTION MNEMONIC	ATTR SIZE	OPERAND ASSEMBLER SYNTAX Source=Src Destination=Dst Src , Dst	OPERATION WORD				WORDS	EXTENSION	CYCLES	OPERATION	STATUS FLAGS			
			15 14 13 12	11 10 9 8	7 6 5 4	3 2 1 0					N	Z	V	C
POP	—	Rx	1	REGx	B	6	1	—	6	(SP)+ → Rx	—	—	—	—
	—	Px	0	PORTx	E	6	1	—	9	(SP)+ → Px				
	—	(Ay)+	1	E	9	0 Ay	1	—	9	(SP)+ → (Ay)+				
POPM	—	REGLIST	1	7	9	6	2	M	9+3n	(SP)+ → REGLIST NOTE: A minimum of 2 registers must be specified and may be specified in any order.	—	—	—	—
PUSH	—	Rx	1	REGx	A	E	1	—	6	Rx → −(SP)	—	—	—	—
	—	Px	0	PORTx	E	E	1	—	9	Px → −(SP)				
	—	−(Ay)	1	1 Ax	9	E	1	—	9	−(Ax) → −(SP)				
	—	#n16	1	E	9	F	2	#	9	#<data> → −(SP)				
PUSHM	—	REGLIST	1	F	9	E	2	M	7+3n	REGLIST → −(SP) NOTE: A minimum of 2 registers must be specified and may be specified in any order.	—	—	—	—
RET	—	—	5	6	D	7	1	—	9	(SP)+ → PC	—	—	—	—
	—	Ax	5	0 Ax	D	7	1	—	9	((Ax))+ → PC				
RETI	—	—	5	F	D	7	1	—	12	(SP)+ → SR (SP)+ → PC	*	*	*	*

INSTRUCTION MNEMONIC	ATTR SIZE	OPERAND ASSEMBLER SYNTAX Source=Src Destination=Dst Src , Dst	OPERATION WORD				WORDS	EXTENSION	CYCLES	OPERATION	STATUS FLAGS			
			15 14 13 12	11 10 9 8	7 6 5 4	3 2 1 0					N	Z	V	C
ROL ROLC ROR RORC		see ASL (page 10)												
SUB SUBC		see ADD (page 8)												
TEST TESTN	.B [.W]	#n16,Ry	7	7	s c1 0 0	REGy	2	#	6	TEST: Src .AND. Dst TESTN: Src .AND. .NOT.(Dst)	*	*	0	0
	.B [.W]	#n16,Py	7	F	s c1 0 1	PORTy	2	#	9					
	.B [.W]	#n16,(Ay)	7	7	s c1 0 1	0 Ay	2	#	9					

The following symbols are used to describe the state of the Status Register flags:

*	Set according to result of operation.
0	Cleared.
1	Set.
—	Not affected.
U	Undefined.

Port Map

Port		Addr	Description	15	14	13	12	11	10	9	8
P0	PH0 PL0	$FC00	16 External I/O or Addr/Data Bus In Expanded Bus Mode								
P1	PH1 PL1	$FC02	16 Ext I/O, Ext Interrupts, Serial Port I/O, Bus Control	LB LDS	HB UDS	R/W	DS BGACK	DTACK	AS	BUSIN BG	BUSOUT BR
P2	PH2 PL2	$FC04	Reserved	◄——— RESERVED ———►							
P3	PH3 PL3	$FC06	Serial Receive and Serial Transmit Buffers	RECEIVE BUFFER							
P4		$FC08	8 External I/O or Timer and Port 0 Handshake	TAO	TBO	TAI	TBI	STRH R/G	STRL	RDYH	RDYL
P5		$FC0A	Reserved	◄——— RESERVED ———►							
P6		$FC0C	Reserved	◄——— RESERVED ———►							
P7		$FC0E	Interrupt Latch Register	RES	NMI	SPARE	XI2I	STRLI	TAOI	TAII	STRHI
P8		$FC10	Interrupt Mask Register	RESERVED		SPARE	XI2M	STRLM	TAOM	TAIM	STRHM
P9		$FC12	Serial I/O Receive Control and Status Register	RE	IS	RW1	RW0	RC	SIS	RESERVED	
P10		$FC14	Serial I/O Transmit Control and Status Register	TE	AT	LM	TW1	TW0	TC	P/S	RES
P11		$FC16	Timer B Latch								
P12		$FC18	Timer A Low Latch								
P13		$FC1A	Timer A High Latch								
P14		$FC1C	Timer Control, Interrupt Edge Select	TEST	RESERVED		XI2C	RESERVED		TCOC	TAM1
P15		$FC1E	Port 0 Handshake mode, Bus Lock, Bus Segment Bits	SEG1	SEG0	BLCK	F/S	PM3	PM2	PM1	PM0
P16		$FC20	Port 0 Direction Control (DDR0)								
P17		$FC22	Port 1 Direction Control (DDR1)								
P18		$FC24	Serial I/O Mode and Sync Register	A/S	WL1	WL0	ST	PAR1	PAR0	TCO	WS
P19		$FC26	Timer C Latch								

Port Map

7	6	5	4	3	2	1	0	Initial Condition
								—
I/O	XI2	XI1	XI0	SI	RCLK	TCLK	SO	—
← RESERVED →								—
TRANSMIT BUFFER								—
← RESERVED →								High Byte 00----00
← RESERVED →								—
← RESERVED →								—
RSCI	RNI	XI1I	TBOI	TBII	XI0I	XMTI	TCI	$0000
RSCM	RNM	XI1M	TBOM.	TBIM	XI0M	XMTM	TCM	$0000
BF	OE	PE	FE	SF/AF	← RESERVED →			$0000
BE	UE	END	← RESERVED →					$00A0
								—
								—
								—
TAM0	TAE	TAIC	TAOC	TBM1	TBM0	TBE	TBIC	$0000
← RESERVED →								Mode pin: VCC-$0C00 CLKOUT-$0E00 GND-$0F00
								$0000
								$0000
SYNC7	SYNC6	SYNC5	SYNC4	SYNC3	SYNC2	SYNC1	SYNC0	$0000
								—

NOTE:
When a reserved bit is read, it is read as a zero.

PORT DESCRIPTIONS

Important note: All port bits that are not explicitly defined are reserved for possible use in future MK68200 family members. If reserved bits are written, they should be written with zero values. When reserved bits are read, they are read as zeros.

PORT 0 — 16 External I/O Bits; read/write
$FC00 (used as address/data lines in external bus configurations)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

PORT 1 —16 External I/O Bits, including Interrupt, Serial I/O,
$FC02 and Bus Control (shown for GP bus); read/write

$\overline{\text{L}}$B	$\overline{\text{H}}$B	R/$\overline{\text{W}}$	$\overline{\text{D}}$S	$\overline{\text{D}}$TACK	$\overline{\text{A}}$S	$\overline{\text{B}}$USIN	$\overline{\text{B}}$USOUT	I/O	XI2	XI1	XI0	SI	RCLK	TCLK	SO
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

PORT 2 —Reserved

PORT 3 —Serial I/O:Transmit Buffer (Low byte; read/write)
$FC06 Receive Buffer (High Byte; read only)

RECEIVE BUFFER								TRANSMIT BUFFER							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

PORT 4 —8 External I/O Pins (Timer and Port 0 Handshake Lines);
$FC08 Inputs: read only
Outputs: read/write

TAO	TBO	TAI	TBI	STRH	STRL	RDYH	RDYL	(reserved)							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Note: STRH is also R/$\overline{\text{G}}$ in the expanded bus modes.

PORT 5 —Reserved

PORT 6 —Reserved

PORT 7 —Interrupt Latch Register; read/write
$FC0E

res	NMI	SPARE	XI2I	STRLI	TAOI	TAII	STRHI	RSCI	RNI	XI1I	TBOI	TBII	XI0I	XMTI	TCI
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

PORT 8 —Interrupt Mask Register; read/write
$FC10

(res)		SPARE	XI2M	STRLM	TAOM	TAIM	STRHM	RSCM	RNM	XI1M	TBOM	TBIM	XI0M	XMTM	TCM
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

INTERRUPT AND RESET VECTORS

VECTOR NUMBER	NAME	MNEMONIC	VECTOR LOCATION	
0	RESET	RESET	$0000	
1	NON-MASKABLE INTERRUPT	NMI	$0002	LEVEL 2
2	SPARE	SPARE	$0004	LEVEL 1
3	EXTERNAL INTERRUPT 2	XI2	$0006	
4	STROBE LOW	STRL	$0008	
5	TIMER A OUTPUT	TAO	$000A	
6	TIMER A INPUT	TAI	$000C	
7	STROBE HIGH	STRH	$000E	
8	RECEIVE SPECIAL CONDITION	RSC	$0010	
9	RECEIVE NORMAL	RN	$0012	
A	EXTERNAL INTERRUPT 1	XI1	$0014	
B	TIMER B OUTPUT	TBO	$0016	
C	TIMER B INPUT	TBI	$0018	
D	EXTERNAL INTERRUPT 0	XI0	$001A	
E	TRANSMIT	XMT	$001C	
F	TIMER C	TC	$001E	

NOTE: Reset and Interrupt sources are listed in order of decreasing priority with RESET having the highest priority.

PORT 9 —Serial I/O Receive Control and Status Register;
$FC12 High byte: control register; read/write
Low byte: status register; read only

RE	IS	RW1	RW0	RC	SIS	(res)		BF	OE	PE	FE	SF/AF	(res)		
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bit Descriptions:

RE
(Receiver Enable control)
0 = Disabled; all status flags cleared.
1 = Enabled.

IS
(Ignore Syncs control)
0 = Disabled; interrupts may occur on all characters received.
1 = Enabled; interrupts cannot occur on sync characters received after the sync match is found.

RW1, RW0
(Receiver Wake-up control)
The receiver wake-up control bits operate as follows.

MODE	RW1	RW0	APPENDED WAKE-UP	BUFFER LOADED	INTERRUPT GENERATED
No Wake up	0	0	no	any character	RN
Wake-up on Any Character	0	1	yes	any character	RN
Wake-up on Address Match	1	0	yes	address match	RSC
Wake-up on Any Address	1	1	yes	any address	RSC

RC
(Receive Clock control)
0 = Selects external receive clock applied on RCLK.
1 = Selects internal clock from the on-chip baud rate generator (Timer C) for the receive clock.

This bit is ignored when either the TCO bit or the LM (Loopback Mode) bit is set.

SIS
(Single Interrupt Select control)

0 = Separate vectors are generated for the Receive Normal and the Receive Special Condition interrupts.
1 = The Receive Normal vector is generated for all receive character interrupts.

BF
(Buffer Full status)

0 = Receive data buffer empty; cleared when receive buffer is read.
1 = Receive buffer full; set when an incoming word is loaded into the receive data buffer.

OE
(Overrun Error status)

0 = No overrun error; cleared when the status register is read.
1 = Overrun error; set when a new word has been received and the previous word has not been read from the receive data buffer.

PE
(Parity Error status)

0 = No Parity Error; cleared when the status register is read.
1 = Parity Error; set when a parity error has been detected on an incoming character in the data stream.

FE
(Frame Error status)

0 = No frame error; cleared when the status register is read.
1 = Frame error; set when a word is transferred to the receive data and no stop bit has been recognized.

This flag applies to async formats only.

SF/AF
(Sync Found or Address Found status)

This flag is used for both sync character match conditions and address found conditions in some wake-up modes as follows.

	MODES			
SS	IS	RW1	RW0	CONDITIONS THAT SET SF/AF
1	x	x	x	Sync Found on any bit boundry
0	1	x	x	unaffected
0	0	0	0	unaffected
0	0	1	1	Any Address
0	0	1	0	Address Match
0	0	0	1	unaffected

PORT 10—Serial I/O Transmit Control and Status Register
$FC14 High byte: control register; read/write
Low byte: status register; read only

TE	AT	LM	TW1	TW0	TC	P/$\overline{S}$	res	BE	UE	END	(reserved)				
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bit Descriptions:

TE
(Transmitter Enable control)
0 = Disable the transmitter; any word being shifted out will continue until completion.
1 = Enable the transmitter.

AT
(Automatic Turn Around control)
0 = No effect on TE or RE.
1 = Causes RE to be set to a "1" and TE to be set to a "0" automatically at the end of a transmission.

LM
(Loopback Mode control)
0 = Disables loopback mode.
1 = Causes the transmitter output to be internally connected to receiver input. Also causes Timer C to be used for both the transmit and receive clocks regardless of the state of TC, RC, TCO, and TCOC.

TW1, TW0
(Transmit Wake-up control)
These bits provide control for wake-up operation as follows.

TW1	TW0	OPERATION
1	0	Transmit Data
1	1	Transmit Address
0	X	No Wake-up

TC
(Transmit Clock control)
0 = Selects the external clock signal applied on TCLK for the transmit clock.
1 = Selects the internal baud rate generator output (Timer C) for the transmit clock.

This bit is ignored if either the TCO bit or the LM bit is set.

$P/\overline{S}$
(Previous/Sync control)

0 = Selects continuous transmission of the contents of the sync character register in the synchronous mode when there is no data to transmit.
1 = Selects continuous transmission of the transmit data buffer in synchronous mode when there is no data to transmit.

BE
(Buffer Empty status)

0 = Transmit Buffer is full; reset to this condition after the transmit buffer is reloaded.
1 = Transmit Buffer is empty; set to this condition after the transmit buffer contents are transferred to the output shift register.

UE
(Underrun Error status)

0 = No underrun error; cleared following a read of the transmit buffer.
1 = Underrun error; set only in the synchronous mode when the last word has been shifted out and transmit buffer has not been reloaded.

END
(End of Transmission status)

0 = No end of transmission; cleared by enabling the transmitter.
1 = End of transmission detected; set when the transmitter is disabled and the last character has been shifted out.

PORT 11— Timer B Latch; read gets counter value;
$FC16 write goes to latch

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

PORT 12—Timer A, Low Latch; read gets counter or latch value;
$FC18 write goes to latch

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

PORT 13—Timer A, High Latch; read gets counter or latch value;
$FC1A write goes to latch

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

PORT 14—Timer Control, Interrupt Edge Select; read/write
$FC1C

TEST	(res)		XI2C	(res)		TCOC	TAM1	TAM0	TAE	TAIC	TAOC	TBM1	TBM0	TBE	TBIC
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bit Descriptions:

TEST (Test mode control)	0 = Normal operation; cleared on user control and on reset. 1 = Selects test mode; not to be used during normal operation.
XI2C (External Interrupt 2 Control)	0 = Interrupt on falling edge of XI2. 1 = Interrupt on rising edge of XI2.
TCOC (Timer C Output Control)	0 = When TCO = 1, TCLK is selected for use as a general purpose I/O pin. 1 = When TCO = 1, TCLK is selected for use as an output for Timer C. TCOC has no effect when TCO = 0.

TAM1, TAM0
(Timer A Mode control)

These bits select the operating mode of Timer A as follows.

TAM1	TAM0	MODE
0	0	Interval
0	1	Event
1	0	Pulse/period 1
1	1	Pulse/period 2

TAE
(Timer A Enable control)

0= Disables Timer A; all Timer A operations are inhibited, and the timer counter is initialized.
1= Enables Timer A; the timer begins operation as defined by the other Timer A control bits.

TAIC
(Timer A Input Control)

0= Selects a negative transition as the active edge for TAI.
1= Selects a positive transition as the active edge for TAI.

TAOC
(Timer A Output Control)

0= Selects TAO as a general purpose output pin.
1= Selects TAO as an ouput pin associated with Timer A; TAO is initialized low when TAOC is a one and TAE is zero.

TBM1, TBM0.
(Timer B Mode control)

These bits select the operating mode of Timer B as described below.

TBM1	TBM0	MODE
0	0	Interval 0 (TBO is not used)
0	1	Interval 1 (TBO is used)
1	0	Retriggerable one-shot
1	1	Non-retriggerable one-shot

TBE
(Timer B Enable control)

0= Disables Timer B; all operations are inhibited, and the timer counter is initialized.
1= Enables Timer B; the timer begins operation as defined by the other Timer B control bits.

TBIC
(Timer B Input Control)

0= Selects a negative transition as active on TBI.
1= Selects a positive transition as active on TBI.

PORT 15—Port 0 Handshake Mode, Fast/Standard, Bus Lock, and Bus
$FC1E Segment Bits; read/write

SEG1	SEG0	BLCK	F/$\overline{S}$	PM3	PM2	PM1	PM0	(reserved)							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bit Descriptions:

SEG1, SEG0 (Segment bits) — Used in the expanded bus mode when a reference is made to the DMA window. The contents of SEG1 and SEG0 are then output on pins AD15 and AD14, respectively.

BLCK (Bus Lock control) — 0 = Disables the bus lock function.
1 = Enables the bus lock function.

F/$\overline{S}$ (Fast/$\overline{\text{Standard}}$ timing control) — 0 = Selects standard timing of read/write cycles on the external bus (4 clock periods).
1 = Selects fast timing of read/write cycles on the external bus (3 clock periods.)

PM0, PM1, PM2, PM3 (Port Mode control) — These bits allow the user to select one of eight different handshaking modes.

PM3	PM2	PM1	PM0	HIGH HANDSHAKE	LOW HANDSHAKE
0	0	0	0	Inactive	PL0 or P0 output
0	0	0	1	PH0 output	PL0 output
0	0	1	0	Inactive	PL0 input
0	0	1	1	PH0 input	PL0 input
0	1	0	0	PH0 input	PL0 ouput
0	1	0	1	Inactive	P0 input (word only)
0	1	1	0	PL0 output	PL0 input (bidirectional)
0	1	1	1	P0 output	P0 input (bidirectional)
1	1	0	0	Inactive	Inactive
1	1	1	0	Full Expansion (62K external bus)	
1	1	1	1	Partial Expansion (16K external bus)	

PORT 16—Port 0 Data Direction Control (DDR0); read/write
$FC20

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

0—Corresponding Port 0 bit is input.
1—Corresponding Port 0 bit is output.

PORT 17—Port 1 Data Direction Control (DDR1); read/write
$FC22

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

0—Corresponding Port 1 bit is input.
1—Corresponding Port 1 bit is output.

PORT 18—Serial I/O Mode and Sync Register; read/write
$FC24

A/$\overline{\text{S}}$	WL1	WL0	ST	PAR1	PAR0	TCO	WS	SYNC7	SYNC6	SYNC5	SYNC4	SYNC3	SYNC2	SYNC1	SYNC0
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bit Descriptions:

A/$\overline{\text{S}}$
(Asynchronous/Synchronous mode control)

0 = Selects synchronous operation for the serial port; transmit and receive clocks are divided by 1.
1 = Selects asynchronous operation for serial port; transmit and receive clocks are divided by 16.

WL1, WLO
(Word Length control)

These two bits select the length of the data word as follows.

WL1	WL0	Word Length
0	0	8 bits
0	1	7 bits
1	0	6 bits
1	1	5 bits

ST
(Stop Bit control)

This bit is only used in the asynchronous mode. It selecfs the number of stop bits transmitted.

ST	Number of Stop Bits
0	1
1	2

PAR1, PAR0
(Parity control)

These two bits provide parity control for both the synchronous and asynchronous modes.

PAR1	PAR0	Parity
0	0	no parity
0	1	fixed "0" parity
1	0	odd parity
1	1	even parity

Note that even parity is defined such that the sum of the data and parity bits is even.

TCO
(Timer C Output mode control)

0= Disables Timer C output mode.
1= Enables Timer C output mode; disables Timer C's use as a baud rate generator when LM = 0; causes transmit and receive clocks to be internally connected to RCLK so that TCLK may be used either as general purpose I/O or as an output for Timer C.

WS

(Wake-up Sense)

The following table lists the effects of the WS bit.

WS	Wake-up bit	Meaning
0	0	Address Word
0	1	Data Word
1	0	Data Word
1	1	Address Word

SYNC7-SYNC0

(Sync character bits)

These eight bits are used to store the sync character or the device address for the wake-up mode.

PORT 19—Timer C Latch; read gets counter, write goes to latch and counter
$FC26

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

ELECTRICAL SPECIFICATIONS

MAXIMUM RATINGS

Temperature Under Bias	−25°C to +100°C
Storage Temperature	−65°C to +150°C
Voltage on Any Pin with Respect to Ground	−0.3 V to +7 V
Power Dissipation	1.5 W

Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other condition above those indicated in the operation sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

MK68200 DC ELECTRICAL CHARACTERISTICS

(V_{CC} = 5.0 V ± 5%, T_a = 0° to 70°C)

SYMBOL	PARAMETER	MIN	MAX	UNITS	TEST CONDITIONS
V_{IL}	Input low voltage; all inputs	−0.3	0.8	V	
V_{IH}	Input high voltage; all inputs	2.0	V_{CC}	V	
V_{OL}	Output low voltage; all outputs		0.4	V	I_{OL} = 2.0 mA
V_{OH}	Output high voltage; all outputs	2.4		V	I_{OH} = −250µA
I_{CC}	Input power supply current		220	mA	Outputs Open
I_{LI}	Input leakage current		±10	µA	V_{IN} = 0 to V_{CC}
I_{LO}	Three-state output leakage current in float		±10	µA	V_{OUT} = 0.4 V to V_{CC}

CAPACITANCE

T_a = 25°C, f = 12 MHz with unmeasured pins returned to ground.

SYMBOL	PARAMETER	MAX	UNIT	TEST CONDITION
C_{IN}	Input Capacitance	10	pF	Unmeasured pins returned to ground
C_{OUT}	Three-state Output Capacitance	10	pF	

MK68200 AC ELECTRICAL SPECIFICATIONS

T_a = 0°C to 70°C, V_{CC} = +5V ±5% unless otherwise specified.
AC measurements are referenced from minimum V_{IH} or maximum V_{IL} for inputs and from minimum V_{OH} or maximum V_{OL} for outputs.

NO.	DESCRIPTION	4 MHz		6MHz		UNITS	NOTES
		MIN	MAX	MIN	MAX		
1	$\overline{\text{RESET}}$ low time	20		20		state times	1
2	CLK 1 width high (external clock input)	45		30		ns	
3	CLK 1 width low (external clock input)	45		30		ns	
4	CLK 1 period (external clock input)	125	1000	83	1000	ns	
5	Crystal input frequency	1.000	8.000	1.000	12.000	MHz	
6	Clock Period (PHI 1)	250		167		ns	
7	PHI 1 low to PHI 1 high	125		83		ns	
8	PHI 1 high to PHI 1 low	125		83		ns	
9	PHI 1 low to CLKOUT low		40		27	ns	
10	PHI 1 high to CLKOUT high		40		27	ns	

MK68200 EXPANDED BUS AC ELECTRICAL SPECIFICATIONS (UPC, GP, AND PRIVATE BUSES)

NO.	DESCRIPTION	4MHz		6MHz		UNITS	NOTES
		MIN	MAX	MIN	MAX		
11	PHI 1 low to R/$\overline{\text{W}}$, $\overline{\text{HB}}$, or $\overline{\text{LB}}$ valid		115		76	ns	2
12	PHI 1 high to $\overline{\text{AS}}$ low		115		76	ns	2
13	PHI 1 low to address valid		115		76	ns	2
14	$\overline{\text{AS}}$ low to address invalid	70		50		ns	2
15	PHI 1 low to tri-state address		90		60	ns	2
16	Tri-state address to $\overline{\text{DS}}$, $\overline{\text{LDS}}$, or $\overline{\text{UDS}}$ starting low (fast cycle)	10		10		ns	2
17	PHI 1 low to $\overline{\text{DS}}$, $\overline{\text{LDS}}$, or $\overline{\text{UDS}}$ low (fast cycle)		165		110	ns	2
18	PHI 1 low to data out valid during write		115		76	ns	2
19	PHI 1 low to R/$\overline{\text{W}}$, $\overline{\text{HB}}$, $\overline{\text{LB}}$ invalid	0		0		ns	2
20	PHI 1 low to address/data bus driven	0		0		ns	2
21	$\overline{\text{AS}}$ low to $\overline{\text{DS}}$, $\overline{\text{LDS}}$, or $\overline{\text{UDS}}$ starting low (fast cycle)	100	225	70	150	ns	2

MK68200 EXPANDED BUS AC ELECTRICAL SPECIFICATIONS (UPC AND GP BUSES)

NO.	DESCRIPTION	4 MHz		6MHz		UNITS	NOTES
		MIN	MAX	MIN	MAX		
22	Tri-state address to $\overline{\text{DS}}$, $\overline{\text{LDS}}$, or $\overline{\text{UDS}}$ starting low (standard cycle)	135		90		ns	
23	PHI 1 high to $\overline{\text{DS}}$, $\overline{\text{LDS}}$, or $\overline{\text{UDS}}$ low (standard cycle)		165		110	ns	2
24	Valid Data Setup to PHI 1 low	10		5		ns	2
25	$\overline{\text{AS}}$ low to $\overline{\text{DS}}$, $\overline{\text{LDS}}$, or $\overline{\text{UDS}}$ starting low (standard cycle)	225	350	150	230	ns	2
26	R/$\overline{\text{W}}$, $\overline{\text{HB}}$, or $\overline{\text{LB}}$ valid to $\overline{\text{AS}}$ starting low	60		60		ns	
27	Address valid to $\overline{\text{AS}}$ starting low	60		60		ns	
28	Irput data hold time from PHI 1 low	45		30		ns	
29	Input data hold time from $\overline{\text{DS}}$, $\overline{\text{LDS}}$, or $\overline{\text{UDS}}$ high	0		0		ns	
30	PHI 1 low to $\overline{\text{DS}}$, $\overline{\text{LDS}}$, or $\overline{\text{UDS}}$ high		180		120	ns	
31	$\overline{\text{DTACK}}$ low setup to PHI 1 high	15		10		ns	
32	$\overline{\text{LDS}}$, $\overline{\text{UDS}}$, or $\overline{\text{DS}}$ high to $\overline{\text{DTACK}}$ high (hold time)	−30		−30		ns	
33	$\overline{\text{LDS}}$, $\overline{\text{UDS}}$, or $\overline{\text{DS}}$ pulse width	240		150		ns	
34	PHI 1 high to $\overline{\text{AS}}$ high		90		60	ns	
35	PHI 1 low to data out invalid	0		0		ns	
36	$\overline{\text{AS}}$ inactive	235		150		ns	
37	$\overline{\text{DS}}$, $\overline{\text{LDS}}$, or $\overline{\text{UDS}}$ high to data out invalid	180		110		ns	
38	$\overline{\text{DS}}$, $\overline{\text{LDS}}$, or $\overline{\text{UDS}}$ high to $\overline{\text{AS}}$ high	5		5		ns	

MK68200 EXPANDED BUS AC ELECTRICAL SPECIFICATIONS (UPC BUS)

NO.	DESCRIPTION	4 MHz		6 MHz		UNITS	NOTES
		MIN	MAX	MIN	MAX		
39	$\overline{\text{BGACK}}$ low to $\overline{\text{BR}}$ high	100	450	100	300	ns	
40	$\overline{\text{BG}}$ low to $\overline{\text{BGACK}}$ low	50	600	50	400	ns	
41	$\overline{\text{BGACK}}$, $\overline{\text{AS}}$, $\overline{\text{DTACK}}$, inactive to $\overline{\text{BGACK}}$ low; $\overline{\text{BG}}$ already low	0	600	0	400	ns	
42	$\overline{\text{BGACK}}$ low to $\overline{\text{AS}}$, $\overline{\text{UDS}}$, $\overline{\text{LDS}}$, or address/data bus driven	40	135	40	90	ns	
43	$\overline{\text{AS}}$, $\overline{\text{LDS}}$, $\overline{\text{UDS}}$ or address/data bus tri-state to $\overline{\text{BGACK}}$ high	0	180	0	120	ns	

MK68211 EXPANDED BUS AC ELECTRICAL SPECIFICATIONS (GP BUS)

NO.	DESCRIPTION	4 MHz		6 MHz		UNITS	NOTES
		MIN	MAX	MIN	MAX		
44	Tri-state $\overline{\text{AS}}$, $\overline{\text{DS}}$, R/$\overline{\text{W}}$, $\overline{\text{LB}}$, $\overline{\text{HB}}$ to $\overline{\text{BUSOUT}}$ low (bus grantor, fast cycle, (no wait states)	175		100		ns	
45	$\overline{\text{BUSIN}}$ low to $\overline{\text{BUSOUT}}$ low (bus grantor, fast cycle, no wait states)		1900		1200	ns	
46	$\overline{\text{BUSOUT}}$ high to $\overline{\text{AS}}$, R/$\overline{\text{W}}$, $\overline{\text{LB}}$, $\overline{\text{HB}}$ driven (bus grantor)	15		15		ns	
47	$\overline{\text{BUSIN}}$ high to $\overline{\text{BUSOUT}}$ high (bus grantor)	520	900	300	600	ns	
48	Tri-state address/data bus to $\overline{\text{BUSOUT}}$ low (bus grantor)	70		70		ns	
49	$\overline{\text{BUSOUT}}$ high to address/data bus driven (bus grantor)	50		50		ns	
50	$\overline{\text{BUSOUT}}$ low to $\overline{\text{AS}}$, $\overline{\text{DS}}$, R/$\overline{\text{W}}$, $\overline{\text{LB}}$, $\overline{\text{HB}}$ driven (bus requestor, $\overline{\text{BUSIN}}$ low)	240		150		ns	
51	$\overline{\text{BUSIN}}$ low to $\overline{\text{AS}}$, $\overline{\text{DS}}$, R/$\overline{\text{W}}$, $\overline{\text{LB}}$, $\overline{\text{HB}}$ driven (bus requestor, $\overline{\text{BUSOUT}}$ low)	270	650	180	500	ns	
52	Tri-state $\overline{\text{AS}}$, $\overline{\text{DS}}$, R/$\overline{\text{W}}$, $\overline{\text{LB}}$, $\overline{\text{HB}}$, to $\overline{\text{BUSOUT}}$ high (bus requestor)	180		100		ns	
53	$\overline{\text{BUSOUT}}$ high to $\overline{\text{BUSIN}}$ high (bus requestor)		530		400	ns	
54	$\overline{\text{BUSIN}}$ low to address/data bus driven (bus requestor)	350		250		ns	
55	Tri-state address/data bus to $\overline{\text{BUSOUT}}$ high (bus requestor)	100		65		ns	

MK68E200 BUS AC ELECTRICAL SPECIFICATIONS (PRIVATE BUS)

NO.	DESCRIPTION	4 MHz		6 MHz		UNITS	NOTES
		MIN	MAX	MIN	MAX		
56	Valid Data Setup to PHI 1 low	30		20		ns	
57	PBR/$\overline{\text{W}}$ valid to $\overline{\text{PBAS}}$ starting low	40		40		ns	
58	Address valid to $\overline{\text{PBAS}}$ starting low	35		35		ns	
59	Input data hold time from PHI 1 low	0		0		ns	
60	Input data hold time from $\overline{\text{PBDS}}$ high	−25		−25		ns	

MK68E200 EXPANDED BUS AC ELECTRICAL SPECIFICATIONS (PRIVATE BUS) (Cont.)

NO.	DESCRIPTION	4 MHz		6 MHz		UNITS	NOTES
		MIN	MAX	MIN	MAX		
61	PHI 1 low to $\overline{\text{PBDS}}$ high		160		105	ns	
62	$\overline{\text{PBDTACK}}$ low setup to PHI 1 high	20		15		ns	
63	PBDS high to $\overline{\text{PBDTACK}}$ high (hold time)	−15		−15		ns	
64	$\overline{\text{PBDS}}$ pulse width	190		125		ns	
65	PHI 1 high to $\overline{\text{PBAS}}$ high		115		75	ns	
66	PHI 1 low to data out invalid	10		10		ns	
67	$\overline{\text{PBAS}}$ inactive	200		135		ns	
68	$\overline{\text{PBDS}}$ high to data out invalid	200		135		ns	
69	$\overline{\text{PBDS}}$ high to $\overline{\text{PBAS}}$ high	15		15		ns	

MK68200 INPUT/OUTPUT AC ELECTRICAL CHARACTERISTICS

NO.	DESCRIPTION		4 MHz		6 MHz		UNITS	NOTES
			MIN	MAX	MIN	MAX		
70	Active and inactive pulse times	For XI2, XI1, STRH, STRL, TAI, TBI, $\overline{\text{NMI}}$	5		5		state times	1
		For XI0	3		3			
71	Input data setup to falling edge of STRH, STRL		15		10		ns	
72	Input data hold from the falling edge of STRH, STRL		60		40		ns	
73	RDYH, RDYL low time		1	3	1	3	state times	1
74	Delay from STRH, STRL high to RDYH, RDYL low			110		75	ns	
75	Delay from data valid to RDYH, RDYL high (output mode)			3		3	state times	1
76	Delay from STRH high to data out (bidirectional mode)			90		60	ns	
77	Port 0 data hold time from STRH low (bidirectional mode)		25		20		ns	
78	Delay to Port 0 float from STRH low (bidirectional mode)			85		55	ns	
79	TCLK,RCLK period (asynchronous)	as input	.250	DC	.167	DC	µs	
		as output	.500	DC	.334	DC		
	TCLK,RCLK period (synchronous)		1.0	DC	.667	DC		
80	TCLK, RCLK width low	as input	1	DC	1	DC	state times	1
		as output	2	DC	2	DC		
81	TCLK, RCLK width high	as input	1	DC	1	DC	state times	1
		as output	2	DC	2	DC		

MK68200 INPUT/OUTPUT AC ELECTRICAL SPECIFICATIONS

NO.	DESCRIPTION		4 MHz		6 MHz		UNITS	NOTES
			MIN	MAX	MIN	MAX		
82	TCLK low to SO delay (sync mode)	TCLK as input	330		220		ns	
		TCLK as output	75		50			
83	SI to RCLK high setup time (sync mode)	RCLK as input	30		20		ns	
		RCLK as output	180		120			
84	SI hold time from RCLK high (sync mode)	RCLK as input	45		30		ns	
		RCLK as output	0		0			

NOTES
1. One state time is equal to one-half of the instruction clock (PHI 1) period.
2. For the private bus case, the signals referenced apply to the equivalent private bus signals.

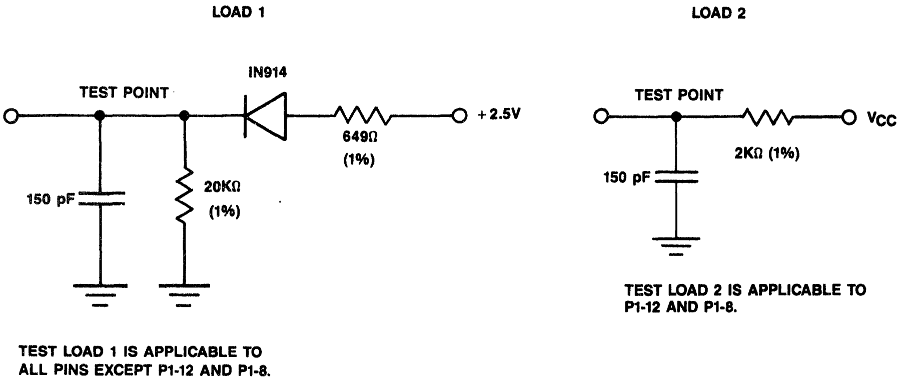

LOAD 1
IN914
TEST POINT
+2.5V
649Ω
(1%)
150 pF
20KΩ
(1%)
TEST LOAD 1 IS APPLICABLE TO ALL PINS EXCEPT P1-12 AND P1-8.
LOAD 2
TEST POINT
Vcc
2KΩ (1%)
150 pF
TEST LOAD 2 IS APPLICABLE TO P1-12 AND P1-8.

Figure 19. Output Test Load

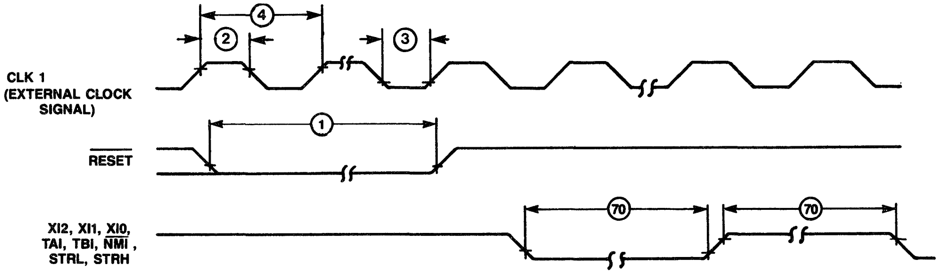


Figure 20. MK68200 AC Timing

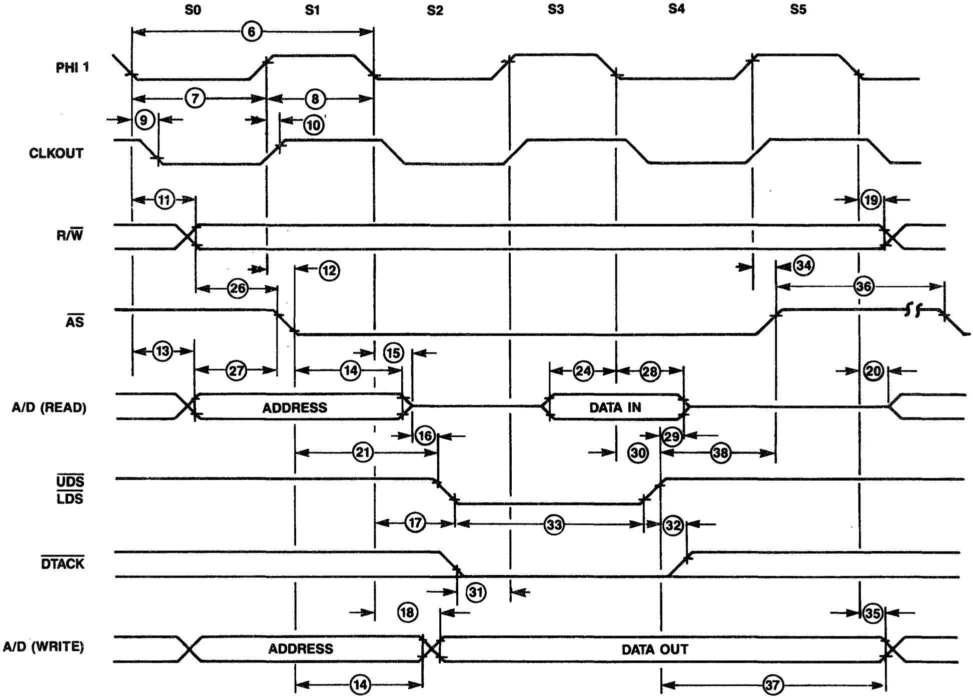


Figure 21. MK68201 UPC Bus Timing (Fast Cycle)

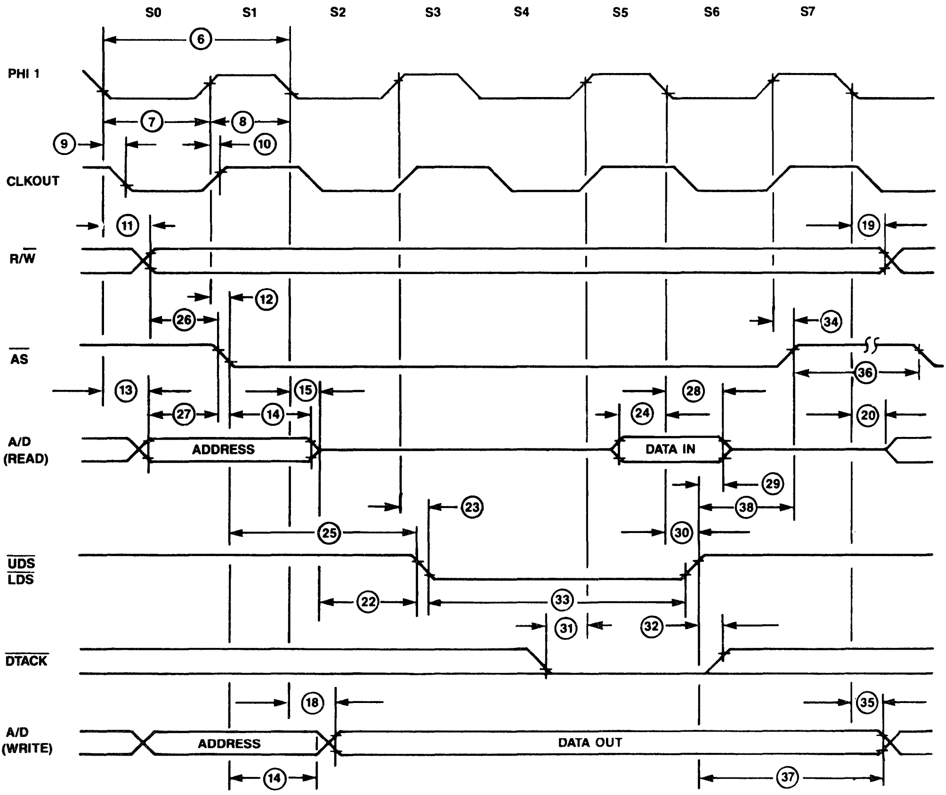


Figure 22. MK68201 UPC Bus Timing (Standard Cycle)

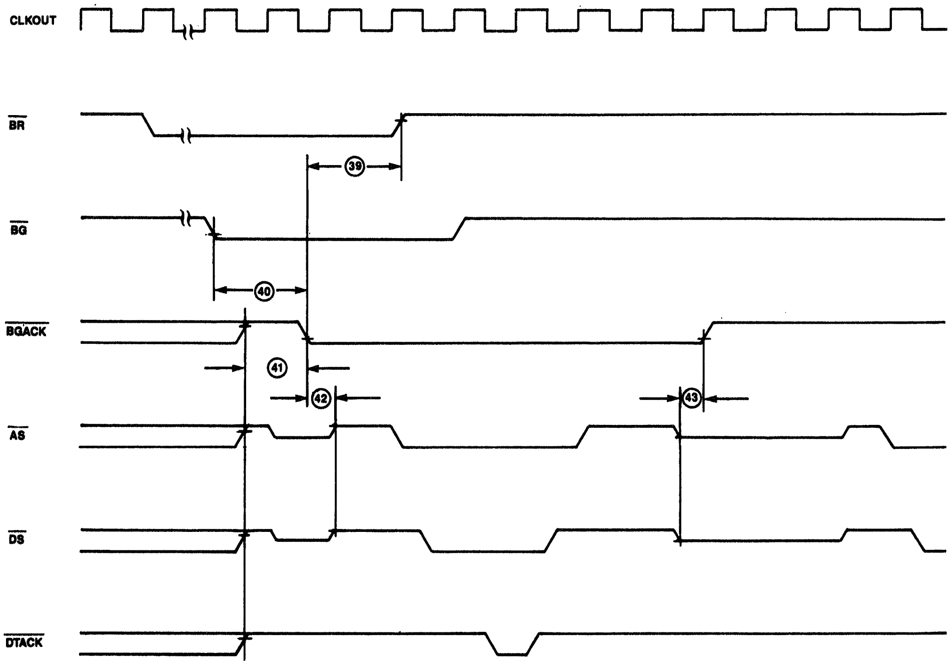


Figure 23. MK68201 UPC Bus Arbitration Timing

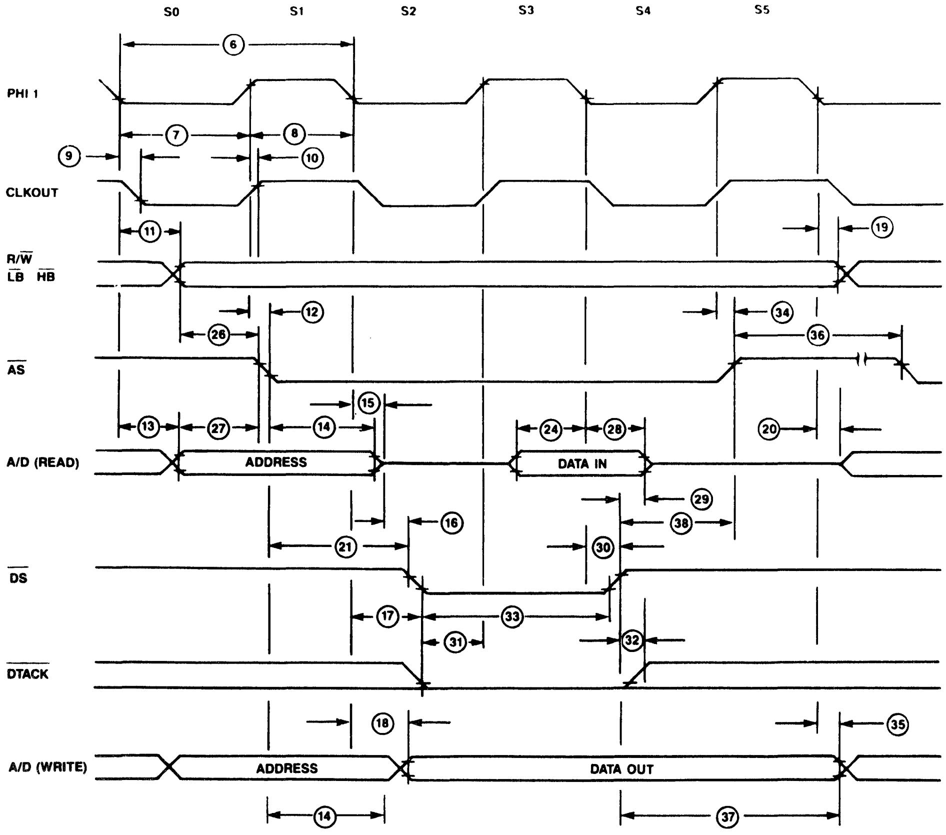


Figure 24. MK68211 GP Bus Timing (Fast Cycle)

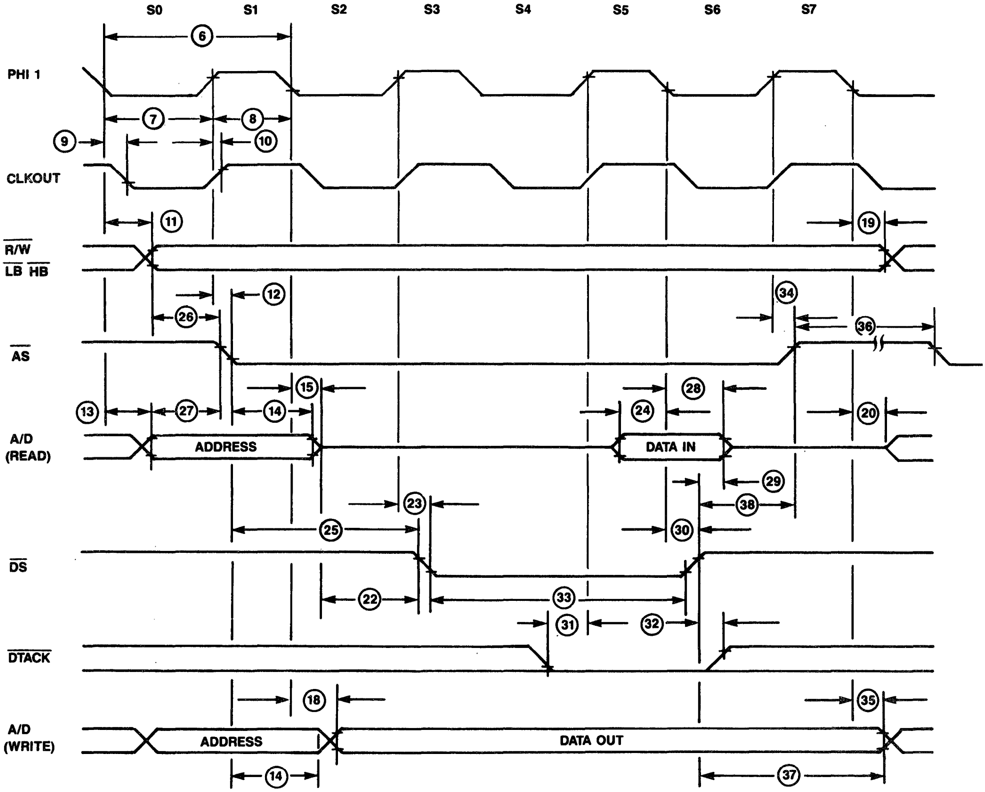


Figure 25. MK68211 GP Bus Timing (Standard Cycle)

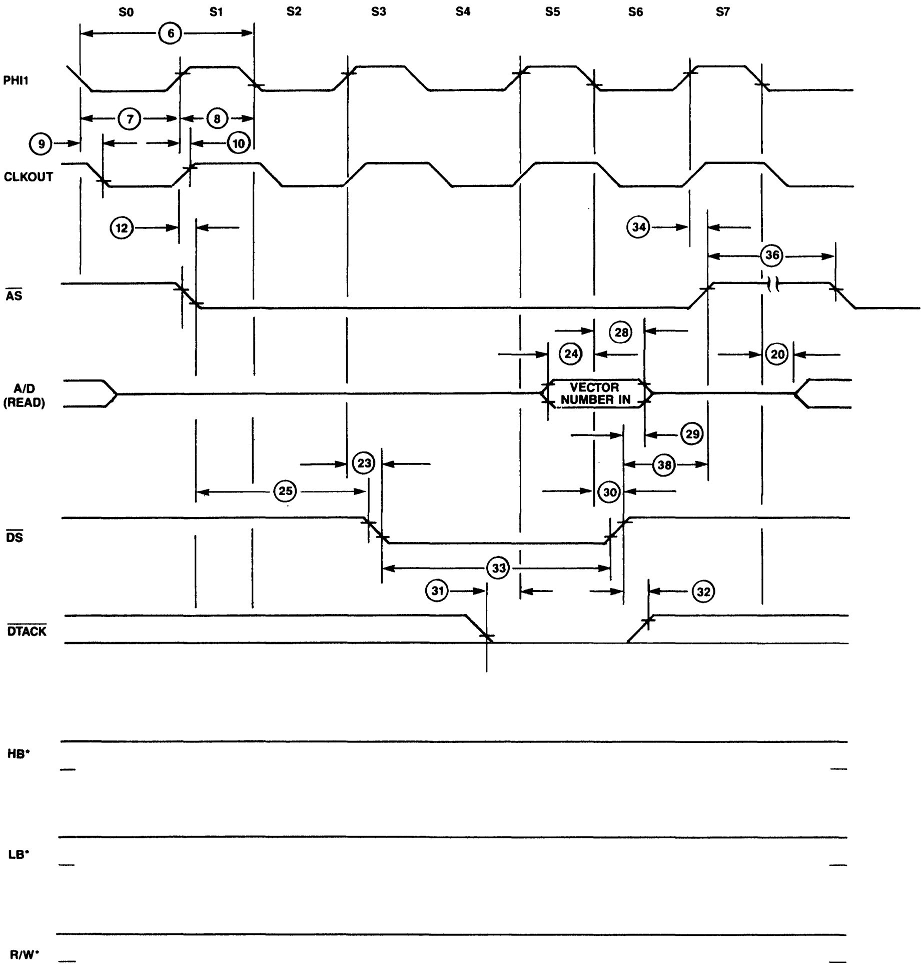


Figure 26. MK68211 GP Bus Timing (Interrupt Acknowledge Timing)

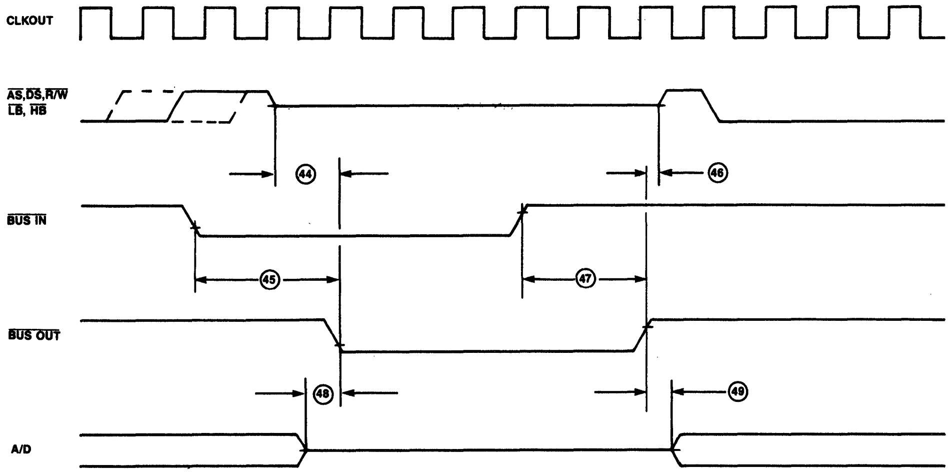


Figure 27. MK68211 GP Bus Arbitration Timing (Bus Grantor)

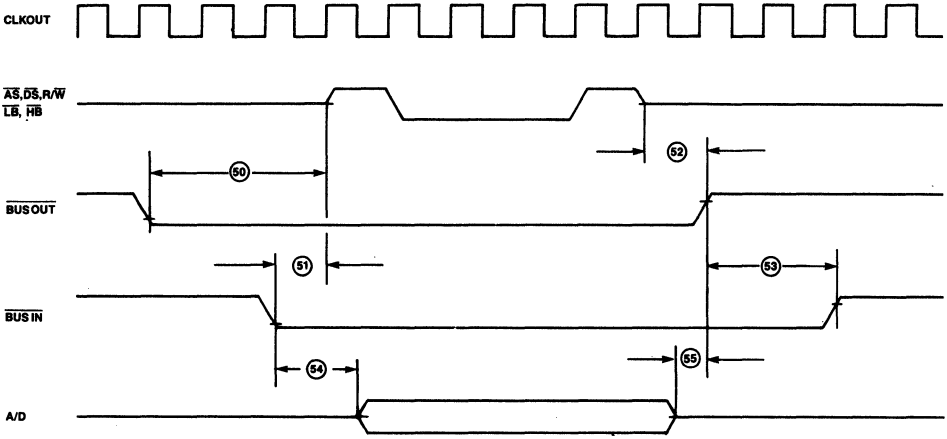


Figure 28. MK68211 GP Bus Arbitration Timing (Bus Requestor)

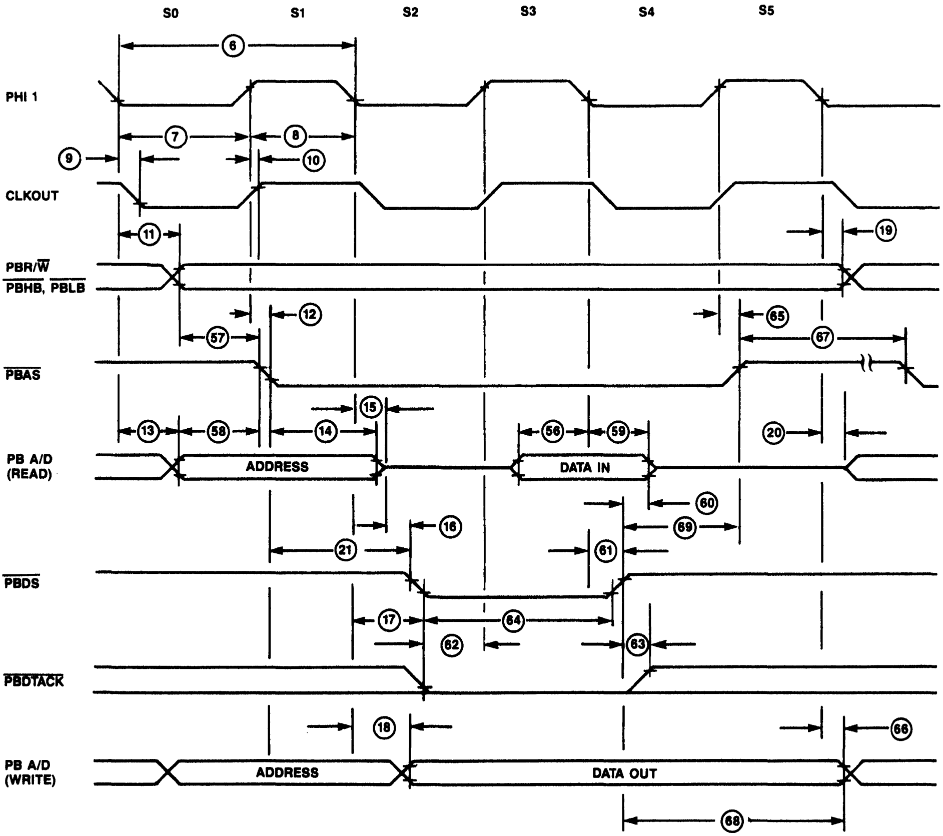


Figure 29. MK68200 Private Bus Timing (Fast Cycle)

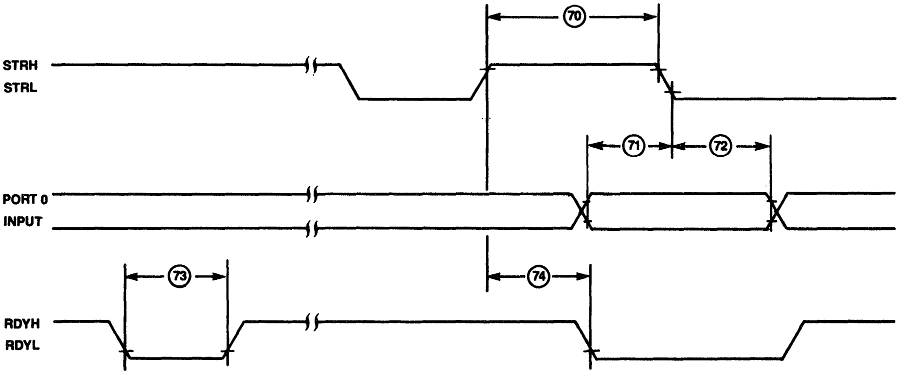


Figure 30. Input/Output AC Timing (Data Input)

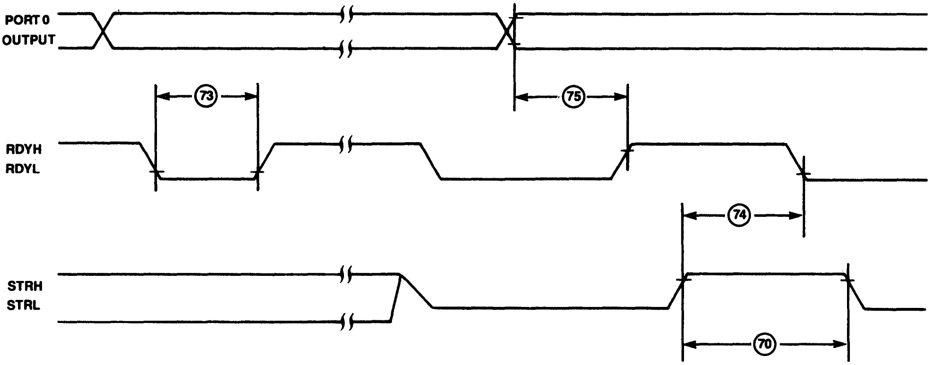


Figure 31. Input/Output AC Timing (Data Output)

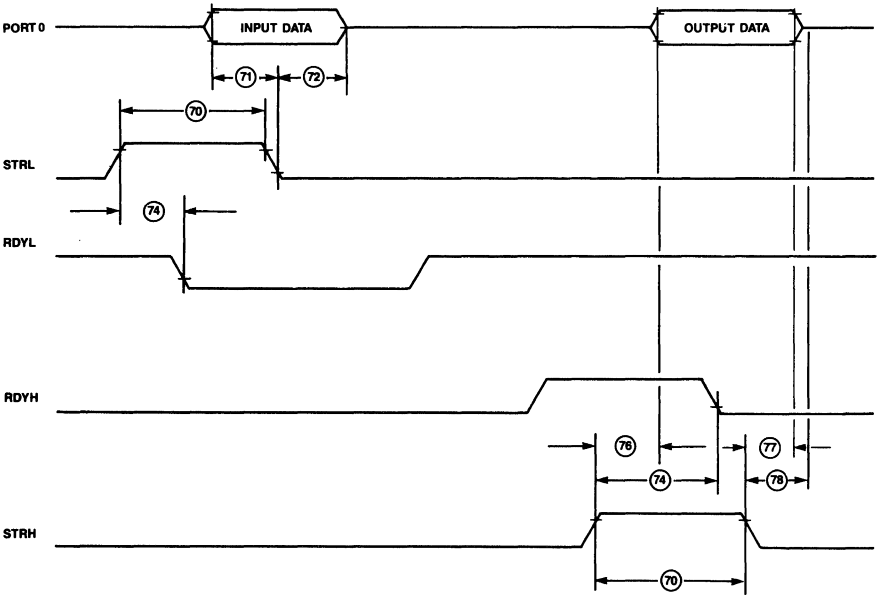


Figure 32. Input/Output AC Timing (Bidirectional I/O)

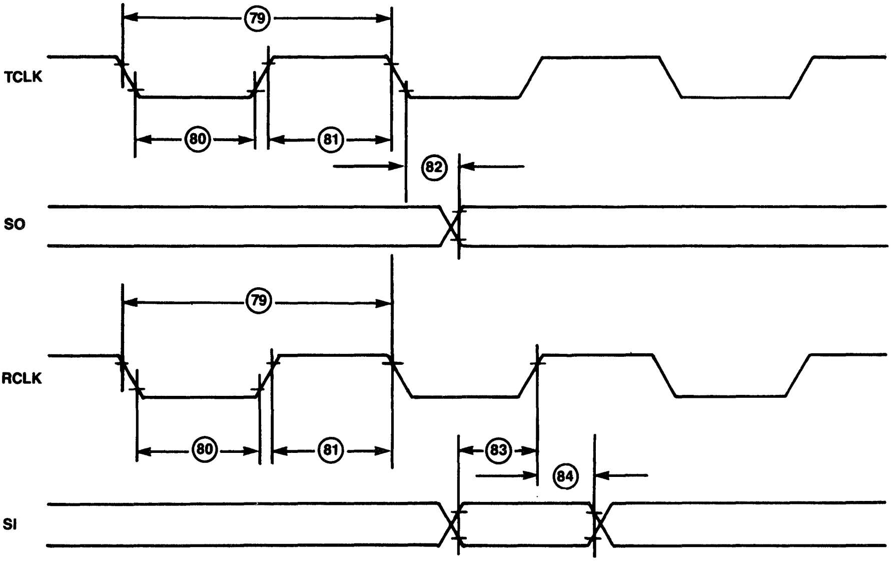
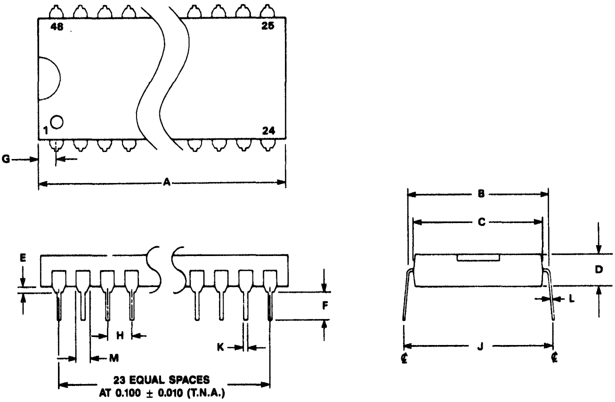


Figure 33. Input/Output AC Timing (Serial I/O)

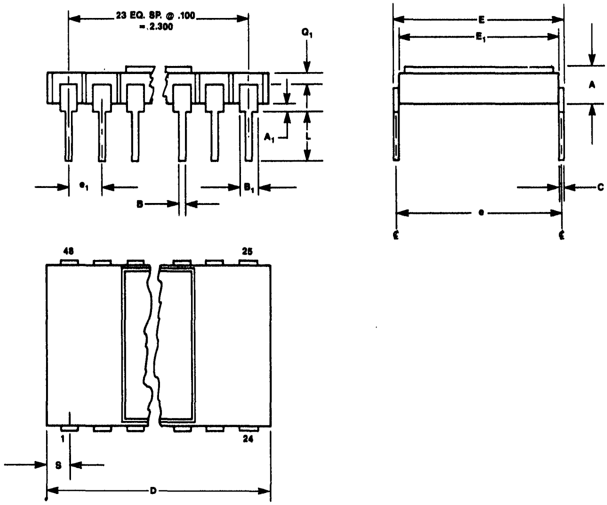
PHYSICAL DIMENSIONS

MK68200 48-Pin Plastic Dual-In-Line Package (N)



	MILLIMETERS		INCHES	
DIM.	MIN	MAX	MIN	MAX
A	61.468	62.738	2.420	2.470
B	14.986	16.256	0.590	0.640
C	13.462	13.97	0.530	0.550
D	3.556	4064	0.140	0.160
E	0.381	1.524	0.015	0.060
F	3048	3.81	0.120	0.150
G	1.524	2.286	0.060	0.090
H	1.186	1.794	0.090	0.110
J	15.24	17.78	0.600	0.700
K	0.381	0.533	0.015	0.021
L	0.203	0.305	0.008	0.012
M	1.143	1.778	0.045	0.070

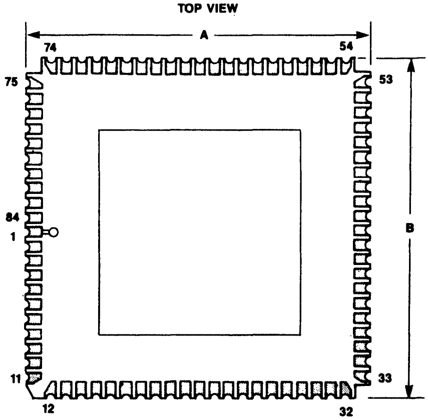
MK68200 48-Pin Ceramic Dual-In-Line Package (P)

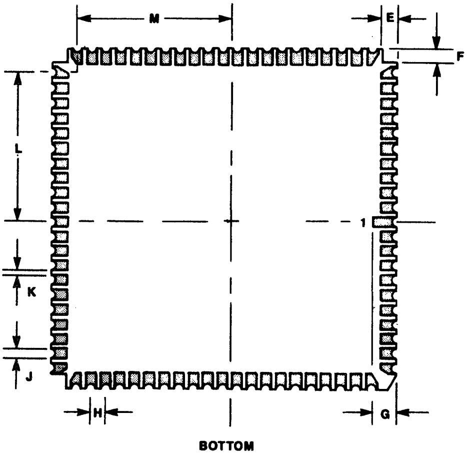


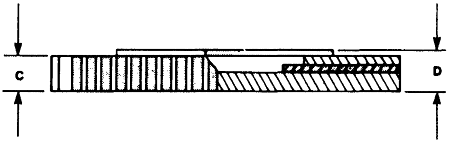
	INCHES		
DIM.	MIN.	MAX.	NOTES
A	.085	.190	
A_1	.020	.070	1
B	.015	.023	
B_1	.038	.060	
C	.008	.012	
D	2.370	2.430	
E	.595	.625	
E_1	.580	.610	
e	.590	.700	2
e_1	.100		
L	.120	.170	
Q_1	.010		
S	.035	065	

NOTES
1. Package stand off to be measured per JEDEC requirements.
2. Measured from centerline to centerline at lead tips.

MK68E200 84-Pin
Ceramic Leadless Chip Carrier (E)







DIM.	INCHES	
	MIN	MAX
A	1.138	1.167
B	1.138	1.167
C	0.070	0.090
D	0.080	0.110
E	0.044	0.056
F	0.044	0.056
G	0.075	0.095
H	0.048	0.052
J	0.033	0.039
K	0.010	0.018
L	0.495	0.505
M	0.495	0.505

ORDERING INFORMATION

There are two types of part numbers for the 68200 family of devices. The generic part number describes the basic device type, the amount of ROM and RAM, the desired package type, temperature range, power supply tolerance, and expandable bus interface type. The device order number indicates the specific mask set Mostek will use to manufacture the device, along with package type, speed grade and temperature range.

Generic Part Number

An example of the generic part number is shown below:

M K 6 8 2 0 1 / 4 4 N - 0 6

Denotes maximum instruction clock frequency.
4 = 4 MHz
6 = 6 MHz

Denotes operating temperature range.
0 = 0°C to +70°C
1 = −40°C to +85°C[3]

Package Type.
P = Ceramic DIP
N = Plastic DIP
E = Leadless Chip Carrier[1]
Q = Plastic Leaded Chip Carrier[3]
G = Pin Grid[1,3]

RAM Designator.
4 = 256 Bytes
8 = 512 Bytes
C = 1024 Bytes[1,3]

ROM Designator.
0 = None[2]
4 = 4K Bytes

Basic Device Type.
68201 = UPC Bus
68211 = GP Bus
68E201 = Emulator with UPC Bus
68E211 = Emulator with GP Bus
68E221 = Emulator with GP/$\overline{\text{UPC}}$ input pin[3]

NOTES

1. Available for emulator only.
2. Must be "0" when specifying the emulator.
3. Contact Mostek for availability.

Device Order Number

An example of the device order number is shown below:

MK41152N-06

Denotes maximum instruction clock frequency.
4 = 4 MHz
6 = 6 MHz

Denotes operating temperature range.
0 = 0°C to +70°C
1 = −40°C to +85°C[3]

Package Type.
P = Ceramic DIP[2]
N = Plastic DIP
E = Ceramic Leadless Chip Carrier[1]
Q = Plastic Leaded Chip Carrier[3]
G = Ceramic Pin Grid Array[1,3]

Version/Code Specific Number.
For MK68201, MK68211:
002 = No ROM version (standard catalog item)
0xx = Reserved for catalog codes
100 - 999 = Customer Code (Mostek Assigned)
For MK68E201, MK68E211, MK68E221:
000 = UPC, 256 bytes RAM, no ROM
010 = GP, 256 bytes RAM, no ROM
020 = GP/UPC, 1024 bytes RAM, no ROM[3]

Basic Device Type.
40 = Emulator Version
41 = UPC, 0 or 4K bytes ROM,
256 bytes RAM
42 = GP, 0 or 4K bytes ROM,
256 bytes RAM

NOTES
1. Available for emulator only.
2. Intended for prototype orders only.
3. Contact Mostek for availability.

PART NUMBER EXAMPLES (A noninclusive list)	

MK68201/44N-04	Device Order Number = MK41XXXN-04
	Speed = 4MHz
	Temperature = 0° to 70°C
	Package = 48 pin plastic
	RAM = 256 bytes
	ROM = 4096 bytes
	Bus = UPC

MK68211/04N-06	Device Order Number = MK42002N-06
	Speed = 6MHz
	Temperature = 0° to 70°C
	Package = 48 pin plastic
	RAM = 256 bytes
	ROM = None
	Bus = GP

MK68E211/04E-14	Device Order Number = MK40010E-14
	Speed = 4MHz
	Temperature = −40° to +85°C
	Package = 84 pin ceramic LCC
	RAM = 256 bytes
	ROM = None
	Bus = GP

MK68E221/0CG-06	Device Order Number = MK40020G-06
	Speed = 6MHz
	Temperature = 0° to 70°C
	Package = 84 lead PGA
	RAM = 1024 bytes
	ROM = None
	Bus = GP/UPC

These specifications are subjet to change without notice.
Please inquire with our sales offices about the availability of the different products.

Printed in France

Ref.DSMK68200

THOMSON COMPONENTS SEMICONDUCTEURS

MK68HC200 FAMILY

16-BIT SINGLE CHIP MICROCOMPUTERS

MK68HC201/MK68HC211/MK68HC221

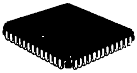
ADVANCE INFORMATION

HCMOS

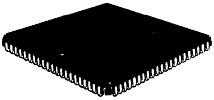
FEATURES

- 16-bit high performance, single-chip microcomputer
- Modular architecture
- Power saving stop and idle modes
- 14 address and data registers
  - — Eight 16-bit or sixteen 8-bit data registers
  - — Six 16-bit address registers
- Advanced 16-bit instruction set
  - — Bit, oyte and word operands
  - — Nine addressing modes
  - — Byte and word BCD arithmetic
- High performance (10 MHz instruction clock)
  - — 0.3 µs register-to-register move or add
  - — 2.1 µs 16 × 16 multiply
  - — 2.3 µs 32/16 divide
- Available with 0 or 4k bytes of ROM
- 256 bytes of RAM
- Three 16-bit timers
  - — Interval modes
  - — Event modes
  - — One-shot modes
  - — Pulse and period measurement modes
- Serial channel
  - — Double buffered receive and transmit
  - — Asynchronous to 625Kbps
  - — Synchronous to 2.5Mbps
  - — Address wake-up recognition and generation
  - — Internal/external baud rate generation
- Parallel I/O on first version
  - — Up to 40 pins
  - — Direction programmable by bit
  - — One 16-bit or 2 8-bit port(s) with handshaking
- Interrupt controller
  - — 16 independent vectors on first version
  - — Expandable to handle 64 independent interrupts
  - — Eight external interrupt sources
  - — One non-maskable interrupt
  - — Individual interrupt masking
- Optional external bus
  - — 16-bit multiplexed address/data bus
  - — Automatic bus request/grant arbitration
  - — Two control bus versions
    - • 68000 compatible bus (UPC) (MK68HC201)
    - • General Purpose bus (GP) (MK68HC211)
- Emulator version available
  - — Added private bus
  - — No on-chip ROM
  - — 512 bytes on-chip RAM
  - — GP or UPC bus version with one part (MK68HC221)
- 16, 20, and 25 MHz time base versions produce 8, 10 and 12.5 MHz instruction clock rates respectively
  - — Crystal or external TTL clock
- Single +5 volt po. er supply
- Plastic or Ceramic Chip carrier packaging

CASES



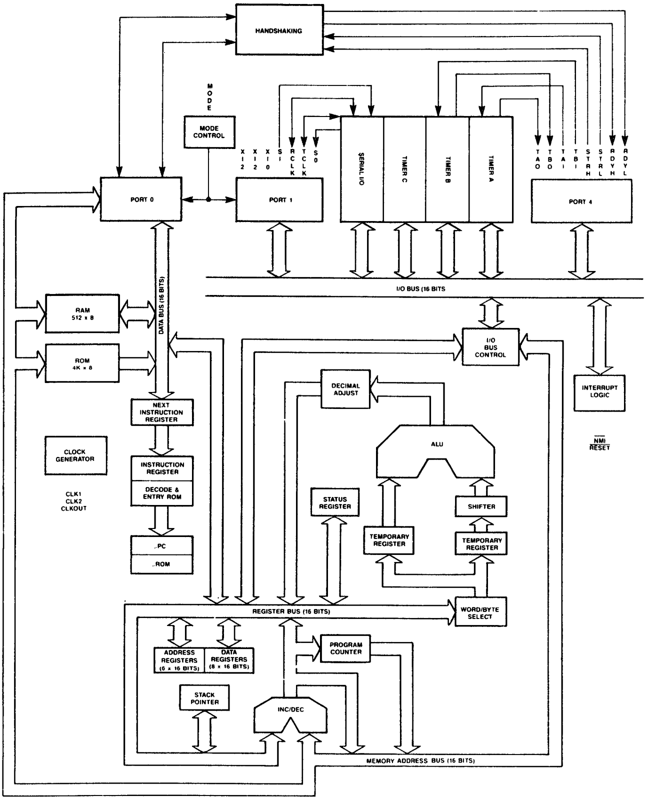
LCC 52



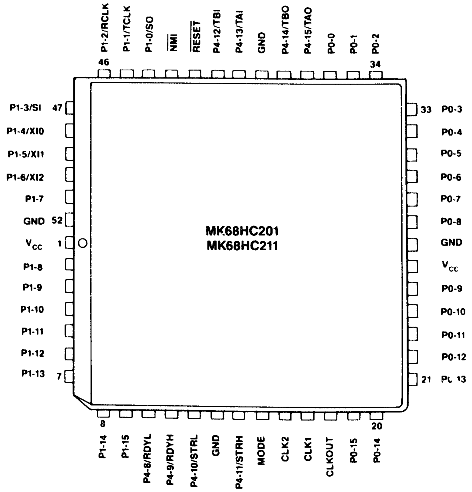
LCC 84

March 1987 - 1/73

BLOCK DIAGRAM



PIN ASSIGNMENT



PIN DESCRIPTION

SINGLE-CHIP DESCRIPTION

Figure 1 illustrates the functions of specific pins for an MK68HC201 or MK68HC211, operating in a single-chip mode. When the device is operating in one of the expanded bus modes, the pins on Port 0 become the multiplexed address/data bus, and the upper half of Port 1 becomes the control signals (GP or UPC) for the bus. The following description applies to the pins only when the device is used in the non-expanded or single-chip mode. Descriptions of the pin functions for the expanded bus modes are in the Expanded Bus Operation section of this data sheet.

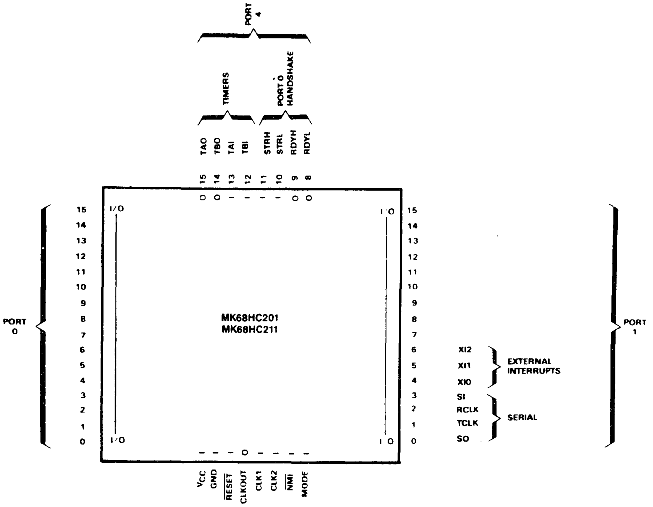


Figure 1. MK68HC201, MK68HC211 Logical Pinout, Single-Chip Mode

MNEMONIC	PIN FUNCTIONS FOR SINGLE-CHIP OPERATION
V_{CC}	Supply voltage 4.5 to 5.5 V
GND	Ground
$\overline{\text{RESET}}$	Reset (input, active low) - $\overline{\text{RESET}}$ input overrides ongoing execution (including interrupts) and resets the chip to its initial power-up condition. $\overline{\text{RESET}}$ cannot be masked.
CLKOUT	Clock Output (output) - CLKOUT will output the instruction clock rate, which is one-half of the frequency provided on CLK1 and CLK2.
CLK1, CLK2	Time base inputs (inputs) - CLK1 and CLK2 may be connected to a crystal, or CLK1 may be connected to an external TTL-compatible oscillator while CLK2 is left floating.
$\overline{\text{NMI}}$	Non-maskable interrupt (input, active low, negative edge triggered) - The $\overline{\text{NMI}}$ request line has a higher priority than all of the maskable interrupts. $\overline{\text{NMI}}$ is always enabled regardless of the state of the L1E (Level 1 Interrupt Enable) bit in the Status Register.
MODE	Mode (input) - The MODE pin has three states, which select fully expanded external bus, partially expanded external bus, or no expanded bus (single-chip configuration).
P0-0 - P0-15	Port 0 (input/output) - Each bit in Port 0 may be individually programmed for general purpose input or output. Port 0 also has several handshaking modes to allow parallel, asynchronous communication with other devices. The high and low bytes may be programmed individually or jointly to be inputs, outputs, or bidirectional.
P1-0 - P1-15	Port 1 (input/output) - Each of the 16 bits in Port 1 may be individually programmed for input or output. Additionally, the lowest seven bits of Port 1 may be programmed to serve specific alternate functions as shown below.
P1-6/XI2	External Interrupt 2 (input, rising or falling edge triggered) - The programmer may select the rising or falling edge as active for XI2.

MNEMONIC	PIN FUNCTIONS FOR SINGLE-CHIP OPERATION
P1-5/XI1	External Interrupt 1 (input, falling edge triggered) - The XI1 may be used to interrupt the MK68HC200 on the falling edge of an input pulse.
P1-4/XI0	External Interrupt 0 (input, low level triggered) - The XI0 interrupt input is level triggered (unlike XI1 and XI2). It may be used to produce an internally vectored interrupt or to cause an external fetch of an interrupt vector number when the MK68HC200 is used in an expanded mode with the GP bus.
P1-3/SI	Serial Input (input, active high) - SI is used to receive serial data when the receiver is enabled.
P1-2/RCLK	Receive Clock (input/output, active high) - Depending on the mode programmed, RCLK can be used by the serial port as either an input or an output pin. When used as an input pin, RCLK provides the receive clock and/or the transmit clock. When RCLK is not providing the transmit or receive clock, it can be used as an output for Timer C. In this mode, the receive clock is being provided by Timer C.
P1-1/TCLK	Transmit Clock (input/output, active high) - Depending on the mode programmed, TCLK can be used by the serial port as either an input or an output pin. When used as an input pin, TCLK provides the transmit clock. When TCLK is not providing the transmit clock, it can be used as an output for the Timer C. In this mode, the transmit clock is being provided by either Timer C or RCLK.
P1-0/SO	Serial Output (output, active high) - SO is used to transmit serial data when the transmitter is enabled.
P4-8 - P4-15	Port 4 (inputs and outputs) - P4-8, P4-9, P4-14, and P4-15 may be used as general purpose outputs, and P4-10, P4-11, P4-12, and P4-13 may be used as general purpose inputs. Interrupts may be generated on the positive transitions on P4-10 and P4-11. Depending on the mode selected, interrupts may be generated on the positive or the negative transitions on P4-12, and they may be generated on the positive, negative or combined transitions on P4-13. Additionally, these bits may be programmed to serve specific alternate functions, as listed below.
P4-15/TAO	Timer A Output (output) - TAO may be programmed for special functions in the interval, event, and pulse/period modes for Timer A. In the interval mode, TAO's state is determined by the Timer A latch (high and low) that is currently active. That is, if the counter is using the high latch for comparison, TAO is high. In the event mode, TAO is initialized to a "1" state and toggles each time the counter matches the Timer A high latch. In the pulse/period modes, TAO is initialized to a "1" state and toggles on positive transitions on TAI.
P4-14/TBO	Timer B Output (output) - TBO may be programmed for special functions in the interval and one-shot modes for Timer B. In the interval mode, TBO is initialized to a "1" state and toggles each time the counter matches the Timer B latch value. In the one-shot modes TBO is initialized to a "1" state, and the counter begins counting in response to the occurrence of an active edge on TBI. TBO will not go low until the counter matches the value loaded into the Timer B latch.
P4-13/TAI	Timer A Input (input, positive and/or negative edge triggered) - TAI may be programmed for special functions in the event mode or the pulse/period modes for Timer A. In the event mode, the counter is incremented on each active transition (positive or negative edge programmable) on TAI. In the pulse/period modes, the counter measures the time during which the signal on TAI remains high and low.
P4-12/TBI	Timer B Input (input, positive or negative edge triggered) - TBI may be programmed for special functions for the Timer B one-shot modes. In the one-shot modes, TBI acts as a trigger input.
P4-11/STRH, P4-10/STRL	Strobe High Byte, Strobe Low Byte (input, active high) - STRH and STRL are both used for input, output and bidirectional handshaking on Port 0. These signals are issued by the peripheral to acknowledge the receipt of data made availa le by the MK68HC200, or are issued by the peripheral to load data from the peripheral into the Port 0 input register.
P4-9/RDYH, P4-8/RDYL	Ready High Byte, Ready Low Byte (output, active high) - RDYH and RDYL are used for input, output, and bidirectional handshaking on Port 0. The ready signal goes active to indicate that peripheral data is stable and ready for transfer to the peripheral or is used when Port 0 is empty and is ready to accept data from the peripheral.

GENERAL DESCRIPTION

MK68HC200 designates a series of new high-performance, 16-bit, single-chip microcomputers from Thomson Components - Mostek Corporation. Implemented in 1.5 micron HCMOS technology, they incorporate an architecture designed for superior performance in computation-intensive control applications. A modern, comprehensive instruction set (which features both high speed execution and code space efficiency) is combined on-chip with extensive, flexible I/O capabilities. On-chip RAM and optional on-chip ROM are provided with a full 64K byte addressing space.

The MK68HC200 can be used to design a true application specific microcontroller. The circuit is partitioned into three major functional blocks: CPU, memory, and I/O. The CPU is the core of the circuit and communicates with the memory via the memory address and data bus, and with the I/O via the I/O bus. New I/O or memory modules can be designed and added to the CPU core to customize the MK68HC200 for a particular application. The initial product offerings in the MK68HC200 family will contain I/O and memory features listed above. This is consistent with the features available on the NMOS MK68200. Future product offerings will contain various assortments of on-chip I/O and memory modules.

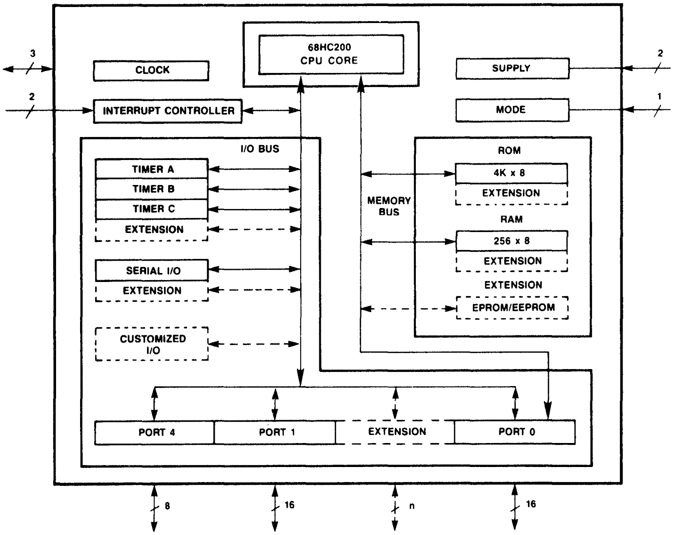


Figure 2. MK68HC200 - Modular Architecture Concept

FUNCTIONAL DESCRIPTION

MK68HC200 APPLICATIONS

The MK68HC200 is designed to serve the needs of a wide variety of control applications, which require high performance operation with a minimal parts count implementation. Industrial controls, instrumentation, and intelligent computer peripheral controls are all examples of applications served by the MK68HC200. High speed mathematical ability, rapid I/O addressing and interrupt response, and powerful bit manipulation instructions provide the necessary tools for these applications. In addition to its single-chip microcomputer configuration, both distributed intelligence and parallel multiprocessing system configurations are supported by the MK68HC200, as illustrated in Figures 9 and10.

In applications requiring loosely-coupled distributed intelligence, several MK68HC200's may be interconnected on a common serial network. The on-chip USART supports a wake-up mode in which an additional bit is appended to the data stream to distinguish a serial data word as address or data. The wake-up logic prevents the serial channel from generating interrupts unless certain criteria have been met. The wake-up options available are: Wake-up on any address or data character, wake-up on any address, or wake-up on address match.

Alternately, the MK68HC200 may be configured as an expandable CPU device which can access external memory and I/O resources. In this operating mode, parallel I/O pins are replaced by multiplexed address/data and control lines. Bus arbitration logic is incorporated on the chip to support a direct interface in parallel shaded bus multiprocessor system configurations. Two versions exist which support two types of control signals present on the expanded bus configuration.

The General Purpose (GP) bus option allows the MK68HC200 to operate either as an executive or a peripheral processor. As an executive processor, the MK68HC200 can control an external system bus and grant the use of it to requesting devices, such as DMA controllers and/or peripheral processors. As a peripheral control processor, the MK68HC200 can provide intelligent local control of an I/O device in a computer system and, thereby, relieve the executive processor of these taks. In this configuration, the MK68HC200 has the capability of effectively performing DMA transfers between system memory and the I/O device. The on-chip resources of ROM, RAM, and I/O are accessed within the MK68HC200 without affecting utilization of the shared system bus. Therefore, only external communications compete for bus bandwidth.

The Universal Peripheral Controller (UPC) bus option supports a direct interface to a 68000 executive processor. Thus, the MK68HC200 can be used as a cost-effective intelligent peripheral controller in 68000 systems. The UPC version's direct bus interface to the 68000 makes the MK68HC200 particularly well-suited for performing many intelligent I/O functions in a 68000 system. For example, since the MK68HC200 includes both a serial channel and an external bus capable of performing DMA transfers, it can be programmed to act as serial protocol controller with DMA capability.

For additional information on the MK68HC200 refer to the MK68HC200 Principles of Operation Manual, publication number 4430196.

PROCESSOR ARCHITECTURE

The MK68HC200 microcomputer contains an advanced processor architecture, combining the best properties of both 8- and 16-bit processors. A large majority of instructions operate on either byte or word operands.

REGISTERS

The MK68HC200 register set includes three system registers, six address registers, and eight data registers. The three 16-bit system registers (Figure 3) include a Program Counter, a Status Register, and a Stack Pointer. The six address registers may be used either for 16-bit data or for memory addressing. The eight 16-bit data registers are used for data and may be referenced as sixteen 8-bit registers, providing great flexibility in register allocation.

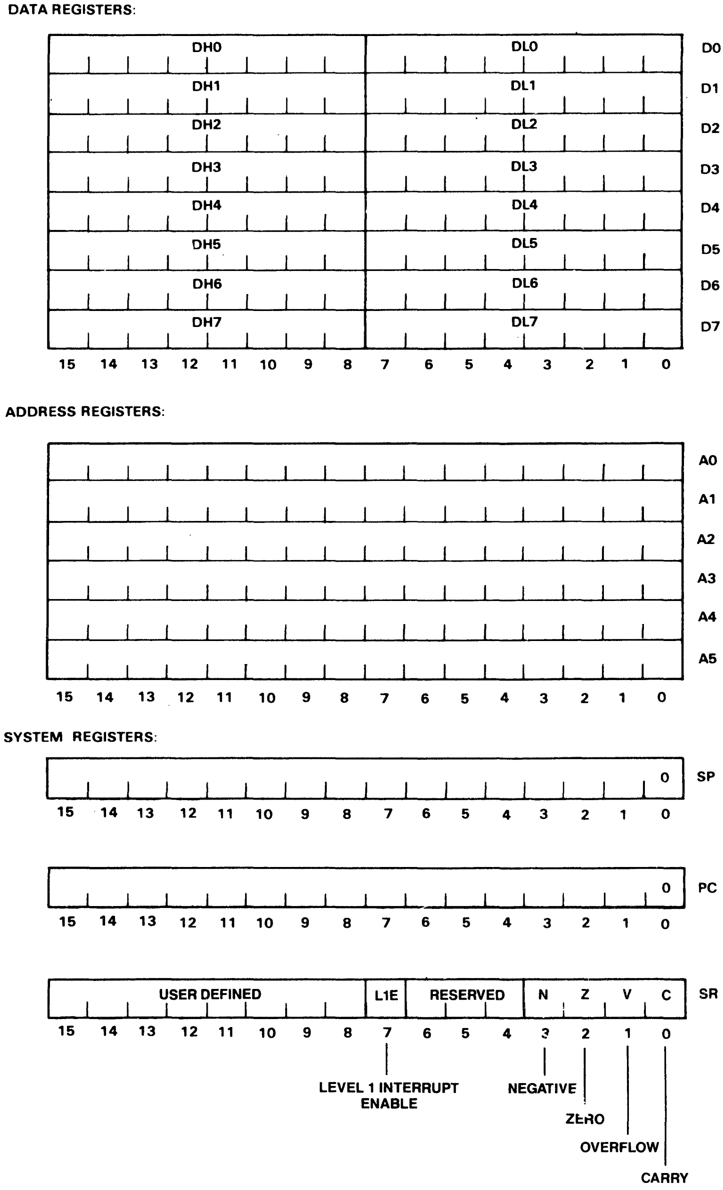


Figure 3 . Register Set

ADDRESSING

The MK68HC200 directly addresses a 64K byte memory space, which is organized as 32K 16-bit words. The memory is byte-addressable, but most transfers occur 16 bits at a time, for increased performance over 8-bit microcomputers. All input/output is memory-mapped, and the on-chip I/O is situated in the top 1K bytes of the address space. In the single-chip mode, all resources including ROM, RAM, and I/O, are accessed via an internal or private bus. The memory map, which is accessed by this bus in the single-chip mode, is depicted in Figure 4. Note on-chip RAM always begins at $FBFF and extends downward. ROM always begins at zero and extends upward.

Nine addressing modes provide ease of access to data in the MK68HC200, as depicted in Table 1. The four register indirect forms utilize the address registers and the Stack Pointer and support many common data structures such as arrays, stacks, queues, and linked lists. I/O Port addressing is a short form addressing mode for the first 16 words of the I/O port space and allows most instructions to access the most often referenced I/O ports in just one word. Many microcomputer applications are I/O intensive and short, fast addressing of I/O has a significant impact on performance.

Table 1. Addressing Modes

Register
Register Indirect
Register Indirect with Post-increment
Register Indirect with Pre-decrement
Register Indirect with Displacement
Program Counter Relative
Memory Absolute
Immediate
I/O Port

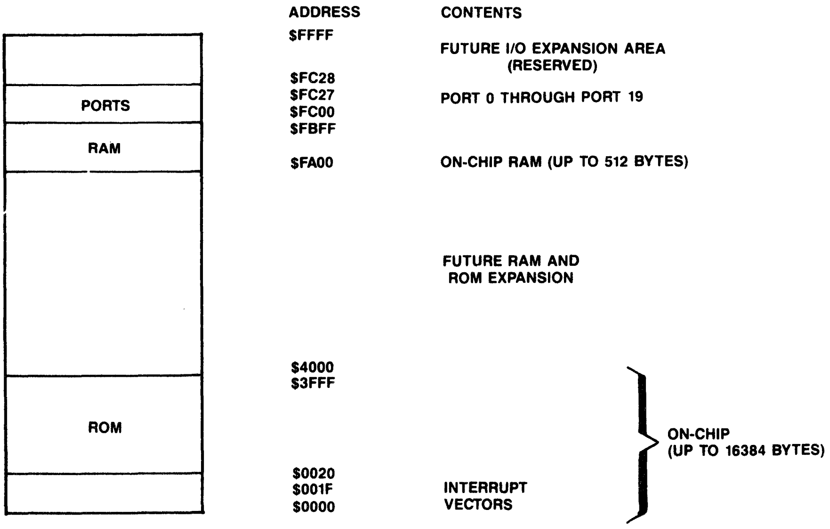


Figure 4. Addressing Space For Single-Chip Configuration

INSTRUCTION SET

The MK68HC200 instruction set has been designed with regularity and ease of programming in mind. In addition, instructions have been encoded to minimize code space, a feature which is especially important in single-chip microcomputers. Small code space is related to execution speed, and most instructions execute in either three or six instruction clock periods. See Table 2.

In addition to operations on bytes and words, the MK68HC200 has rapid bit manipulation instructions that can operate on registers, memory, and ports. The bit to be affected may be an immediate operand of the instruction, or it may be dynamically specified in a register. Operations available include bit set, clear, test, change, and exchange; and all bit operations perform a bit test as well. Since each instruction is indivisible, this provides the necessary test-and-set function for the implementation of semaphores.

The MOVE group of instructions has the most extensive capabilities. A wide variety of addressing mode combinations is supported including memory-to-memory transfers. A special move multiple is included to save and restore a specified portion of the registers rapidly.

In total, the MK68HC200 instruction set provides a programming environment, similar to the 68000, which has been optimized for the needs of the single-chip microcomputer marketplace. A summary of the instruction set is provided in Table 3.

Table 2. Instruction Execution Times

Instruction Type	Clock Periods	Execution Time with 8 MHz Clock (μs)	Execution Time with 10 MHz Clock (μs)	Execution Time with 12.5 MHz Clock (μs)
Move Register-to-register	3	0.38	0.30	0.24
Add Register-to-register (binary or BCD)	3	0.38	0.30	0.24
Move Memory-to-register	6	0.75	0.60	0.48
Add Register-to-memory	9	1.13	0.90	0.72
Multiply (16 × 16)	21	2.63	2.10	1.68
Divide (32/16)	23	2.88	2.30	1.84
Move Multiple (save or restore all registers)	55	6.88	5.50	4.40

Table 3. Instruction Set Summary

INSTRUCTION	DESCRIPTION
ADD	ADD
ADD.B	ADD BYTE
ADDC	ADD WITH CARRY
ADDC.B	ADD WITH CARRY BYTE
AND	LOGICAL AND
AND.B	LOGICAL AND BYTE
ASL	ARITHMETIC SHIFT LEFT
ASL.B	ARITHMETIC SHIFT LEFT BYTE
ASR	ARITHMETIC SHIFT RIGHT
ARS.B	ARITHMETIC SHIFT RIGHT BYTE
BCHG	BIT CHANGE
BCLR	BIT CLEAR
BEXG	BIT EXCHANGE
BSET	BIT SET
BTST	BIT TEST
CALLA	CALL ABSOLUTE
CALLR	CALL RELATIVE
CLR	CLEAR
CLR.B	CLEAR BYTE
CMP	COMPARE
CMP.B	COMPARE BYTE
DADD	DECIMAL ADD
DADD.B	DECIMAL ADD BYTE
DADDC	DECIMAL ADD WITH CARRY
DADDC.B	DECIMAL ADD WITH CARRY BYTE
DI	DISABLE INTERRUPTS
DIVU	DIVIDE UNSIGNED
DJNZ	DECREMENT COUNT AND JUMP IF NON-ZERO
DJNZ.B	DECREMENT COUNT BYTE AND JUMP IF NON-ZERO
DNEG	DECIMAL NEGATE
DNEG.B	DECIMAL NEGATE BYTE
DNEGC	DECIMAL NEGATE WITH CARRY
DNEGC.B	DECIMAL NEGATE WITH CARRY BYTE
DSUB	DECIMAL SUBTRACT
DSUB.B	DECIMAL SUBTRACT BYTE
DSUBC	DECIMAL SUBTRACT WITH CARRY
DSUBC.B	DECIMAL SUBTRACT WITH CARRY BYTE
EI	ENABLE INTERRUPTS
EOR	EXCLUSIVE OR
EOR.B	EXCLUSIVE OR BYTE
EXG	EXCHANGE
EXG.B	EXCHANGE BYTE
EXT	EXTEND SIGN
HALT	HALT
IDLE	IDLE

INSTRUCTION	DESCRIPTION
JMPA	JUMP ABSOLUTE
JUMPR	JUMP RELATIVE
LIBA	LOAD INDEXED BYTE ADDRESS
LINK	LINK
LIWA	LOAD INDEXED WORD ADDRESSED
LSR	LOGICAL SHIFT RIGHT
LSR.B	LOGICAL SHIFT RIGHT BYTE
MOVE	MOVE
MOVE.B	MOVE BYTE
MOVEM	MOVE MULTIPLE REGISTERS
MOVEM.B	MOVE MULTIPLE REGISTERS BYTE
MULS	MULTIPLY SIGNED
MULU	MULTIPLY UNSIGNED
NEG	NEGATE
NEG.B	NEGATE BYTE
NEGC	NEGATE WITH CARRY
NEGC.B	NEGATE WITH CARRY BYTE
NOP	NO OPERATION
NOT	ONE'S COMPLEMENT
NOT.B	ONE'S COMPLEMENT BYTE
OR	LOGICAL OR
OR.B	LOGICAL OR BYTE
POP	POP
POPM	POP MULTIPLE REGISTERS
PUSH	PUSH
PUSHM	PUSH MULTIPLE REGISTERS
RET	RETURN FROM SUBROUTINE
RETI	RETURN FROM INTERRUPT
ROL	ROTATE LEFT
ROL.B	ROTATE LEFT BYTE
ROLC	ROTATE LEFT THROUGH CARRY
ROLC.B	ROTATE LEFT THROUGH CARRY BYTE
ROR	ROTATE BYTE
ROR.B	ROTATE RIGHT BYTE
RORC	ROTATE RIGHT THROUGH CARRY
RORC.B	ROTATE RIGHT THROUGH CARRY BYTE
STOP	STOP
SUB	SUBTRACT
SUB.B	SUBTRACT BYTE
SUBC	SUBTRACT WITH CARRY
SUBC.B	SUBTRACT WITH CARRY BYTE
TEST	TEST
TEST.B	TEST BYTE
TESTN	TEST NOT
TESTN.B	TEST NOT BYTE
UNLINK	UNLINK

INPUT/OUTPUT ARCHITECTURE

The I/O capabilities of the MK68HC200 are extensive, encompassing timers, a serial channel, parallel I/O, and an interrupt controller. All of these devices are accessible to the programmer as ports within the top 1K bytes of the address space, and the most commonly accessed ports may be accessed with the short port addressing mode. A description of these ports is given in Table 4.

In total, 40 pins of the 48 are used for I/O, and their functions are highly programmable by the user. In particular, many pins can perform multiple functions, and the programmer selects which ones are to be used. For example, TAI may be used as an input for Timer A, an interrupt source, or a general purpose input pin. The interrupt source may be selected simultaneously with either of the other functions.

Table 4. Port Descriptions

PORT	ADDRESS	READ/WRITE	BYTE-ADDRESSABLE	FUNCTION
0	$FC00	READ/WRITE	YES	16 EXTERNAL I/O PINS OR ADDRESS/DATA BUS
1	$FC02	READ/WRITE	YES	16 EXTERNAL I/O PINS (INCLUDING INTERRUPT, SERIAL I/O PINS, AND BUS CONTROL)
2	$FC04	—	—	(RESERVED)
3	$FC06	LOW BYTE: READ/WRITE HIGH BYTE: READ	YES	SERIAL TRANSMIT (LOW BYTE) AND RECEIVE (HIGH BYTE) BUFFER
4	$FC08	INPUTS: READ ONLY OUTPUTS: READ/WRITE	NO	8 EXTERNAL I/O PINS (TIMER CONTROL AND PORT 0 HANDSHAKE CONTROL)
5	$FC0A	—	—	(RESERVED)
6	$FC0C	—	—	(RESERVED)
7	$FC0E	READ/WRITE	NO	INTERRUPT LATCH REGISTER
8	$FC10	READ/WRITE	NO	INTERRUPT MASK REGISTER
9	$FC12	STATUS: READ ONLY CONTROL: READ/WRITE	NO	SERIAL I/O RECEIVE CONTROL AND STATUS
10	$FC14	STATUS: READ ONLY CONTROL: READ/WRITE	NO	SERIAL I/O TRANSMIT CONTROL AND STATUS
11	$FC16	READ GETS COUNTER WRITE GOES TO LATCH	NO	TIMER B LATCH
12	$FC18	READ GETS COUNTER OR LATCH WRITE GOES TO LATCH	NO	TIMER A, LOW LATCH
13	$FC1A	READ GETS COUNTER OR LATCH WRITE GOES TO LATCH	NO	TIMER A, HIGH LATCH
14	$FC1C	READ/WRITE	NO	TIMER AND HANDSHAKE CONTROL
15	$FC1E	STATUS: READ ONLY CONTROL: READ/WRITE	NO	EXPANDED BUS CONTROL AND STATUS
16	$FC20	READ/WRITE	NO	PORT 0 DIRECTION CONTROL (DDR0)
17	$FC22	READ/WRITE	NO	PORT 1 DIRECTION CONTROL (DDR1)
18	$FC24	READ/WRITE	NO	SERIAL I/O MODE AND SYNC REGISTER
19	$FC26	READ GETS COUNTER WRITE GOES TO LATCH AND COUNTER	NO	TIMER C LATCH

TIMERS

The MK68HC200 includes three on-chip timers, each with unique features. They are denoted Timer A, Timer B, and Timer C. All three timers are a full 16 bits in width, and count at the instruction clock rate of the MK68HC200 processor. Thus, this rate provides a resolution equal to the instruction clock period (tc) of the MK68HC200. The maximum count interval is equal to tc $*2^{16}$. Each timer has the capability to interrupt the processor when it matches a predetermined value stored in an associated latch.

Timer A is capable of operating in interval, event, or two pulse/period modes. There is one 16-bit counter and two 16-bit latches, a high latch (Port 13), and a low latch (Port 12), associated with Timer A. Once Timer A is initialized in the interval mode, the counter is reset, then increments at the instruction clock rate until the value loaded into the high latch is reached. The counter is then reset, increments until the low latch value is reached, and the cycle is repeated. In the event mode, the counter is incremented for every active edge on TAI (programmable as positive or negative) until the value in the high latch is reached. The counter is then reset, and the cycle repeats. In the pulse/period modes, the times are measured during which the applied pulse stays high and low. The counter is reset on the occurrence of any transition on TAI, and increments at the instruction clock rate until the occurrence of the next transition. The value in the counter at the end of the high level or low level time is loaded into the appropriate latch. Interrupts may be generated each time the counter reaches the high latch or low latch value in the interval mode or when the counter reaches the high latch in the event mode. Also, an interrupt is generated whenever the counter overflows. See the Pin Description section of the data sheet for TAI and TAO functions in the various Timer A modes.

Timer B is capable of operating in interval or one-shot modes. There is one 16-bit counter and one 16-bit latch (Port 11) associated with Timer B. In the interval mode, the counter is initially reset and incremented at the instruction clock rate until the value in the latch is reached. The counter is then reset, and the cycle repeats. In the one-shot modes, the counter begins incrementing in response to an active transition (programmable as positive or negative) on TBI. The counter is reset when the value in the Timer B latch is reached. In the retriggerable one-shot mode, active transitions on TBI always cause the counter to reset and begin incrementing. In the non-retriggerable one-shot mode, active transitions on TBI have no effect until the counter reaches the latch value. Interrupts may be generated each time the counter reaches the latch value. See the Pin Description section of this data sheet for TBI and TBO functions in the various Timer B modes.

Timer C has a 16-bit down counter and latch (Port 19) associated with it and operates only in the interval mode. The output of Timer C toggles each time the counter value rolls over from 0 to the latch value and may be used to internally supply the baud rate clock for the serial port. Also, an interrupt may be generated each time the counter rolls over to the latch value. Timer C may be output on the TCLK pin (P1-3), depending on the mode programmed.

A detailed description of the Timer Control Port is given on the next page.

Table 5. Timer Modes

Timer	Modes
A	Interval
A	Event
A	Pulse Width and Period Measurement
B	Interval
B	Retriggerable One-shot
B	Non-retriggerable One-shot
C	Interval
C	Baud Rate Generation

PORT 14—Timer Control Register; read/write
$FC1C

RESERVED					TCE	TCOC	TAM1	TAM0	TAE	TAIC	TAOC	TBM1	TBM0	TBE	TBIC
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

TCE
(Timer C Enable control)

0 = Disables Timer C; all operations are inhibited, and the timer counter is initialized.
1 = Enables Timer C; the timer begins operation as defined by the other Timer C control bits.

TCOC
(Timer C Output control)

0 = When TCO (Port 18) = 1, TCLK is selected for use as a general purpose I/O pin.
1 = When TCO (Port 18) = 1, TCLK is selected for use as an output for Timer C.

TCOC has no effect when TCO (Port 18) = 0.

TAM1, TAM0
(Timer A Mode control)

These bits select the operating mode of Timer A as follows.

TAM1	TAM0	MODE
0	0	Interval
0	1	Event
1	0	Pulse/period 1
1	1	Pulse/period 2

TAE
(Timer A Enable control)

0= Disables Timer A; all Timer A operations are inhibited, and the timer counter is initialized.
1= Enables Timer A; the timer begins operation as defined by the other Timer A control bits.

TAIC
(Timer A Input control)

0= Selects a negative transition as the active edge for TAI.
1= Selects a positive transition as the active edge for TAI.

TAOC
(Timer A Output control)

0= Selects TAO as a general purpose output pin.
1= Selects TAO as an ouput pin associated with Timer A; TAO is initialized low when TAOC is a one and TAE is zero.

TBM1, TBM0
(Timer B Mode control)

These bits select the operating mode of Timer B as described below.

TBM1	TBM0	MODE
0	0	Interval 0 (TBO is not used)
0	1	Interval 1 (TBO is used)
1	0	Retriggerable one-shot
1	1	Non-retriggerable one-shot

TBE (Timer B Enable control)	0= Disables Timer B; all operations are inhibited, and the timer counter is initialized. 1= Enables Timer B; the timer begins operation as defined by the other Timer B control bits.
TBIC (Timer B Input control)	0= Selects a negative transition as active on TBI. 1= Selects a positive transition as active on TBI.

INTERRUPT CONTROLLER

The MK68HC200 interrupt controller provides rapid service of up to 15 interrupt sources, each with a unique internal vector. The lowest 16 words of the address space contain the starting addresses of the service routines of each potential interrupt source and reset, as shown in Figure 5.

Interrupt sources and RESET are prioritized in the order shown in Figure 5, with RESET having the highest priority. When an interrupt is pending it sets the corresponding bit in the interrupt latch located in Port 7. NMI is the only non-maskable interrupt. All of the other sources share an interrupt enable bit in the processor Status Register. This bit is automatically cleared whenever an interrupt is acknowledged. Also, each of these sources has a corresponding individual mask bit located in Port 8. This feature allows selective masking of particular interrupts, including the ability to choose (with minimal softwware overhead) any priority scheme desired. In fact, 15 levels of nested priority may be programmed.

Note that the XI2 interrupt is detected on either a rising or falling edge, depending upon the status of the XI2C bit (bit 12 in Port 14). An interrupt will be generated on the falling edge if this bit is set to a "0"; however, if the bit is set to a "1", an interrupt will be generated on the rising edge.

PORT 7 —Interrupt Latch Register; read/write
$FC0E

res	NMI	SPARE	XI2I	STRLI	TAOI	TAII	STRHI	RSCI	RNI	XI1I	TBOI	TBII	XI0I	XMTI	TCI
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

PORT 8 —Interrupt Mask Register; read/write
$FC10

(res)		SPARE	XI2M	STRLM	TAOM	TAIM	STRHM	RSCM	RNM	XI1M	TBOM	TBIM	XI0M	XMTM	TCM
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

VECTOR NUMBER	NAME	MNEMONIC	VECTOR LOCATION	
0	RESET	RESET	0000	LEVEL 2
1	NON-MASKABLE INTERRUPT	NMI	0002	
2	SPARE	SPARE	0004	LEVEL 1
3	EXTERNAL INTERRUPT 2	XI2	0006	
4	STROBE LOW	STRL	0008	
5	TIMER A OUTPUT	TAO	000A	
6	TIMER A INPUT	TAI	000C	
7	STROBE HIGH	STRH	000E	
8	RECEIVE SPECIAL CONDITION	RSC	0010	
9	RECEIVE NORMAL	RN	0012	
A	EXTERNAL INTERRUPT 1	XI1	0014	
B	TIMER B OUTPUT	TBO	0016	
C	TIMER B INPUT	TBI	0018	
D	EXTERNAL INTERRUPT 0	XI0	001A	
E	TRANSMIT	XMT	001C	
F	TIMER C	TC	001E	

Figure 5. Interrupt and Reset Vectors

SERIAL CHANNEL

The serial channel on the MK68HC200 (Figure 6) is a full-duplex USART with double buffering on both transmit and receive. Port 3 High Byte is the Receive Buffer, and Port 3 Low Byte is the Transmit Buffer. Word length, parity, stop bits, and modes are fully programmable. The asynchronous mode supports bit rates up to 781 Kbps with an external clock and up to 390Kbps with an internal clock. The byte synchronous mode operates up to 3.125Mbps with either an internal or an external clock.

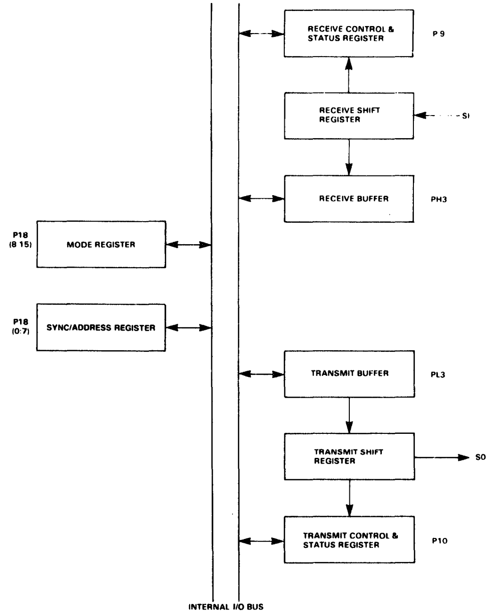


Figure 6. Serial Channel

In addition to the typical USART functions, the serial channel can operate in a special wake-up mode with a wake-up bit appended to each data word, as illustrated in Figure 7. This wake-up bit is used to differentiate normal data words and special address words. The receiver can be programmed to receive only address words or only address words with a specific data value. In this way, the processor can be interrupted only when it receives its particular address and can then change mode to receive the following data words. Wake-up capability is especially useful when several MK68HC200 microcomputers are interconnected on one serial link.

A detailed description of the serial channel control ports is given on the following pages.

STARTt	DATA LSB MSB	PARITY (OPTIONAL)	WAKE-UP (OPTIONAL)	STOPt

Figure 7. Serial Frame Format

PORT 9 —Serial I/O Receive Control and Status Register;
$FC12 High byte: control register; read/write
Low byte: status register; read only

RE	IS	RW1	RW0	RC	SIS	(res)		BF	OE	PE	FE	SF/AF	BRK	(res)	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bit Descriptions:

RE
(Receiver Enable control)
0 = Disabled; all status flags cleared.
1 = Enabled.

IS
(Ignore Syncs control)
0 = Disabled; interrupts may occur on all characters received.
1 = Enabled; interrupts cannot occur on sync characters received after the sync match is found.

RW1, RW0
(Receiver Wake-up control)
The receiver wake-up control bits operate as follows.

MODE	RW1	RW0	APPENDED WAKE-UP	BUFFER LOADED	INTERRUPT GENERATED
No Wake up	0	0	no	any character	RN
Wake-up on Any Character	0	1	yes	any character	RN
Wake-up on Address Match	1	0	yes	address match	RSC
Wake-up on Any Address	1	1	yes	any address	RSC

RC
(Receive Clock control)
0 = Selects external receive clock applied on RCLK.
1 = Selects internal clock from the on-chip baud rate generator (Timer C) for the receive clock.

This bit is ignored when either the TCO bit or the LM (Loopback Mode) bit is set.

SIS
(Single Interrupt Select control)

0 = Separate vectors are generated for the Receive Normal and the Receive Special Condition interrupts.
1 = The Receive Normal vector is generated for all receive character interrupts.

BF
(Buffer Full status)

0 = Receive data buffer empty, cleared when receive buffer is read.
1 = Receive buffer full; set when an incoming word is loaded into the receive data buffer.

OE
(Overrun Error status)

0 = No overrun error; cleared when the status register is read.
1 = Overrun error; set when a new word has been received and the previous word has not been read from the receive data buffer.

PE
(Parity Error status)

0 = No Parity Error; cleared when the status register is read.
1 = Parity Error; set when a parity error has been detected on an incoming character in the data stream.

FE
(Frame Error status)

0 = No frame error; cleared when the status register is read.
1 = Frame error; set when a word is transferred to the receive data and no stop bit has been recognized.

The FE flag applies to async formats only.

SF/AF
(Sync Found or Address Found status)

This flag is used for both sync character match conditions and address found conditions in some wake-up modes as follows. SS stands for Sync Search mode.

MODES				
SS	**IS**	**RW1**	**RW0**	**CONDITIONS THAT SET SF/AF**
1	x	x	x	Sync Found on any bit boundry
0	1	x	x	unaffected
0	0	0	0	unaffected
0	0	1	1	Any Address
0	0	1	0	Address Match
0	0	0	1	unaffected

PORT 10—Serial I/O Transmit Control and Status Register
$FC14 High byte: control register; read/write
Low byte: status register; read only

TE	AT	LM	TW1	TW0	TC	$P/\overline{S}$	res	BE	UE	END	(reserved)				
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bit Descriptions:

TE
(Transmitter Enable control)

0 = Disable the transmitter; any word being shifted out will continue until completion.
1 = Enable the transmitter.

AT
(Automatic Turn Around control)

0 = No effect on TE or RE.
1 = Causes RE to be set to a "1" and TE to be set to a "0" automatically at the end of a transmission.

LM
(Loopback Mode control)

0 = Disables loopback mode.
1 = Causes the transmitter output to be internally connected to receiver input. Also causes Timer C to be used for both the transmit and receive clocks regardless of the state of TC, RC, TCO, and TCOC.

TW1, TWO
(Transmit Wake-up control)

These bits provide control for wake-up operation as follows.

TW1	TW0	OPERATION
1	0	Transmit Data
1	1	Transmit Address
0	X	No Wake-up

TC
(Transmit Clock control)

0 = Selects the external clock signal applied on TCLK for the transmit clock.
1 = Selects the internal baud rate generator output (Timer C) for the transmit clock.

The TC bit is ignored if either the TCO bit or the LM bit is set.

$P/\overline{S}$
(Previous/Sync control)

0 = Selects continuous transmission of the contents of the sync character register in the synchronous mode when there is no data to transmit.
1 = Selects continuous transmission of the transmit data buffer in synchronous mode when there is no data to transmit.

BE
(Buffer Empty status)

0 = Transmit Buffer is full; reset to this condition after the transmit buffer is reloaded.
1 = Transmit Buffer is empty; set to this condition after the transmit buffer contents are transferred to the output shift register.

UE
(Underrun Error status)

0 = No underrun error; cleared following a read of the transmit buffer.
1 = Underrun error; set only in the synchronous mode when the last word has been shifted out and transmit buffer has not been reloaded.

END
(End of Transmission status)

0 = No end of transmission; cleared by enabling the transmitter.
1 = End of transmission detected; set when the transmitter is disabled and the last character has been shifted out.

PORT 18—Serial I/O Mode and Sync Register; read/write
$FC24

A/$\overline{S}$	WL1	WL0	ST	PAR1	PAR0	TCO	WS	SYNC7	SYNC6	SYNC5	SYNC4	SYNC3	SYNC2	SYNC1	SYNC0
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bit Descriptions:

A/$\overline{S}$
(Asynchronous/$\overline{\text{Synchronous}}$ mode control)

0 = Selects synchronous operation for the serial port; transmit and receive clocks are divided by 1.
1 = Selects asynchronous operation for serial port; transmit and receive clocks are divided by 16.

WL1, WL0
(Word Length control)

These two bits select the length of the data word as follows.

WL1	WL0	Word Length
0	0	8 bits
0	1	7 bits
1	0	6 bits
1	1	5 bits

ST
(Stop Bit control)

This bit is only used in the asynchronous mode. It selects the number of stop bits transmitted.

ST	Number of Stop Bits
0	1
1	2

PAR1, PAR0
(Parity control)

These two bits provide parity control for both the synchronous and asynchronous modes.

PAR1	PAR0	Parity
0	0	no parity
0	1	fixed "0" parity
1	0	odd parity
1	1	even parity

Note that even parity is defined such that the sum of the data and parity bits is even.

TCO
(Timer C Output mode control)

0= Disables Timer C output mode.
1= Enables Timer C output mode; disables Timer C's use as a baud rate generator when LM = 0; causes transmit and receive clocks to be internally connected to RCLK so that TCLK may be used either as general purpose I/O or as an output for Timer C.

WS
(Wake-up Sense)

The following table lists the effects of the WS bit.

WS	Wake-up bit	Meaning
0	0	Address Word
0	1	Data Word
1	0	Data Word
1	1	Address Word

SYNC7-SYNC0
(Sync character bits)

These eight bits are used to store the sync character or the device address for the wake-up mode.

PARALLEL I/O AND HANDSHAKING

Two 16-bit ports, P0 and P1, may be used for parallel I/O. If individual bits are desired, each of the 32 bits may be separately defined an input or output. This is achieved by setting the corresponding bits in the Data Direction Registers, Port 16 (Data Direction Register for Port 0) and Port 17 (DDR for Port 1). Bits may be grouped to provide the exact data widths desired.

Eight additional I/O bits are provided in Port 4. Bits 15, 14, 9 and 8 are output only, and bits 13, 12, 11 and 10 are input only.

Port 0 has the additional capability of operating under the control of external handshaking signals. Eight-bit or sixteen-bit sections of P0 may be individually controlled as input, output or bidirectional I/O. This is achieved by programming the handshake control bits as detailed below.

PORT 14— Handshake Control Register; read/write
$FC1C

HSM2	HSM1	HSM0	res	HSE	TIMER CONTROL										
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bit Descriptions:

HSE
(Handshake enable control)
0 = Handshaking is disabled.
1 = Handshaking is enabled.

HSM2, HSM1, HSM0
(Handshake Mode control)
The handshake mode bits operate as follows:

HSM2	HSM1	HSM0	HIGH HANDSHAKE	LOW HANDSHAKE
0	0	0	Inactive	PLO or PO output
0	0	1	PHO output	PLO output
0	1	0	Inactive	PLO input
0	1	1	PHO input	PLO input
1	0	0	PHO input	PLO output
1	0	0	Inactive	PO input (word only)
1	1	0	PLO output	PLO input (bidirectional)
1	1	1	PO output	PO input (bidirectional)

Two pairs of Ready and Strobe signals, which are available as programmable options on Port 4, provide the necessary control for handshaking.

P4-9/RDYH, P4-8/RDYL
(Ready High Byte, Ready Low Byte)
Output, active high. RDYH and RHYL are used for input, output, and bidirectional handshaking on Port 0.

1) Output mode: The ready signal goes active to indicate that the Port 0 output register has been loaded, and the peripheral data is stable and ready for transfer to the peripheral device.

2) Input mode: The ready signal is active when the Port 0 input register is empty and is ready to accept data from the peripheral device.

3) Bidirectional mode: The RDYH signal is active when data is available in Port 0 output register for transfer to the peripheral device. In this mode, data is not placed on the Port 0 data bus unless STRH is active. The RDYL signal is active when the Port 0 input register is empty and is ready to accept data from the peripheral device.

P4-11/STRH, P4-10/STRL
(Strobe High Byte, Strobe Low Byte)
Input, active high. STRH and STRL are both used for input, output, and bidirectional handshaking on Port 0.

1) Output Mode; The positive edge of this strobe is issued by the peripheral to acknowledge the receipt of data made available by the MK68HC200.

2) Input mode: The strobe is issued by the peripheral to load data from the peripheral into the Port 0 input register. Data is latched into the MK68HC200 on the negative edge of this signal.

3) Bidirectional mode: When the STRH signal is active, data from the Port 0 output register is gated onto the Port 0 bidirectional data bus.

The negative edge of STRH acknowledges the receipt of the output data. The negative edge of the signal applied to the STRL signal is used to latch input data into Port 0.

EXPANDED BUS OPERATION

When it is necessary to expand beyond the on-chip complement of RAM, ROM, or I/O, or when operation in a parallel multiprocessing system is desired, the MK68HC200 may be placed in an external bus mode. The MODE pin is used to select the expansion capability on power-up and reset to one of the following states:

MODE PIN
V_{CC} - No expansion (single chip mode)
GND - Partial Expansion
CLKOUT - Full Expansion

By programming the appropriate bits in Port 15 (which are described below), the MK68HC200 may be reconfigured dynamically. In an expansion mode Port 0 becomes the 16-bit multiplexed address/data bus and eight bits from Port 1 become control signals which handle data transfer and bus arbitration. Sixteen lines are still available for I/O functions, including eight lines from Port 1 and all eights lines of Port 4. See figure 8 for the expanded bus pinout. The following page describes the functions of the expanded bus pins.

PORT 15— Expanded bus control and status register
$FCIE High byte: read/write
Low byte: read only

SEG1	SEG0	BLCK	F/$\overline{S}$	EXP	F/$\overline{P}$	(res)		UPC/$\overline{GP}$	R/$\overline{G}$	(reserved)					
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bit Descriptions:

SEG1, SEG0
(Segment bits)
Used in the expanded bus mode when a reference is made to the DMA window. The contents of SEG1 and SEG0 are then output on pins AD15 and AD14, respectively.

BLCK
(Bus Lock control)
0 = Disables the bus lock function.
1 = Enables the bus lock function.

F/$\overline{S}$
(Fast/$\overline{\text{Standard}}$ timing control)
0 = Selects standard timing of read/write cycles on the external bus (4 clock periods).
1 = Selects fast timing of read/write cycles on the external bus (3 clock periods.)

EXP
(Expanded Mode control)
0 = Expanded mode is disabled.
1 = Expanded mode is enabled.

F/$\overline{P}$
(Full/$\overline{\text{Partial}}$ control)
0 = Partial expand when EXP bit is set to 1.
1 = Full expand when EXP bit is set to 1.

UPC/$\overline{GP}$
(UPC/$\overline{GP}$ status)
0 = Part is programmed in GP mode.
1 = Part is programmed in UPC mode.

R/$\overline{G}$
(Request/$\overline{\text{Grant}}$ status)
0 = Part is programmed bus grantor.
1 = Part is programmed bus r questor.

MNEMONIC	PIN FUNCTIONS FOR EXPANDED BUS OPERATION (Common for GP and UPC options)
R/$\overline{\text{W}}$	Read/Write (output, active high and low) - R/$\overline{\text{W}}$ determines whether a read or a write is being performed during the current bus cycle. It is stable for the entire bus operation. A high signal denotes a read, and a low signal denotes a write.
$\overline{\text{DTACK}}$	Data Transfer Acknowledge (input, active low) - When the addressed device has either placed the requested read data on the bus or taken the write data from the bus, $\overline{\text{DTACK}}$ should be brought low to signify completion. The data portion of the bus cycle will be extended indefinitely until this signal is asserted. For systems using the GP bus, in which no devices need wait states, $\overline{\text{DTACK}}$ may be strapped low.
$\overline{\text{AS}}$	Address Strobe (output, active low) - $\overline{\text{AS}}$ is used to signify that the address is stable on the multiplexed bus. $\overline{\text{AS}}$ is high at the beginning of each bus cycle, goes low after the address has stabilized, and returns to the high state near the end of the bus cycle.

MNEMONIC	PIN FUNCTIONS FOR UPC BUS OPERATION
$\overline{\text{UDS}}$	Upper Data Strobe (output, active low) - $\overline{\text{UDS}}$ is used to signify the data portion of the bus cycle for the upper byte of the data bus. For read operations, $\overline{\text{UDS}}$ should be used by the external device to gate its most significant byte onto the multiplexed address/data bus. For writes, $\overline{\text{UDS}}$ signifies that the lower byte of the bus contains valid data to be written from the processor.
$\overline{\text{LDS}}$	Lower Data Strobe (output, active low) - $\overline{\text{LDS}}$ is used to signify the data portion of the bus cycle for the lower byte of the data bus. For read operations, $\overline{\text{LDS}}$ should be used by the external device to gate its least significant byte onto the multiplexed address/data bus. For writes, $\overline{\text{LDS}}$ signifies that the lower byte of the bus contains valid data to be written from the processor.
$\overline{\text{BR}}$	Bus Request (output, active low, open drain) - $\overline{\text{BR}}$ goes low when the MK68HC200 requires external bus master status.
$\overline{\text{BG}}$	Bus Grant (input, active low) - $\overline{\text{BG}}$ notifies that the MK68HC200 has been granted the external bus master status.
$\overline{\text{BGACK}}$	Bus Grant Acknowledge (output, active low, open drain) - The MK68HC200 will assert $\overline{\text{BGACK}}$ when it assumes mastership of the system bus.

MNEMONIC	PIN FUNCTIONS FOR GP BUS OPERATION
P4-11/R/$\overline{\text{G}}$	Request/$\overline{\text{Grant}}$ (input) - During reset, P4-11 served as the R/$\overline{\text{G}}$ input (0 = bus grantor, 1 = bus requestor). Following reset, and at all times during program execution, P4-11 may be used as a general purpose input pin.
$\overline{\text{DS}}$	Data Strobe (output, active low) - $\overline{\text{DS}}$ is used to signify the data portion of the bus cycle. For read operations, $\overline{\text{DS}}$ should be used by the external device to gate its contents onto the multiplexed address/data bus. For writes, $\overline{\text{DS}}$ signifies that valid data from the processor is on the bus.
$\overline{\text{HB}}$	High Byte (output, active low) - $\overline{\text{HB}}$ signifies that the upper byte of the data is to be read or written. $\overline{\text{HB}}$ remains active for the entire bus cycle.
$\overline{\text{LB}}$	Low Byte (output, active low) - $\overline{\text{LB}}$ signifies that the lower byte of the data bus is to be read or written). $\overline{\text{LB}}$ remains active for the entire bus cycle.
$\overline{\text{BUSIN}}$	Bus Input (input, active low) - $\overline{\text{BUSIN}}$ provides either bus request or bus grant. When the MK68HC200 is the bus grant device, its $\overline{\text{BUSIN}}$ signal is a bus request input from a requesting device on the bus. When the MK68HC200 is a bus request device, its $\overline{\text{BUSIN}}$ signal is a bus grant from the granting device on the bus.
$\overline{\text{BUSOUT}}$	Bus Output (output, active low) - $\overline{\text{BUSOUT}}$ provides the opposite function of $\overline{\text{BUSIN}}$. When $\overline{\text{BUSIN}}$ is a bus request signal, $\overline{\text{BUSOUT}}$ is a corresponding bus grant, and vice versa.

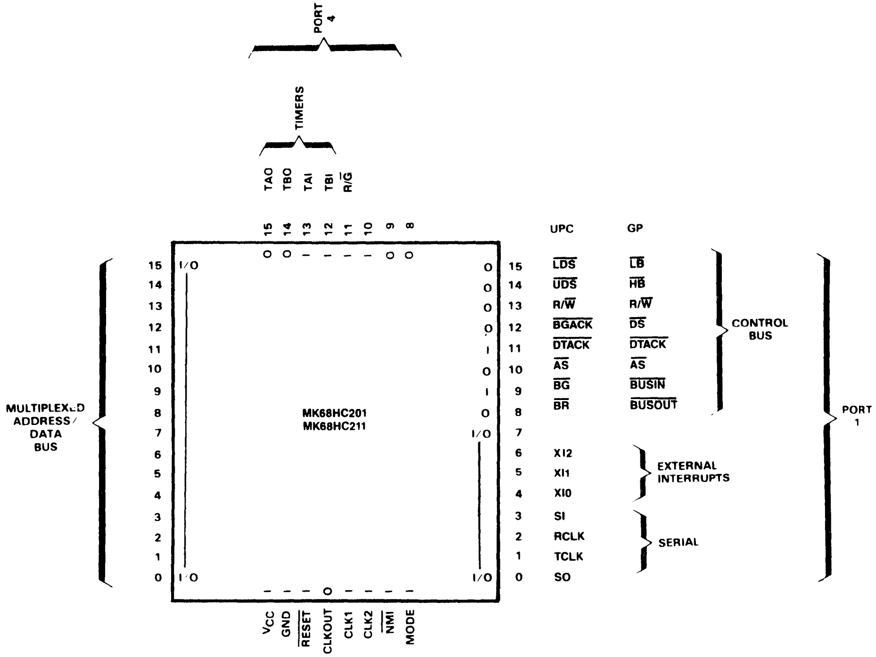


Figure 8. MK68HC201, MK68HC211 Logical Pinout Expanded Bus

As shown in figure 8, two different control bus versions are available: a Universal Peripheral Controller (UPC), and which generates 68000 compatible bus signals, and a General Purpose (GP) bus, which can be used to interface to a wide variety of existing microprocessor buses. With the selection of an expanded bus mode, the MK68HC200 can act either as a general purpose CPU chip (bus grant device) or as an intelligent peripheral I/O controller to a host CPU (bus request device). These two system configurations are illustrated in figures 9 and 10.

With the GP bus option, the user may configure the MK68HC200 in either of the two ways shown in figures 9 and 10. As a host CPU (Figure 9), the MK68HC200 bus arbitration logic causes the device to act as the system bus grantor. In other words, the MK68HC200 would have control of the system bus and would grant its use to DMA devices or peripheral CPUs. Alternately the MK68HC200 may be configured as a peripheral CPU (Figure 10) that must issue a request to the bus grant device before being allowed to use the system bus. The selection of one of these two configurations is accomplished by the P4-11 pin at reset time. During reset, P4-11 serves as the R/G input (0 = bus grantor, 1 = bus requestor). Following reset and at all times during program execution, P4-11 may be used as a general purpose input pin.

With the GP bus operating in the host CPU configuration, the MK68HC200 may be used to interface with external memory and I/O devices in a manner that is analagous to any general purpose microprocessor. Additionally, the MK68HC200 retains its on chip RAM and I/O resources, with on-chip ROM as an option,

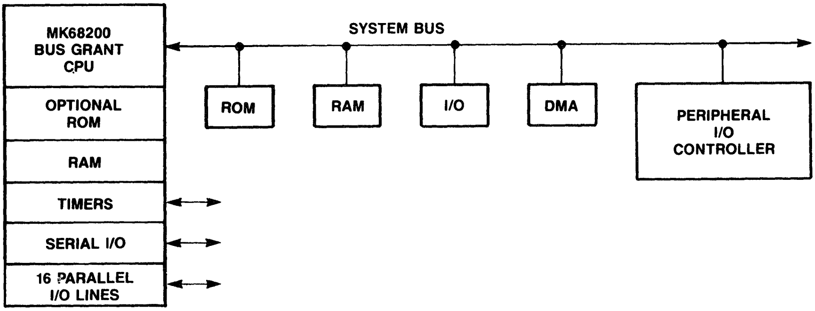


Figure 9. Host CPU Hardware Configuration

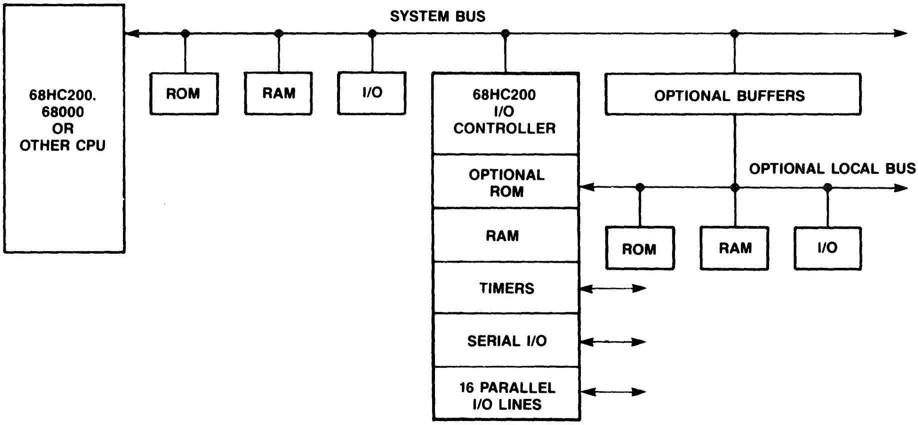


Figure 10. Peripheral I/O Controller Configuration

depending on the expansion configuration selected. BUSIN and BUSOUT are used to perform the bus arbitration handshake function, where BUSIN acts as the bus request input and BUSOUT as the bus grant output.

In the full expansion configuration, any on-chip ROM is disabled, and program memory starting at location $0000 is located off-chip and is addressed via the expanded bus, as shown in Figure 12. In effect, the internal bus from locations $0000-$FAFF is mapped onto the external bus. In the partially expanded configuration (Figure 11), on-chip ROM may be accessed on the internal bus. To gain greater addressability in the partial expansion configuration, a scheme is implemented to allow access of a full 64K-byte address space in four segments on the expanded system bus through the 16K byte "window" on the internal bus. Basically, the most significant two bits of address on the expanded bus are replaced with two user-defined segment bits available to the programmer in the expanded bus control and status register, Port 15.

As a peripheral I/O controller, the MK68HC200 operates as a bus requestor that gains mastership of the system bus from the bus grant CPU. The GP bus version may be selected to implement this system configuration in cases where an interface to a general purpose CPU is desired. In this case, the BUSIN and BUSOUT lines are again used to perform the bus arbitration handshake function, where BUSOUT now acts as bus request output, and BUSIN acts as bus grant input. In this configuration, the MK68HC200 can conceivably act as a complete peripheral I/O control subsystem on a single chip, with 16 lines of I/O and its on-chip ROM, RAM, timers, and serial I/O performing the necessary interface to the I/O device. The UPC bus version provides the peripheral I/O control function with a direct interface to a 68000 bus grant CPU. Note that the UPC bus version can operate only as a bus request device. Once the MK68HC200 has gained mastership of the system bus via the 68000 bus arbitration handshake lines (BR, BG, and BGACK), it may proceed to perform DMA transfers and communicate with system memory or other I/O devices in the system.

As in the case of the GP bus grant configuration, the portion of the internal (or private) bus address space that is mapped onto the expanded bus when the part is operating as either a GP or a UPC bus request device is determined by the expansion configuration selected. In the partial expansion bus requestor case, the resulting memory map is identical to that shown for the GP bus grant configuration in Figure 11. During the time the MK68HC200 is executing its programs from ROM and accessing internal RAM and I/O resources, the expanded bus is held in a tri-state condition. The bus arbitration logic within the MK68HC200 monitors each memory reference to detect external bus addresses (referenced in segments via the 16K byte DMA window). Whenever such an external reference occurs, the logic automatically holds the processor in a wait state as it proceeds to obtain mastership of the bus. When use of the system bus is obtained, the processor is allowed to continue the reference. This procedure is transparent to the programmer. In case of successive external references, the expanded bus is retained until an internal reference is encountered.

Finally, if the on-chip resources are insufficient to perform the control task in the bus requestor configuration, the internal bus address range (excluding on-chip RAM, I/O) may be mapped onto an external local bus, which is physically the same as the system bus but logically separated with bus buffers. This is the full expansion bus requestor configuration. The memory map for this configuration is shown in Figure 13. The bus arbitration sequence is performed only when the system bus is referenced through the DMA window. In this manner, the I/O subsystem is isolated from the host CPU.

When operating as a bus request device, it is possible to retain the external bus for an indefinite duration by using a bus lock feature. This will help facilitate the transfer of large blocks of data. Thus, the on-chip bus arbitration logic allows (with a minimum of hardware and software overhead) a maximum of concurrent processing in parallel, multiprocessing configurations. The bus lock feature may be used by the MK68HC200 in a bus grantor mode to keep any peripheral from gaining mastership of the bus.

In any of the GP expanded bus modes, the MK68HC200 may respond to peripheral devices on the expanded bus which generate an interrupt request on XI0. The MK68HC200 will obtain the XI0 interrupt vector number from the requesting peripheral on the bus during an interrupt acknowledge cycle. When responding to an interrupt on XI0, the MK68HC200 will wait for the bus arbitration logic to gain control of the bus and then asserts neither HB nor LB while asserting AS to signify that an interrupt acknowledge cycle is in progress. The X10 interrupt will be the lowest priority interrupt when operating in any of the GP expanded bus modes.

There is a user-programmable speed selection associated with the read and write cycles for both the UPC and GP mask option parts. A bit in the expanded bus control and status register, Port 15, allows the user to select either the standard or the fast read/write cycle on the expanded bus. The standard bus cycle is four clock periods, while the fast bus cycle is three clock periods.

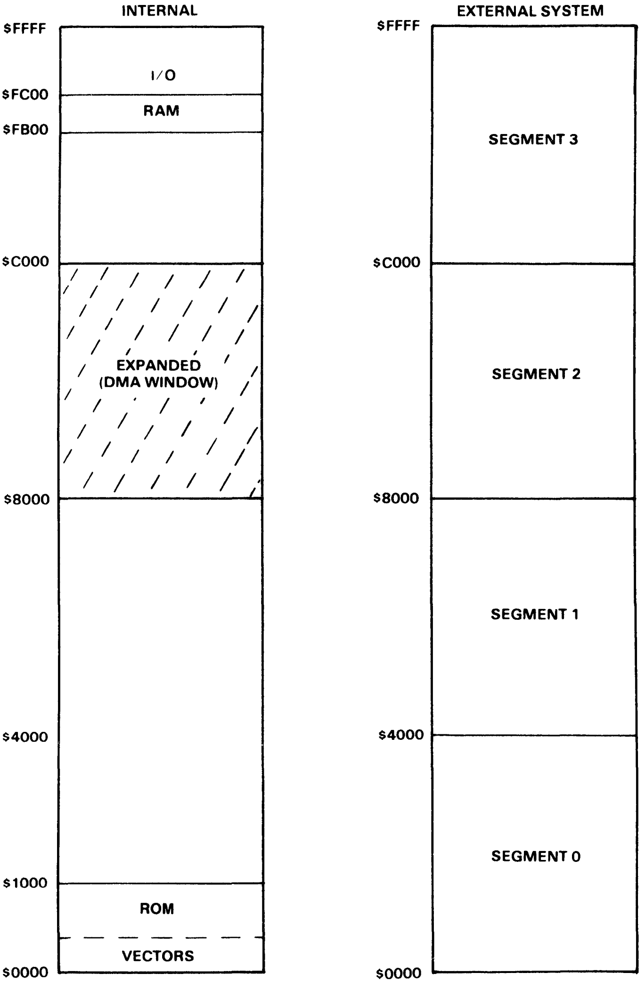


Figure 11. Parial Expansion Memory Map
(256 byte RAM, 4K byte ROM version shown)

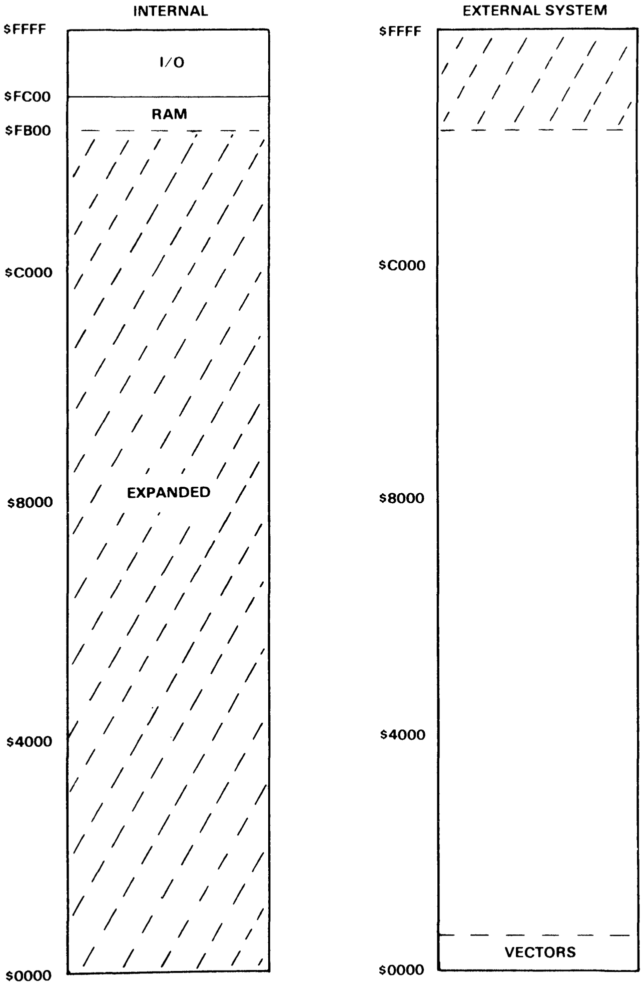


Figure 12 . Full Expansion Bus Grantor Memory Map
(256 byte RAM version shown)

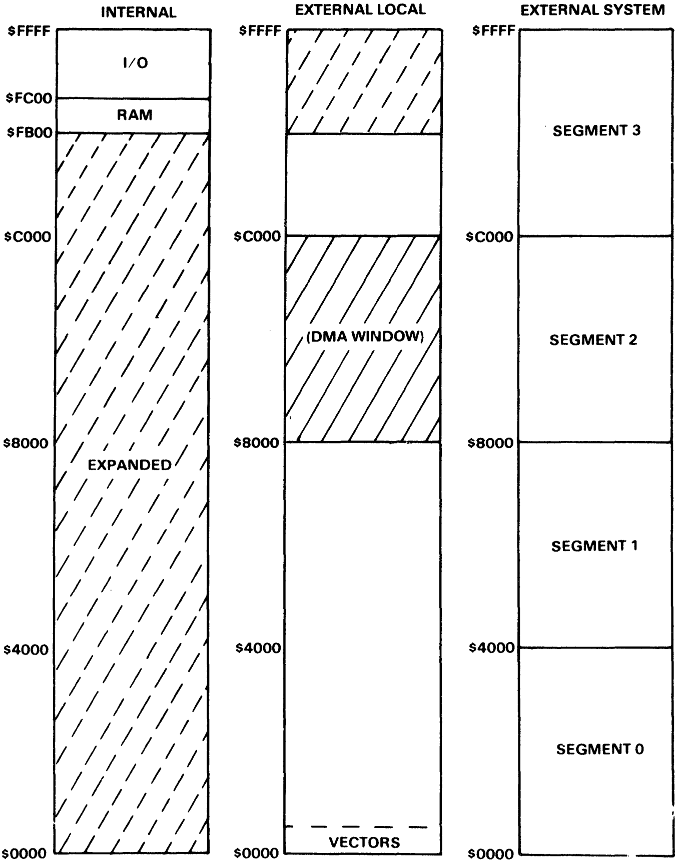


Figure 13. Full Expansion Bus Requestor Memory Map
(256 byte RAM version shown)

EMULATOR VERSION

The emulator versions of the MK68HC200 are available in 84-pin, leadless or leaded chip carrier packages. Figure 14 illustrates the logical pinout of the emulator version. The emulator versions have no on-chip ROM, but instead include a second complete bus, referred to as the private bus. The private bus includes a multiplexed address/data bus as well as bus control signals. There are 22 pins associated with the private bus. All 40 I/O port pins that exist on the 48-pin versions are available to the user for configuration either as general purpose or special I/O pins, or as expanded bus pins.

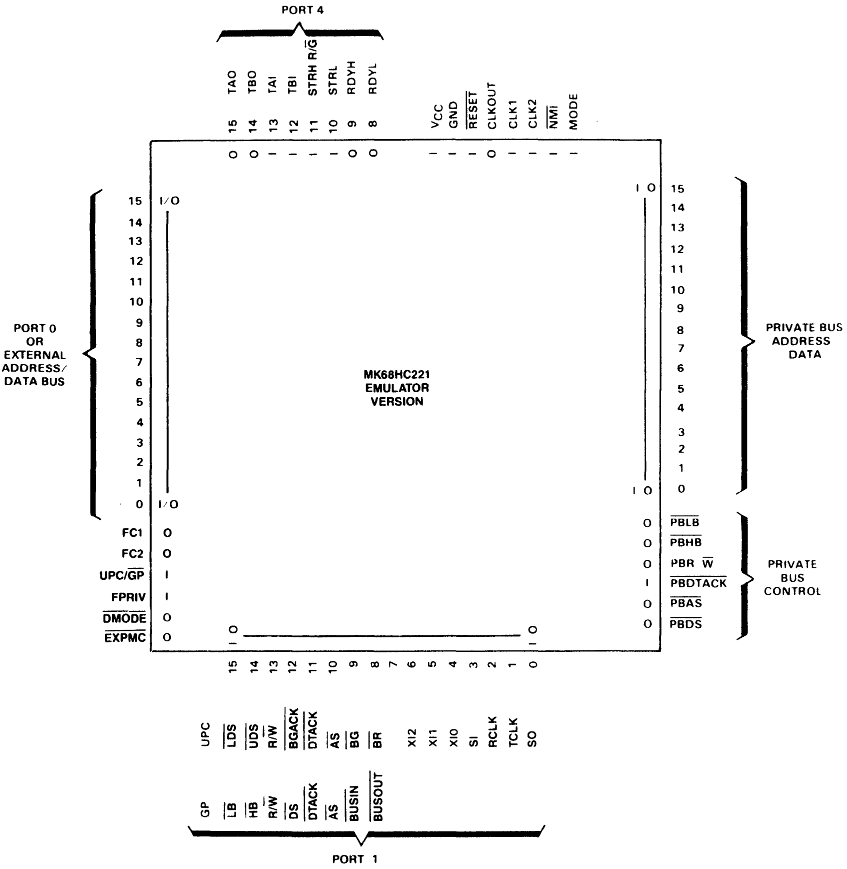


Figure 14. Logical Pinout for MK68HC221

PRIVATE BUS OPERATION

The address/data lines and control signals that constitute the private bus are functionally equivalent to the internal signals used to access internal resources on the ROM versions of the MK68HC200. Thus, the private bus may be used to interface to EPROM memory in emulating mask ROM versions of the MK68HC200. Alternately, any combination of ROM, RAM, and I/O may reside on the private bus.

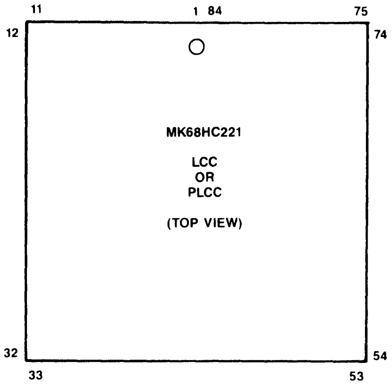
The address that is generated on the private bus is identical to that which is internally generated for 48-pin versions. When the part is used in a configuration that supports system bus addressing through the DMA window, any references in this region of the memory map produce an address on the private bus identical to that specified by the programmer. In other words, the segment bits have no effect on the private bus address. The $\overline{\text{DMODE}}$ pin will go active during a reference to the DMA windov. Write data appears on the private bus pins for all write operations, regardless of whether the reference is on-chip or off-chip. The MK68HC221 emulator version reads data from the private bus unless data is read from on-chip RAM, I/O, or the external bus formed by the Port 0 and Port 1 I/O pins.

The I/O port range of the memory map ($FC00-$FFFF) is actually subdivided into space which is exclusively reserved for on-chip I/O ($FC00-$FDFF) and space which is exclusively reserved for in-circuit-emulator use ($FE00-$FFFF). The user should ensure that no external devices reside in the in-circuit-emulator area.

The private bus interface is the same as that for the GP expanded bus. All read/write transfers made exclusively on the private bus are three clock periods, regardless of the state of the Fast/$\overline{\text{Standard}}$ (F/$\overline{\text{S}}$) bus timing selection bit. In systems using the expanded bus, the user should be sure to tie the FPRIV pin low so that expanded bus operation is not effected. The user should ignore all activity on the private bus while accesses are in progress on the expanded bus. Care should also be taken that no external devices reside on the private bus in the memory space intended for expanded bus accesses.

There are six additional control pins available on the emulator version that are not on the ROM version Five of these pins are meant to be used by the development system. FC1, FC2, $\overline{\text{DMODE}}$, and $\overline{\text{EXPMC}}$ are used to define the memory cycle currently being executed FPRIV will affect the memory cycle currently being executed. These signals are made available to simplify the design of the development system. Using these signals, the development system only has to interface to the private bus and not also to the expanded bus. The user might also be able to use these signals to simplify his design, however care should be taken when using FPRIV since this input will affect expanded bus memory cycles.

<table>
<tr><th>MNEMONIC</th><th>ADDITIONAL PIN FUNCTIONS FOR THE EMULATOR</th></tr>
<tr><td>UPC/$\overline{\text{GP}}$</td><td>UPC/$\overline{\text{GP}}$ (input, active high and low). This pin is used to select either the UPC or GP control bus configuration for the expanded bus. (1 = UPC bus, 0 = GP bus). It is sampled only when reset is active</td></tr>
<tr><td>FC1, FC2</td><td>Function Code 1. Function Code 2 (outputs, active high and low). These pins (FC1 and FC2) define the memory cycle currentiy being executed. They are valid during the time private bus address strobe ($\overline{\text{PBAS}}$) is active. The cycle types are interrupt, data, branch, and program fetch. The branch cycle is defined as the first program fetch after a branch occurs. A branch can occur as a result of a jump or call instruction, or an interrupt. For internal interrupts, the interrupt cycles are defined as the two writes to the stack and the read of the vector location which occur during the interrupt acknowledge routine. For external interrupts, the interrupt cycles are defined as the 3 cycles above plus the read of the vector number. The interrupt cycle is a special case of the data cycle. The function code pins are defined below.
<table>
<tr><th>TYPE OF CYCLE</th><th>FC1</th><th>FC2</th></tr>
<tr><td>Interrupt</td><td>0</td><td>0</td></tr>
<tr><td>Data</td><td>0</td><td>1</td></tr>
<tr><td>Branch</td><td>1</td><td>0</td></tr>
<tr><td>Program Fetch</td><td>1</td><td>1</td></tr>
</table>
</td></tr>
<tr><td>$\overline{\text{DMODE}}$</td><td>DMA Mode (output, active low). This pin goes low when the segment bits are being output on AD14 and AD15 on the expanded bus. (The address output on the private bus will not contain the segment bits.) $\overline{\text{DMODE}}$ is stable for the entire bus operation.</td></tr>
<tr><td>$\overline{\text{EXPMC}}$</td><td>Expanded Memory Cycle (output, active low). This pin goes low when the expanded bus is being accessed. $\overline{\text{EXPMC}}$ is stable for the entire bus operation.</td></tr>
<tr><td>FPRIV</td><td>Force Private (input, active high). This pin is used to force the MK68HC200 to read data from the private bus when the address is actually located on the expanded bus. In normal operation this pin should be tied low and the expanded bus operation will be unaffected.</td></tr>
</table>

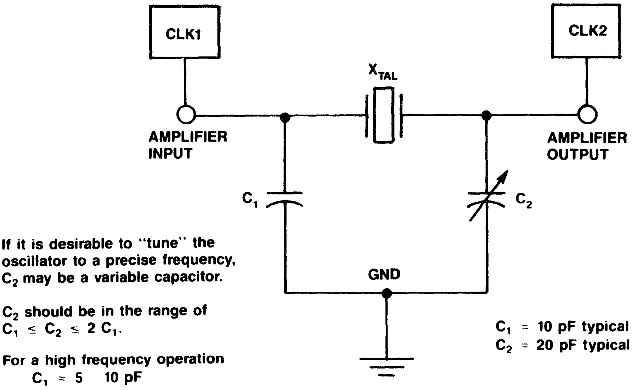


LCC	FUNCTION	LCC	FUNCTION	LCC	FUNCTION	LCC	FUNCTION
1	P1-4/XIO	22	PBAS	43	P0-7	64	PB-8
2	P1-5/XI1	23	PBDS	44	P0-6	65	PB-9
3	P1-6/XI2	24	P4-8/RDYL	45	P0-5	66	PB-10
4	P1-7	25	P4-9/RDYH	46	P0-4	67	PB-11
5	P1-8	26	P4-10/STRL	47	P0-3	68	PB-12
6	P1-9	27	P4-11/STRH	48	P0-2	69	PB-13
7	P1-10	28	MODE	49	P0-1	70	PB-14
8	P1-11	29	CLK2	50	P0-0	71	PB-15
9	EXPMC	30	CLK1	51	FC2	72	P4-13/TAI
10	VCC	31	CLKOUT	52	DMODE	73	P4-12/TBI
11	NO CONNECT	32	FC1	53	FPRIV	74	VCC
12	GROUND	33	VCC	54	VCC	75	GROUND
13	UPC/GP	34	GROUND	55	GROUND	76	P4-15/TAO
14	P1-12	35	P0-15	56	PB-0	77	P4-14/TBO
15	P1-13	36	P0-14	57	PB-1	78	RESET
16	P1-14	37	P0-13	58	PB-2	79	NMI
17	P1-15	38	P0-12	59	PB-3	80	P1-0/SO
18	PBLB	39	P0-11	60	PB-4	81	P1-1/TCLK
19	PBHB	40	P0-10	61	PB-5	82	P1-2/RCLK
20	PBR/W	41	P0-9	62	PB-6	83	P1-3/SI
21	PBDTACK	42	P0-8	63	PB-7	84	VCC

Figure 15 . MK68HC221 Pin Assignment, Emulator Version

CRYSTAL SELECTION

The wide frequency range of crystals that can be chosen for the MK68HC200 offers the user a large degree of flexibility. To aid in the selection of a suitable crystal, the suggestions shown in Figure 16 should be considered by the user. The MK68HC200 offers an output pin that will provide a system clock signal at one-half of the crystal frequency.



If it is desirable to "tune" the oscillator to a precise frequency, C_2 may be a variable capacitor.

C_2 should be in the range of $C_1 \leq C_2 \leq 2 C_1$.

For a high frequency operation $C_1 \approx 5 \quad 10$ pF

$C_1 = 10$ pF typical
$C_2 = 20$ pF typical

Figure 16. Crystal Connection

ASSEMBLER DIRECTIVES

Directive	Function	Assembler Syntax		
DC	Define constant	[label:]	DC[.size]*[1]	expr {,expr}
DS	Define storage	[label:]	DS[.size]*[1]	expr
DUP	Duplicate constant block	[label:]	DUP[.size]*[1]	length, value
END	Program end		END	[start address]
EQU	Equate symbol value	label:	EQU	expr
FAIL	Programmer generated error		FAIL	expr
FORMAT	Format the source listing		FORMAT	
IDNT	Generate module ID	module_name:	IDNT	version, revision
LIST	Enable the assembly listing		LIST	
LLEN	Specify line length		LLEN	length
NOFORMAT	Do not format listing		NOFORMAT	
NOLIST	Disable assembly listing		NOLIST	
NOOBJ	Disable object code generation		NOOBJ	
NOPAGE	Suppress paging		NOPAGE	
OFFSET	Define Offsets		OFFSET	expr
OPT	Assembler output options		OPT	option[2] {, option}
ORG	Define absolute origin		ORG	expr
PAGE	Eject a page in the listing		PAGE	
REG	Define register list	reg_list_name:	REG[.size]	register list
SECTION	Define relocatable program section	[section_name:]	SECTION	number
SET	Set symbol value	label:	SET	expr
SPC	Space between source lines		SPC	number
TTL	Specify heading title string		TTL	title string
XDEF	External symbol definition		XDEF	symbol {, symbol}
XREF	External symbol reference		XREF	[sect no:] symbol - {,[sect no]: symbol}

NOTES:

1. .size = .B or .W (byte or word size)
2. Options for the OPT directive include:

CEX	Print DC expansions
NOCEX	Do not print DC expansions (default)
CL	Print conditional assembly directives (default)
NOCL	Do not print conditional assembly directives
CRE	Print cross-reference table
IMM.L	Forces immediate operands for arithmetic instructions ADD, SUB, DADD, and DSUB to use the long instruction form
IMM.S	Allows the assembler to select automatically the short form of the arithmetic instructions for small immediate values (0-15) (default)
MC	Print macro calls (default)
NOMC	Do not print macro calls
MD	Print macro definitions (default)
NOMD	Do not print macro definitions
MEX	Print macro expansions
NOMEX	Do not print macro expansions (default)
O	Create object module (default)
NOO	Do not create object module
STR	Print code generated by structured statements
NOSTR	Do not print code generated by structured statements (default)

GENERAL SYMBOL DEFINITIONS

SYMBOL	GENERAL SYMBOL DEFINITIONS
Rn	General Purpose Registers - D0-D7, A0-A5, SP, SR, DH0-DH7, DL0-DL7.
RPn	Register Pairs - D0-D1, D2-D3, D4-D5, D6-D7, A0-A1, A2-A3, A4-A5.
An	Address Registers - A0-A5, SP.
Pn	Ports - P0-P15, PH0-PH3, PL0-PL3.
CC	Condition Code - See Table.
d16	16-Bit Address Displacement Field In Words.
d13	13-Bit Address Displacement Field In Bytes.
d9	9-Bit Address Displacement Field In Bytes.
d8	8-Bit Address Displacement Field In Bytes.
#nx	Immediate Data Field - x Number of Bits.[1]
$\boxed{\text{s}}$	Size Bit - '1' = Word, '0' = Byte.
REGn	4-Bit Register Field - See Table.
$\boxed{\text{PORTn}}$	4-Bit Port Field - See Table.
$\boxed{\text{An}}$	3-Bit Address Register Field - See Table.
$\boxed{\text{PRTn}}$	3-Bit Port Field - See Table
$\boxed{\text{RGn}}$	3-Bit Register Pair Field - See Table.
M	Register Mask Field - See Table.
$\boxed{\text{COND}}$	Condition Code Field - See Table.
$\boxed{\text{c3}}$	3-Bit Class Field - See Table.
$\boxed{\text{c2}}$	2-Bit Class Field - See Table
$\boxed{\text{c1}}$	1-Bit Class Field—See Table
a	Address Field - 16 Bits.
#	Immediate Data Field.
n	3-Bit Shift Field - $2 \leq n \leq 7$.
b#	4-Bit Bit Select Field.
d	Displacement Field.
.B	Byte Attribute.
.W	Word Attribute.
.L	Long Attribute.
.S	Short Attribute.
[ ]	Optional Field.

NOTE

[1] When using the byte format of an instruction with a 16-bit immediate data field, both the high and low byte of the data field must contain the same 8-bit data.

FIELD DEFINITIONS

REGn 4-Bit Register Map					
Register		Bit Field	Register		Bit Field
D0	DH0	0 0 0 0	A0	DL0	1 0 0 0
D1	DH1	0 0 0 1	A1	DL1	1 0 0 1
D2	DH2	0 0 1 0	A2	DL2	1 0 1 0
D3	DH3	0 0 1 1	A3	DL3	1 0 1 1
D4	DH4	0 1 0 0	A4	DL4	1 1 0 0
D5	DH5	0 1 0 1	A5	DL5	1 1 0 1
D6	DH6	0 1 1 0	SP	DL6	1 1 1 0
D7	DH7	0 1 1 1	SR	DL7	1 1 1 1

PORTn 4-Bit Port Map				
Port		Bit Field	Port	Bit Field
P0	PH0	0 0 0 0	P8	1 0 0 0
P1	PL0	0 0 0 1	P9	1 0 0 1
P2	PH1	0 0 1 0	P10	1 0 1 0
P3	PL1	0 0 1 1	P11	1 0 1 1
P4	PH2	0 1 0 0	P12	1 1 0 0
P5	PL2	0 1 0 1	P13	1 1 0 1
P6	PH3	0 1 1 0	P14	1 1 1 0
P7	PL3	0 1 1 1	P15	1 1 1 1

An 3-Bit Addr Reg Map	
Register	Bit Field
A0	0 0 0
A1	0 0 1
A2	0 1 0
A3	0 1 1
A4	1 0 0
A5	1 0 1
SP	1 1 0

PTRn 3-Bit Port Map	
Port	Bit Field
PH0	0 0 0
PL0	0 0 1
PH1	0 1 0
PL1	0 1 1
PH2	1 0 0
PL2	1 0 1
PH3	1 1 0
PL3	1 1 1

RGn 3-Bit Reg Pair Map	
Register	Bit Field
D0-D1	0 0 0
D2-D3	0 0 1
D4-D5	0 1 0
D6-D7	0 1 1
A0-A1	1 0 0
A2-A3	1 0 1
A4-A5	1 1 0

M—REGISTER MASK MAP FOR MOVEM, PUSHM, AND POPM

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Inc Word	SR	SP	A5	A4	A3	A2	A1	A0	D7	D6	D5	D4	D3	D2	D1	D0
Dec Word	D0	D1	D2	D3	D4	D5	D6	D7	A0	A1	A2	A3	A4	A5	SP	SR
Inc Byte	DH7	DL7	DH6	DL6	DH5	DL5	DH4	DL4	DH3	DL3	DH2	DL2	DH1	DL1	DH0	DL0
Dec Byte	DL0	DH0	DL1	DH1	DL2	DH2	DL3	DH3	DL4	DH4	DL5	DH5	DL6	DH6	DL7	DH7

$\boxed{\text{COND}}$ CONDITION CODE TABLE

Condition Code	Bit Field	Description	Test
Z EQ	0 0 0 0	Zero Equal	Z
MI	0 0 0 1	Minus	N
LO[2] CS	0 0 1 0	Lower Carry Set	C
VS	0 0 1 1	Overflow Set	V
GE[2]	0 1 0 0	Greater than or Equal	$\overline{N \text{ .EOR. } V}$
GT[2]	0 1 .0 1	Greater than	$\overline{Z}$ .AND. ($\overline{N \text{ .EOR. } V}$)
HI[2]	0 1 1 0	Higher	$\overline{C}$ .AND. $\overline{Z}$
F[1]	0 1 1 1	False	Always False
NE NZ	1 0 0 0	Not Equal Not Zero	$\overline{Z}$
PL	1 0 0 1	Plus	$\overline{N}$
HS[2] CC	1 0 1 0	Higher or Same Carry Clear	$\overline{C}$
VC	1 0 1 1	Overflow Clear	$\overline{V}$
LT[2]	1 1 0 0	Less than	N .EOR. V
LE[2]	1 1 0 1	Less than or Equal	Z .OR. (N .EOR. V)
LS[2]	1 1 1 0	Lower or Same	C .OR. Z
T[1]	1 1 1 1	True	Always True

NOTES:

1. The assembler does not recognize the T and F condition codes.
2. LT, LE, GT, and GE are used for unsigned conditions; LO, L' HI, and HS are for unsigned conditions.

INSTRUCTION CLASS FIELDS									
c3 - 3-Bit Field			c2 - 2-Bit Field			c1 - 1-Bit Field			
Bit Field	Shift Instr	Bit[1] Instr	Bit Field	Arith[2] Instr	Logical Instr	Bit Field	Arith[2] Instr	Test Instr	Neg[2] Instr
0 0 0	ROR	BSET	0 0	ADDC	OR	0	ADD	TESTN	NEGC
0 0 1	ROL	BCHG	0 1	SUBC	EOR	1	SUB	TEST	NEG
0 1 0	RORC	BCLR	1 0	ADD	AND				
0 1 1	ROLC	BTST	1 1	SUB	—				
1 0 0	ASR	—							
1 0 1	ASL	—							
1 1 0	LSR	—							
1 1 1	—	BEXG							

NOTES:

1. The bit fields do not apply to bit instructions using a port operand.
2. These fields also apply to BCD instructions.

INSTRUCTION FORMAT

15 0

PREFIX WORD **(used only in some forms of the decimal and bit instructions)**
OPERATION WORD **(contains the opcode and possibly the operands)**
EXTENSION WORD **(optional - specified by the operation word to be immediate operand, mask field, displacement or absolute address)**

INSTRUCTION MNEMONIC	ATTR SIZE	OPERAND ASSEMBLER SYNTAX Source = Src Destination = Dst Src . Dst	OPERATION WORD 15 14 13 12	11	10 9 8	7	6 5	4	3	2 1 0	WORDS	EXTENSION	CYCLES	OPERATION	STATUS FLAGS N	Z	V	C
ADD ADDC SUB SUBC	B [W]	Ry.Rx	2		REGx	s	:2	0		REGy	1	—	3	ADD Src · Dst · Dst	*	*	*	*
	B [W]	(Ay).Rx	2		REGx	s	c2	1	0	Ay	1	–	6	ADDC Src · Dst · C · Dst				
	B [W]	d16(Ay).Rx	2		REGx	s	c2	1	1	Ay	2	d	9	SUB Dst Src · Dst				
	B [W]	Addr Rx	2		REGx	s	c2	1		F	2	a	9	SUBC Dst Src C · Dst				
	B [W]	#n16 Rx	2		REGx	s	c2	1		7	2	#	6	Note For addressing modes #n.Rx and #n.(Ax) with the ADD and SUB instructions. the assembler uses the short version for immediate values < 4 bits				
	B [W]	Ry (Ax)	3	0	Ax	s	c2	0		REGy	1	-	9					
	B [W]	(Ax) (Ay)	3	0	Ax	s	c2	1	0	Ay	1	..	12					
	B [W]	#n16.(Ax)	3	0	Ax	s	c2	1		7	2	#	12					
	B [W]	(Ax) · (Ay) ·	3	1	Ax	s	c2	1	0	Ay	1		12					
	B [W]	#n16.(Ax) ·	3	1	Ax	s	c2	1		7	2	#	12					
	B [W]	(Ax) (Ay)	3	1	Ax	s	c2	1	1	Ay	1		12					
	B [W]	#n16. (Ax)	3	1	Ax	s	c2	1		F	2	#	12					
	B [W]	Ry.d16(Ax)	3	1	Ax	s	c2	0		REGy	2	d	12					
	B [W]	Ry.Addr	3		F	s	c2	0		REGy	2	a	12					

INSTRUCTION MNEMONIC	ATTR SIZE	OPERAND ASSEMBLER SYNTAX Source Src Destination Dst Src . Dst	OPERATION WORD 15 14 13 12	11	10 9 8	7	6 5	4	3	2 1 0	WORDS	EXTENSION	CYCLES	OPERATION	STATUS FLAGS N	Z	V	C
ADD SUB	B [W]	#n4 Rx	B		REGx	s	1 c1	0		#	1		3	ADD Src + Dst · Dst	*	*	*	*
	B [W]	#n4.(Ax)	B	0	Ax	s	1 c1	1		#	1		9					
	B [W]	#n4.d16(Ax)	B	1	Ax	s	1 c1	1		#	2	d	12	SUB Dst Src · Dst				
	B [W]	#n4.Addr	B		F	s	1 c1	1		#	2	a	12					
AND · EOR OR	B [W]	Ry.Rx	6		REGx	s	c2	0		REGy	1		3	AND Src AND Dst Dst	*	*	0	0
	B [W]	Py.Rx	5		REGx	s	c2	0		PORTy	1		6					
	B [W]	(Ay) Rx	6		REGx	s	c2	1	0	Ay	1		6	EOR Src EOR Dst · Dst				
	B [W]	d16(Ay) Rx	6		REGx	s	c2	1	1	Ay	2	d	9					
	B [W]	Addr Rx	6		REGx	s	c2	1		F	2	a	9	OR Src OR Dst Dst				
	B [W]	#n16.Rx	6		REGx	s	c2	1		7	2	#	6					
	B [W]	Ry.Px	4		PORTx	s	c2	0		REGy	1		9					
	[W]	#n16.Px	5		PORTx	1	c2	1		9	2	#	12					
	B [W]	Ry.(Ax)	7	0	Ax	s	c2	0		REGy	1		9					
Continued on next page	B [W]	(Ax).(Ay)	7	0	Ax	s	c2	1	0	Ay	1		12					

INSTRUCTION MNEMONIC	ATTR SIZE	OPERAND ASSEMBLER SYNTAX Source · Src Destination · Dst Src , Dst	OPERATION WORD 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0	WORDS	EXTENSION	CYCLES	OPERATION	STATUS FLAGS			
								N	Z	V	C
AND EOR OR (cont.)	B [W]	#n16.(Ax)	7 0 Ax s c2 1 7	2	#	12	AND Src AND Dst Dst	*	*	0	0
	B [W]	(Ax) · (Ay) ·	7 1 Ax s c2 1 0 Ay	1	--	12					
	B [W]	#n16.(Ax) ·	7 1 Ax s c2 1 7	2	#	12	EOR Src EOR Dst · Dst				
	B [W]	(Ax) (Ay)	7 1 Ax s c2 1 1 Ay	1	--	12					
	B [W]	#n16. (Ax)	7 1 Ax s c2 1 F	2	#	12	OR Src OR Dst · Dst				
	B [W]	Ry.d16(Ax)	7 1 Ax s c2 0 REGy	2	d	12					
	B [W]	Ry.Addr	7 F s c2 0 REGy	2	a	12					

INSTRUCTION MNEMONIC	ATTR SIZE	OPERAND ASSEMBLER SYNTAX Source · Src Destination · Dst Src , Dst	OPERATION WORD 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0	WORDS	EXTENSION	CYCLES	OPERATION	STATUS FLAGS			
								N	Z	V	C
ASL ASR LSR ROL ROLC ROR RORC	B [W]	[#1].Rx	B REGx s 0 c3 0 0 1	1	—	3	Dst · Dst SHIFT	*	*	*	*
	B [W]	#n3.Rx	B REGx s 0 c3 n	1	—	4 + n*	C ◄ · ASL ◄ · O O · ► LSR · ► C	NOTE The ASR. LSR ROR and RORC instructions will clear the V status bit			
	B [W]	(Ax)	B 0 Ax s 0 c3 0 0 0	1	—	9	► ASR ► C				
	B [W]	d16(Ax)	B 1 Ax s 0 c3 0 0 0	2	d	12	ROTATE C ◄ · ROL ◄				
	B [W]	Addr	B F s 0 c3 0 0 0	2	a	12	C ◄ · ROLC ◄ ► ROR · ► C ► RORC · ► C *NOTE 2 < n < 7				

INSTRUCTION MNEMONIC	ATTR SIZE	OPERAND ASSEMBLER SYNTAX Source = Src Destination = Dst Src , Dst	OPERATION WORD 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0	WORDS	EXTENSION	CYCLES	OPERATION	STATUS FLAGS			
								N	Z	V	C
BCHG BCLR BEXG BSET BTST	—	#n4.Rx	4 REGx c3 1 b#	1	—	3	BCHG: Dst(b#) - C 1 — Dst(b#) - Dst(b#)	*	*	0	*
	—	#n4.(Ax)	5 0 Ax c3 1 b#	1	—	9*		If Dst(b#) is in the low byte of the SR,then the other SR bits are unchanged			
	—	#n4.d16(Ax)	5 1 Ax c3 1 b#	2	d	12*	BCLR: Dst(b#) - C 0 - Dst(b#)				
	—	#n4.Addr	5 F c3 1 b#	2	a	12*					
	—	Ry.Rx	0 REGy F 7	2	—	6	BEXG: Dst(b#) - C				
			4 REGx c3 1 0								
	—	Ry.(Ax)	0 REGy F 7	2	—	12*	BSET: Dst(b#) - C 1 - Dst(b#)				
			5 0 Ax c3 1 0								
	—	Ry.d16(Ax)	0 REGy F 7	3	d	15*	BTST: Dst(b#) - C				
			5 1 Ax c3 1 0								
	—	Ry.Addr	0 REGy F 7	3	a	15*	*NOTE: The BTST instruction execute n 3 less clock cycles.				
			5 F c3 1 0								

INSTRUCTION MNEMONIC	ATTR SIZE	OPERAND ASSEMBLER SYNTAX Source = Src Destination = Dst Src , Dst	OPERATION WORD				WORDS	EXTENSION	CYCLES	OPERATION	STATUS FLAGS			
			15 14 13 12	11 10 9 8	7 6 5 4	3 2 1 0					N	Z	V	C
BCHG	—	#n4,Px	4	PORTx	B	b#	1	—	9	Dst(b#) → C 1 — Dst(b#) → Dst(b#)	*	*	0	*
	—	Ry,Px	0	REGy	F	7	2	—	12					
			4	PORTx	B	0								
BCLR	—	#n4,Px	4	PORTx	D	b#	1	—	9	Dst(b#) → C 0 → Dst(b#)				
	—	Ry,Px	0	REGy	F	7	2	—	12					
			4	PORTx	D	0								
BEXG	—	#n4,Px	7	PORTx	E	b#	1	—	9	Dst(b#) ↔ C				
	—	Ry,Px	0	REGy	F	7	2	—	12					
			7	PORTx	E	0								
BSET	—	#n4,Px	4	PORTx	9	b#	1	—	9	Dst(b#) → C 1 → Dst(b#)				
	—	Ry,Px	0	REGy	F	7	2	—	12					
			4	PORTx	9	0								
BTST Continued on next page	—	#n4,Px	7	PORTx	6	b#	1	—	6	Dst(b#) - C				

INSTRUCTION MNEMONIC	ATTR SIZE	OPERAND ASSEMBLER SYNTAX Source = Src Destination = Dst Src , Dst	OPERATION WORD				WORDS	EXTENSION	CYCLES	OPERATION	STATUS FLAGS			
			15 14 13 12	11 10 9 8	7 6 5 4	3 2 1 0					N	Z	V	C
BTST (cont.)	—	Ry,Px	0	REGy	F	7	2	—	9	Dst (b#) - C	*	*	0	*
			7	PORTx	6	0								
CALLA	[.L]	(Ax) (unconditional)	5	0 Ax	D	F	1	—	9	PC + 2 - (SP) (Ax) - PC	—	—	—	—
	[.L]	Addr (unconditional)	5	F	B	F	2	a	9	PC + 4 - (SP) Addr - PC				
	[.L]	CC, Addr	5	COND	B	F	2	a	F:6 T:12	If COND is true, PC + 4 - (SP); Addr - PC				
										NOTE The initial PC value is the location of the CALLA instruction				
CALLR	[.S]	d13 (unconditional)	F		d		1	—	10	PC · 2 - (SP) PC · 4 · 2·(d) - PC	—	—	—	—
	L	d16 (unconditional)	5	F	9	F	2	d	9	PC · 4 - (SP) PC · 4 · d - PC				
	L	CC, d16	5	COND	9	F	2	d	F:6 T:12	If COND is true, PC · 4 - (SP); PC · 4 · d - PC				
										NOTE The initial PC value is the location of the CALLR instruction The displacement, d13, is in signed magnitude represe⁻tation, and the displace ent, d16, is in two's co..plement representation				

INSTRUCTION MNEMONIC	ATTR SIZE	OPERAND ASSEMBLER SYNTAX Source = Src Destination = Dst	OPERATION WORD				WORDS	EXTENSION	CYCLES	OPERATION	STATUS FLAGS			
		Src , Dst	15 14 13 12	11 10 9 8	7 6 5 4	3 2 1 0					N	Z	V	C
CLR	.B [.W]	Rx	1	REGx	s 0 1 1	7	1	—	3	0 → Dst	—	—	—	—
	.B [.W]	Px	1	PORTx	s 0 1 0	7	1	—	6					
	.B [.W]	(Ax)	1	0 Ax	s 0 1 1	F	1	—	6					
	.B [.W]	d16(Ax)	1	1 Ax	s 0 1 1	F	2	d	9					
	.B [.W]	Addr	1	F	s 0 1 1	F	2	a	9					
CMP	.B [.W]	Ry,Rx	6	REGx	s 1 1 0	REGy	1	—	3	Dst — Src	*	*	*	*
	.B [.W]	Py,Rx	5	REGx	s 1 1 0	PORTy	1	—	6					
	.B [.W]	(Ay),Rx	6	REGx	s 1 1 1	0 Ay	1	—	6					
	.B [.W]	(Ay) + ,Rx	4	REGx	s 1 1 0	0 Ay	1	—	6					
	.B [.W]	– (Ay),Rx	4	REGx	s 1 1 0	1 Ay	1	—	6					
	.B [.W]	d16(Ay),Rx	6	REGx	s 1 1 1	1 Ay	2	d	9					
	.B [.W]	Addr,Rx	6	REGx	s 1 1 1	F	2	a	9					
	.B	#n8,Rx	C	REGx	#		1	—	3					
Continued on next page	[.W]	#n16,Rx	6	REGx	F	7	2	#	6					

INSTRUCTION MNEMONIC	ATTR SIZE	OPERAND ASSEMBLER SYNTAX Source = Src Destination = Dst	OPERATION WORD				WORDS	EXTENSION	CYCLES	OPERATION	STATUS FLAGS			
		Src , Dst	15 14 13 12	11 10 9 8	7 6 5 4	3 2 1 0					N	Z	V	C
CMP (cont.)	.B	#n8,Px	D	0 PRTx	#		1	—	6	Dst — Src	*	*	*	*
	[W]	#n16,Px	4	PORTx	E	F	2	#	9					
	B [.W]	(Ax),(Ay)	7	0 Ax	s 1 1 1	0 Ay	1	—	9					
	.B [.W]	#n16,(Ax)	7	0 Ax	s 1 1 1	7	2	#	9					
	.B [.W]	(Ax) + ,(Ay) +	7	1 Ax	s 1 1 1	0 Ay	1	—	9					
	.B [.W]	#n16,(Ax) +	7	1 Ax	s 1 1 1	7	2	#	9					
	.B [.W]	(Ax), (Ay)	7	1 Ax	s 1 1 1	1 Ay	1	—	9					
	.B [.W]	#n16, – (Ax)	7	1 Ax	s 1 1 1	F	2	#	9					
DADD DADDC DSUB DSUBC	.B [.W]	Ry,Rx	A	REGx	s c2 0	REGy	1	—	3	DADD. BCD: [Src + Dst → Dst]	U	*	U	*
	B [.W]	(Ay),Rx	A	REGx	s c2 1	0 Ay	1	—	6	DADDC BCD: [Src + Dst + C → Dst]				
	.B [.W]	d16(Ay),Rx	A	REGx	s c2 1	1 Ay	2	d	9	DSUB BCD: [Ds Src - Dst]				
	B [W]	Addr,Rx	A	REGx	s c2 1	F	2	a	9					
Continued on next page	.B [.W]	#n16,Rx	A	REGx	s c2 1	7	2	#	6	DSUBC BCD [Dst Src C - Dst]				

INSTRUCTION MNEMONIC	ATTR SIZE	OPERAND ASSEMBLER SYNTAX Source = Src Destination = Dst Src , Dst	OPERATION WORD 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0	WORDS	EXTENSION	CYCLES	OPERATION	N	Z	V	C
DADD DADDC DSUB DSUBC (cont.)	.B [.W]	Ry,(Ax)	0 F F F	2	—	12		U	*	U	*
			3 0 Ax s c2 0 REGy								
	.B [.W]	(Ax),(Ay)	0 F F F	2	—	15	NOTE For addressing modes #n,Rx and #n,(Ax) with the DADD and DSUB instructions, the assembler uses the short version for immediate values < 4 bits				
			3 0 Ax s c2 1 0 Ay								
	B [.W]	#n16,(Ax)	0 F F F	3	#	15					
			3 0 Ax s c2 1 /								
	.B [.W]	(Ax) + ,(Ay) +	0 F F F	2	—	15					
			3 1 Ax s c2 1 0 Ay								
	.B [.W]	#n16,(Ax) +	0 F F F	3	#	15					
			3 1 Ax s c2 1 7								
	.B [.W]	− (Ax), − (Ay)	0 F F F	2	—	15					
			3 1 Ax s c2 1 1 Ay								
	.B [.W]	#n16, − (Ax)	0 F F F	3	#	15					
Continued on next page			3 1 Ax s c2 1 F								

INSTRUCTION MNEMONIC	ATTR SIZE	OPERAND ASSEMBLER SYNTAX Source = Src Destination = Dst Src . Dst	OPERATION WORD 15 14 13 12 11 10 9 8 7 6 5 4,3 2 1 0	WORDS	EXTENSION	CYCLES	OPERATION	N	Z	V	C
DADD DADDC DSUB DSUBC (cont)	B [W]	Ry.d16(Ax)	0 F F F	3	d	15		U	*	U	*
			3 1 Ax s c2 0 REGy								
	B [W]	Ry Addr	0 F F F	3	a	15					
			3 F s c2 0 REGy								
DADD DSUB	B [W]	#n4.Rx	0 F F F	2	—	6	DADD: BCD: [Src + Dst → Dst]	U	*	U	*
			B REGx s 1 c1 0 #								
	B [W]	#n4.(Ax)	0 F F F	2	—	12	DSUB: BCD: [Dst − Src → Dst]				
			B 0 Ax s 1 c1 1 #								
	B [W]	#n4.d16(Ax)	0 F F F	3	d	15					
			B 1 Ax s 1 c1 1 #								
	.B [.W]	#n4.Addr	0 F F F	3	a	15					
			B F s 1 c1 1 #								
DI	--	—	4 F 5 7	1	—	3	0 → L1E : Disable Interrupts	—	—	—	—

INSTRUCTION MNEMONIC	ATTR SIZE	OPERAND ASSEMBLER SYNTAX Source = Src Destination - Dst Src , Dst	OPERATION WORD				WORDS	EXTENSION	CYCLES	OPERATION	STATUS FLAGS			
			15 14 13 12	11 10 9 8	7 6 5 4	3 2 1 0					N	Z	V	C
DIVU	–	Rx.RPy	·5	REGx	D	RGy 0	1	-	F 6 T 23	RPy(even) RPy(odd)/Rx · RPy(even) REM · RPy(odd) MSW even LSW odd	–	–	*	–
DJNZ	B [W]	Rx.d8	9	REGx	s	d	1	—	F 6 T 9	Rx 1 · Rx If Rx ≠ 0 then PC · 4 [2·(d)] · PC NOTE The initial PC value is the location of the DJNZ instruction The displacement value d is a magnitude	—	—	—	—
DNEG DNEGC	B [W]	Ry	0	F	F	F	2	—	6	DNEG BCD [0 Dst - Dst]	U	*	U	*
			3	7	s c1 1 0	REGy								
	B [W]	(Ay)	0	F	F	F	2	--	12	DNEGC BCD [0 Dst C · Dst]				
			3	7	s c1 1 1	0 Ay								
	B [W]	d16(Ay)	0	F	F	F	3	d	15					
			3	7	s c1 1 1	1 Ay								
	B [W]	Addr	0	F	F	F	3	a	15					
			3	7	s c1 1 1	F								
DSUB DSUBC		see DADD												

INSTRUCTION MNEMONIC	ATTR SIZE	OPERAND ASSEMBLER SYNTAX Source=Src Destination=Dst Src , Dst	OPERATION WORD				WORDS	EXTENSION	CYCLES	OPERATION	STATUS FLAGS			
			15 14 13 12	11 10 9 8	7 6 5 4	3 2 1 0					N	Z	V	C
EI	—	—	4	F	1	7	1	—	3	1 → L1E Enable Interrupts	—	—	—	—
EOR		see AND (page 9)												
EXG	.B [.W]	Rx,Ry	0	REGx	s 1 0 0	REGy	1	—	4	Src → Dst	—	—	—	—
	.B [.W]	Rx,(Ay)	0	REGx	s 1 0 1	0 Ay	1	—	9					
	.B [.W]	Rx,(Ay) +	0	REGx	s 1 1 1	0 Ay	1	—	9					
	.B [.W]	Rx, – (Ay)	0	REGx	s 1 1 1	1 Ay	1	—	9					
	.B [.W]	Rx,d16(Ay)	0	REGx	s 1 0 1	1 Ay	2	d	12					
	.B [.W]	Rx,Addr	0	REGx	s 1 0 1	F	2	a	12					
EXT	—	Rx	0	REGx	D	7	1	—	3	SIGN EXTEND: SignBit(RLx) → RHx	—	—	—	—
HALT	—	—	1	7	3	F	1	—	—	PC + 2 → PC: Fetch continuously	—	—	—	—
IDLE	—	—	1	7	1	F	1	—	—	PC + 2 → PC; Pause; CPU clocks off, I/O still active	—	—	—	—
JMPA	[.L]	(Ax) (unconditional)	5	1 Ax	D	F	1	—	6	(Ax) → PC	—	—	—	—
	[.L]	Addr (unconditional)	5	F	B	7	2	a	6	Addr →				
	[.L]	CC,Addr	5	COND	B	7	2	a	F:6 T:9	If COND is true, Addr → PC				
LINK	—	Ax,#d16	5	1 Ax	9	D	2	a	9	Ax → -(SP); SP → Ax; SP+d → SP	—	—	—	—

INSTRUCTION MNEMONIC	ATTR SIZE	OPERAND ASSEMBLER SYNTAX Source = Src Destination = Dst Src , Dst	OPERATION WORD 15 14 13 12	 11 10 9 8	 7 6 5 4 3 2 1 0	WORDS	EXTENSION	CYCLES	OPERATION	STATUS FLAGS N	 Z	 V	 C
JMPR	[.S]	d9 (unconditional)	8	F	d	1	–	7	PC · 4 · 2·(d) · PC				
	[S]	CC.d9	8	COND	d	1	–	F 4 T7	If COND is true. PC · 4 · 2·(d) · PC				
	.L	d16 (unconditional)	5	F	9 7	2	d	9	PC · 4 · d · PC				
	L	CC.d16	5	COND	9 7	2	d	F 6 T9	If COND is true. PC · 4 · d · PC				
									NOTE The initial PC value is the location of the JMPR instruction The displacement. d16. is in two's complement representation. and the displacement. d9. is in signed magnitude representation				
LIBA LIWA	—	#d16(Rx).Az	1	REGx	s 1 Az 1 1 1	2	#	6	Rx[·2] · d · Az				
	—	#d16(Ay.Rx).Az	1	REGx	s 1 Az Ay	2	#	6	Ay · (Rx[·2]) · d · Az				
									NOTE [·2] used for LIWA only. REGx and Az must not refer to same register when using 2nd addressing mode				
LSR		see ASL (page 10)											
MOVE	.B [.W]	Ry.Rx	0	REGx	s 0 0 0 REGy	1	—	3	Src · Dst	—	—	—	—
	.B [.W]	Py.Rx	0	REGx	s 0 1 1 PORTy	1	—	6					
Continued on next page	.B [.W]	(Ay).Rx	0	REGx	s 0 0 1 0 Ay	1	—	6					

INSTRUCTION MNEMONIC	ATTR SIZE	OPERAND ASSEMBLER SYNTAX Source = Src Destination = Dst Src . Dst	OPERATION WORD 15 14 13 12	 11 10 9 8	 7 6 5 4 3 2 1 0	WORDS	EXTENSION	CYCLES	OPERATION	STATUS FLAGS N	 Z	 V	 C
MOVE (cont)	B [W]	(Ay) · Rx	1	REGx	s 0 1 1 0 Ay	1	—	6	Src · Dst	—	—	—	—
	B [W]	(Ay) Rx	1	REGx	s 0 1 1 1 Ay	1	—	6					
	B [W]	d16(Ay) Rx	0	REGx	s 0 0 1 1 Ay	2	d	9					
	B [W]	Addr Rx	0	REGx	s 0 0 1 F	2	a	9					
	B	#n8 Rx	E	REGx	#	1	—	3					
	[W]	#n16 Rx	0	REGx	9 7	2	#	6					
	B [W]	Ry Px	0	PORTx	s 0 1 0 REGy	1	—	6					
	B [W]	(Ay) · Px	0	PORTx	s 1 1 0 0 Ay	1	—	9					
	B	#n8 Px	D	1 PRTx	#	1	—	6					
	[W]	#n16 Px	0	PORTx	E 7	2	#	9					
	B [W]	Ry (Ax)	1	0 Ax	s 0 0 0 REGy	1	—	6					
	B [W]	(Ax) (Ay)	1	0 Ax	s 0 0 1 0 Ay	1	—	9					
	B [W]	#n16 (Ax)	1	0 Ax	s 0 0 1 7	2	#	9					
Continued on next page	B [W]	Rx (Ay) ·	1	REGx	s 0 1 0 0 Ay	1	—	6					

INSTRUCTION MNEMONIC	ATTR SIZE	OPERAND ASSEMBLER SYNTAX Source = Src Destination = Dst Src , Dst	OPERATION WORD				WORDS	EXTENSION	CYCLES	OPERATION	STATUS FLAGS			
			15 14 13 12	11 10 9 8	7 6 5 4	3 2 1 0					N	Z	V	C
MOVE (cont.)	.B [W]	(Ax) + ,(Ay) +	1	1 Ax	s 0 0 1	0 Ay	1	—	9	Src → Dst	—	—	—	—
	.B [W]	#n16,(Ax) +	1	1 Ax	s 0 0 1	7	2	#	9					
	B [W]	Rx, – (Ay)	1	REGx	s 0 1 0	1 Ay	1	—	6					
	B [W]	Px, – (Ay)	0	PORTx	s 1 1 0	1 Ay	1	—	9					
	.B [W]	– (Ax), – (Ay)	1	1 Ax	s 0 0 1	1 Ay	1	—	9					
	B [W]	#n16, – (Ax)	1	1 Ax	s 0 0 1	F	2	#	9					
	B [W]	Ry,d16(Ax)	1	1 Ax	s 0 0 0	REGy	2	d	9					
	B [W]	Ry,Addr	1	F	s 0 0 0	REGy	2	a	9					
MOVEM	B [W]	(Ay) + ,REGLIST	1	7	s 0 0 1	0 Ay	2	M	9 + 3n	(Ay) + → REGLIST	—	—	—	—
	B [W]	– (Ay),REGLIST	1	7	s 0 0 1	1 Ay	2	M	9 + 3n	– (Ay) → REGLIST				
	B [W]	REGLIST,(Ay) +	1	F	s 0 0 1	0 Ay	2	M	7 + 3n	REGLIST → (Ay) +				
	B [W]	REGLIST, – (Ay)	1	F	s 0 0 1	1 Ay	2	M	7 + 3n	REGLIST → – (Ay)				
										NOTE A minimum of 2 registers must be specified and may be specified in any order				

INSTRUCTION MNEMONIC	ATTR SIZE	OPERAND ASSEMBLER SYNTAX Source = Src Destination = Dst Src , Dst	OPERATION WORD				WORDS	EXTENSION	CYCLES	OPERATION	STATUS FLAGS			
			15 14 13 12	11 10 9 8	7 6 5 4	3 2 1 0					N	Z	V	C
MULS	—	Rx,RPy	5	REGx	B	RGy 0	1	—	21	Rx • RPy(even) → RPy(even), RPy(odd) MSW → even; LSW → odd	—	—	—	—
MULU	—	Rx,RPy	5	REGx	9	RGy 0	1	—	21	Rx • RPy(even) → RPy(even), RPy(odd) MSW → even, LSW → odd	—	—	—	—
NEG NEGC	B [W]	Ry	3	7	s c1 1 0	REGy	1	—	3	2's Complement	*	*	*	*
	.B [W]	(Ay)	3	7	s c1 1 1	0 Ay	1	—	9	NEG 0 Dst → Dst				
	.B [W]	d16(Ay)	3	7	s c1 1 1	1 Ay	2	d	12	NEGC 0 Dst C → Dst				
	.B [W]	Addr	3	7	s c1 1 1	F	2	a	12					
NOP	—	—	0	0	0	0	1	—	3	PC + 2 → PC	—	—	—	—
NOT	.B [W]	Ry	3	7	s 0 0 0	REGy	1	—	3	1's Complement	*	*	0	0
	B [W]	(Ay)	3	7	s 0 0 1	0 Ay	1	—	9	NOT(Dst) → Dst				
	B [W]	d16(Ay)	3	7	s 0 0 1	1 Ay	2	d	12					
	B [W]	Addr	3	7	s 0 0 1	F	2	a	12					
OR		see AND (page 9)												

INSTRUCTION MNEMONIC	ATTR SIZE	OPERAND ASSEMBLER SYNTAX Source - Src Destination - Dst Src , Dst	OPERATION WORD 15 14 13 12	 11 10 9 8	 7 6 5 4	 3 2 1 0	WORDS	EXTENSION	CYCLES	OPERATION	STATUS FLAGS N	 Z	 V	 C
POP	—	Rx	1	REGx	B	6	1	—	6	(SP) · · Rx				
	—	Px	0	PORTx	E	6	1	—	9	(SP) · · Px				
	—	(Ay) +	1	E	9	0 Ay	1	—	9	(SP) · · (Ay) ·				
POPM	—	REGLIST	1	7	9	6	2	M	9 · 3n	(SP) · · REGLIST NOTE A minimum of 2 registers must be specified and may be specified in any order				
PUSH	—	Rx	1	REGx	A	E	1	·	6	Rx · (SP)				
	—	Px	0	PORTx	E	E	1		9	Px · (SP)				
	—	(Ay)	1	1 Ax	9	E	1	·	9	(Ax) · (SP)				
	—	#n16	1	E	9	F	2	#	9	#· data · · (SP)				
PUSHM	—	REGLIST	1	F	9	E	2	M	7 · 3n	REGLIST · (SP) NOTE A minimum of 2 registers must be specified and may be specified in any order				
RET	—	—	5	6	D	7	1	—	9	(SP) · · PC	—	—	—	—
	—	Ax	5	0 Ax	D	7	1	—	9	((Ax)) · · PC				
RETI	—	—	5	F	D	7	1	—	12	(SP) · · SR (SP) · · PC	*	*	*	*

INSTRUCTION MNEMONIC	ATTR SIZE	OPERAND ASSEMBLER SYNTAX Source=Src Destination=Dst Src . Dst	OPERATION WORD 15 14 13 12	 11 10 9 8	 7 6 5 4	 3 2 1 0	WORDS	EXTENSION	CYCLES	OPERATION	STATUS FLAGS N	 Z	 V	 C
ROL ROLC ROR RORC		see ASL												
STOP	—	—	1	7	9	F	1	—	—	Pause: CPU & I/O CLOCKS OFF	—	—	—	—
SUB SUBC		see ADD												
TEST TESTN	.B [.W]	#n16.Ry	7	7	s c1 0 0	REGy	2	#	6	TEST: Src .AND. Dst	*	*	0	0
	.B [.W]	#n16.Py	7	F	s c1 0 1	PORTy	2	#	9	TESTN: Src .AND. NOT.(Dst)				
	.B [.W]	#n16.(Ay)	7	7	s c1 0 1	0 Ay	2	#	9					
UNLK	—	Ax	5	1 Ax	9	5	1	—	6	Ax → SP; (SP) + → Ax	—	—	—	—

The following symbols are used to describe the state of the Status Register flags:
- * Set according to result of operation.
- 0 Cleared.
- 1 Set.
- — Not affected.
- U Undefined.

Port Map

Port		Addr	Description	15	14	13	12	11	10	9	8
P0	PH0 PL0	$FC00	16 External I/O or Addr/Data Bus In Expanded Bus Mode								
P1	PH1 PL1	$FC02	16 Ext I/O, Ext Interrupts, Serial Port I/O, Bus Control	LB LDS	HB UDS	R/W	DS BGACK	DTACK	AS	BUSIN BG	BUSOUT BR
P2	PH2 PL2	$FC04	Reserved	◄— RESERVED —►							
P3	PH3 PL3	$FC06	Serial Receive and Serial Transmit Buffers	RECEIVE BUFFER							
P4		$FC08	8 External I/O or Timer and Port 0 Handshake	TAO	TBO	TAI	TBI	STRH R/G	STRL	RDYH	RDYL
P5		$FC0A	Reserved	◄— RESERVED —►							
P6		$FC0C	Reserved	◄— RESERVED —►							
P7		$FC0E	Interrupt Latch Register	RES	NMI	SPARE	XI2I	STRLI	TAOI	TAII	STRHI
P8		$FC10	Interrupt Mask Register	RESERVED		SPARE	XI2M	STRLM	TAOM	TAIM	STRHM
P9		$FC12	Serial I/O Receive Control and Status Register	RE	IS	RW1	RW0	RC	SIS	RESERVED	
P10		$FC14	Serial I/O Transmit Control and Status Register	TE	AT	LM	TW1	TW0	TC	P/S	RES
P11		$FC16	Timer B Latch								
P12		$FC18	Timer A Low Latch								
P13		$FC1A	Timer A High Latch								
P14		$FC1C	Timer and Handshake Control Register	HSM2	HSM1	HSM0	XI2C	HSE	TCE	TCOC	TAM1
P15		$FC1E	Expanded Bus Control and Status Register	SEG1	SEG0	BLCK	F/S	EXP	F/P	TEST 1*	TEST 0*
P16		$FC20	Port 0 Direction Control (DDR0)								
P17		$FC22	Port 1 Direction Control (DDR1)								
P18		$FC24	Serial I/O Mode and Sync Register	A/S	WL1	WL0	ST	PAR1	PAR0	TCO	WS
P19		$FC26	Timer C Latch								

***NOTE:** Both test bits should remain a 0 during normal operation.

Port Map

7	6	5	4	3	2	1	0	Initial Condition
								—
I/O	XI2	XI1	XI0	SI	RCLK	TCLK	SO	—
RESERVED								—
TRANSMIT BUFFER								—
RESERVED								High Byte 00----00
RESERVED								—
RESERVED								—
RSCI	RNI	XI1I	TBOI	TBII	XI0I	XMTI	TCI	$0000
RSCM	RNM	XI1M	TBOM	TBIM	XI0M	XMTM	TCM	$0000
BF	OE	PE	FE	SF/AF	BRK	RESERVED		$0000
BE	UE	END	RESERVED					$00A0
								—
								—
								—
TAM0	TAE	TAIC	TAOC	TBM1	TBM0	TBE	TBIC	S0000
UPC/GP	R/G	RESERVED						Mode pin: VCC-S0400 CLKOUT-S0C00 GND-S0800
								S0000
								S0000
SYNC7	SYNC6	SYNC5	SYNC4	SYNC3	SYNC2	SYNC1	SYNC0	S0000
								—

NOTE:
When a reserved bit is read, it is read as a zero.

ELECTRICAL SPECIFICATIONS

ABSOLUTE MAXIMUM RATINGS

Voltage on any pin relative to GND . . . . . −1.0 V to +7.0 V
Total Device Power Dissipation . . . . . 1 Watt
Ambient Storage Temperature . . . . . −65°C to +150°C
Ambient Operating Temperature ($T_L < T_A < T_H$) . . . . .
Commercial MK68HC2xx/xxx-Cx . . . . . −0°C to +70°C
Industrial MK68HC2xx/xxx-Vx . . . . . −40°C to +85°C
Military MK68HC2xx/xxx-Mx . . . . . −55°C to +125°C

Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other condition above those indicated in the operation sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

MK68200 DC ELECTRICAL CHARACTERISTICS

(V_{CC} = 5.0 V ± 10%, $T_A = T_L$ to T_H)

SYMBOL	PARAMETER	MIN	MAX	UNITS	TEST CONDITIONS
V_{IL}	Input low voltage; all inputs	−0.3	0.8	V	
V_{IH}	Input high voltage; all inputs except CLK1	2.0	V_{CC} + 0.3	V	
V_{IH}	Input high voltage; CLK1	0.7 × V_{CC}	V_{CC} + 0.3	V	
V_{OL}	Output low voltage; all outputs		0.4	V	I_{OL} = 2.0 mA
V_{OH}	Output high voltage; all outputs	2.4		V	I_{OH} = −250μA
V_{OH}	Output high voltage; all outputs	V_{CC} −0.1		V	I_{OH} = −10μA
I_{LI}	Input leakage current		±1	υA	V_{IN} = 0 to V_{CC}
I_{LO}	Three-state output leakage current in float		±10	μA	V_{OUT} = 0.4 V to V_{CC}
I_{CC1}	Input power supply current CLKOUT = 12.5 MHz CLKOUT = 10 MHz CLKOUT = 8 MHz CLKOUT = 0.5 MHz		 75 65 55 15	 mA mA mA mA	Outputs Open
I_{CC2}	Power supply current in IDLE mode. CLKOUT = 12.5 MHz CLKOUT = 10 MHz CLKOUT = 8 MHz CLKOUT = 0.5 MHz		 20 20 20 10	 mA mA mA mA	Outputs Open. All I/O functions active.
I_{CC3}	Power supply current in STOP mode		5	mA	Outputs Open

CAPACITANCE

T_A = 25°C

SYMBOL	PARAMETER	MAX	UNIT	TEST CONDITION
C_{IN}	Input Capacitance	10	pF	Unmeasured pins returned to ground
C_{OUT}	Three-state Output Capacitance	10	pF	

MK68HC200 AC ELECTRICAL SPECIFICATIONS

$T_A = T_L$ to T_H, $V_{CC} = +5V \pm 10\%$ unless otherwise specified.
AC measurements are referenced from minimum V_{IH} or maximum V_{IL} for inputs and from minimum V_{OH} or maximum V_{OL} for outputs.

		8 MHz		10 MHz		12.5 MHz			
NO.	DESCRIPTION	MIN	MAX	MIN	MAX	MIN	MAX	UNITS	NOTES
1	$\overline{\text{RESET}}$ low time	20		20		20		state times	1
2	CLK 1 width high (external clock input)	22		18		14		ns	
3	CLK 1 width low (external clock input)	22		18		14		ns	
4	CLK 1 period (external clock input)	62	1000	50	1000	40	1000	ns	
5	Crystal input frequency	1.000	16.000	1.000	20.000	1.000	25.000	MHz	
6	Clock Period (PHI 1) (tc)	125		100		80		ns	
7	PHI 1 low to PHI 1 high	62		50		40		ns	
8	PHI 1 high to PHI 1 low	62		50		40		ns	
9	PHI 1 low to CLKOUT low		20		16		13	ns	
10	PHI 1 high to CLKOUT high		20		16		13	ns	

MK68HC200 EXPANDED BUS AC ELECTRICAL SPECIFICATIONS (UPC, GP, AND PRIVATE BUSES)

		8 MHz		10 MHz		12.5 MHz			
NO.	DESCRIPTION	MIN	MAX	MIN	MAX	MIN	MAX	UNITS	NOTES
11	PHI 1 low to R/$\overline{\text{W}}$, $\overline{\text{HB}}$, $\overline{\text{LB}}$, FC1, FC2, $\overline{\text{EXPMC}}$, $\overline{\text{DMODE}}$ Valid		58		46		37	ns	2
12	PHI 1 high to $\overline{\text{AS}}$ low		58		46		37	ns	2
13	PHI 1 low to address valid		60		50		40	ns	2
14	$\overline{\text{AS}}$ low to address invalid	35		30		24		ns	2
15	PHI 1 low to tri-state address		45		36		29	ns	2
16	Tri-state address to $\overline{\text{DS}}$, $\overline{\text{LDS}}$, or $\overline{\text{UDS}}$ starting low (fast cycle)	10		10		10		ns	2
17	PHI 1 low to $\overline{\text{DS}}$, $\overline{\text{LDS}}$, or $\overline{\text{UDS}}$ low (fast cycle)		83		66		53	ns	2
18	PHI 1 low to data out valid during write		60		50		40	ns	2
19	PHI 1 low to R/$\overline{\text{W}}$, $\overline{\text{HB}}$, $\overline{\text{LB}}$, FC1, FC2, $\overline{\text{EXPMC}}$, $\overline{\text{DMODE}}$ invalid	0		0		0		ns	2
20	PHI 1 low to address/data bus driven	0		0		0		ns	2
21	$\overline{\text{AS}}$ low to $\overline{\text{DS}}$, $\overline{\text{LDS}}$, or $\overline{\text{UDS}}$ starting low (fast cycle)	50	113	42	90	33	72	ns	2
85	$\overline{\text{AS}}$ high to FPRIV invalid	0		0		0		ns	
86	Address valid to FPRIV valid (fast cycle)		121		94		75	ns	
87	Valid data setup to $\overline{\text{DS}}$, $\overline{\text{UDS}}$ or $\overline{\text{LDS}}$ high during write (fast cycle)	65		50		40		ns	2

MK68HC200 EXPANDED BUS AC ELECTRICAL SPECIFICATIONS (UPC AND GP BUSES)

		8 MHz		10 MHz		12.5 MHz			
NO.	DESCRIPTION	MIN	MAX	MIN	MAX	MIN	MAX	UNITS	NOTES
22	Tri-state address to DS, LDS, or UDS starting low (standard cycle)	67		54		43		ns	
23	PHI 1 high to DS, LDS, or UDS low (standard cycle)		83		66		53	ns	2
24	Valid Data Setup to PHI 1 low	5		5		5		ns	2
25	AS low to DS, LDS, or UDS starting low (standard cycle)	112	175	90	138	72	111	ns	2
26	R/W, HB, or LB valid to AS starting low	45		36		29		ns	
27	Address valid to AS starting low	45		36		29		ns	
28	Input data hold time from PHI 1 low	22		18		14		ns	
29	Input data hold time from DS, LDS, or UDS high	0		0		0		ns	
30	PHI 1 low to DS, LDS, or UDS high		90		72		58	ns	
31	DTACK low or FPRIV valid setup to PHI 1 high	7		6		5		ns	
32	LDS, UDS, or DS high to DTACK high (hold time)	0		0		0		ns	
33	LDS, UDS, or DS pulse width	120		90		72		ns	
34	PHI 1 high to AS high		45		36		29	ns	
35	PHI 1 low to data out invalid	0		0		0		ns	
36	AS inactive	117		90		72		ns	
37	DS, LDS, or UDS high to data out invalid	90		66		53		ns	
38	DS, LDS, or UDS high to AS high	5		5		5		ns	
88	Address valid to FPRIV valid (standard cycle)		246		194		155	ns	
89	Valid data setup to DS, LDS, or UDS during write (standard cycle)	190		150		120		ns	2
90	DS, LDS or UDS low to data in valid (fast cycle)		37		29		22	ns	2
91	DS, LDS or UDS low to data in valid (standard cycle)		100		79		62	ns	2
92	Address valid to data in valid (fast cycle)		185		145		115	ns	
93	Address valid to data in valid (standard cycle)		310		245		195	ns	

MK68HC200 EXPANDED BUS AC ELECTRICAL SPECIFICATIONS (UPC BUS)

		8 MHz		10 MHz		12.5 MHz			
NO.	DESCRIPTION	MIN	MAX	MIN	MAX	MIN	MAX	UNITS	NOTES
39	BGACK low to BR high	100	225	75	180	60	144	ns	
40	BG low to BGACK low	50	300	50	240	40	192	ns	
41	BGACK, AS, DTACK, inactive to BGACK low; BG already low	0	300	0	240	0	192	ns	
42	BGACK low to AS, UDS, LDS, R/W or address/data bus driven	40	68	30	54	25	43	ns	
43	AS, LDS, UDS, R/W or address/data bus tri-state to BGACK high	0	90	0	72	0	58	ns	

MK68HC200 EXPANDED BUS AC ELECTRICAL SPECIFICATIONS (GP BUS)

		8 MHz		10 MHz		12.5 MHz			
NO.	**DESCRIPTION**	**MIN**	**MAX**	**MIN**	**MAX**	**MIN**	**MAX**	**UNITS**	**NOTES**
45	$\overline{\text{BUSIN}}$ low to $\overline{\text{BUSOUT}}$ low (bus grantor, fast cycle, no wait states)		950		720		576	ns	
46	$\overline{\text{BUSOUT}}$ high to $\overline{\text{AS}}$, $\overline{\text{DS}}$, R/$\overline{\text{W}}$, $\overline{\text{LB}}$, $\overline{\text{HB}}$ or address/data bus driven (bus grantor)	15		15		15		ns	
47	$\overline{\text{BUSIN}}$ high to $\overline{\text{BUSOUT}}$ high (bus grantor)	260	450	180	360	144	288	ns	
48	Tri-state $\overline{\text{AS}}$, $\overline{\text{DS}}$, R/$\overline{\text{W}}$, $\overline{\text{LB}}$, $\overline{\text{HB}}$ or address /data bus to $\overline{\text{BUSOUT}}$ low (bus grantor)	50		42		33		ns	
50	$\overline{\text{BUSOUT}}$ low to $\overline{\text{AS}}$, $\overline{\text{DS}}$, R/$\overline{\text{W}}$, $\overline{\text{LB}}$, $\overline{\text{HB}}$ or address/data bus driven (bus requestor, $\overline{\text{BUSIN}}$ low)	120		90		72		ns	
51	$\overline{\text{BUSIN}}$ low to $\overline{\text{AS}}$, $\overline{\text{DS}}$, R/$\overline{\text{W}}$, $\overline{\text{LB}}$, $\overline{\text{HB}}$ or address/data bus driven (bus requestor, $\overline{\text{BUSOUT}}$ low)	135	325	108	300	86	240	ns	
53	$\overline{\text{BUSOUT}}$ high to $\overline{\text{BUSIN}}$ high (bus requestor)		275		240		192	ns	
55	Tri-state $\overline{\text{AS}}$, $\overline{\text{DS}}$, R/$\overline{\text{W}}$, $\overline{\text{HB}}$, $\overline{\text{LB}}$ or address/data bus to $\overline{\text{BUSOUT}}$ high (bus requestor)	50		39		31		ns	

MK68HC200 BUS AC ELECTRICAL SPECIFICATIONS (PRIVATE BUS)

		8 MHz		10 MHz		12.5 MHz			
NO.	**DESCRIPTION**	**MIN**	**MAX**	**MIN**	**MAX**	**MIN**	**MAX**	**UNITS**	**NOTES**
56	Valid Data Setup to PHI 1 low	15		12		10		ns	
57	PBR/$\overline{\text{W}}$ valid to $\overline{\text{PBAS}}$ starting low	30		24		19		ns	
58	Address valid to $\overline{\text{PBAS}}$ starting low	27		21		17		ns	
59	Input data hold time from PHI 1 low	0		0		0		ns	
60	Input data hold time from $\overline{\text{PBDS}}$ high	0		0		0		ns	

MK68HC200 EXPANDED BUS AC ELECTRICAL SPECIFICATIONS (PRIVATE BUS) (Cont.)

		8 MHz		10 MHz		12.5 MHz			
NO.	DESCRIPTION	MIN	MAX	MIN	MAX	MIN	MAX	UNITS	NOTES
61	PHI 1 low to $\overline{\text{PBDS}}$ high		80		63		50	ns	
62	$\overline{\text{PBDTACK}}$ low setup to PHI 1 high	10		9		7		ns	
63	$\overline{\text{PBDS}}$ high to $\overline{\text{PBDTACK}}$ high (hold time)	−15		−15		−15		ns	
64	$\overline{\text{PBDS}}$ pulse width	95		75		60		ns	
65	PHI 1 high to $\overline{\text{PBAS}}$ high		58		45		36	ns	
66	PHI 1 low to data out invalid	10		10		10		ns	
67	$\overline{\text{PBAS}}$ inactive	100		81		65		ns	
68	$\overline{\text{PBDS}}$ high to data out invalid	100		81		65		ns	
69	$\overline{\text{PBDS}}$ high to $\overline{\text{PBAS}}$ high	15		15		15		ns	
94	$\overline{\text{PBDS}}$ low to data in valid		27		27		37	ns	
95	Address valid to data in valid		159		159		238	ns	

MK68HC20C INPUT/OUTPUT AC ELECTRICAL CHARACTERISTICS

			8 MHz		10 MHz		12.5 MHz			
NO.	DESCRIPTION		MIN	MAX	MIN	MAX	MIN	MAX	UNITS	NOTES
70	Active and inactive pulse times	For XI2, XI1, STRH, STRL, TAI, TBI, NMI	5		5		5		state times	1
		For XI0	3		3		3			
71	Input data setup to falling edge of STRH, STRL		7		6		5		ns	
72	Input data hold from the falling edge of STRH, STRL		30		24		19		ns	
73	RDYH, RDYL low time		1	3	1	3	1	3	state times	1
74	Delay from STRH, STRL high to RDYH, RDYL low			55		45		36	ns	
75	Delay from data valid to RDYH, RDYL high (output mode)			3		3		3	state times	1
76	Delay from STRH high to data out (bidirectional mode)			45		36		29	ns	
77	Port 0 data hold time from STRH low (bidirectional mode)		15		12		10		ns	
78	Delay to Port 0 float from STRH low (bidirectional mode)			43		33		26	ns	
79	TCLK,RCLK period (asynchronous)	as input	.125	DC	.100	DC	.080	DC	µs	
		as output	.250	DC	.200	DC	.160	DC		
	TCLK,RCLK period (synchronous)		.500	DC	.400	DC	.320	DC		
80	TCLK, RCLK width low or high (asynchronous)	as input	1	DC	1	DC	1	DC	state	1
		as output	2	DC	2	DC	[illegible]	DC	times	
	TCLK, RCLK width low or high (synchronous)	as input	3	DC	3	DC	3	DC	state	1
		as output	4	DC	4	DC	4	DC	times	

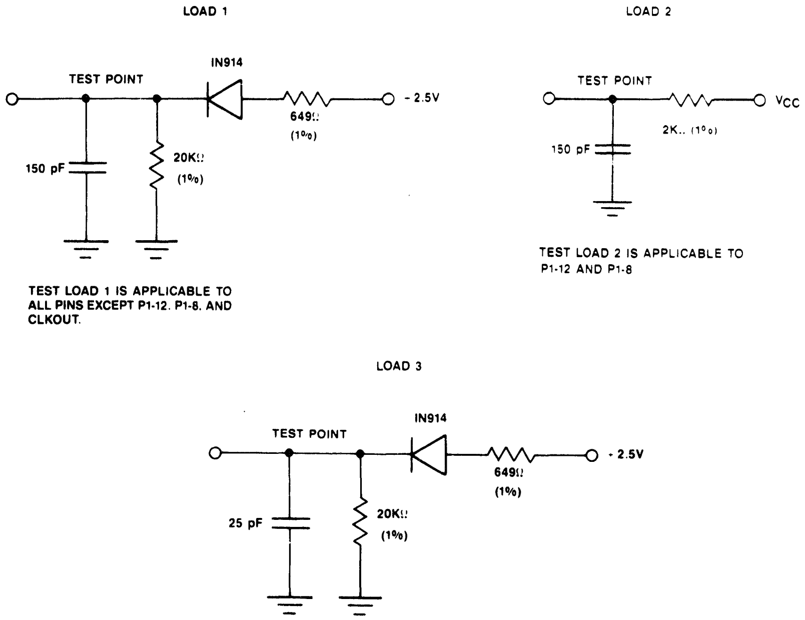
MK68HC200 INPUT/OUTPUT AC ELECTRICAL SPECIFICATIONS

			8 MHz		10 MHz		12.5 MHz			
NO.	DESCRIPTION		MIN	MAX	MIN	MAX	MIN	MAX	UNITS	NOTES
82	TCLK low to SO delay (sync mode)	TCLK as input	165		132		106		ns	
		TCLK as output	37		30		24			
83	SI to RCLK high setup time (sync mode)	RCLK as input	15		12		10		ns	
		RCLK as output	90		72		58			
84	SI hold time from RCLK high (sync mode)	RCLK as input	22		18		14		ns	
		RCLK as output	0		0		0			

NOTES

1 One state time is equal to one-half of the instruction clock (PHI 1) period

2 For the private bus case, the signals referenced apply to the equivalent private bus signals



TEST LOAD 1 IS APPLICABLE TO ALL PINS EXCEPT P1-12, P1-8, AND CLKOUT.

TEST LOAD 2 IS APPLICABLE TO P1-12 AND P1-8

TEST LOAD 3 IS APPLICABLE TO CLKOUT.

Figure 17 . Output Test Load

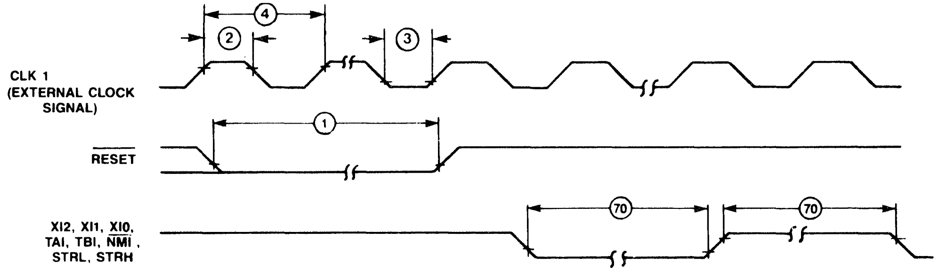

CLK 1 (EXTERNAL CLOCK SIGNAL)
RESET
XI2, XI1, XI0, TAI, TBI, NMI, STRL, STRH
4
2
3
1
70
70

Figure 18. MK68HC200 AC Timing

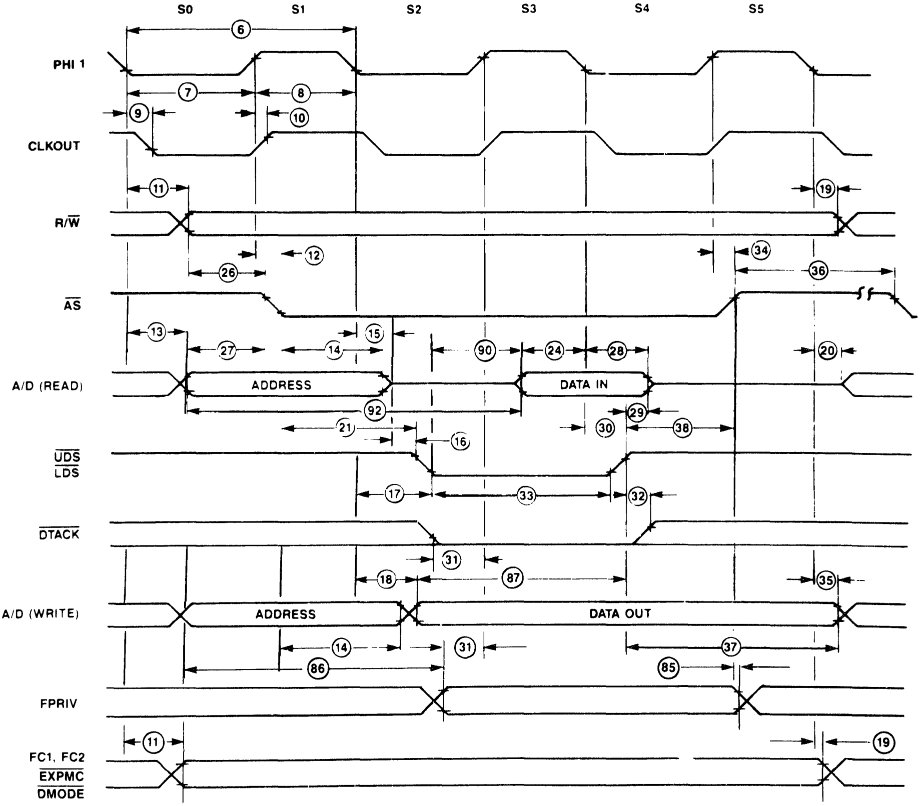


Figure 19. MK68HC201 UPC Bus Timing (Fast Cycle)

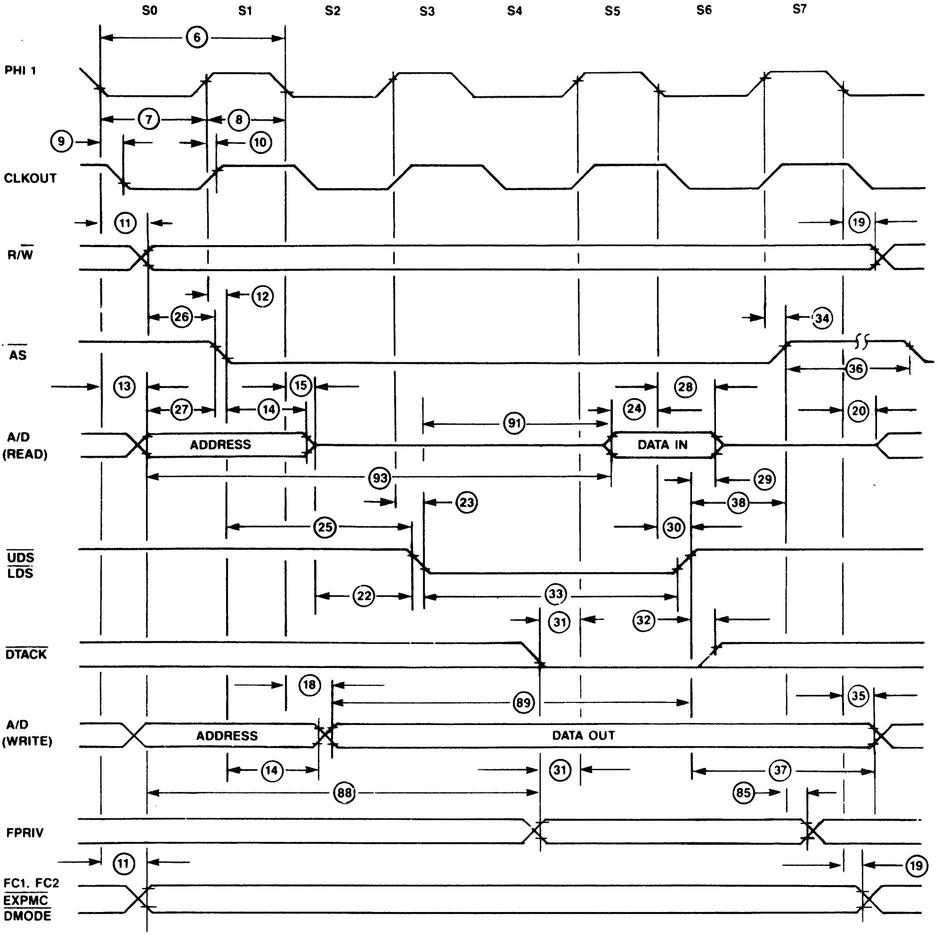


Figure 20. MK68HC201 UPC Bus Timing (Standard Cyυie)

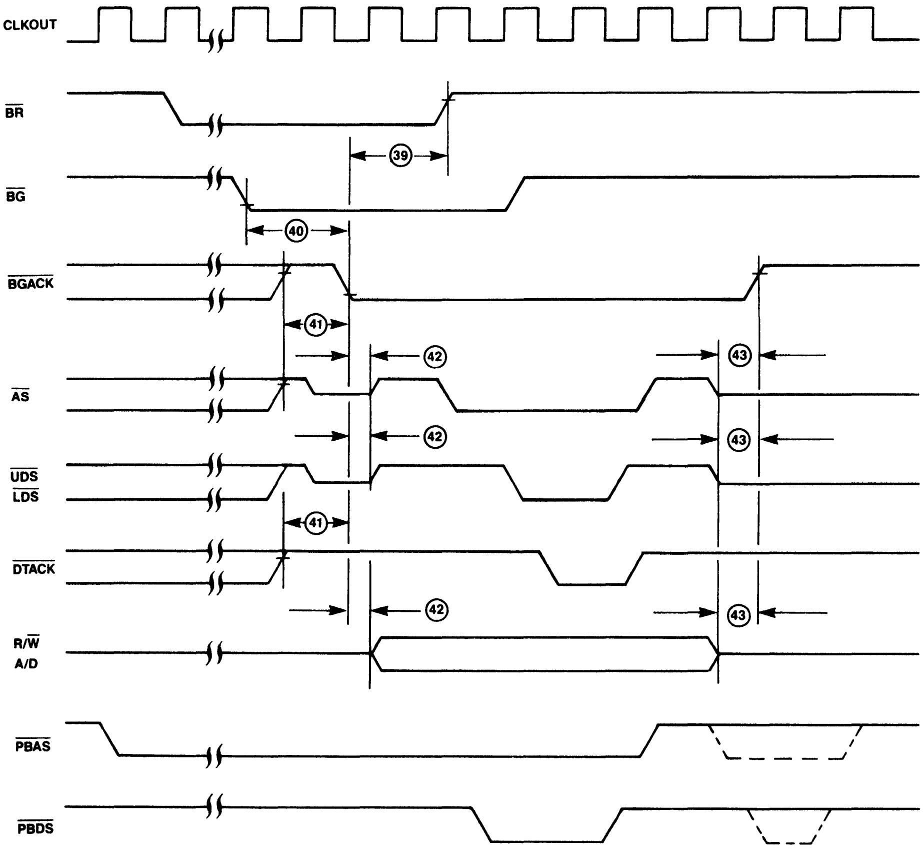


Figure 21. MK68HC201 UPC Bus Arbitration Timing

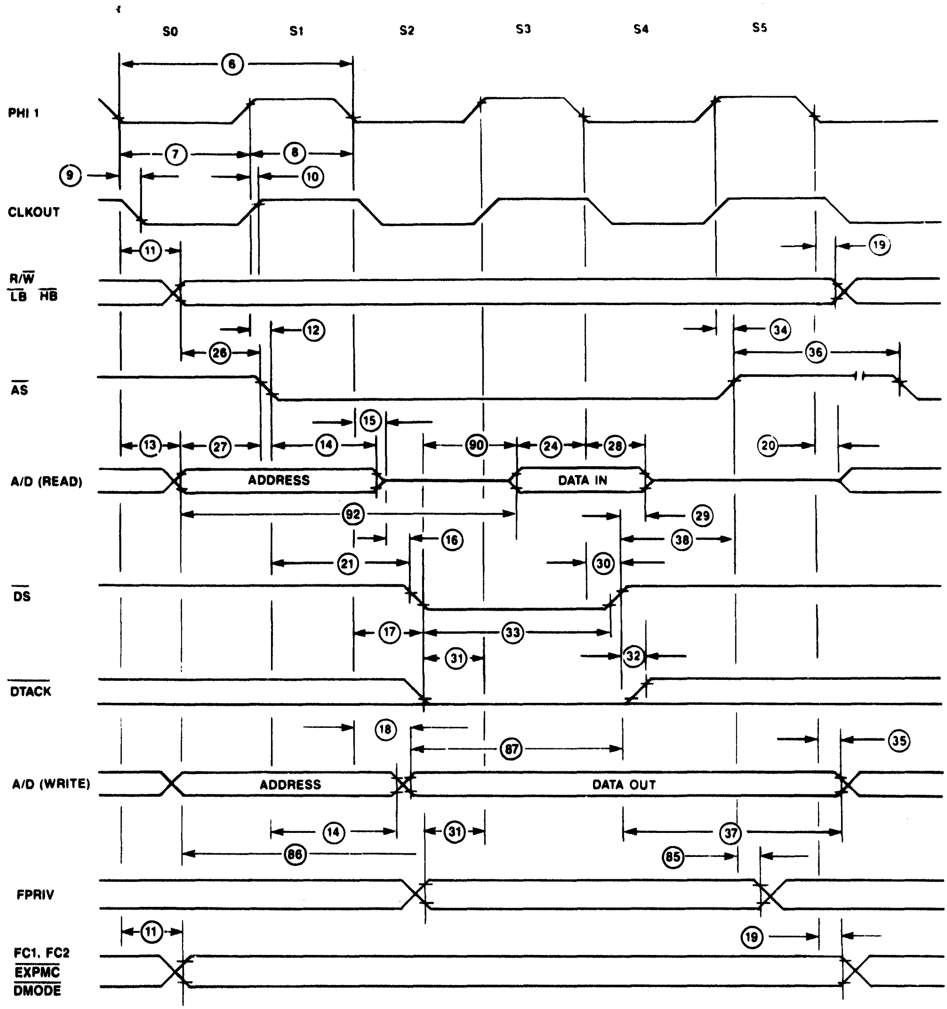


Figure 22. MK68HC211 GP Bus Timing (Fast Cycle)

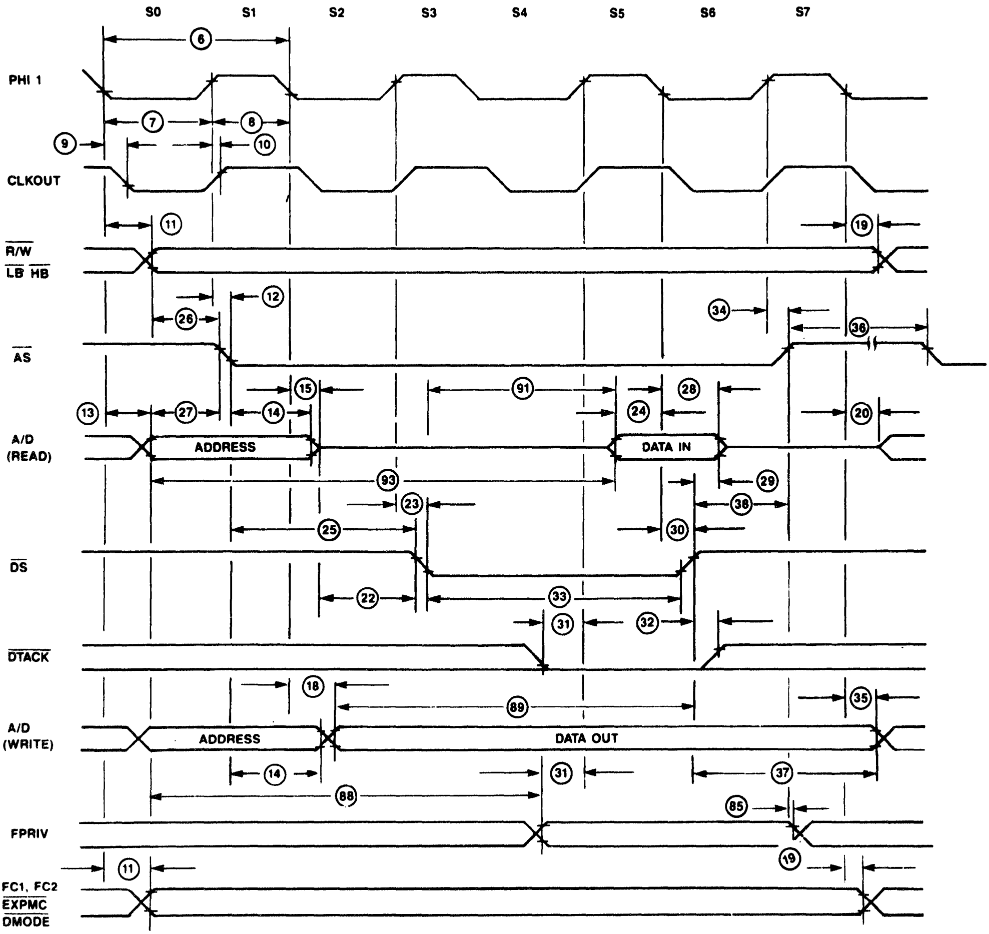


Figure 23 . MK68HC211 GP Bus Timing (Standard Cycle)

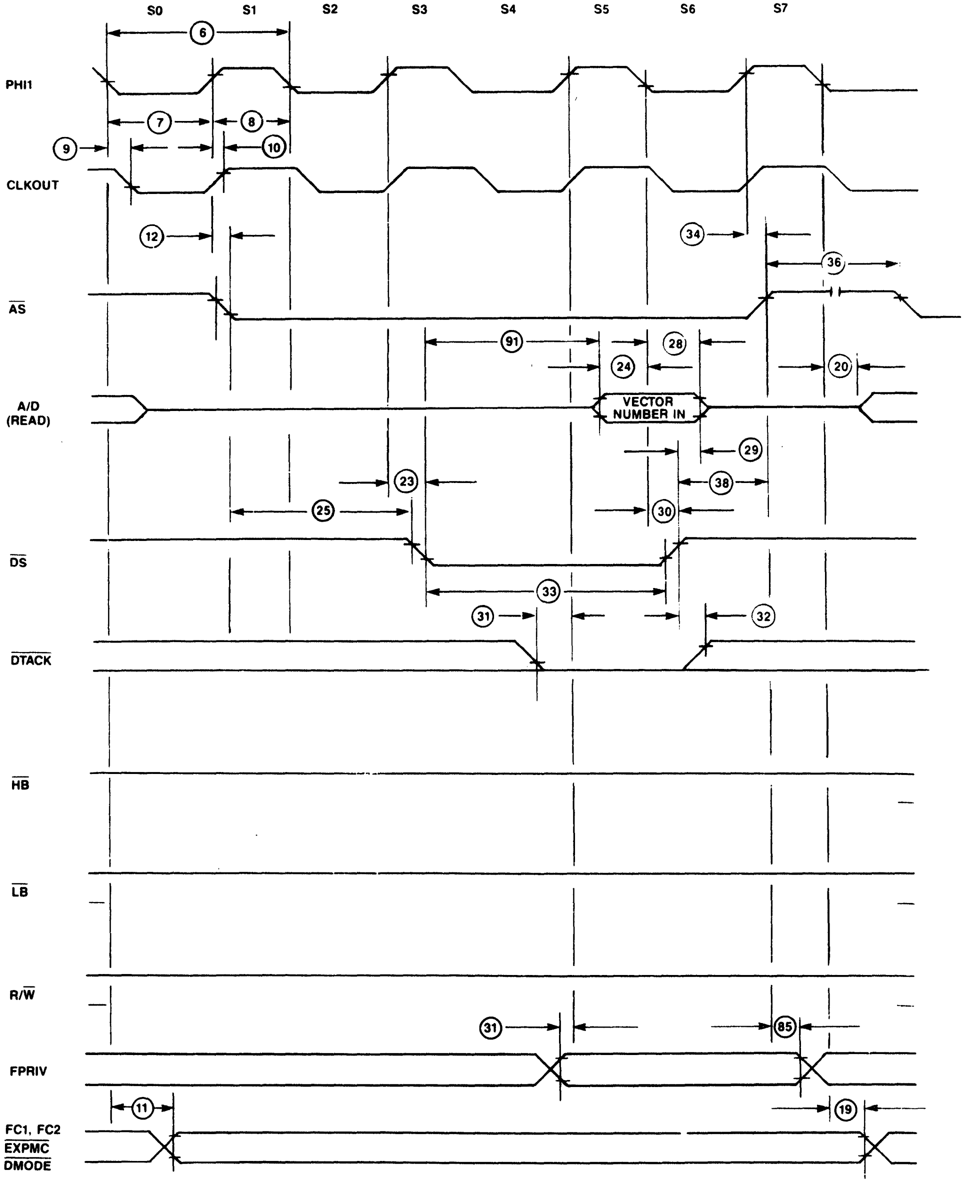


Figure 24. MK68HC211 GP Bus Timing (Interrupt Acknowledge Timing)

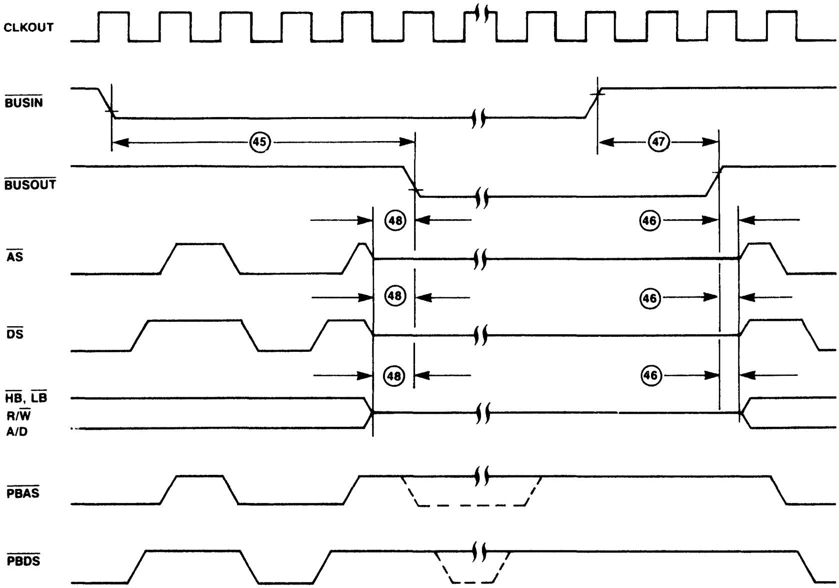


Figure 25. MK68HC211 GP Bus Arbitration Timing (Bus Grantor)

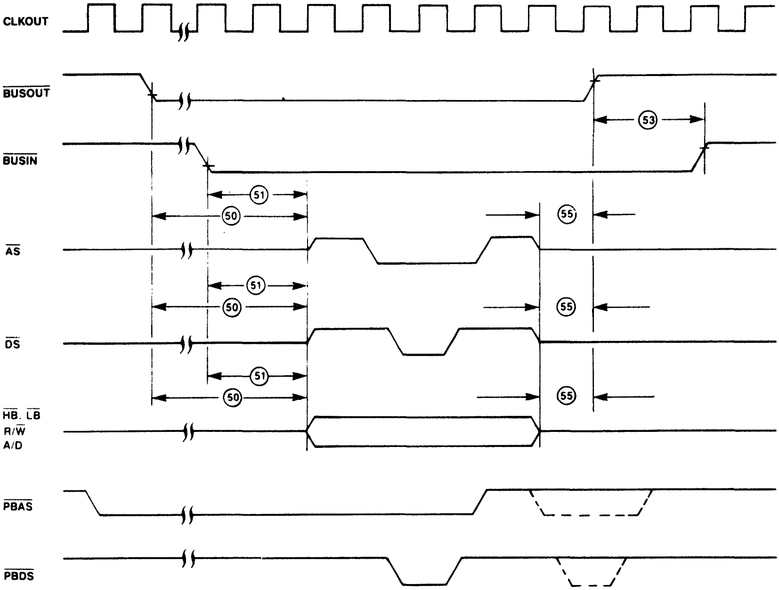


Figure 26. MK68HC211 GP Bus Arbitration Timing (Bus Requestor)

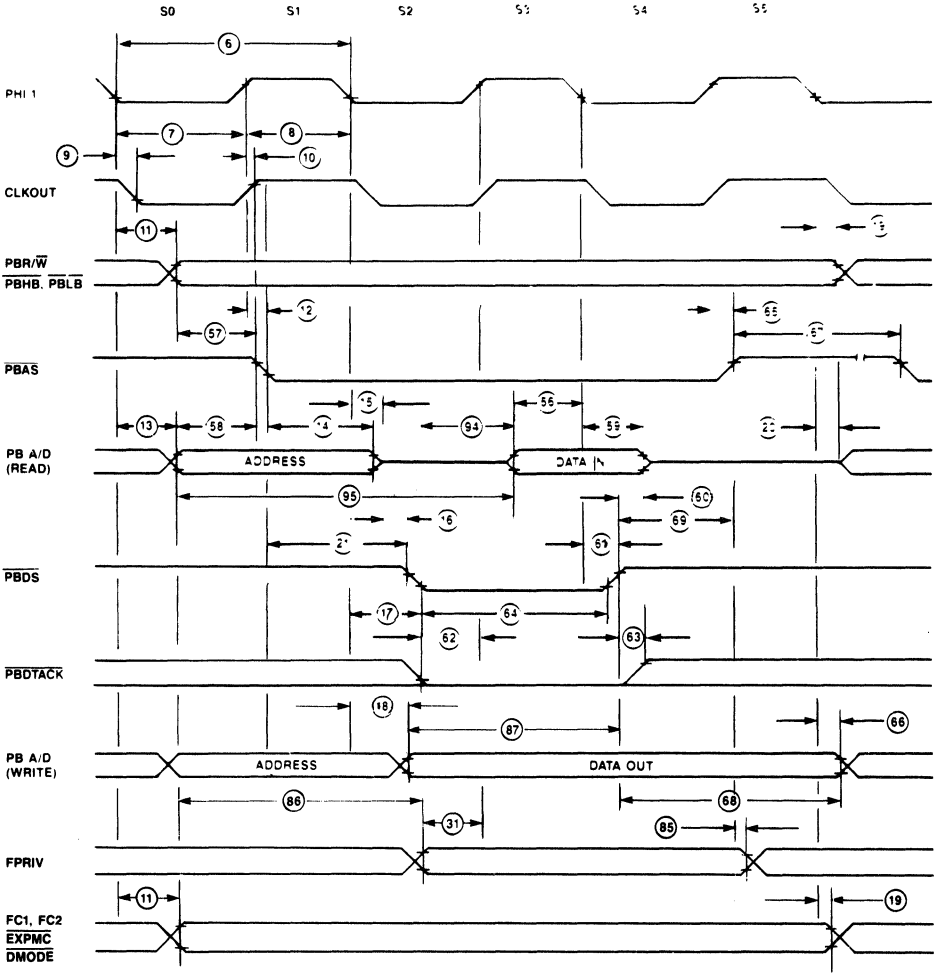


Figure 27. MK68HC221 Private Bus Timing (Fast Cycle)

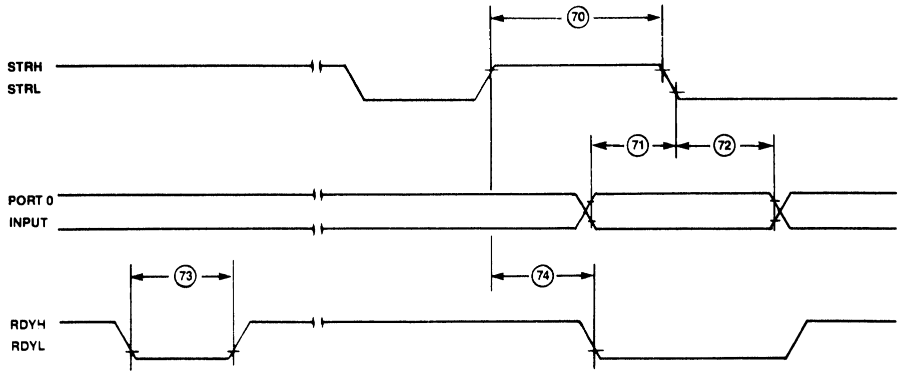


Figure 28. Input/Output AC Timing (Data Input)

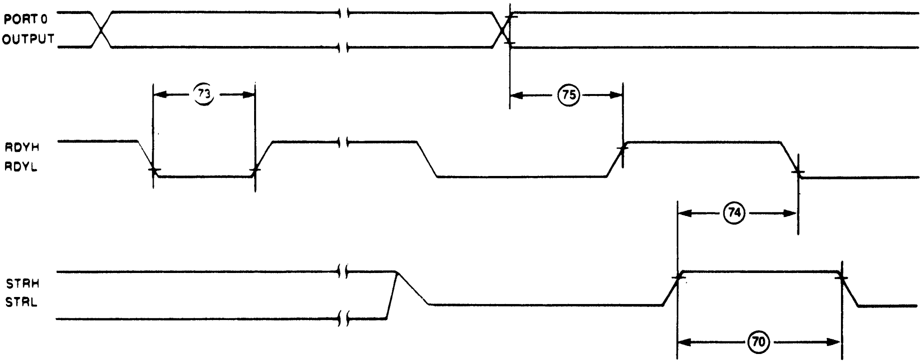


Figure 29. Input/Output AC Timing (Data Output)

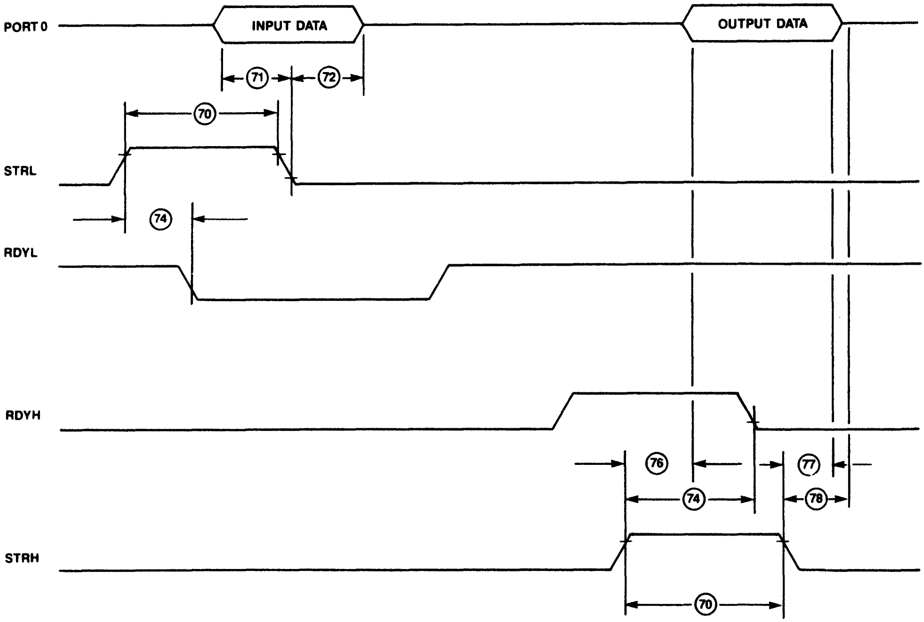


Figure 30 . Input/Output AC Timing (Bidirectional I/O)

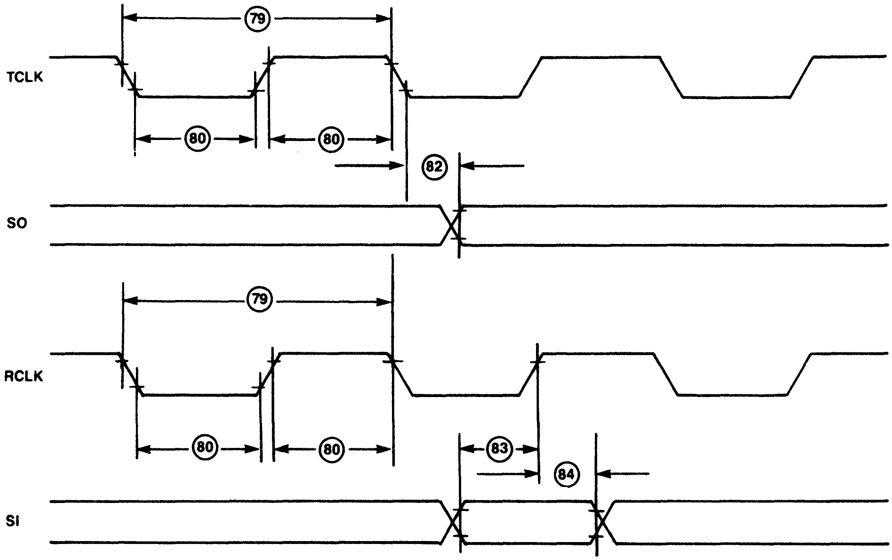
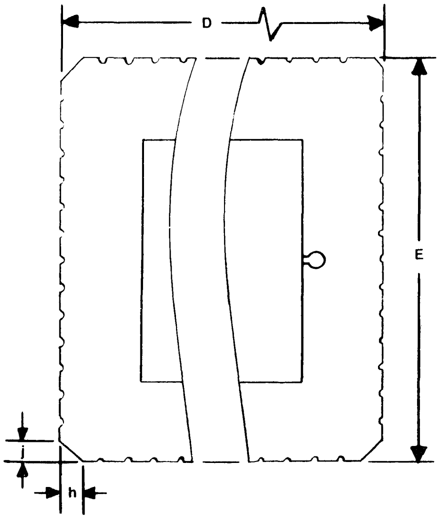


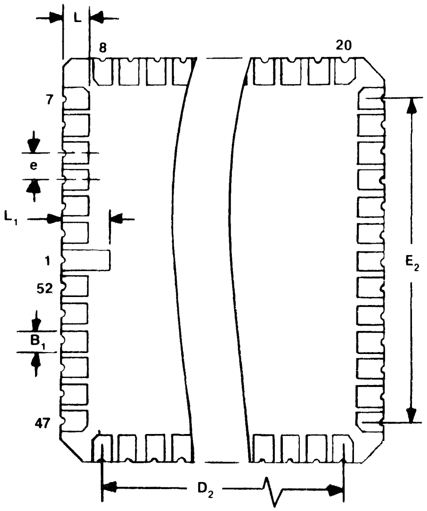
Figure 31. Input/Output AC Timing (Serial I/O)

PHYSICAL DIMENSIONS

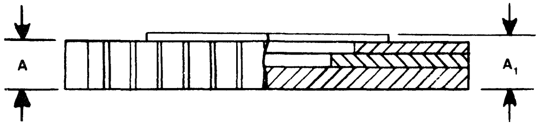
MK68HC200 52-Pin
Ceramic Leadless Chip Carrier (E)



TOP VIEW



BOTTOM VIEW

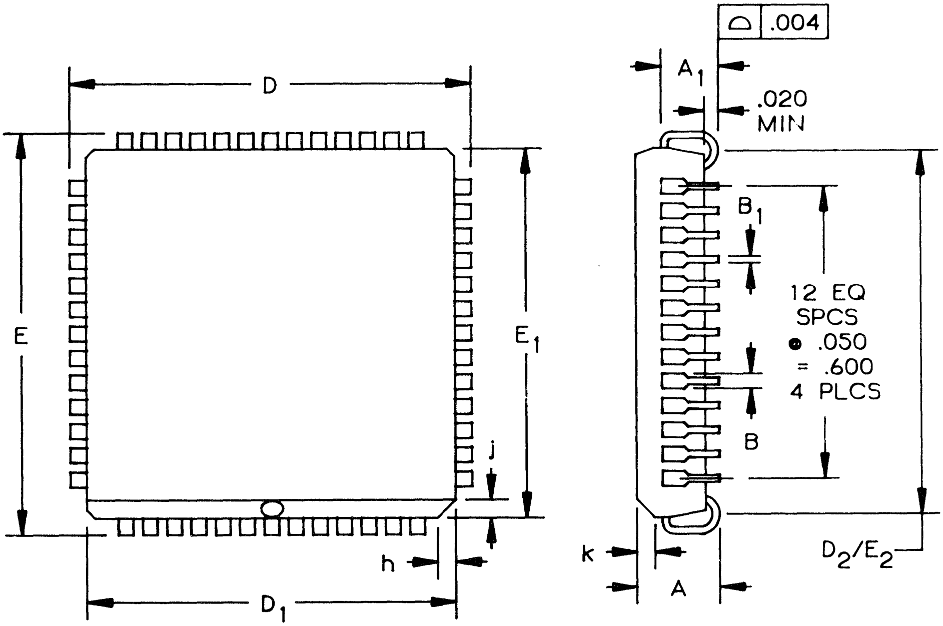


	INCHES		
DIM.	MIN.	MAX.	NOTES
A	.070	.095	
A_1	.080	.110	
B_1	.022	.028	
D	.739	.761	
D_2	.590	.610	
E	.739	.761	
E_2	.590	.610	
e	.048	.052	
h	.035	.045	
j	.035	.045	
L	.045	.055	
L_1	.075	.095	

1 INCH 2.54 CENTIMETERS

NOTES:
1. Body material shall be ceramic.
2. Plating shall be gold over nickel as specified in the detail specification

MK68HC200 52-Pin
Plastic Leaded Chip Carrier (Q)



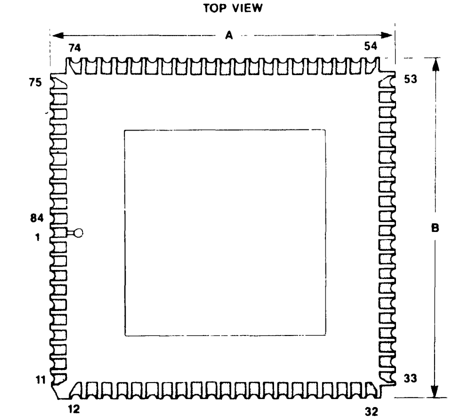
DIM.	INCHES		NOTES
	MIN.	MAX.	
A	.165	.185	2
A_1	.090	.120	2
B	.026	.032	2
B_1	.013	.021	2
D	.785	.795	
D_1	.750	.756	
D_2	.690	.730	
E	.785	.795	
E_1	.750	.756	
E_2	.690	.730	
h	.042	.060	
j	.042	.060	
k	.042	.056	

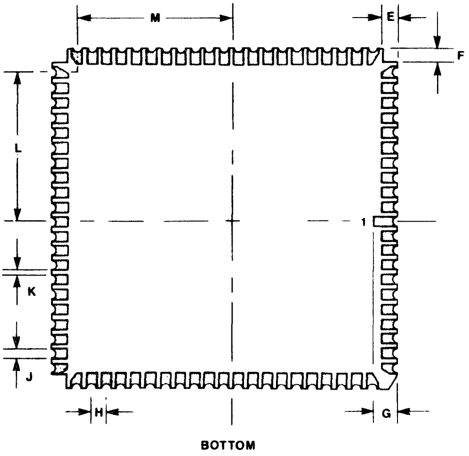
1 INCH = 2.54 CENTIMETERS

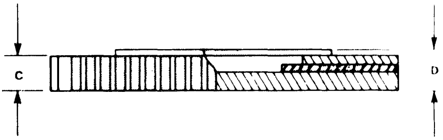
NOTES:

1. LEAD FINISH TO BE SPECIFIED PER CUSTOMER AGREEMENT.
2. WHEN SOLDER DIP LEAD FINISH IS SPECIFIED, THE MAXIMUM LIMIT SHALL BE INCREASED BY .003.

MK68HC221 84-Pin
Ceramic Leadless Chip Carrier (E)



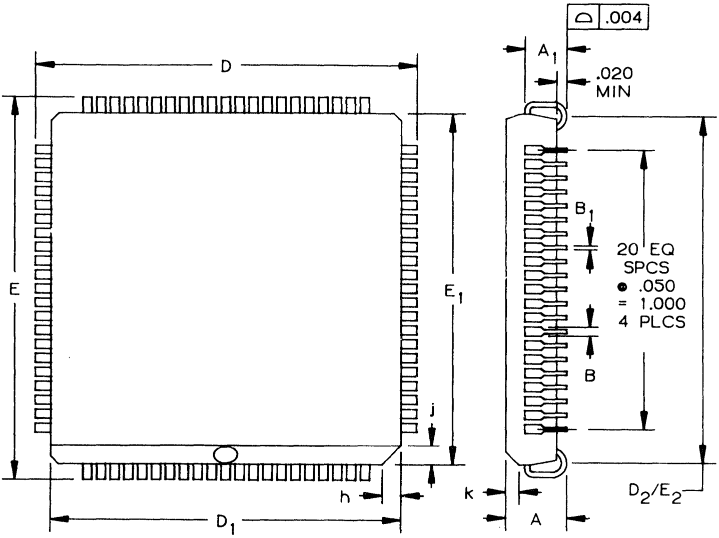




DIM.	INCHES	
	MIN	MAX
A	1.138	1.167
B	1.138	1.167
C	0.070	0.090
D	0.080	0.110
E	0.044	0.056
F	0.044	0.056
G	0.075	0.095
H	0.048	0.052
J	0.033	0.039
K	0.010	0.018
L	0.495	0.505
M	0.495	0.505

1 INCH = 2.54 CENTIMETERS

MK68HC221 84-Pin
Plastic Leaded Chip Carrier (Q)



DIM.	INCHES		NOTES
	MIN.	MAX.	
A	.165	.200	2
A_1	.090	.130	2
B	.026	.032	2
B_1	.013	.021	2
D	1.185	1.195	
D_1	1.150	1.158	
D_2	1.090	1.130	
E	1.185	1.1.95	
E_1	1.150	1.158	
E_2	1.090	1.130	
h	.042	.060	
j	.042	.060	
k	.042	.056	

1 INCH = 2.54 CENTIMETERS

NOTES:

1. LEAD FINISH TO BE SPECIFIED PER CUSTOMER AGREEMENT.
2. WHEN SOLDER DIP LEAD FINISH IS SPECIFIED, THE MAXIMUM LIMIT SHALL BE INCREASED BY .003.

ORDERING INFORMATION

There are two types of part numbers for the MK68HC200 family of devices. The generic part number describes the basic device type, the amount of ROM and RAM, the desired package type, temperature range, power supply tolerance, and expandable bus interface type. The device order number indicates the specific mask set Mostek will use to manufacture the device, along with package type, speed grade and temperature range.

Generic Part Number

An example of the generic part number is shown below:

MK68HC201 / 4 4 N - 0 6

Denotes maximum instruction clock frequency.
8 = 8 MHz
10= 10 MHz
12= 12.5 MHz

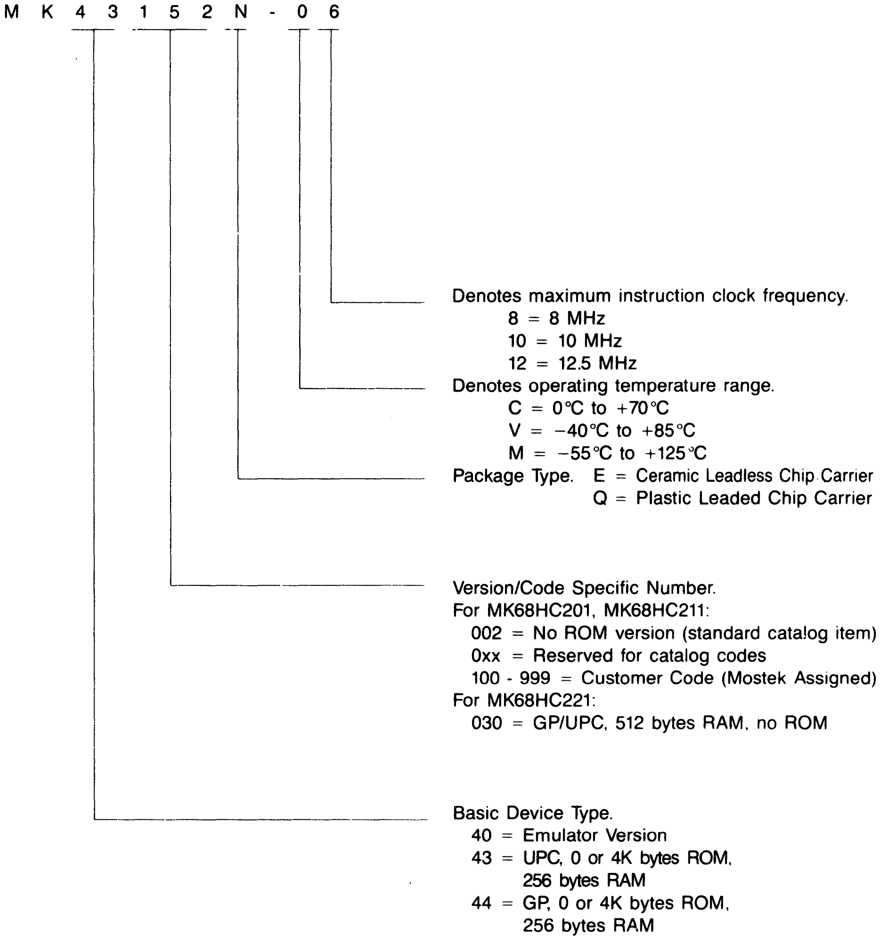
Denotes operating temperature range.
C = 0°C to +70°C
V = −40°C to +85°C
M = −55°C to +125°C

Package Type.	E	=	Leadless Chip Carrier
	Q	=	Plastic Leaded Chip Carrier
RAM Designator.	4	=	256 Bytes[1]
	8	=	512 Bytes[2]
ROM Designator.	0	=	None[3]
	4	=	4K Bytes
Basic Device Type.	68HC201	=	UPC Bus
	68HC211	=	GP Bus
	68HC221	=	Emulator with UPC/$\overline{\text{GP}}$ input pin

NOTES
[1] Must be "4" when specifying the ROM version
[2] Must be "8" when specifying the emulator version
[3] Must be "0" when specifying the emulator version

Device Order Number

An example of the device order number is shown below:



M K 4 3 1 5 2 N - 0 6

Denotes maximum instruction clock frequency.
8 = 8 MHz
10 = 10 MHz
12 = 12.5 MHz

Denotes operating temperature range.
C = 0°C to +70°C
V = −40°C to +85°C
M = −55°C to +125°C

Package Type. E = Ceramic Leadless Chip Carrier
Q = Plastic Leaded Chip Carrier

Version/Code Specific Number.
For MK68HC201, MK68HC211:
002 = No ROM version (standard catalog item)
0xx = Reserved for catalog codes
100 - 999 = Customer Code (Mostek Assigned)
For MK68HC221:
030 = GP/UPC, 512 bytes RAM, no ROM

Basic Device Type.
40 = Emulator Version
43 = UPC, 0 or 4K bytes ROM,
256 bytes RAM
44 = GP, 0 or 4K bytes ROM,
256 bytes RAM

PART NUMBER EXAMPLES (A noninclusive list)	

MK68HC201/44Q-C10	Device Order Number = MK43XXXQ-C10
	Speed = 10MHz
	Temperature = 0° to 70°C
	Package = 52 pin plastic LCC
	RAM = 256 bytes
	ROM = 4096 bytes
	Bus = UPC

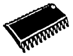
MK68HC211/04Q-C8	Device Order Number = MK44002Q-C8
	Speed = 8MHz
	Temperature = 0° to 70°C
	Package = 52 pin plastic LCC
	RAM = 256 bytes
	ROM = None
	Bus = GP

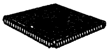
MK68HC221/08E-M12	Device Order Number = MK40030E-M12
	Speed = 12.5MHz
	Temperature = −55°C to +125°C
	Package = 84 pin ceramic LCC
	RAM = 512 bytes
	ROM = None
	Bus = GP/UPC

MK68HC201/44Q-V8	Device Order Number = MK43XXXQ-V8
	Speed = 8MHz
	Temperature = −40 to +85°C
	Package = 52 pin plastic LCC
	RAM = 256 bytes
	ROM = 4096 bytes
	Bus = UPC

These specifications are subjet to change without notice.
Please inquire with our sales offices about the availability of the different products.

Surface Mounted Devices

4-BIT MCUs	SO 24  CB-524	PLCC 28 CB-520	Page
ET9320		●	1-45
ET9321	●	●	1-45
ET9322	●	●	1-45
ET9420		●	1-45
ET9421	●	●	1-45
ET9422	●	●	1-45
ETC9310	●	●	1-5
ETC9320		●	1-5
ETC9321	●	●	1-5
ETC9322	●	●	1-5
ETC9344		●	1-69
ETC9345	●	●	1-69
ETC9410	●	●	1-5
ETC9420		●	1-5
ETC9421	●	●	1-5
ETC9422	●	●	1-5
ETC9444		●	1-69
ETC9445	●	●	1-69
ETL9310	●	●	1-21
ETL9311	●	●	1-21
ETL9320		●	1-21
ETL9321	●	●	1-21
ETL9322	●	●	1-21
ETL9344		●	1-115
ETL9345	●	●	1-115
ETL9410	●	●	1-21
ETL9411	●		1-21
ETL9420		●	1-21
ETL9421	●	●	1-21
ETL9422	●	●	1-21
ETL9444		●	1-115
ETL9445	●	●	1-115

8-BIT MCUs	PLCC 28 CB-520	PLCC 44 CB-521	PLCC 52 	PLCC 84 LCCC 84 	Page
EF6801		●			2-7
EF6801U4		●			2-47
EF6804J2	●				2-91
EF6804P2	●				2-135
EF6805P2	●				2-225
EF6805P6	●				2-249
EF6805R2		●			2-273
EF6805R3		●			2-301
EF6805U2		●			2-329
EF6805U3		●			2-355
EF68HC04P3	●				2-181

16-BIT MCUs

MK68211			●		3-5
MK68E201				●	3-5
MK68E211				●	3-5
MK68E221				●	3-5
MK68HC201			●		3-77
MK68HC211			●		3-77
MK68HC221				●	3-77

SURFACE MOUNTED DEVICES: today's solution for state-of-the-art system designs.

Today's trend toward light weight system designs with high component density allows Surface Mounting Technology to revolutionize manufacturing in the Electronics industry.

Reduction in board assembly cost by as much as 40% and in board size by as much as 50% is a goal that can be reached throught the utilization of Surface Mounted Devices:
Active Semiconductors (SO IC's) - SO Discretes and Chip carriers as well as passive resistors and capacitor chips.

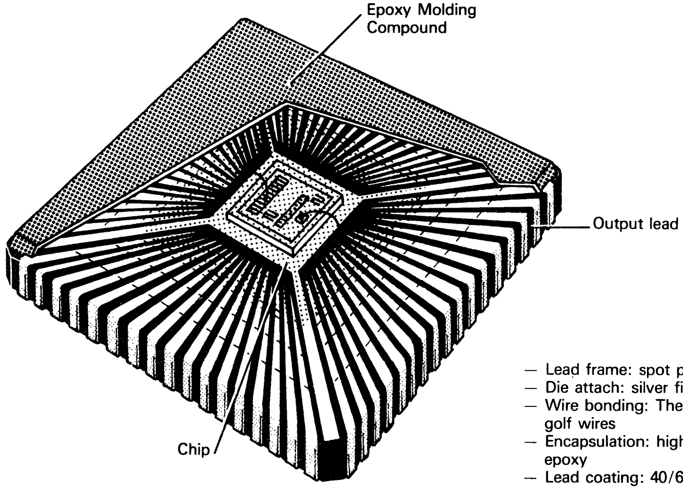
Today system designers can select package outlines that meet state-of-the-art weight/space ratio requirements while enhancing electrical performance.

By 1990, through widely accepted data, one can foresee above 50% of the world wide demand to be Surface Mounted Devices.

Included in the THOMSON SEMICONDUCTEURS family of Surface Mounted Devices are integrated circuits (SO8 - SO28), Plastic Leaded Chip Carriers (PLCC) from 18 to 84 pins, Leadlen Ceramic Chip Carrier's (LCCC) from 20 to 68 pads, as well as discrete diodes and transistors in SOT 23 (TO-236).

Capacitor chips are also available in THOMSON LCC. Compared to conventional types THOMSON SC's SMD have the following features:

- Compact design enabling high packing density and significant reduction in board size and weight, for instance in consumer electronics, telecommunications and automotives.
- Easy and low cost handling through automated high speed pick and place machinery.
- Mounting capability on both sides of all types of substrates (Ceramic or PC boards) using all current methods, such as wave soldering reflow and vapor phase technics.
- Same electrical characteristics (same dice) as conventional packages, with improved high frequency, high speed switching performances due to lower lead inductance and capacitance.
- Optimized way to package VLSI circuits by utilizing plastic chip carrier along with SO packages, leading to a major advantage over chip and wire assembly processes.



- Lead frame: spot plated copper
- Die attach: silver filled compound
- Wire bonding: Thermosonic ball bonding of golf wires
- Encapsulation: high performance non inflammable epoxy
- Lead coating: 40/60 Sn-Pb solder dipping.

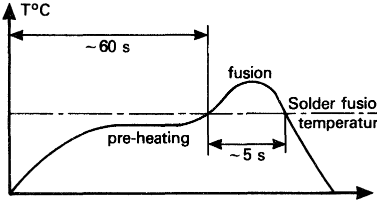
MOUNTING METHODS

PLCC and LCCC may be mounted on either PC boards or substrates, employing different soldering methods.

Reflow process

- This method permits the soldering of all components in a single operation. In this case, solder cream is used either by screen printing or employing a pneumatic gun. After the application of solder cream the components are mounted and the circuit is guided through an infrared or convection oven which allows the solder to melt.

In order to avoid solder spreading risks and to position the components, the board should be pre-screen printed with an insulating material.



- An other way to solder coat the board is to dip it into a soldering bath or through wave soldering. Then, flux is applied by a brush or immersion and the components are placed.

The soldering is done by reflow through an oven as above.
The result obtained by both methods is identical. However, in this case, a complete tinning of the circuit is necessary.

Wave soldering process

The components are glued to substrate by means of an insulated glue. The board is dipped in flux and then goes through the wave soldering process.

Hot air soldering

In this system, the soldering iron must be replaced by a hot air nozzle. Employing this method, an improvement in solder temperature control is achieved.
It is also easy to correct some soldering failures with this method.

Heated collet

This method is used for rework and soldering PCC on the versus side of the board after wave solder.

Vapor phase soldering (VPS)

This method uses the heat of the vapor of a boiling inert fluorinated fluid. Soldering is accomplished by either screening tin layers or by electroplating solder paste.
Components are then mounted in appropriate locations and soldered to the circuit by dipping it into the vapor of boiling inert fluorinated fluid.
The main advantages of this method are:

- Uniform workpiece temperature whatever the difference in size of the components involved
- Accurate temperature without any elaborate control (temperature of boiling inert fluid)
- Dipping time may be very short and as a consequence better soldering results are obtained
- Soldering takes place in an oxygen-free atmosphere which permits:
  — to use less active soldering flux, and
  — to avoid soldering flux oxidation (easy cleaning of remaining flux).

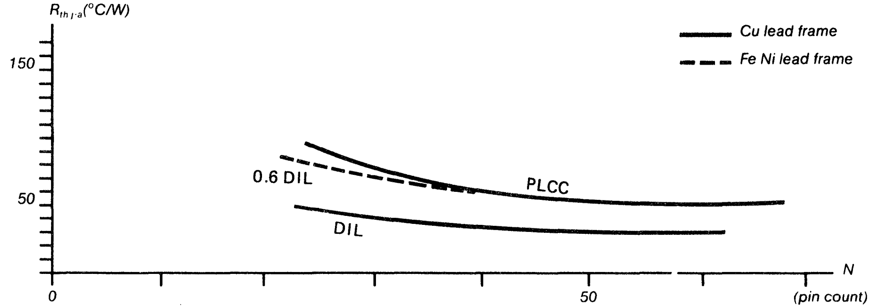
Remarks

Various types of high speed automatic mounting equipments for passive and active devices are available contributing to considerable reduction in production costs.

THERMAL CHARACTERISTICS

Thermal performances of SMD are dependent upon substrate material, size, mounting process, chip area, die attach and lead frame material characteristics.

For PLCC, copper is selected in order to improve the thermal characteristics. Therefore thermal resistance approaching that of a conventional Fe Ni 42 DIL can be obtained.



Junction-ambient air thermal resistance, components reported on classical PC board (FR 4, 1.3 mm thickness, 50 x 50 mm size).

Allowable power dissipation can reach 350 mW according to device type, refer to data sheet.

Development and emulation tool

HOMSON
COMPONENTS
EMICONDUCTEURS

INICE 4-8

DEVELOPMENT AND EMULATION TOOL

INICE4-8 is a support for:

- EF6800 and EF6809 MPU Families
- EF6805, EF6804 and TS68HC04 MCU Families
- ETC 9400 and ETL 9400 MCU Families
- TS94000 MCU Family.

SOFTWARE DEVELOPMENT ON IBM-PC

DEVELOPMENT WITH INICE 4-8

INICE 4-8 is a powerful, lowcost and user friendly stand alone development tool, fully dedicated to support the complete range of 4 and 8 bit MCU's manufactured by THOMSON-SC.

INICE 4-8 offers 2 RS 232 C Serial communication lines:
- one to interface to a host computer IBM-PC (1) which provides functions of human interface and software development.
- the other is used for auxiliary systems such as a line printer. Eprom programmer or centralized mainframe.

The whole development station comprises the host computer IBM-PC, INICE 4-8 and a probe dedicated for each MPU and MCU family.

SOFTWARE DEVELOPMENT ON IBM-PC

The choice of a standard personnal computer for Software purposes (MS DOS operating system) and for hardware (IBM-PC bus) is an important key to obtaining a full evolutive and open system.

It offers access to a very large Software library.

Full screen editor: word, wordstar... (2)
Full range of development software
- MPU and MCU cross assembler
- linking editor.

(1) Adaptation for Vax computer will be available in 1987.
(2) Software available in library for IBM-PC.

INICE 4-8 INTERACTIVE SOFTWARE

Software supplied to control INICE 4-8 offers the ability to use the advantages of the powerful architecture hardware.

- Meaningful mnemonics
- Wide choice for characterising values (Decimal, Hexa, ASCII)
- Aids to the operator when in difficulty
- Full range of debugging aides
  - On line assembler/disassembler
  - 8 hardware breakpoints
  - 8 software breakpoints
  - Logic analyser triggered breakpoints
  - Execution program step by step in RAM and ROM area
  - Echo on line printer
  - Load and save program through RS 232 C connection
  - Memory check
  - Operator keyboard entry request can be chained in a command file
  - Self test of the entire system INICE 4-8 + emulator probe at power up
  - Possibility to temporarily leave control of emulation session to access DOS routines, through the emulation session keeps acting.

IBM-PC: is a registered trade mark of International Business MachineCorp.

TSTMU 94

EMULATION WITH INICE 4-8 (*)

An advanced architecture for total transparency and dynamic emulation.

Thanks to the dual processor architecture (System bus and emulation bus), the emulated MPU or MCU operates independently of the other system processors.

Therefore, it is possible during execution, to list or modify emulator probe control parameters.

REALTIME EMULATION

With the transparency feature, the emulated processor operates at 100% of the maximum speed specified by the manufacturer.

(*) Each emulator probe is delivered with an interactive software specific to each MPU or MCU family.

OPEN ENDED

- Switching from one MPU or MCU to another by changing only the emulator probe
- Program execution without existing final application
- Shared overlay memory (16 Kbytes) in blocks of 256 bytes
- Write protect facility for individual blocks
- 8 hardware and 8 software breakpoints over entire emulation space.

LOGIC ANALYSER

The logic analyser operates with 1024 words of 32 bits (16 address, 8 data bits, 8 available to user).

With sampling and trigger or recognition of maskable words. Five operating modes are available.

USER'S SELECTION GUIDE

Family	MPU or MCU	Development System	Emulator Chip	Emulator Board	INICE Interactive software*	Emulator Probe
EF6800	EF6800	TST/N48			TSR 6800	EFT EMU0
	EF6802	—			—	—
EF6809	EF6809	—			TSR 6809	EFT EMU9
	EF6809E	—			—	—
EF6805	EF6805P2	—	MC68705P3		TSR 6805	EFT MPU5
		—	MC68705P3		—	—
	EF6805U2	—	MC68705U3		—	—
	EF6805R2	—	MC6805R2		—	—
	EF6805U3	—	MC68705U3		—	—
	EF6805R3	—	MC68705R3		—	—
EF6804	EF6804P2	—	MC68704P3		TSR 6804	EFT MUP4
	EF68HC04P3	—		TSEMU04	—	—
ET9400	ETC9410/11	—			TSR 6804	TSTMU94
	ETC9420/21	—			—	—
	ETC9444/45	—			—	—
	ETL9410/11/13	—			—	—
	ET(L)9420/21/22	—			—	—
	ETL9444/45	—			—	—
TS94000	TS94110	—			TSR 94000	TST MU94
	TS94120	—			—	—

* Cross assembler included

EMU-04

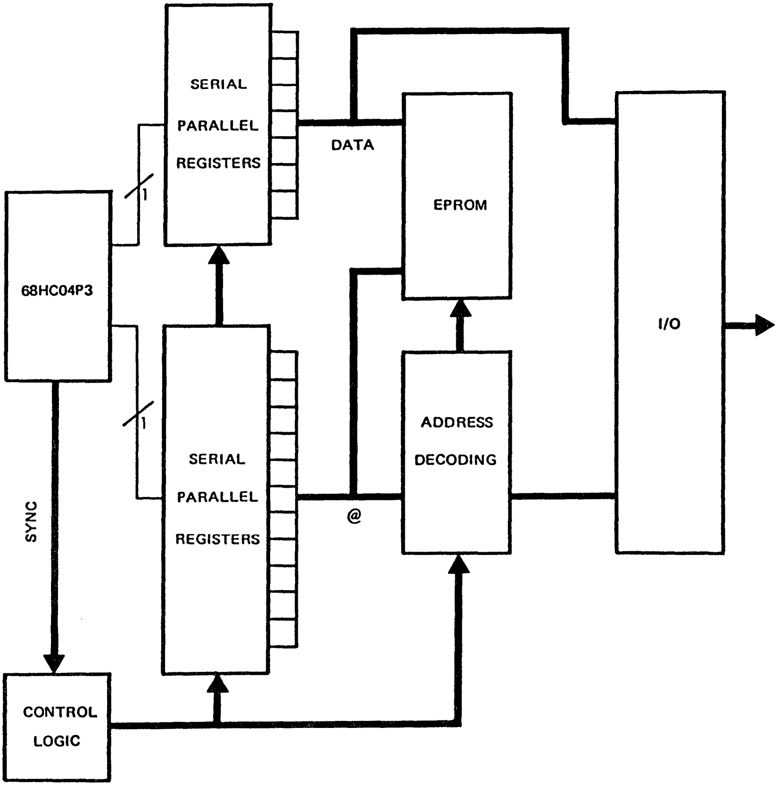
PIGGY-BACK EMULATION TOOL

The EMU-04 is a printed board module, build around EF68HC04P3, and a standard EPROM.

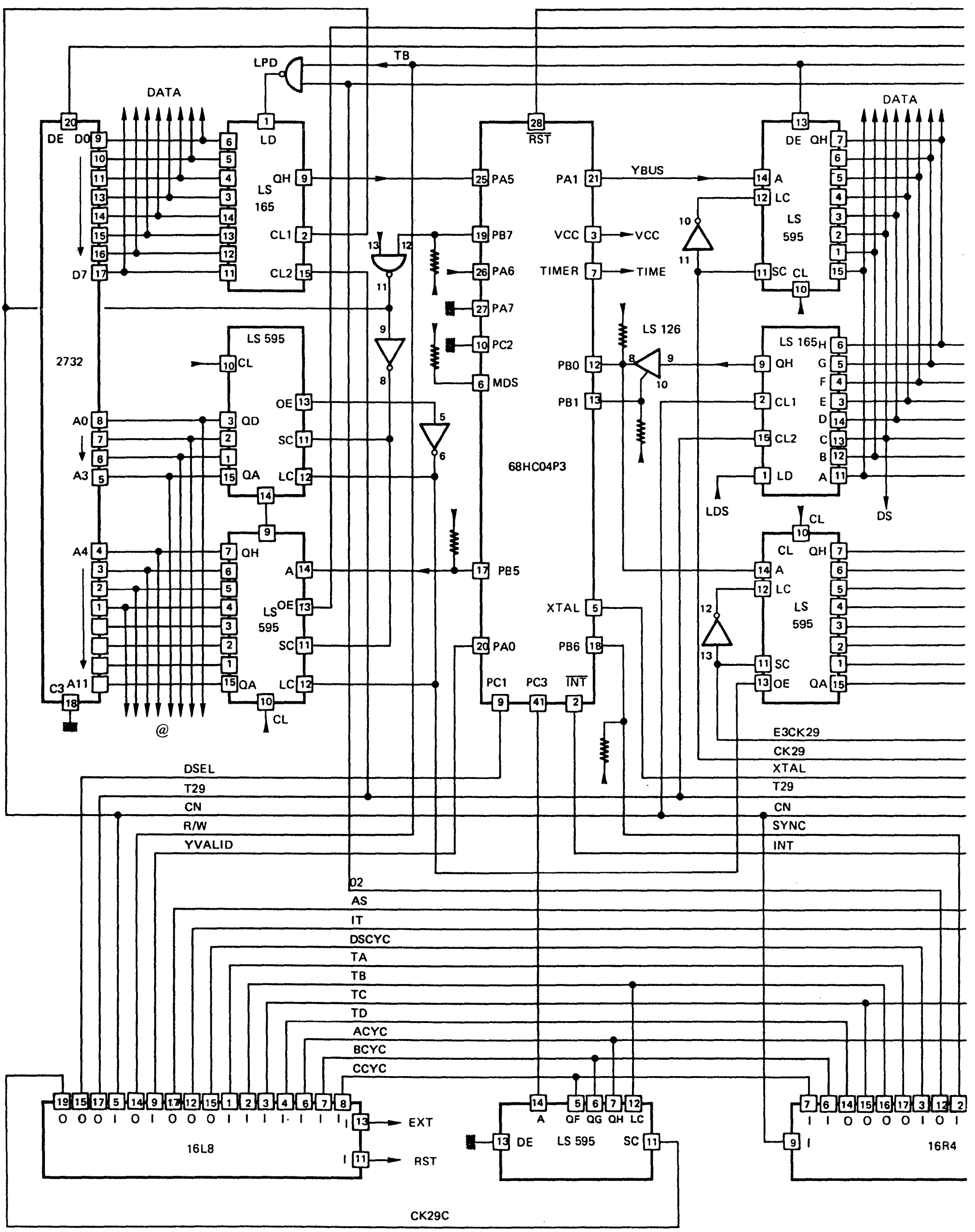
It fully emulates de EF68HC04P3 in the final applications, with the customer program stored in the board EPROM

The EMU-04 includes MICRO, RAM, STACK, I/O parts EPROM Data and Program Space.

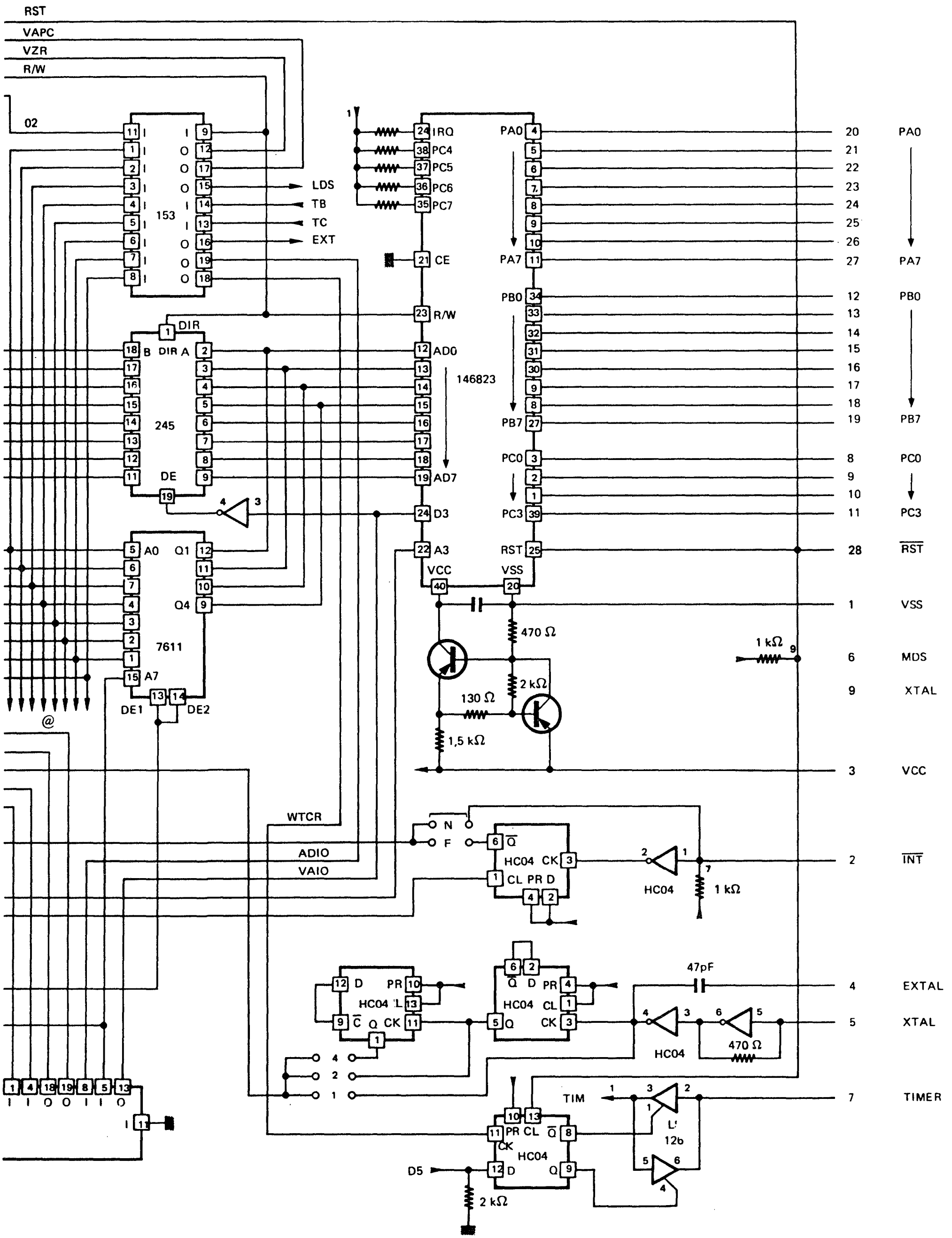
The following block diagram discribes it.



DIAGRAM



LPD
TB
DATA
DE
D0
D7
2732
A0
A3
A4
A11
C3
LD
QH
LS
165
CL1
CL2
LS 595
CL
QD
QA
OE
SC
LC
A
595
@
28
RST
PA5
PB7
PA6
PA7
PC2
MDS
68HC04P3
PB5
PA0
PC1
PC3
INT
PA1
YBUS
VCC
TIMER
TIME
PB0
PB1
LS 126
XTAL
PB6
DATA
DE QH
LS
595
SC CL
LS 165
H
G
F
E
D
C
B
A
LDS
DS
E3CK29
CK29
XTAL
T29
CN
SYNC
INT
DSEL
T29
CN
R/W
YVALID
02
AS
IT
DSCYC
TA
TB
TC
TD
ACYC
BCYC
CCYC
O O O I O I O O O I I I I I I I
16L8
EXT
RST
LS 595
QF QG QH LC
DE
SC
I I O O O O O I O I
16R4
CK29C


RST
VAPC
VZR
R/W
02
153
LDS
TB
TC
EXT
245
DIR
B DIR A
DE
7611
A0
Q1
Q4
A7
DE1
DE2
@
146823
IRQ
PC4
PC5
PC6
PC7
CE
R/W
AD0
AD7
D3
A3
VCC
VSS
PA0
PA7
PB0
PB7
PC0
PC3
RST
470 Ω
2 kΩ
130 Ω
1,5 kΩ
1 kΩ
20 PA0
21
22
23
24
25
26
27 PA7
12 PB0
13
14
15
16
17
18
19 PB7
8 PC0
9
10
11 PC3
28 RST
1 VSS
6 MDS
9 XTAL
3 VCC
2 INT
4 EXTAL
5 XTAL
7 TIMER
WTCR
ADIO
VAIO
N
F
HC04
Q
CK
CL PR D
D
PR
CL
C
47pF
470 Ω
4
2
1
TIM
L!
12b
PR CL
CK
D5
2 kΩ
1 I O O I I O
I

SOFTWARE

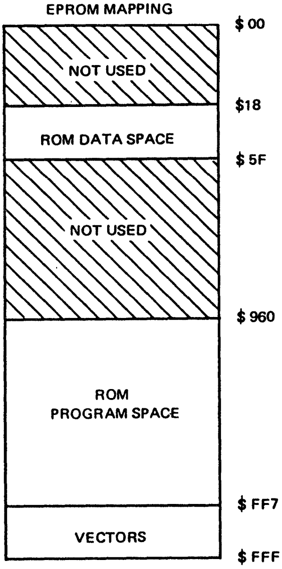
Instructions STOP & WAIT correctly work on piggy-back. Due to PAL & LS components of P.C. board, supply current does not meet Micro's spec. It stands around 800 mA as it does in RUN mode.

2732 EPROM have been chosen (acces time <X 300 nS) but 2764 or 27128 ones may be used, taking account than:
- upper address bit must be correctly setted
- program codes must stand in the lower 4 Kbytes space
- one have to wire a socket adaptor between EPROM & piggy-back EPROM socket.

DESCRIPTION

All EF68HC04P3 fonctions are restored. Thus application is emulated in real time and user program may be modified.

The DATA space and PROGRAM space are in EPROM space. ADDRESS $00 to $17 and $60 to $95 F are unknown.



CLOCK

Instead of using EF68HC04P3 one can plug the piggy-back and use application clock, either quartz (pin EXTAL-XTAL) or CMOS levels (pin XTAL).

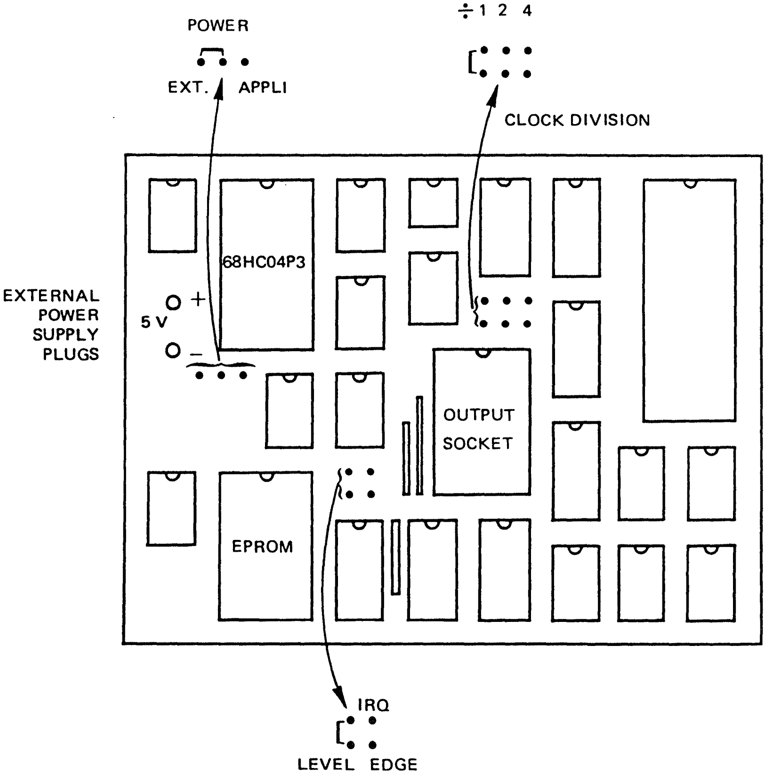
SELECTABLE OPTIONS

As EF68HC04P3 device you can choose maskable option on piggy-back board:
- CLOCK: division by 1,2,4
- IRQ: level or negative going edge.

Two ways are available to supply the piggy-back:
- Application supply (pin 3 of EF68CH02P3 socket)
- External power supply through DC supply plugs on piggy-back board.

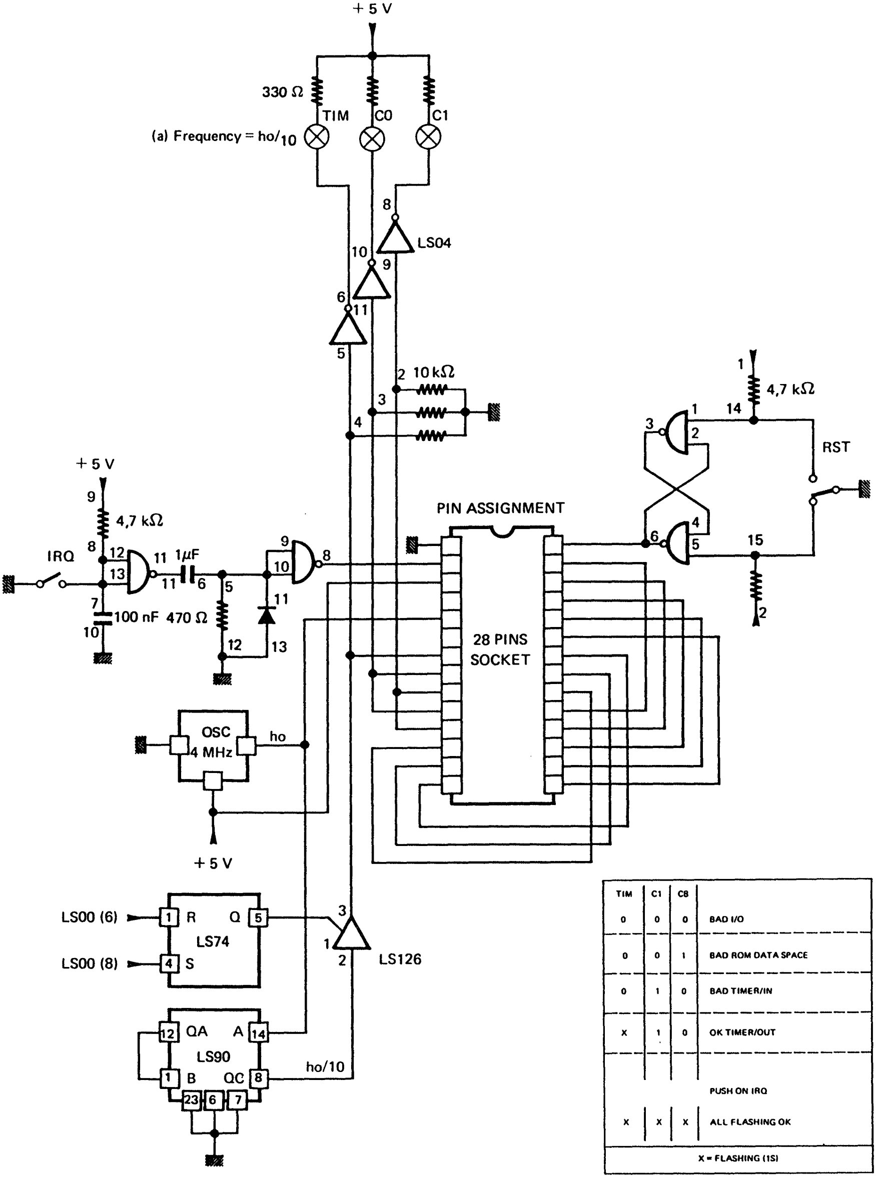
Due to bipolar components V supply must be 5 volts. (0,8 A).

The following diagram shows where jumpers are located.



SELF-CHECK

A self-check program is available in EPROM to verify piggy-back. To use it you must connect pins according to the following diagram. (Piggy-back self-check program is different than monochip one).



TIM	C1	C8	
0	0	0	BAD I/O
0	0	1	BAD ROM DATA SPACE
0	1	0	BAD TIMER/IN
X	1	0	OK TIMER/OUT
			PUSH ON IRQ
X	X	X	ALL FLASHING OK
X = FLASHING (1S)			

THOMSON SEMICONDUCTEURS SALES HEADQUARTERS :
43, AVENUE DE L'EUROPE - 78140 VELIZY-VILLACOUBLAY - FRANCE
TEL. (33 1) 39 46 97 19 / TELEX 204 780 F / TELEFAX. (33 1) 39 46 52 64

INTERNATIONAL SALES NETWORK

SUBSIDIARIES (March 1987)

AUSTRALIA

THOMSON COMPONENTS (AUSTRALASIA) PTY Ltd
Suite 211 - Edgecliff centre
203-233 New South Head Road
SYDNEY NSW 2027
Tel. (61 2) 327.13.18 Telex : AA 126911 TCAUS
Telefax. (61 2) 327.61.76

AUSTRIA

THOMSON-CSF Elektronische Anlagen GmbH
Lützowgasse 12-14
1140 WIEN
Tel. (43 222) 94.62.82 Telex : 135572 TCSF WA

BELGIUM and The NETHERLANDS

THOMSON-CSF S.A.-N.V.
363 Avenue Louise B.P. 10
B-1050 BRUSSELS
Tel. (32 2) 648.64.85 Telex : 24005 THBXLB
Telefax. (32 2) 648.19.13

BRAZIL

THOMSON COMPONENTES DO BRASIL
Avenida Roque Petroni JR
BROOKLIN - CEP 04707
SAO PAULO
Tel. (55 11) 542.47.22 Telex : 1124226 TCSF BR
Telefax. (55 11) 61.50.18

FAR EAST ASIA

THOMSON-CSF FAR EAST Ltd
92, Granville Road
Energy Plaza Building
Units 1.2.3 A, 6th Floor
Tsimshatsui East
KOWLOON - HONG KONG
Tel. (852 3) 721.96.82 Telex : 40766 HX TCFE
Telefax. (852 3) 722.00.44

FRANCE

THOMSON SEMICONDUCTEURS
43, Avenue de l'Europe
78140 VÉLIZY-VILLACOUBLAY
Tel. (33 1) 39.46.97.19 Telex : TCSF 204 780 F
Telefax. (33 1) 39.46.52.64

GERMANY (WEST)

THOMSON BAUELEMENTE GmbH
Perchtinger Str 3
Postfach 70.19.09
8000 MUNCHEN 70
Tel. (49 89) 78.79.0 Telex : 522916 CSFD
Telefax. (49 89) 787.91.45

ITALY

THOMSON COMPONENTI
Via M. Gioia 72
20125 MILANO
Tel. (39 2) 699.41 Telex : 330.301 TOMCO-I
Telefax. (39 2) 607.05.27

THOMSON COMPONENTI
Via Sergio I, 32
00165 ROMA
Tel. (39 6) 63.90.248 Telex : 620683
Telefax. (39 6) 63.90.207

JAPAN

THOMSON JAPAN
Semiconductor division
TBR Bldg 715 Kojimachi 5-7
CHIYODA-KU, TOKYO 102
Tel. (81 3) 264.6348 Telex : 2324241 THCSF. J.
Telefax. (81 3) 234.7539

SOUTH EAST ASIA

THOMSON-CSF S.E. ASIA PTE. LTD.
996 Bendermeer Road # 05-05
Kallang Basin Industrial Estate
SINGAPORE 1233
Tel. (65) 295.31.24 Telex : RS 36124 TC SEA
Telefax. (65) 296.87.15

SPAIN

THOMSON COMPONENTES Y TUBOS
Calle Albacete, 5
MADRID 28027
Tel. (34 1) 405.16.15 Telex : 46033
Telefax. (34 1) 404.60.02

THOMSON COMPONENTES Y TUBOS
Poligono Industrial Fontsanta Montilla, 5
San Juan Despi
BARCELONA
Tel. (34 3) 373.30.11 Telex : 53077
Telefax. (34 3) 373.38.50

SWEDEN

THOMSON KOMPONENTER SKANDINAVIAN AB
Sandhamnsgatan 65 Box 27257
S-10253 STOCKHOLM
Tel. (46 8) 63.50.60 Telex : 12078 THCSF S
Telefax. (46 8) 63.28.28

UNITED KINGDOM and IRELAND

THOMSON COMPONENTS Ltd
Ringway House Bell Road
DANNESHILL
BASINGSTOKE - HANTS RG 24-0QG
Tel. (44 256) 29.155 Telex : 858865
Telefax. (44 256) 23.172

U.S.A. (see complete network in the following pages)

THOMSON COMPONENTS MOSTEK CORPORATION
1310 Electronics Drive
CARROLLTON, Texas 75006
Tel. (1) 214/466-6000 Telex : 730643
Telefax. (1) 214/466-7602

For all other countries :
see THOMSON Composants Export network on the following pages.

INTERNATIONAL SALES NETWORK

DISTRIBUTORS (March 1987)

AUSTRALIA

CONSULAUST INTERNATIONAL PTY LTD
P.O. Box 357
734 Riversdale Road
CAMBERWELL, VIC 3124
Tel. (61 3) 836.25.66 Telex : 37455 CONAUS AA
Telefax. (61 3) 836.17.64

3, Rd Floor
10, Bridge Street
GRANVILLE NSN 2142 / SYDNEY
Tel. (61 2) 637.25.58 Telex : 23861

PROMARK ELECTRONICS PTY LTD
Suite 208, 6-8 Clarke Street
CROWS NEST, NSW 2065
Tel. (61 2) 439.64.77 Telex : AA 20474 PROMAR
Telefax. (61 2) 436.08.63

Suite 3, 366 Whitehorse road
NUNAWADING, VIC 3131
Tel. (61 3) 878.12.55 Telex : AA 35045 PROMEL
Telefax. (61 3) 842.64.60

BELGIUM AND THE NETHERLANDS

INELCO
94, avenue des Croix de guerre
B-1120 BRUSSELS
Tel. (32 2) 216.01.60 Telex : 64475
Telefax. (32 2) 216.61.50

ITT MULTICOMPONENTS
15, rue Antoine-Court
B-1080 BRUSSELS
Tel. (32 2) 424.28.00 Telex : 22901

ALCOM ELECTRONICS
Esse Baan 1
2908 LJ CAPELLE A/D IJSSEL
Tel. (31 10) 51.95.33 Telex : 26160
Telefax. (31 10) 45.86.48

ITT MULTICOMPONENTS
PB 345
2700 AH ZOETERMEER
Tel. (31 1794) 10141 Telex : 32360
Telefax. (31 1794) 16544

DENMARK

SCAN SUPPLY A/S
Nannasgade 18
DK 2200 - COPENHAGEN
Tel. (45 1) 83.50.90 Telex : SCALPY DK
Telefax. (45 1) 83.25.40

FINLAND

A LÄIKKÔ OY
Kolmas Linja 16B22
SF - 00530 HELSINKI
Tel. (358 0) 75.04.14 Telex : 125200 TOPCO SF
Telefax. (358 0) 70.17.396

INSELE OY
Kumpulantie 1
SF-00520 HELSINKI
Tel. (358 0) 75.06.00 Telex : 122217
Telefax. (358 0) 73.57.74

FRANCE

Région parisienne

ALMEX
48, rue de l'Aubépine
B.P. 102
Z.I. 92160 ANTONY
Tel. (1) 46.66.21.12 Telex : 250067
Telefax. (1) 46.66.60.28

C.G.E. COMPOSANTS / GECODIS
32, rue Grange -Dame-Rose
Z.I. de Vélizy
92360 MEUDON LA FORÊT
Tel. (1) 46.30.24.34
Telefax. (1) 46.30.00.81

MECODIS
31, rue de Metz
94700 MAISONS ALFORT
Tel. (1) 43.78.18.18 Telex : 262061
Telefax. (1) 48.93.13.13

PEP
6/8, rue Ambroise Croizat
Z.I. des Glaises
91120 PALAISEAU
Tel. (1) 64.47.00.31 Telex : 603341
Telefax. (1) 64.47.00.84

RTF DIFFUSION
59-63, rue Desnouettes
75015 PARIS
Tel. (1) 45.31.16.50 Telex : 260832

SILEC/CODICOM
52, rue des Carrières
B.P. 43
94222 CHARENTON LE PONT
Tel. (1) 43.75.95.92 Telex : 680363

TCC-DIS
30, avenue de l'Épi d'Or
94800 VILLEJUIF
Tel. (1) 46.87.22.24 Telex : 261049

TEKELEC/AIRTRONIC
2-6, avenue Salvador-Allende - Bât. C
93804 EPINAY CEDEX
Tel. (1) 48.21.60.44 Telex : 630260

Centre de l'agora
523, place des Terrasses
91034 EVRY CEDEX
Tel. (1) 60.77.82.66 Telex : 691158

Burêau des Arcades
424 La Closerie - Bât. A
Clos Mont d'Est
93160 NOISY-LE-GRAND
Tel. (1) 43.04.62.00 Telex : 220368

1, rue Carle Vernet
B.P. 2
92310 SEVRES
Tel. (1) 45.34.75.35 Telex : 2045[illegible]2
Telefax. (1) 45.07.21.91

5, allée du Bourbonnais
78310 MAUREPAS
Tel. (1) 30.62.00.58 Telex : 698121

Nord

C.G.E. COMPOSANTS
Parc des Moulins
2, rue de la Créativité
59650 VILLENEUVE D'ASCQ
Tel. 20.67.04.04

HEXA COMPOSANTS / DIRECT
Immeuble Le Tertial
1, rue des Promenades
B.P. 11
59110 LA MADELEINE
Tel. 20.55.58.03 Telex : 120934

SILEC/SIDE
Avenue Robert-Schuman
C2 Résidence de l'Europe
59370 MONS-EN-BAROEUL
Tel. 20.04.75.08 Telex : 172017

TEKELEC/AIRTRONIC
Immeuble Moulin 2
5, rue du Colibri
59650 VILLENEUVE-D'ASCQ
Tel. 20.05.17.00 Telex : 160011

Est

C.G.E. COMPOSANTS
27, avenue Kléber
68000 COLMAR
Tel. 89.41.15.43 Telex : 880951

HEXA COMPOSANTS / SELFCO
31, rue du Fossé-des-Treize
67000 STRASBOURG
Tel. 88.22.08.88 Telex : 890706

SILEC/DEL
Rue de l'Escault
Z.I. St-Apollinaire
21000 DIJON
Tel. 80.71.57.45 Telex : 350833

SILEC/SLRD
36, rue des Jardins
B.P. 1
57023 METZ CEDEX 2
Tel. 87.32.53.12 Telex : 660177

SILEC/CODICOM
24, rue des Prés l'Évêque
10000 TROYES
Tel. 25.82.17.32 / 27.82.17.43

TEKELEC/AIRTRONIC
8, rue Gustave-Adolphe-Hirn
67000 STRASBOURG
Tel. 88.22.31.51 Telex : 880765

Rhône-Alpes

ALMEX
Norly 2
Route des Peupliers
69750 DARDILLY
Tel. 78.66.00.66 Telex : 375187

C.G.E. COMPOSANTS
101, rue de Dieu
69100 VILLEURBANNE
Tel. 78.68.32.29

PEP
191 C, avenue Saint-Exupéry
69500 BRON
Tel. 78.00.70.02 Telex : 340855

RTF DIFFUSION
Le Chatelet
5, place Carnot
42000 SAINT-ETIENNE
Tel. 77.74.61.49 Telex : 300921

SILEC
8, rue des Frères L.-et-E. Bertrand
69632 VENISSIEUX CEDEX
Tel. 78.00.86.97 Telex : 340189

SILEC/DEL
Immeuble Le Zodiac
40, avenue de la Maveria
74000 ANNECY-LE-VIEUX
Tel. 50.23.17.29 Telex : 309261

Z.A.C. du Rondeau
Impasse du Docteur-Pascal
38431 ECHIROLLES CEDEX
Tel. 76.22.05.09 Telex : 960938

TCC-DIS
12 A, rue de Rupetit
69740 GENAS
Tel. 78.90.30.79

TEKELEC/AIRTRONIC
22, rue de la Baïsse
69100 VILLEURBANNE
Tel. 78.84.06.08 Telex : 370481

Zirst 4
Chemin des Prés
38240 MEYLAN CEDEX
Tel. 76.41.11.36 Telex : 980207

Ouest

ALMEX
Immeuble Armorique
171, rue de Vern
35022 RENNES
Tel. 99.51.66.16 Telex : 741034
Telefax. 99.51.79.91

C.G.E. COMPOSANTS
9, rue du Général-Nicolet
B.P. 1466
35015 RENNES CEDEX
Tel. 99.50.40.40 Telex : 740311

HEXA COMPOSANTS
7, rue Mogador
76000 ROUEN
Tel. 35.89.57.11

HEXA COMPOSANTS / DIRECT
B.P. 4012
76021 ROUEN CEDEX
Tel. 35.98.17.98 Telex : 770842
Telefax. 35.88.50.81

RTF DIFFUSION
Rue des Landelles
Z.I. Sud-Est
35510 CESSON SEVIGNE
Tel. 99.41.81.21 Telex : 741239

SILEC/SIDE
Résidence Front-de-Seine
41, quai du Havre
76000 ROUEN
Tel. 35.98.22.99 Telex : 172017

SILEC/RIME
Rue de la Dutée
B.P. 38
44800 SAINT-HERBLAIN
Tel. 40.46.12.00 Telex : 710084

TEKELEC/AIRTRONIC
20, avenue de Crimée
B.P. 2246
35022 RENNES CEDEX
Tel. 99.50.62.35 Telex : 740414

Centre et Centre-Ouest

C.G.E. COMPOSANTS
Z.I. Allée de la Détente
86361 CHASSENEUIL-DU-POITOU
Tel. 49.52.88.88 Telex : 791525

SILEC/AUVERLEC
Z.I. 2, rue de l'Industrie
B.P. 2
63800 COURNON-D'AUVERGNE
Tel. 73.84.76.62 Telex : 390926
Telefax. 73.84.95.26

Aquitaine-Midi-Pyrénées

ALMEX
Immeuble «Centreda»
Avenue Didier-Daurat
31700 BLAGNAC
Tel. 61.71.11.22 Telex : 521370

C.G.E. COMPOSANTS
B.P. 81
33605 PESSAC CEDEX
Tel. 56.36.40.40 Telex : 550696
Telefax. 56.07.64.41

55, avenue Louis-Bréguet
31400 TOULOUSE
Tel. 61.20.82.38 Telex : 530957

HEXA COMPOSANTS / SODIMEP
Z.I. du Palays
B.P. 4345
31038 TOULOUSE CEDEX
Tel. 61.54.34.54 Telex : 530737
Telefax. 61.34.95.33

Cedex 52
Rue Edouard Faure
33083 BORDEAUX CEDEX
Tel. 56.39.93.42

SILEC/SPELEC
55, boulevard de Thibaud
31084 TOULOUSE CEDEX
Tel. 61.41.05.00 Telex : 530777

TEKELEC/AIRTRONIC
Immeuble «Le Montesquieu»
Avenue Président-Kennedy
33700 MERIGNAC
Tel. 56.34.84.11 Telex : 550589

22/24, boulevard Thibaud
31084 TOULOUSE CEDEX
Tel. 61.40.83.94 Telex : 520374

Provence-Côte d'Azur

C.G.E. COMPOSANTS
Bureau 3000
Avenue Eugène-Donadei
Bât. B
06700 ST-LAURENT-DU-VAR
Tel. 93.07.77.67 Telex : 461481

HEXA COMPOSANTS / DIMEL
B.P. 1153
83058 TOULON CEDEX
Tel. 94.41.49.63 Telex : 430093
Telefax. 94.46.54.66

TCC-DIS/SRD
Z.A. l'agavon
Avenue Lamartine
13170 LES PENNES-MIRABEAU
Tel. 42.02.91.08 Telex : 440076

TEKELEC/AIRTRONIC
Tertia 2
B.P. 77
13762 LES MILLES CEDEX
Tel. 42.24.40.45 Telex : 440928

GERMANY AND AUSTRIA

ELECDIS RUGGABER GMBH
Hertichstr. 41
7250 LEONBERG / STUTTGART
Tel. (49 7152) 602.0 Telex : 724192

ECS FREHSDORF GMBH
Carl-Zeiss-Str. 3
2085 QUICKBORN
Tel. (49 4106) 700.50 Telex : 213693

JERMYN GMBH
Schulstr. 84
6277 BAD CAMBERG
Tel. (49 6434) 230 Telex : 415257
Telefax. (49 6434) 23-289

METRONIK GMBH
Leonhardsweg 2
8025 UNTERHACHING
Tel. (49 89) 61108-0 Telex : 17/897434
Telefax. (49 89) 6116468

RSC-HALBLEITER UND
Elektronische Bauelemente GMBH
Industriestr. 2
7536 ISPRINGEN / PFORZHEIM
Tel. (49 7231) 801-0 Telex : 783650
Telefax. (49 7231) 82282

SETRON SCHIFFER ELECTRONIC GMBH
Theodor-Heuss-Str. 4b
3300 BRAUNSCHWEIG
Tel. (49 531) 80011 Telex : 952812
Telefax. (49 531) 85910

SPOERLE ELEKTRONIK KG
Max-Plank-Str. 1-3
6072 DREIEICH
Tel. (49 6103) 304-0 Telex : 417983
Telefax. (49 6103) 304201

ELBATEX GES.M.B.H.
Eitnergasse 6
A-1232 WIEN
Tel. (43 222) 86.32.11-0 Telex : 133128
Telefax. (43 222) 86.32.11-200

GREECE

MAKONIK A. LUCINI AND CO.OE
90 Archilleus Street
KALLITHEA - ATHENS
Tel. (30) 941.93.29 Telex : 219150 MAKO GR

HONG KONG

JARDINE INDUSTRIAL SUPPLIES
20 th Floor, Sing Pao Building
101, King's Road, North Point,
HONG KONG
Tel. (852 5) 716213 Telex : 60004 JISHK HX
Telefax. (852 5) 807.29.30

INDONESIA

PT GENERAL EKA ENGINEER'S CORP LTD
JLN Tanah Abang IV NO 15
JAKARTA
Tel. (6221) 361608 Telex : 46761 TTGE IA

ITALY

ADVECO
Via-Lattuada, 20
20135 MILANO
Tel. (39 2) 545.64.65 Telex : 340116

CAMEL
Via Tiziano 18
20145 - MILANO
Tel, (39 2) 498.14.81 Telex : 325237
Telefax. (39 2) 481.86.37

COMPREL S.P.A.
Viale F. Testi 115
20092 CINISELLO BALSAMO (MI)
Tel. (39 2) 612.06.41/5 Telex : 332484
Telefax. (39 2) 612.81.58

ELEDRA
Via Giacomo Watt, 27
20143 MILANO
Tel. (39 2) 81.82.1 Telex : 332332
Telefax. (39 2) 81.82.211

LASI
Viale F. Testi 126
20092 - CINISELLO BALSAMO (MI)
Tel. (39 2) 244.00.12-212 Telex : 352040
Telefax. (39 2) 248.77.17

MECOM
Via Ognissanti, 83
35129 PADOVA
Tel. (39 2) 807.03.22 Telex : 430602
Telefax. (39 049) 807.02.78

RECOM
Via E. Collamarini, 22
40138 BOLOGNA
Tel. (39 051) 53.48.83 Telex : 511818
Telefax. (39 051) 53.12.17

RG2
Via Rosa Luxemburg 12/14
10093 COLLEGNO, TORINO
Tel. (39 011) 71.22.89 Telex : 224335

JAPAN

AMSC CO, LTD
Asahiseimei 6-7, Nakamachi 1-chome,
Musashino-Sh 1,
TOKYO 180
Tel. (81 422) 54.6800
Telefax. (81 422) 54.6179

Toabo 2nd Bldg 23, Awajimachi 3-chome,
Higashiku
OSAKA 541
Tel. (81 06) 222.1031

INTERNIX INC.,
Shinjuku Hamada Bldg
7-4-7, Nishi Shinjuku
SHINJUKU-KU, TOKYO 160
Tel. (81 3) 369-1101 Telex : J26733
Telefax. (81 3) 366-8566

MCM JAPAN LTD
Maison L-203
1-1-6, Sakurashinmachi-Setagaya-ku
TOKYO 154
Tel. (81 3) 705.5611 Telex : 02466416MCM JPNJ
Telefax. (81 3) 705.5617

MICRO SUMMIT K.K.
NO.5 MORI Bldg
1-17-1 Toranomon
MINATO-KU, TOKYO 105
Tel. (81 3) 593-2210
Telefax. (81 3) 593-2206

TOMEN ELECTRONICS CORP.
1-1 Uchisaiwai-cho 2 chome
CHIYODA-KU, TOKYO 100
Tel. (81 3) 506-3671
Telefax. (81 3) 506-3497

MALAYSIA

COSMOS COMPANY
79 B Jalan Jejaka 9
Taman Maluri, cheras
55100 KUA LA LUMPUR
Tel. (60 03) 985.1812 Telex : MA 21527 COSMOS

NEW ZEALAND

HENLYNN DISTRIBUTORS LTD
Po. Box 12 550
PENROSE AUCKLAND
Tel. (64 9) 592453 Telex : NZ 63677
Telefax : (64 9) 592561

NORWAY

SATT ELECTRONICS A/S
P.O Box 70
N-1322 HOVIK
Tel. (47 2) 12.36.00 Telex : 72558
Telefax. (47 2) 12.32.10

PHILIPPINES

EDGE-HORTH MARKETING CORPORATION
5TH FLR, Pacific Bank Bldg,
O, Paredes Street, C.P.O. Box 2 '29
BINONDO, MANILA
Tel. (63) 471541/8 Telex : RCA 27993 EDAMC PH

PORTUGAL

ELECTROQUIPO
15, rua S. José
1100 LISBOA
Tel. (351) 37.03.17
Telex : 18123 Electroquipo

SINGAPORE

CONY ELECTRONICS (S) PRIVATE LTD
10 Jalan Besar # 03-25
Sim Lim Tower
SINGAPORE 0820
Tel. (65) 296.21.11 Telex : RS 34804 CONY
Telefax. (65) 296.03.39

SOUTH KOREA

BUK SUNG TRADING COMPANY LTD
# 502 Samyung Building
159-22 Docksan-4 Dong
KURO-KU, SEOUL
Tel. (82 2) 854.1362/3/4 Telex : BUSUCO K 26925
Telefax. (82 2) 862.1273

SPAIN

SAGITRON, S.A.
Castelló, 25
28001 MADRID
Tel. (34 1) 402.60.85 Telex : 43819
Telefax. (34 1) 275.40.23

Ronda General Mitre, 200
08006 BARCELONA
Tel. (34 3) 212.65.83 Telex : 97317

VECTOR ESPANA, S.A.
Torre de Madrid
Plaza de Espana, 18
28008 MADRID
Tel. (34 1) 247.72.96 Telex : 49331

Muntaner, 281
08021 BARCELONA
Tel. (34 3) 200.37.44 Telex : 98343

SWEDEN

FERTRONIC AB
Box 3035/Dalvagen 12
S-17103 SOLNA
Tel. (46 8) 83.00.60 Telex : 11181 FERTRON S
Telefax. (46 8) 83.28.20

TH'S ELEKTRONIK AB
Box 3027
Arrendevagen 36
S-16303 - SPANGA
Tel. (46 8) 36.29.70 Telex : 11145
Telefax. (46 8) 761.30.65

SWITZERLAND

MODULATOR S.A.
Konizstrasse 194
CH-3097 BERN-LIEBEFELD
Tel. (41 31) 59.22.22 Telex : 911 859
Telefax.(41 31) 21.02.85

TAIWAN

HUGE CO LTD
10F, n° 95, SEC 4
Nanking E Road
TAIPEI/TAIWAN
Tel. (886 2) 771.5373-5 Telex : 25619 Huge Co
Telefax. (886 2) 717.4114

THAILAND

CHOAKCHAÏ ELECTRONIC SUPPLIES LTD.PART.
128/21-23 THANON ATSADANG
BANGKOK 10200
Tel. (66 2) 222.3921/221.9075 Telex : 84809 CESLP TH
Telefax. (66 2) 224.7639

UNITED KINGDOM AND IRELAND

ABACUS ELECTRONICS LTD
Abacus House - Bone Lane
Newbury
BERKSHIRE RG14 5SF
Tel. (44 635) 33311/5 Telex : 847589

ADDON DEVICES
11 Shield Road
Ashford Industrial Estate
ASHFORD MIDDLESEX TW 15 1AU
Tel. (44 7842) 47141/2/3 Telex : 923802

ANZAC COMPONENTS LTD
822 Yeovill Road
Slough Trading Estate
SLOUGH SL1 4JA
Tel. (44 6286) 4701 Telex : 847949

BLOOMER ELECTRONIC COMPONENTS
9-10 Carn industrial Estate
Portadown Co. Armagh
NORTHERN IRELAND BT 63 5RH
Tel. (353 762) 339818 Telex : 748054

HAWKE ELECTRONICS
Amotex House
45 Hanworth Road
Sunbury on Thames
MIDDLESEX TW 16 5DA
Tel. (44 1) 979.7799 Telex : 723592

HITEK
Ditton Walk
Beadle Estate
CAMBRIDGE CB 5 8QD
Tel. (44 223) 214722 Telex : 817347
Telefax. (44 223) 214365

HYCOMP
11 Shield Road
Ashford Industrial Estate
ASHFORD MIDDLESEX TW 15 1AU
Tel. (44 7842) 46273 Telex : 923802

POLAR ELECTRONICS LTD
Europa House/West Street
Leighton Buzzard
BEDFORSHIRE
Tel. (353 525) 377093 Telex : 825238

PRONTO ELECTRONICS
399-425 Eastern Avenue
GANTS HILL - ILFORD
ESSEX
Tel. (44 1) 554622 Telex : 8954213

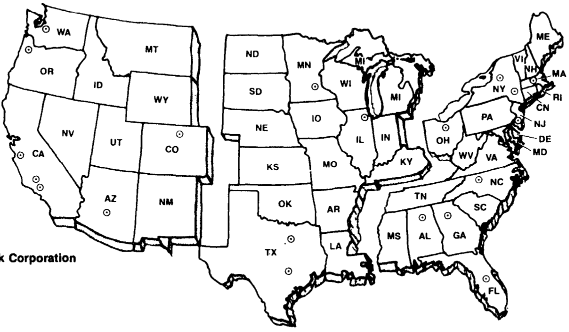
SEMICONDUCTOR SPECIALISTS
Carroll House
159, High Street
West Drayton / MIDDLESEX
Tel. (44 8954) 46415 Telex : 21958

TRANSWORLD SCIENTIFIC
Richardson Street
HIGH WYCOMBE
BUCKINGHAMSHIRE HP11 2QH
Tel. (353 494) 36381 Telex : 837236

THOMSON COMPONENTS MOSTEK CORPORATION SALES HEADQUARTERS :
1310 ELECTRONICS DRIVE - CARROLLTON, TEXAS 75006
TEL. (1) 214/466-6000 / TELEX : 730643 / TELEFAX. (1) 214/466-7602

U.S. AND CANADIAN SALES NETWORK

SALES OFFICES (March 1987)



TECHNICAL SUPPORT:

Thomson Components - Mostek Corporation
16 Commerce Drive
Montgomeryville, PA 18936-1002
215/362-8500
FAX 215/362-1293

WESTERN AREA:

Thomson Components - Mostek Corporation
2540 Mission College Blvd.
Suite 104
Santa Clara, CA 95054
408/970-8585
FAX 408-970-8737

Thomson Components - Mostek Corporation
18004 Skypark Circle
Suite 140
Irvine, CA 92714
714/250-0455
FAX 714/261-1505

Thomson Components - Mostek Corporation
6203 Variel Ave.
Unit A, P.O. Box 4051
Woodland Hills, CA 91367
818/887-1010
FAX 818/702-0725

Thomson Components - Mostek Corporation
1107 North East 45th St.
Suite 411
Seattle, WA 98105
206/632-0245
FAX 206/633-5413

Thomson Components - Mostek Corporation
601 South Bowen St.
Longmont, CO 80501
303/449-9000
FAX 303/651-7676

Thomson Components - Mostek Corporation
7950 East Redfield Rd.
Suite 160
Scottsdale, AZ 85260
602/998-1580
FAX 602/483-2303

Thomson Components - Mostek Corporation
7155 SW Varns St.
Tigard, OR 97223-8057
503/620-5517
FAX 503/639-3591

CENTRAL AREA:

Thomson Components - Mostek Corporation
1310 Electronics
MS1137
Carrollton, TX 75006
214/466-8844
TWX 910-860-5437

Thomson Components - Mostek Corporation
6100 Green Valley Drive
Suite 130
Bloomington, MN 55438
612/831-2322
FAX 612/831-8195

Thomson Components - Mostek Corporation
1827 Walden Office Square
#430
Schaumburg, IL 60173
312/397-6550
FAX 312/397-4066

Thomson Components - Mostek Corporation
3215 Steck Ave.
Suite 202
Austin, TX 78758
512/451-4061
TWX 910-874-2007

Thomson Components - Mostek Corporation
26677 W. 12 Mile Rd.
Suite #141
Southfield, MI 48034
313/354-5840
FAX 313/354-3370

CANADA

Thomson CSF Canada
Semiconductor Division
1000 Sherbrooke West
Suite 2340
Montreal, Quebec H3A 3G4
514/288-4148
FAX 514/288-8987

Thomson Components - Mostek Corporation
44 Rosebud Ave.
Brampton, Ontario L6X 2W5
416/454-5252
FAX 416/454-4328

EASTERN AREA:

Thomson Components - Mostek Corporation
83 Cambridge Street
Suite 2A
Burlington, MA 01803
617/273-3310
FAX 617/272-2467

Thomson Components - Mostek Corporation
The Pavilions at Greentree
Route #73, Suite 101
Marlton, NJ 08053
609/596-9200
FAX 609/424-7437

Thomson Components - Mostek Corporation
4414 Evangel Cr. #C
Huntsville, AL 35816
205/830-9036
FAX 205/830-9038

Thomson Components - Mostek Corporation
4651 Crossroads Park
Liverpool, NY 13088
315/457-2160
FAX 315/457-2164

Thomson Components - Mostek Corporation
387 Hooker Avenue
Office No. 2
Poughkeepsie, NY 12603
914/454-8813
FAX 914/454-1320

Thomson Components - Mostek Corporation
5890 Sawmill Rd.
Suite 204
Dublin, Ohio 43017
614/761-0676
FAX 614/761-2305

Thomson Components - Mostek Corporation
3200 Northline Avenue
Forum VI, Suite 501
Greensboro, NC 27408
919/282-4307

Thomson Components - Mostek Corporation
1014 Park Forest Court
Lilburn, GA 30247
404/925-2324

CORPORATE HEADQUARTERS

Thomson Components - Mostek Corporation
1310 Electronics - MS1137
Carrollton, TX 75006
214/466-6000

FOR ALL OTHER COUNTRIES
Thomson Semiconducteurs
43, Avenue de L'Europe
78140 Vélizy-Villacoublay/France
Tél: (1) 39 46 97 19/Télex: 204 780 F
or contact Corporate Headquarters

U.S. AND CANADIAN SALES NETWORK REPRESENTATIVES (March 1987)

CONNECTICUT
Scientific Components*
1185 S. Main St.
Cheshire, CT 06410
(203) 272-2963
FAX (203) 271-3048

FLORIDA
Sales Engineering Concepts, Inc.
926 Great Pond Dr.
Suite 2002
Altamonte Springs, FL 32714
(305) 682-4800
FAX (305) 682-6491

Sales Engineering Concepts, Inc.*
1000 S. Federal Hwy.
Suite 204
Deerfield Beach, FL 33441
(305) 426-4601
TWX 510-600-7740

ILLINOIS
John G. Twist Co.*
1301 Higgins Road
Elk Grove Village, IL 60007
(312) 593-0200
TWX 510-601-0552

INDIANA
M/S Sales Associates, Inc.*
7319 W. Jefferson Blvd.
Ft. Wayne, IN 46804
(219) 436-3023
FAX (219) 436-3026

M/S Sales Associates, Inc.
1425 E. 86th St.
Indianapolis, IN 46240
(317) 257-8916

KANSAS
Rush & West Associates
107 N. Chester Street
Olathe, KS 66061
(913) 764-2700
TWX 910-380-8110

MARYLAND
Tri-Mark Inc.*
1410 Crain Hwy. NW
Glen Burnie, MD 21061
(301) 761-6000
FAX (301) 761-6006

MASSACHUSETTS
A/D Nova Sales, Inc.*
2 Milita Drive
Lexington, MA 02173
(617) 861-1820

MICHIGAN
Electronic Sources, Inc.*
8014 W. Grand River
Suite 6
Brighton, MI 48116
(313) 227-3598
FAX (313) 227-5655

MINNESOTA
Horizon
8053 East Bloomington Freeway
Bloomington, MN 55420
(612) 884-6515
FAX (612) 888-3073

MISSOURI
Rush & West Associates
2170 Mason Rd.
St. Louis, MO 63131
(314) 965-3322
TWX 910-752-653
TELEX 752 653

NEW JERSEY
Tritek Sales, Inc.*
21 E. Euclid Ave.
Haddonfield, NJ 08033
(609) 429-1551
TWX 215-627-0149 (Philadelphia Line)
TWX 710-896-0881

NE Components*
189-191 Godwin Ave.
Wyckoff, NJ 07481
(201) 848-1100

NEW YORK
Empire Technical Assoc.*
33 West State St.
Suite 211B
Binghamton, NY 13901
(607) 772-0651

GT Sales*
34 Grand Blvd.
Brentwood, NY 11717
(516) 231-0270

OHIO
Five Star Electronics*
6200 S.O.M. Center Road
Suite B 21
Solon, OH 44139
(216) 349-1611

WISCONSIN
Heartland Technical Marketing
3846 Wisconsin Ave.
Milwaukee, WI 53208
(414) 931-0606

CANADA
Solution Electronic Sales*
100A 3380 Maquinna Dr.
Vancouver, B.C. V5S 4C9
(604) 438-0679

* Home Office

U.S. AND CANADIAN SALES NETWORK

DISTRIBUTORS (March 1987)

ALABAMA
Marshall Industries
3313 S. Memorial Pkwy.
Huntsville, AL 35801
(205) 881-9235

Quality Components, S.E.
4900 University Square #207
Huntsville, AL 35817
(205) 830-1881

Pioneer Technologies Group
4825 University Square
Huntsville, AL 35805
(205) 837-9300

Schweber Electronics
4930-A Corporate Drive
Huntsville, AL 35805
(205) 895-0480

ARIZONA
Kierulff Electronics
4134 E. Wood Street
Phoenix, AZ 85040
(602) 437-0750

Marshall Industries
835 West 22nd St.
Tempe, AZ 85282
(602) 968-6181

Schweber Electronics
11049 North 23rd Drive
Suite 100
Phoenix, AZ 85029
(602) 997-4874

ARKANSAS
See Oklahoma

CALIFORNIA
Integrated Electronics Corp.
7000 Franklin Blvd., Suite 625
Sacramento, CA 95823
(916) 424-5297

Integrated Electronics Corp.
2170 Paragon Drive
San Jose, CA 95131
(408) 435-1000

ITAL Sales
15405 Proctor Avenue
City of Industry, CA 91745
(818) 968-8515

Kierulff Electronics
9800 Variel Avenue
Chatsworth, CA 91311
(818) 407-2500

Kierulff Electronics
5650 Jillson St.
Los Angeles, CA 90040
(213) 725-0325

Kierulff Electronics
8797 Balboa Avenue
San Diego, CA 92123
(619) 278-2112

Kierulff Electronics
1180 Murphy Avenue
San Jose, CA 95131
(408) 971-2600

Kierulff Electronics
14101 Franklin Avenue
Tustin, CA 92680
(714) 731-5711

Marshall Industries
One Morgan
Irvine, CA 92715
(714) 458-5395

Marshall Industries
9710 DeSoto Avenue
Chatsworth, CA 91311
(818) 407-0101

Marshall Industries
3039 Kilgore Ave., #140
Rancho Cordova, CA 95670
(916) 635-9700

Marshall Industries
10105 Carroll Canyon Rd.
San Diego, CA 92131
(619) 578-9600

Marshall Industries
336 Los Coches St.
Milpitas, CA 95035
(408) 943-4600

Schweber Electronics
21139 Victory Blvd.
Conoga Park, CA 91303
(818) 999-4702

Schweber Electronics
1225 West 190th Street
Suite 360
Gardena, CA 90248
(213) 327-8409

Schweber Electronics
17822 Gillette Avenue
Irvine, CA 92714
(714) 863-0264
FAX (714) 863-0200 (X500)

Schweber Electronics
1771 Tribute Rd. Suite B
Sacramento, CA 95815
(916) 929-9732
FAX (916) 929-5608

Schweber Electronics
6750 Nancy Ridge Drive
San Diego, CA 92121
(619) 450-0454
TWX 910-335-1155

Schweber Electronics
90 E. Tasman Drive
San Jose, CA 95131
(408) 946-7171

Zeus Components
1130 Hawk Circle
Anaheim, CA 92807
(714) 632-6880
TWX 910-591-1696
FAX (714) 630-8770

Zeus Components
1580 Old Oakland Road
Suite C205/C206
San Jose, CA 95131
(408) 998-5121
TWX 408-628-96083
FAX (408) 998-0285

COLORADO
Integrated Electronics Corp.
5750 N. Logan Street
Denver, CO 80216
(303) 292-6121

Kierulff Electronics
7060 South Tuscon Way
Englewood, CO 80112
(303) 790-4444

Marshall Industries
7000 North Broadway
Denver, CO 80221
(303) 427-1818

Schweber Electronics
8955 E. Nicholas, Bldg. 2
Englewood, CO 80221
(303) 799-0258

CONNECTICUT
Greene-Shaw
1475 Whalley Avenue
New Haven, CT 06525
(203) 397-0710
TWX 92-2498

Marshall Industries
20 Sterling Drive
Barnes Industrial Park, N.
P.O. Box 200
Wallingford, CT 06492-0200
(203) 265-3822

Pioneer-Standard
112 Main Street
Norwalk, CT 06851
(203) 853-1515
TWX 710-468-3373
FAX (203) 838-9901

Schweber Electronics
Commercial Industrial Park
Finance Drive
Danbury, CT 06810
(203) 748-7080
TWX 710-456-9405

DELAWARE
See New Jersey
Pennsylvania

FLORIDA
All American Semiconductor
16251 N.W. 54th Avenue
Miami, FL 33014
(305) 621-8282
800-327-6237

Marshall Industries
4205 34th St., S.W.
Orlando, FL 32811
(305) 841-1878
(305) 841-1878

Pioneer Technologies Group
337 S. North Lake, #1000
Altamonte Springs, FL 32701
(305) 834-9090
TWX 810-853-0284

Pioneer Technologies Group
674 S. Military Trail
Deerfield Beach, FL 33441
(305) 428-8877
TWX 510-955-9653

Schweber Electronics
215 North Lake Blvd.
Altamonte Springs, FL 32701
(305) 331-7555
TWX 510-954-0304

Schweber Electronics
3665 Park Central Blvd. N.
Pompano Beach, FL 33064
(305) 977-7511
TWX 510-954-0304

Zeus Components
1750 West Broadway
Suite 114
Oviedo, FL 32765
(305) 365-3000
TWX 910-380-7430
FAX (305) 365-2356

GEORGIA
Dixie Electronics
1234 Gordon Park Road
Augusta, GA 30901
(404) 722-2055

Pioneer Technologies Group
3100 F. Northwoods Place
Norcross, GA 30071
(404) 448-1711
TWX 810-766-4515

Quality Components
6145 N. Belt Parkway #B
Norcross, GA 30071
(404) 449-9508
TWX 510-601-5297
629-32421

Schweber Electronics
303 Research Drive
Suite 210
Norcross, GA 30092
(404) 449-9170
TWX 810-766-1592

ILLINOIS
Advent Electronics
7110-16 N. Lyndon St.
Rosemont, IL 60018
(312) 297-6200

Kierulff Electronics
1140 W. Thorndale Ave.
Itasca, IL 60143
(312) 250-0500

Marshall Industries
1261 Wiley Road
#F
Schaumburg, IL 60195
(312) 490-0155

Pioneer-Standard
1551 Carmen Drive
Elk Grove Village, IL 60007
(312) 437-9680

Schweber Electronics
904 Cambridge Drive
Elk Grove Village, IL 60007
(312) 364-3750
TWX 910-222-3453

U.S. AND CANADIAN SALES NETWORK

DISTRIBUTORS (March 1987)

INDIANA
Advent Electronics
8446 Moller Road
Indianapolis, IN 46268
(317) 872-4910
TWX 810-341-3228

Marshall Industries
6990 Corporate Dr.
Indianapolis, IN 46278
(317) 297-0483

Pioneer-Standard
6408 Castleplace Drive
Indianapolis, IN 46250
(317) 849-7300
TWX 810-260-1794

IOWA
Advent Electronics
682 58th Avenue, Ct. SW
Cedar Rapids, IA 52404
(319) 363-0221
TWX 910-525-1337

Schweber Electronics
5270 North Park Place, NE
Cedar Rapids, IA 52402
(319) 373-1417

KANSAS
Marshall Industries
8321 Melrose Dr.
Lenexa, KS 66214
(913) 492-3121

Schweber Electronics
10300 West 103rd Street
Suite 200
Overland Park, KS 66214
(913) 492-2922

KENTUCKY
See Indiana

LOUISIANA
See Texas

MAINE
See Massachusetts

MARYLAND
Marshall Industries
8445 Helgerman Court
Gaithersburg, MD 20877
(301) 840-9450

Pioneer Technologies Group
9100 Gaither Road
Gaithersburg, MD 20877
(301) 921-0660
TWX 710-828-0545

Schweber Electronics
9330 Gaither Road
Gaithersburg, MD 20877
(301) 840-5900
TWX 710-828-9749

Zeus Components
8930-A Route 108
Columbia, MD 21045
(301) 997-1118
TWX 910-380-3554
FAX (301) 964-9784

MASSACHUSETTS
Greene-Shaw
70 Bridge Street
Newton, MA 02195
(617) 969-8900
TWX 92 2498

Kierulff Electronics
13 Fortune Drive
Billerica, MA 01821
(617) 667-8331

Lionex Corporation
36 Jonspin Road
Wilmington, MA 01887
(617) 657-5170
FAX (617) 657-6008

Marshall Industries
One Wilshire Road
Burlington, MA 01803
(617) 272-8200

Schweber Electronics
25 Wiggins Avenue
Bedford, MA 01730
(617) 275-5100
TWX 710-326-0268

Zeus Components
429 Marrett Road
Lexington, MA 02173
(617) 863-8800
TWX 710-326-7604
FAX (617) 863-8807

MICHIGAN
Advent Electronics
24713 Crestview Ct.
Farmington Hills, MI 48018
(313) 477-1650

Pioneer-Standard
4505 Broadmoor Avenue SE
Grand Rapids, MI 49508
(616) 698-1800
TWX 510-600-8456

Pioneer-Standard
13485 Stamford
Livonia, MI 48150
(313) 525-1800
TWX 810-242-3271

Schweber Electronics
12060 Hubbard Ave. CN3306
Livonia, MI 48150
(313) 525-8100
TWX 810-242-2983

MINNESOTA
Kierulff Electronics
7667 Cahill Road
Edina, MN 55435
(612) 941-7500

Marshall Industries
3800 Annapolis Lane
Plymouth, MN 55441
(612) 559-2211

Pioneer Standard
10203 Bren Road East
Minnetonka, MN 55343
(612) 935-5444
TWX 910-576-2738

Schweber Electronics
7424 W. 78th Street
Edina, MN 55435
(612) 941-5280
TWX 910-576-3167

MISSISSIPPI
See Texas

MISSOURI
Kierulff Electronics
11804 Borman Drive
St. Louis, MO 63146
(314) 997-4956
TWX 910-762-0721

Schweber Electronics
502 Earth City Expressway
Suite 203
Earth City, MO 63045
(314) 739-0526
TWX 43-4065

MONTANA
See California

NEBRASKA
See Iowa

NEW HAMPSHIRE
Schweber Electronics
Bedford Farms Bldg. #2
Manchester, NH 03102
(603) 625-2250
TWX 710-220-7572
FAX (603) 625-5710

NEW JERSEY
Kierulff Electronics
37 Kulick Road
Fairfield, NJ 07006
(201) 575-6750

Marshall Industries
101 Fairfield Rd.
Fairfield, NJ 07006
(201) 882-0320

Pioneer-Standard
45 Route 46
Pine Brook, NJ 07058
(201) 575-3510
TWX 710-734-4382

Schweber Electronics
18 Madison Road
Fairfield, NJ 07006
(201) 227-7880
TWX 710-734-4305

Solid State
46 Farrand Street
Bloomfield, NJ 07003
(201) 429-8700
TWX 710-994-4780
FAX (201) 429-8683

NEW YORK
Add Electronics
7 Adler Drive
E. Syracuse, NY 13057
(315) 437-0300

Nu-Horizons Electronics
6000 New Horizons Blvd.
N. Amityville, NY 11701
(516) 226-6000

Pioneer-Standard
840 Fairport Park
Fairport, NY 14450
(716) 381-7070
TWX 510-253-7001
FAX (716) 381-5955

Pioneer-Standard
1806 Vestal Pkwy. East
Vestal, NY 13850
(607) 748-8211
TWX 510-252-0893

Pioneer-Standard
Crossways Park West
Woodbury, NY 11797
(516) 921-8700
TWX 510-221-2184
FAX (516) 921-2143

Schweber Electronics
3 Townline Circle
Rochester, NY 14623
(716) 424-2222
TWX 710-541-0601

Schweber Electronics
Jericho Turnpike
Westbury, NY 11590
(516) 334-7474
TWX 510-220-1365

Zeus Components
100 Midland Avenue
Port Chester, NY 10573
(914) 937-7400
TWX 710-567-1248
FAX (914) 937-2553

NORTH CAROLINA
Dixie Electronics
2220 S. Tryon Street
Charlotte, NC 28234
(704) 377-4348

Dixie Electronics
1021 R. Burke St.
Winston-Salem, NC 27102
(919) 724-5961

Hammond Electronics
2923 Pacific Avenue
Greensboro, NC 27406
(919) 275-6391
TWX 628-94645

Pioneer Technologies Group
9801 A Southern Pine Blvd.
Charlotte, NC 28210
(704) 527-8188
TWX 810-621-0366

Quality Components, S.E.
2940-15 Trawick Road
Raleigh, NC 27604
(919) 876-7767

Schweber Electronics
5285 North Blvd.
Raleigh, NC 27604
(919) 876-0000
TWX 510-928-0531

NORTH DAKOTA
See Minnesota

OHIO
Kieruff Electronics
476 Windsor Park Drive
Dayton, OH 45459
(513) 439-0045

Marshall Industries
6212 Executive Blvd.
Dayton, OH 45424
(513) 236-8088

Marshall Industries
59058 Harper Road
Solon, OH 44139
(216) 248-1788

U.S. AND CANADIAN SALES NETWORK

DISTRIBUTORS (March 1987)

OHIO (cont.)
Pioneer-Standard
4800 East 131st Street
Cleveland, OH 44105
(216) 587-3600
TWX 810-421-0011

Pioneer-Standard
4433 Interpoint Blvd.
Dayton, OH 45424
(513) 236-9900
TWX 810-459-1622

Schweber Electronics
23880 Commerce Park Rd.
Beachwood, OH 44122
(216) 464-2970
TWX 810-427-9441

Schweber Electronics
7865 Paragon Road
Suite 210
Dayton, OH 45459
(513) 439-1800

Zeus (Televox)
2593 Lance Drive
Dayton, OH 45409
(513) 294-4499
TWX 75-9251
FAX (513) 294-6620

OKLAHOMA
Quality Components
9934 East 21st South
Tulsa, OK 74129
(918) 664-8812
TWX 910-860-5459
629-28599

Schweber Electronics
4815 South Sheridan
Fountain Plaza, Suite 109
Tulsa, OK 74145
(918) 622-8000

OREGON
Almac Electronics Corp.
1885 N.W. 169th Place
Beaverton, OR 98006
(503) 629-8090
FAX (503) 645-0611
TWX 910-467-8743

Kierulff Electronics
14273 N.W. Science Park Drive
Portland, OR 97229
(503) 641-9150

Marshall Industries
8333 S.W. Cirrus Dr.
Beaverton, OR 97005
(503) 644-5050

PENNSYLVANIA
Almo Electronics, Inc.
9815 Roosevelt Blvd.
Philadelphia, PA 19114
(215) 698-4063
TLX 476-1218
FAX (215) 969-6768

Pioneer-Standard
259 Kappa Drive
Pittsburgh, PA 15238
(412) 782-2300
TWX 710- 795-3122

Pioneer Technologies Group
261 Gibraltar Road
Horsham, PA 19044
(215) 674-4000
TWX 510-665-6778

Schweber Electronics
900 Business Center Dr.
Horsham, PA 19044
(215) 441-0600
TWX 510-665-6540

Schweber Electronics
1000 R.I.D.C. Plaza
Suite 203
Pittsburgh, PA 15238
(412) 782-1600
TWX 810-427-9441

RHODE ISLAND
See Massachusetts
New York

SOUTH CAROLINA
Dixie Electronics
1900 Barnwell Street
Columbia, SC 29202
(803) 779-5332
TLX 810-666-2620
FAX (803) 765-9276

Dixie Electronics
531 E. Palmetto Street
Florence, SC 29503
(803) 669-8201

Dixie Electronics
4909 Pelham Rd.
Greenville, SC 29606
(803) 297-1435

Dixie Electronics
#6 Pepperhill Square
7525 Brandywine Road
N. Charleston, SC 29410
(803) 552-2671

SOUTH DAKOTA
See Minnesota

TENNESSEE
Dixie Electronics
Box 8215 Suncrest Drive
Gray, TN 37615
(615) 477-3838

Dixie Electronics
6408 Clinton Highway
Knoxville, TN 27912
(615) 938-4131

TEXAS
Kierulff Electronics
9610 Skillman Ave.
Dallas, TX 75243
(214) 343-2400

Marshall Industries
2045 Chenault St.
Carrollton, TX 75006
(214) 233-5200
FAX (214) 770-0675

Pioneer-Standard
13710 Omega Road
Dallas, TX 75234
(214) 386-7300
TWX 910-860-5563

Pioneer-Standard
9901 Burnet Road
Austin, TX 78758
(512) 835-4000
TWX 910-874-1323

Pioneer-Standard
5853 Point West Drive
Houston, TX 77036
(713) 988-5555
TWX 910-881-1606

Quality Components
4257 Kellway Circle
Addison, TX 75001
(214) 733-4300
TWX 910-860-5459

Quality Components
1005 Industrial Blvd.
Sugarland, TX 77478
(713) 240-2255
TWX 629 27026

Quality Components
2120 M. Braker Lane
Austin, TX 78758
(512) 835-0220
TWX 324930

Schweber Electronics
4202 Beltway Drive
Dallas, TX 75234
(214) 661-5010
TWX 910-860-5493

Schweber Electronics
6300 La Calma Drive
Suite 240
Austin, TX 78752
(512) 458-8253
TWX 910-874-2045

Schweber Electronics
10625 Richmond, Suite 100
Houston, TX 77042
(713) 784-3600
TWX 910-881-4836

Zeus Components
1800 North Glenville
Suite 120
Richardson, TX 75081
(214) 783-7010
TWX 910-867-9422
FAX (214) 234-4385

UTAH
Integrated Electronics Corp.
101 N. 700 West
N. Salt Lake City, UT 84054
(801) 298-1869

Kierulff Electronics
1846 Parkway Blvd.
Salt Lake City, UT 84119
(801) 973-6913

Marshall Industries
3501 South Main St.
Salt Lake City, UT 84115
(801) 261-0901

VERMONT
See New York

VIRGINIA
See Maryland

WASHINGTON
Almac Electronics Corp.
14360 S.E. Eastgate Way
Bellevue, WA 98007
(206) 643-9992
TWX 910-444-2067
FAX (206) 746-7425

Almac Electronics Corp.
East 10905 Montgomery
Spokane, WA 99206
(509) 924-9500
TWX 510-773-1855
FAX (509) 928-6096

Kierulff Electronics
19450 68th Ave.
South Kent, WA 98032
(206) 575-4420

Marshall Industries
14102 N.E. 21st St.
Bellevue, WA 98007
(206) 747-9100

WASHINGTON D.C.
See Maryland

WEST VIRGINIA
See Ohio
Pennsylvania
Maryland

WISCONSIN
Kierulff Electronics
2238-E West Bluemound Road
Waukesha, WI 53186
(414) 784-8160

Marshall Industries
235 North Executive Dr.
#305
Brookfield, WI 53005
(414) 797-8400

Schweber Electronics
3050 South Calhoun Rd.
New Berlin, WI 53151
(414) 784-9020

WYOMING
See Oregon
Washington

CANADA
R.A.E. Industrial
3455 Gardner Court
Burnaby, B.C.
(604) 291-8866
TWX 610-929-3065

R.A.E. Industrial
11680 170th Street
Edmonton, Alberta
T5S 1J7
(403) 451-4001
TWX 037-2653

Zentronics
8 Tilbury Court
Brampton, Ontario
L6T 3T4
(416) 451-9600
TWX 06-97678
FAX (416) 451-8320

Zentronics
3300-14 Ave., NE Bay #1
Calgary, Alberta
T2A 6J4
(403) 272-1021

Zentronics
155 Colonnade, S. #17/18
Nepean, Ontario
K2E 7K1
(613) 226-8840
TWX 06-97698

U.S. AND CANADIAN SALES NETWORK

DISTRIBUTORS (March 1987)

CANADA (cont.)
Zentronics
11400 Bridgeport Rd. #108
Richmond, B.C.
V6X 1T2
(604) 273-5575
TWX 04-355844

Zentronics
817 McCaffrey Street
St. Laurent, Quebec
H4T 1N3
(514) 737-9700
TWX 05-824826

Zentronics
564 Weber Street, N. #10
Waterloo, Ontario
N21 5C6
(519) 884-5700
TWX 06-97678

Zentronics
590 Berry Street
Winnipeg, Manitoba
R3H 0S1
(204) 775-8661
TWX 06-97678

Future Electronics
3220 5th Avenue, N.E.
Calgary, Alberta
T2A 5N1
(403) 235-5325

Future Electronics
82 St. Regis Crescent N.
Downsview, Ontario
M3J 1Z3
(416) 638-4771
TWX 610-491-1470
FAX (416) 638-2936

Future Electronics
5312 Calgary Trail South
Edmonton, Alberta
T6H 4J8
(403) 438-2858

Future Electronics
Hymus Blvd.
Pointe Claire
Montreal, Quebec
H9R 5C7
(514) 694-7710
TWX 610-421-3251 or
610-421-3500
FAX (514) 695-3707 or
(514) 694-0062

Future Electronics
Baxter Center
1050 Baxter Road
Ottawa, Ontario
K2C 3P2
(613) 820-8313
TWX 610-563-1697
FAX (613) 820-3271

Future Electronics
1695 Boundary Road
Vancouver, B.C.
B5K 4X7
(604) 294-1166
TLX 04354744
FAX (604) 294-1206

Future Electronics
444 Sharon Bay
Winnipeg, Manitoba
R2G 0H7
(604) 294-1166 (Vancouver)

THOMSON COMPOSANTS EXPORT SALES HEADQUARTERS :
101, BD MURAT - 75781 PARIS CEDEX 16 - FRANCE
TEL. (33 1) 47.43.96.40 / TELEX : TCEX 648782 F

SALES NETWORK

REPRESENTATIVES (March 1987)

ARGENTINE

STE CORTE & CIA, S.R.L.
Santiago del Estero
643 4° Piso
1075 BUENOS AIRES
Tel. (54 1) 38.00.10 7/8/9 Telex : 21992 CORTE

INDE

MICRONIC DEVICES
«Arun Complex»
N° 65, DVG road Basavanagudi
BANGALORE 560 004
Tel. (91 812) 600.631 Telex : 8458332

PERMEDIA ASSOCIATES
14 Dream Land House 1st Floor
1/18 - B, Asaf Ali Road
NEW DELHI 110 002
Tel. (91 11) 73.23.47 - (91 11) 73.38.39
Telex : 03162136 PAAS IN

MEXIQUE

CYDESA
Prol de Jade 2 Col. Estrella
C.P. 07810
MEXICO D.F.
Tel. (52 5) 537.38.32 - (52 5) 577.18.83 Telex : 1773944

INTERTRON
Cerrada de Palomas N° 7
Col Reforma Social
11650 MEXICO D.F.
Tel. (52 5) 540.07.15 - (52 5) 540.29.69 Telex : 1763254

PEROU

INTERSA
Raimundo Carcano 995
Urb. Santa Catalina
LIMA 13
Tel. (51 14) 724.921 Telex : 25257 PE PB SIS

R.S.A.

PACE ELECTRONICS
Components PTY
P.O. Box 701
ISANDO 1600 TRANSVAAL R.S.A
Tel. (27) 11.974.1211 (6) Telex : 429023

TURQUIE

BARKEY SANAYI MALZEMELERI
Temsilcilik ltd STI
P.K. 667 SISLI ISTAMBUL
Tel. (90 11) 47.97.40 - (90 11) 48.91.47 - (90 11) 40.14.83
Telex : 26900 NED TR

URUGUAY

MURACCIOLE LTDA
Colonia 844 Piso 3
Casilla 430
MONTEVIDEO
Tel. (59 8) 90.08.32 - (59 8) 90.21.37 Telex : 23079 UY TENIP

SALES NETWORK

DISTRIBUTORS (March 1987)

ARGENTINE

ELECTRONICA ELEMON S.A.
Monroe 4566
1431 BUENOS AIRES
Tel. (54 1) 51.42.06 - (54 1) 52.73.35 Telex : 22204

ELKO S.R.L.
Belgrano 1661
1093 BUENOS AIRES
Tel. (54 1) 45.35.59 - (54 1) 45.11.01 Telex : 17561 ELKO

CHILI

VICTRONICS
Casilla 283 V Correo 21
SANTIAGO 01
Tel. (56 2) 392.787 - (56 2) 330.232 Telex : 340168 VICTORCK

CASA ROYAL
Libertador B. O'Higgins 845
SANTIAGO
Tel. (56 2) 39.15.24 - (56 2) 38.10.37 Telex : 340517 ROYAL CK

INDE

THOMSON CSF/COMPOSANTS EXPORT
(délégation)
C578 Defence Colony
NEW DELHI 110024
Tel. (91 11) 62.55.98 - (91 11) 61.15.98
Telex : 3166642 TCSF IN

MEXIQUE

PROVEEDORA ELECTRONICA
Prol Moctezuma Ote 24
Col. Romero de Terreros
P.O. Box 21139
04310 MEXICO D.F.
Tel. (52 5) 554.83.00 Telex : 01772402 PROVME

DICOPEL S.A.
Tochtli 368 Fracc
Ind San Antonio
Azcapotzalco
02760 MEXICO DF
Tel. (52 5) 561.32.11 - Telex : 1773790 DICOMF

ELECTRONICA STEREN S.A.
Rep. de El Salvador N° 28 A y B
06000 MEXICO D.F.
Tel. (52 5) 512 06 99 - (52 5) 512 03 20
Telex : 01771977 STERNE

SEMICONDUCTORES PROFESIONALES
Insurgentes sur 1748 403
01030 MEXICO D.F.
Tel. (52 5) 534.17.42 - (52 5) 534.79.80
Telex : 1771700 BRASME

URUGUAY

ENEKA
Avenida Gral Rondeau 1534
P.O. Box 695
MONTEVIDEO
Tel. (59 8) 90.79.44 - (59 8) 90.79.96 Telex : 22270

VENEZUELA

BENAVIDES
Avilanes A Rio
Residencias Kamarata
Local 4 al 7
Apartado postal 20249
La Gandelaria
CARACAS 1020 A
Tel. (56 2) 571.03.96 - (56 2) 571.27.10 Telex : 28450 PBVENVC

LOGIBYTE
Av. Civdad Universitaria
Edif. Los Moriches
Apartado Los Chaguaranos 47776
CARACAS 1041 A
Tel. (56 2) 662.19.80 - (56 2) 662.38.41 Telex : 24581 LOGIB VC

CREDITS
Typesetting
Technical artwork and lay-out
Technical support and supervision
SEDOC

PRINTED IN FRANCE BY
SEDOC

Dépot légal : 2eme trimestre 1987

THOMSON
COMPONENTS
SEMICONDUCTEURS

43, AVENUE DE L'EUROPE 78140 VELIZY-VILLACOUBLAY / FRANCE
TEL. : (1) 39.46.97.19 / TELEX : TCSF 204 780 F THSC VELIZY / TELECOPIE : (1) 39.46.52.64
SALES NETWORK INSIDE

Ref. DBMCU - 120FFTTC