82C206
Integrated Peripheral Controller

Data Book

OPTi Confidential

Revision: 0.1 (2ND DRAFT)
912-3000-026
August 9, 1996 (2ND DRAFT)

Disclaimer

OPTi Inc. makes no representations or warranties with respect to the design and documentation herein described and especially disclaims any implied warranties of merchantability or fitness for any particular purpose. Further, OPTi Inc. reserves the right to revise the design and associated documentation and to make changes from time to time in the content without obligation of OPTi Inc. to notify any person of such revisions or changes.

Note: Before designing contact OPTi for latest Product Alerts, Applications Notes, and Errata for this product line.

OPTi Inc.
888 Tasman Drive
Milpitas, CA 95035
Tel: (408) 486-8000
Fax: (408) 486-8001
WWW: http://www.opti.com/

Table of Contents

Table of Contents (cont.)

Table of Contents (cont.)

Table of Contents (cont.)

List of Figures

List of Figures (cont.)

List of Tables

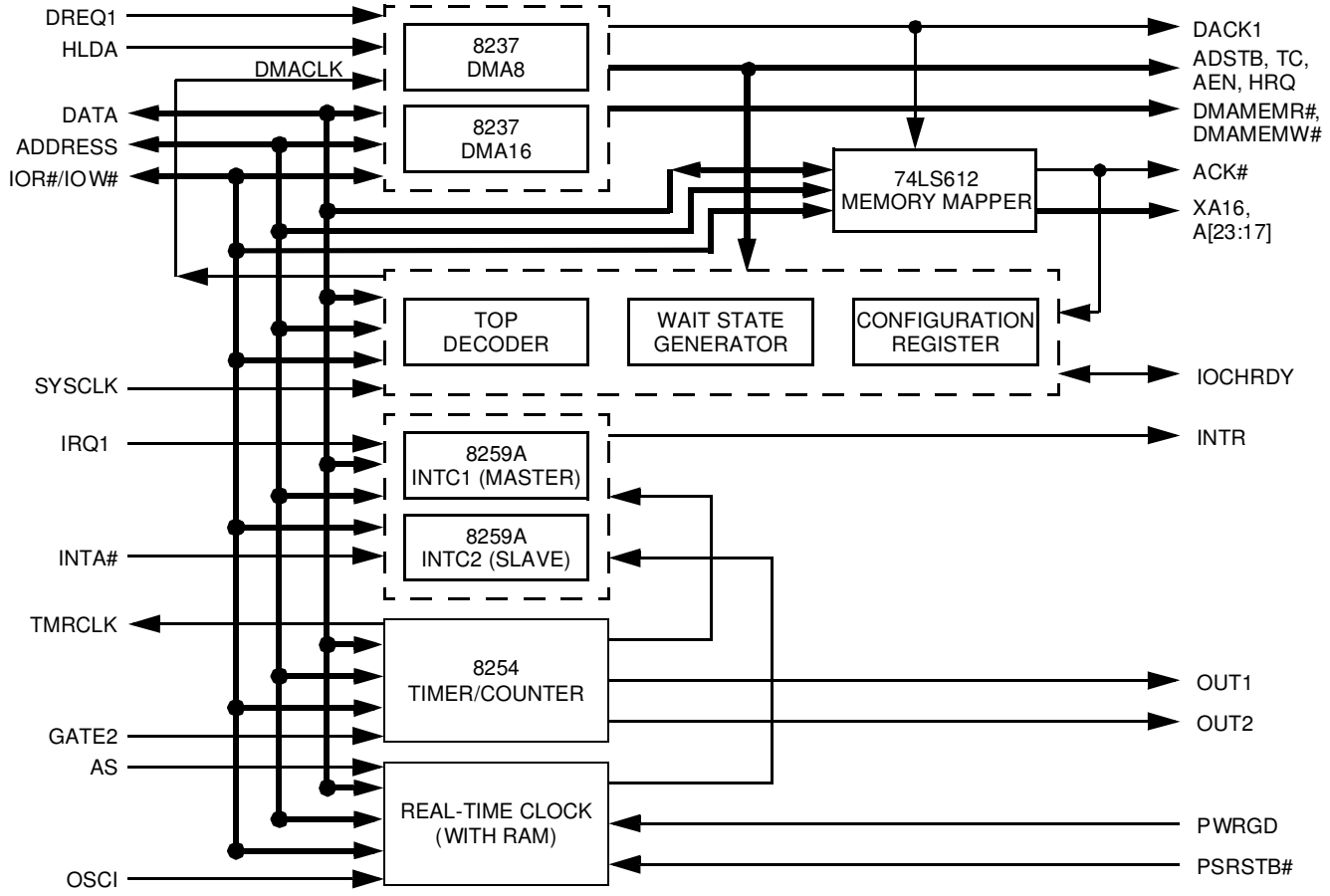
List of Tables (cont.)

Integrated Peripheral Controller

1.0 Features

- Fully compatible with PC/AT-based systems
- Includes:
  - two 8237 DMA controllers
  - two 8259A interrupt controllers
  - one 8254 timer/counter
  - one 146818A-compatible real-time clock
  - an additional 64 bytes of CMOS RAM
  - one 74LS612 memory mapper
- Provides:
  - Seven DMA channels
  - 13 interrupt request channels
  - two timer/counter channels
- Four DMA transfer modes supported:
  - Single
  - Block
  - Demand
  - Cascade
- Special Commands provided for ease of programming:
  - Clear byte pointer flip-flop
  - Set byte pointer flip-flop
  - Master clear
  - Clear request mask register
  - Clear mode register counter
- Contains 114 bytes of CMOS RAM
- 8MHz DMA clock with programmable internal divider for 4MHz operation
- 16MB DMA address space
- Programmable wait states for DMA cycles
- Reduced recovery time (120ns) between I/O operations
- 84-pin PLCC (plastic leadless chip carrier) or 100-pin QFP (quad flat pack)

Figure 1-1 82C206 Block Diagram



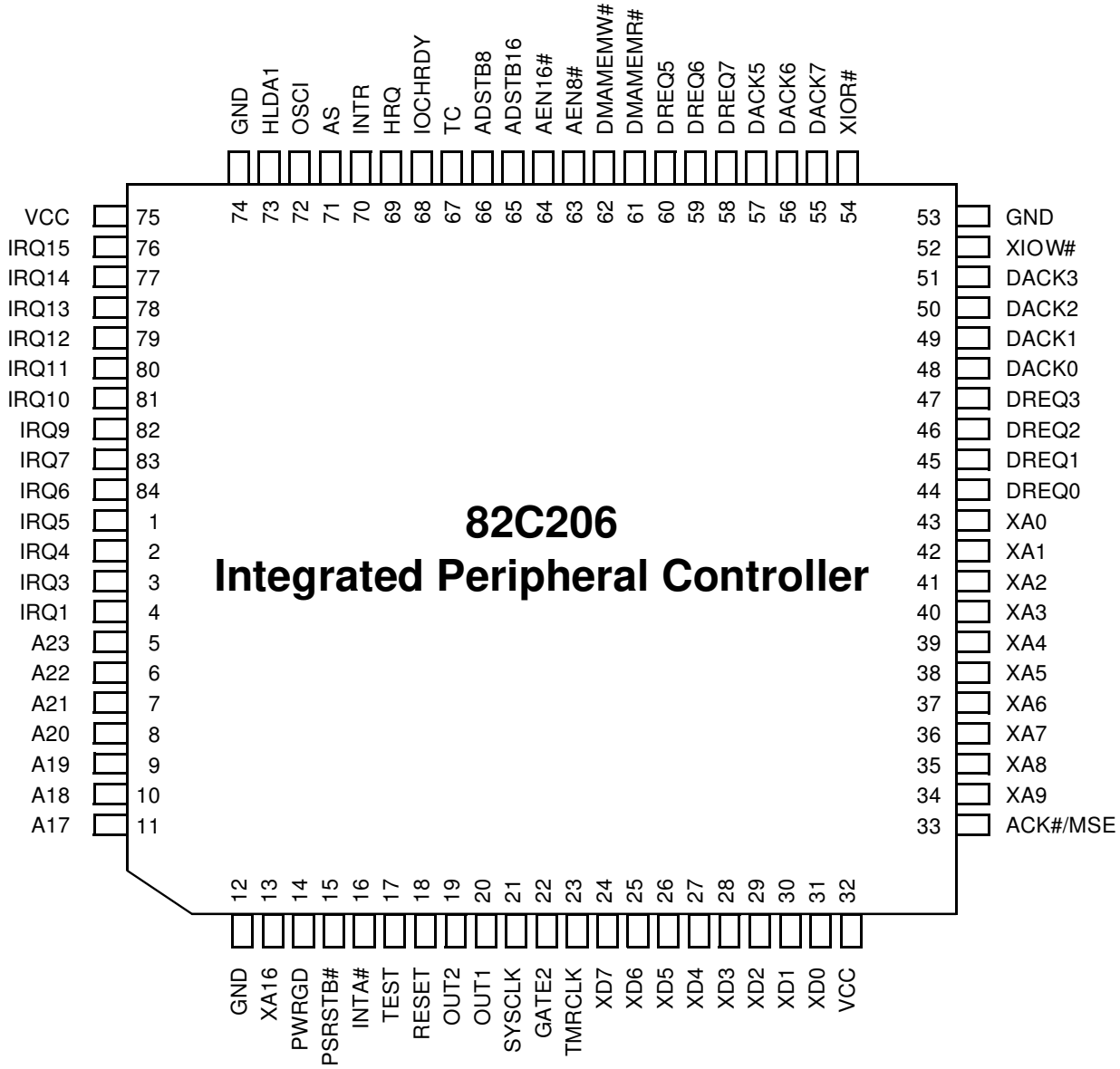
2.0 Overview

OPTi's 82C206 Integrated Peripheral Controller (IPC) is a single-chip integration of all the main peripheral parts attached to the X bus of PC/AT™ architecture. The 82C206 replaces the following peripheral logic on motherboards: two 8237 DMA controllers, two 8259A interrupt controllers, one 8254 timer/counter, one 146818A-compatible real-time clock, and one 74LS612 memory mapper.

The 82C206 also offers several enhanced features and improved speed performance. These include an additional 64 bytes of user definable CMOS RAM in the real-time clock and drastically reduces the recovery time for the 8237, 8259A, and 8254. Programmable wait state operation is provided for DMA and CPU I/O cycles accessing the chip. The 82C206 also provides programmable 4MHz or 8MHz DMA clock selection. The 82C206 is packaged in an 84-pin PLCC (plastic leaded chip carrier) or a 100-pin PQFP (plastic quad flat pack).

3.0 Signal Definitions

Figure 3-1 84-Pin PLCC Pin Diagram



3.1 Pin Cross-Reference Lists

Table 3-1 84-Pin PLCC Numerical-Cross Reference List

Pin No.	Pin Name	Pin No.	Pin Name	Pin No.	Pin Name
1	IRQ5	29	XD2	57	DACK5
2	IRQ4	30	XD1	58	DREQ7
3	IRQ3	31	XD0	59	DREQ6
4	IRQ1	32	VCC	60	DREQ5
5	A23	33	ACK#/MSE	61	DMAMEMR#
6	A22	34	XA9	62	DMAMEMW#
7	A21	35	XA8	63	AEN8#
8	A20	36	XA7	64	AEN16#
9	A19	37	XA6	65	ADSTB16
10	A18	38	XA5	66	ADSTB8
11	A17	39	XA4	67	TC
12	GND	40	XA3	68	IOCHRDY
13	XA16	41	XA2	69	HRQ
14	PWRGD	42	XA1	70	INTR
15	PSRSTB#	43	XA0	71	AS
16	INTA#	44	DREQ0	72	OSCI
17	TEST	45	DREQ1	73	HLDA
18	RESET	46	DREQ2	74	GND
19	OUT2	47	DREQ3	75	VCC
20	OUT1	48	DACK0	76	IRQ15
21	SYSCLK	49	DACK1	77	IRQ14
22	GATE2	50	DACK2	78	IRQ13
23	TMRCLK	51	DACK3	79	IRQ12
24	XD7	52	XIOW#	80	IRQ11
25	XD6	53	GND	81	IRQ10
26	XD5	54	XIOR#	82	IRQ9
27	XD4	55	DACK7	83	IRQ7
28	XD3	56	DACK6	84	IRQ6

Table 3-2 84-Pin PLCC Alphabetical Cross-Reference List

Pin Name	Pin Number	Pin Name	Pin Number	Pin Name	Pin Number
A17	11	DREQ7	58	RESET	18
A18	10	GATE2	22	SYSCLK	21
A19	9	GND	12	TEST	17
A20	8	GND	53	TC	67
A21	7	GND	74	TMRCLK	23
A22	6	HLDA	73	VCC	32
A23	5	HRQ	69	VCC	75
ACK#/MSE	33	INTA#	16	XA0	43
ADSTB8	66	INTR	70	XA1	42
ADSTB16	65	IOCHRDY	68	XA2	41
AEN8#	63	IRQ1	4	XA3	40
AEN16#	64	IRQ3	3	XA4	39
AS	71	IRQ4	2	XA5	38
DACK0	48	IRQ5	1	XA6	37
DACK1	49	IRQ6	84	XA7	36
DACK2	50	IRQ7	83	XA8	35
DACK3	51	IRQ9	82	XA9	34
DACK5	57	IRQ10	81	XA16	13
DACK6	56	IRQ11	80	XD0	31
DACK7	55	IRQ12	79	XD1	30
DMAMEMR#	61	IRQ13	78	XD2	29
DMAMEMW#	62	IRQ14	77	XD3	28
DREQ0	44	IRQ15	76	XD4	27
DREQ1	45	PSRSTB#	15	XD5	26
DREQ2	46	PWRGD	14	XD6	25
DREQ3	47	OSCI	72	XD7	24
DREQ5	60	OUT1	20	XIOR#	54
DREQ6	59	OUT2	19	XIOW#	52

Figure 3-2 100-Pin PQFP Pin Diagram

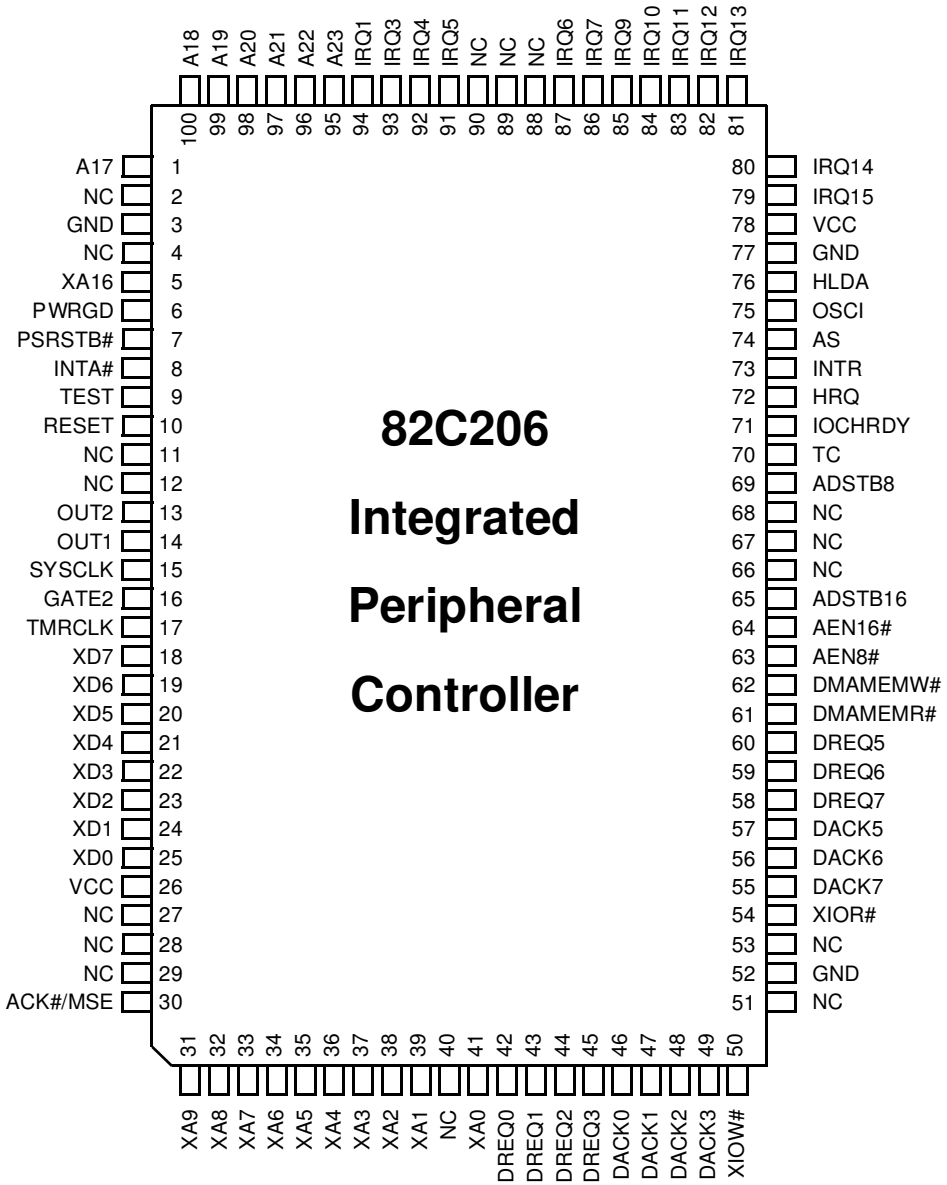


Table 3-3 100-Pin QFP Numerical Cross-Reference Lists

Pin No.	Pin Name	Pin No.	Pin Name	Pin No.	Pin Name	Pin No.	Pin Name
1	A17	26	VCC	51	NC	76	HLDA
2	NC	27	NC	52	GND	77	GND
3	GND	28	NC	53	NC	78	VCC
4	NC	29	NC	54	XIOR#	79	IRQ15
5	XA16	30	ACK#/MSE	55	DACK7	80	IRQ14
6	PWRGD	31	XA9	56	DACK6	81	IRQ13
7	PSRSTB#	32	XA8	57	DACK5	82	IRQ12
8	INTA#	33	XA7	58	DREQ7	83	IRQ11
9	TEST	34	XA6	59	DREQ6	84	IRQ10
10	RESET	35	XA5	60	DREQ5	85	IRQ9
11	NC	36	XA4	61	DMAMEMR#	86	IRQ7
12	NC	37	XA3	62	DMAMEMW#	87	IRQ6
13	OUT2	38	XA2	63	AEN8#	88	NC
14	OUT1	39	XA1	64	AEN16#	89	NC
15	SYSCLK	40	NC	65	ADSTB16	90	NC
16	GATE2	41	XA0	66	NC	91	IRQ5
17	TMRCLK	42	DREQ0	67	NC	92	IRQ4
18	XD7	43	DREQ1	68	NC	93	IRQ3
19	XD6	44	DREQ2	69	ADSTB8	94	IRQ1
20	XD5	45	DREQ3	70	TC	95	A23
21	XD4	46	DACK0	71	IOCHRDY	96	A22
22	XD3	47	DACK1	72	HRQ	97	A21
23	XD2	48	DACK2	73	INTR	98	A20
24	XD1	49	DACK3	74	AS	99	A19
25	XD0	50	XIOW#	75	OSCI	100	A18

Table 3-4 100-Pin QFP Alphabetical Cross-Reference List

Pin Name	Pin No.	Pin Name	Pin No.	Pin Name	Pin No.	Pin Name	Pin No.
A17	1	DREQ3	45	IRQ15	79	TEST	9
A18	100	DREQ5	60	NC	2	TMRCLK	17
A19	99	DREQ6	59	NC	4	VCC	26
A20	98	DREQ7	58	NC	11	VCC	78
A21	97	GATE2	16	NC	12	XA0	41
A22	96	GND	3	NC	27	XA1	39
A23	95	GND	52	NC	28	XA2	38
ACK#/MSE	30	GND	77	NC	29	XA3	37
ADSTB8	69	HLDA	76	NC	40	XA4	36
ADSTB16	65	HRQ	72	NC	51	XA5	35
AEN8#	63	INTA#	8	NC	53	XA6	34
AEN16#	64	INTR	73	NC	66	XA7	33
AS	74	IOCHRDY	71	NC	67	XA8	32
DACK0	46	IRQ1	94	NC	68	XA9	31
DACK1	47	IRQ3	93	NC	88	XA16	5
DACK2	48	IRQ4	92	NC	89	XD0	25
DACK3	49	IRQ5	91	NC	90	XD1	24
DACK5	57	IRQ6	87	OSCI	75	XD2	23
DACK6	56	IRQ7	86	OUT1	14	XD3	22
DACK7	55	IRQ9	85	OUT2	13	XD4	21
DMAMEMR#	61	IRQ10	84	PSRSTB#	7	XD5	20
DMAMEMW#	62	IRQ11	83	PWRGD	6	XD6	19
DREQ0	42	IRQ12	82	RESET	10	XD7	18
DREQ1	43	IRQ13	81	SYSCLK	15	XIOR#	54
DREQ2	44	IRQ14	80	TC	70	XIOW#	50

3.2 Signal Descriptions

Signal Name	Pin No. 84 PLCC	Pin No. 100 PQFP	Signal Type	Signal Description
3.2.1 Clock & Control Interface Signals				
SYSCLK	21	15	I	Clock: This input is used to generate the timing signals for DMA operation and can be driven to a 10MHz frequency. The internal programmable clock used for DMA cycles is either SYSCLK or SYSCLK/2.
OSCI	72	75	I	Oscillator Input: Used to generate the time-base for the real-time clock function. An external square waves of 32.768kHz may be connected to this pin.
RESET	18	10	I	Reset: This active high input affects the following registers: DMA Controller: Clears the command, status, request, temporary registers, and the byte pointer flip-flop. Sets the mask register. After RESET, the DMA controller is in the idle state. Interrupt Controller: Clears the edge sense circuit, mask registers, and all ICW4 functions. IRQ0 is assigned the highest priority. The slave address is set to 7. The Special Mask Mode is disabled and status read is set to IRR.
IOCHRDY	68	71	I/O	I/O Channel Ready: When an input, this signal is used to extend the memory read or write pulses for the DMA controller to access slow memories or I/O devices. It must satisfy setup and hold times with respect to the DMA internal clock in order to work reliably. A low on IOCHRDY causes the internal DMA READY signal to go low asynchronously. When IOCHRDY goes high, one DMA clock cycle will elapse before the internal DMA READY signal goes high. When an output, this signal is an open drain output and provides an active low output whenever an 82C206 internal register is accessed. It will remain low for a pre-programmed number of DMA internal clock cycles (as controlled by bits 7 and 6 of the 82C206 Configuration Register) and then goes high. In this way, IOCHRDY can insert wait states (as counted by the DMA internal clock cycles) when the CPU accesses the 82C206's internal registers. This pin must be pulled up by an external resistor. In a PC/AT-based design, this pin should be wire-OR'd to the PC/AT's IOCHRDY signal.

Signal Name	Pin No. 84 PLCC	Pin No. 100 PQFP	Signal Type	Signal Description
XD[7:0]	24:31	18:25	I/O	X Data Bus bits 7 through 0: These tristate bidirectional pins are directly connected to the XD bus in a PC/AT-based design. During CPU I/O read cycles, these output pins are used to read out the contents of the 82C206's internal registers. During CPU I/O write cycles, they are input pins that allow the CPU to program the contents of the 82C206's internal registers. During DMA cycles, the most significant eight bits of the address are output on to these pins to be strobed into an external latch via pins ADSTB8 or ADSTB16. During DMA memory-to-memory transfers, data from the memory comes into the DMA controller via the XD data bus and stores in the internal temporary register during read from the memory partial cycle. In the write to memory partial cycle, the data stored in the temporary register will output via the XD data bus and write into the new memory location. During an interrupt acknowledge cycle, the interrupt controllers output the interrupt vector byte via through the XD data bus. The XD data bus pins are also used as the multiplexed address/data bus for the real-time clock (RTC) and CMOS RAM accesses.
XA[9:0]	34, 35:43	31:39, 41	I, I/O	X Address Bus bits 9 through 0: These pins are directly connected to the XA bus in PC/AT-based designs. XA[8:0] are bidirectional pins and XA9 is an input only pin. During CPU I/O accesses to the 82C206, XA[9:0] are used for addressing the Configuration Register and the internal registers of the 8237, 8259A, 8254, RTC, CMOS RAM, and 74LS612. During a CPU cycle, XA[3:0] are used by the CPU to address the registers of the DMA controller corresponding to DMA Channels 3 through 0, while XA[4:1] are used to address Channels 7 through 5. During a DMA cycle, XA[7:0] are outputs and carry address information for DMA Channels 3 through 0 and XA[8:1] for Channels 7 through 5.
XIOR#	54	54	I/O	X I/O Read - A bidirectional active low tristate pin used as a control signal. In a non-DMA or non-interrupt cycle, XIOR# is an input control signal used by the CPU to read the 82C206's internal registers. In an active DMA cycle, it is an output control signal used by the DMA controller to access data from a peripheral during a DMA write memory transfer.
XIOW#	52	50	I/O	X I/O Write - A bidirectional active low tristate pin used as a control signal. In a non-DMA or non-interrupt cycle, XIOW# is an input control signal used by the CPU to read the 82C206's internal registers. In an active DMA cycle, it is an output control signal used by the DMA controller to write data from a peripheral during a DMA read memory transfer.
DMAMEMR#	61	61	O	DMA Memory Read - An active low tristate output pin used to access data from a selected memory location during DMA memory read or memory-to-memory transfer cycles.

Signal Name	Pin No. 84 PLCC	Pin No. 100 PQFP	Signal Type	Signal Description
DMAMEMW#	62	62	O	DMA Memory Write - An active low tristate output pin used to write data from a selected memory location during DMA memory write or memory-to-memory transfer cycles.
HLDA	73	76	I	Hold Acknowledge - An active high signal from the System Controller to indicate that the CPU has relinquished control of the system bus.
HRQ	69	72	O	Hold Request: This active high output signal to the System Controller is used to request control of the system bus. When a DREQ occurs and the corresponding mask bit is clear, or a software DMA request is made, the DMA controller issues HRQ to the System Controller. After the CPU releases the system bus, the System Controller then issues a HLDA1 back to the 82C206 if DMA has been permitted to control the system bus.
DREQ[7:5], DREQ[3:0]	58:60 47:44	58:60, 45:42	I	DMA Request bits 7 through 5 and 3 through 0: These pins are asynchronous DMA channel request inputs for each DMA channel. In fixed priority, DREQ0 has the highest priority and DREQ7 has the lowest. A peripheral device will activate a DREQ line if it needs a DMA service. DACK will acknowledge the recognition of the DREQ request. DREQ must be sustained until the corresponding DACK goes active. DREQ will not be recognized while the DMA clock is stopped. Unused DREQ inputs should be inactive and the corresponding mask bit should be set to avoid an undesired DMA function. The polarity of DREQ is programmable. RESET initializes these lines to be active high. DREQ[3:0] support 8-bit transfers between 8-bit I/O devices and 8- or 16-bit system memory. DREQ[7:5] support 16-bit transfers between 16-bit I/O devices and 16-bit system memory. DREQ4 is not externally available and is used to cascade DREQ[3:0].
TC	67	70	O	Terminal Count: An active high signal that indicates the completion of DMA services. A pulse is generated by the DMA controller when the terminal count for any channel is reached except for Channel 0 in memory-to-memory transfer cycles. During a memory-to-memory transfer, TC will be generated when the terminal count for Channel 1 occurs. When a TC pulse occurs, the DMA controller terminates the service and if auto-initialize is enabled, the base registers will be written to the current registers of that channel. The mask bit and TC bit in the Status Register will be set for the currently active channel unless the channel is programmed for auto-initialization. In this case, the mask bit remains clear.
DACK[7:5], DACK[3:0]	55:57 51:48	55:57, 49:46	O	DMA Acknowledge bits 7 through 5 and 3 through 0: These bits are used to notify the individual peripherals when one has been granted a DMA cycle. Because these signals are used internally for cascading the DMA channels and for DMA page register selection, they must be programmed active low and cannot be changed. A reset initializes them to become active low.

Signal Name	Pin No. 84 PLCC	Pin No. 100 PQFP	Signal Type	Signal Description
ADSTB8	66	69	O	Address Strobe 8: An active high output signal used to latch the upper address bytes XA[15:8] for 8-bit peripheral devices. During DMA block transfers, ADSTB8 will only be issued when the upper address byte must be updated, thus speeding up transfer through elimination of S1 states of DMA cycles. ADSTB8 is active for DMA Channels 3 through 0.
ADSTB16	65	65	O	Address Strobe 16: An active high output signal used to latch the upper address bytes XA[16:9] for 16-bit peripheral devices. During DMA block transfers, ADSTB16 will only be issued when the upper address byte must be updated, thus speeding up transfer through elimination of S1 states of DMA cycles. ADSTB16 is active for DMA Channels 7 through 5
AEN8#	63	63	O	Address Enable 8: This active low output is used to enable the latch of the upper address byte XA[15:8] for 8-bit peripheral devices. It is inactive when an external bus master controls the system bus. AEN8# is active for DMA Channels 3 through 0.
AEN16#	64	64	O	Address Enable 16: This active low output is used to enable the latch of the upper address byte XA[16:9] for 16-bit peripheral devices. It is inactive when an external bus master controls the system bus. AEN16# is active for DMA Channels 7 through 5.
ACK#/MSE	33	30	I	Acknowledge or Module Select Enable: This pin is a dual function input signal. When high, it enables the chip selection function on one of the modules of the 82C206 for the CPU programming functions. When low, the 82C206 is essentially disconnected from the system bus and is capable of performing an active DMA or an interrupt cycle. In a PC/AT-based architecture design, it is tied to the ACK# signal of the main board.
A[23:17]	11:5	95:100, 1	O	Address Bus bits 23 through 17: These are tristate output pins and are the upper seven bits of the DMA page register.
XA16	13	5	O	X Address Bus bit 16 - This tristate output pin is the least significant bit (LSB) of the DMA page register and is used for DMA transfers for 8-bit peripheral devices only. XA16 is not used for 16-bit DMA transfers as XA[16:9] is provided by demultiplexing the data bus.
IRQ[15:9], IRQ[7:3], IRQ1	76:82, 83, 84 1:3	79:85, 86, 87, 91:94	I	Interrupt Request bits 15 through 9, 7 through 3, and 1: These signals are asynchronous inputs. When the 8259 is operating in an edge triggered mode, an interrupt request is executed by raising an IRQ input low-to-high and holding it high until it is acknowledged by the CPU. When the 8259 is operating in a level triggered mode, an interrupt request is executed by raising an IRQ input high and holding it high until it is acknowledged by the CPU.
INTA#	16	8	I	Interrupt Acknowledge: An active low input used to enable the interrupt controllers to output the vector data on the data bus by an interrupt acknowledge sequence from the CPU.

Signal Name	Pin No. 84 PLCC	Pin No. 100 PQFP	Signal Type	Signal Description
INTR	70	73	O	Interrupt Request: An active high output pin which is connected to the CPU's interrupt pin and is used to interrupt the CPU when an interrupt request occurs.
TMRCLK	23	17	I	Timer Clock: This is the input clock for the 8254's Counters 0, 1, and 2. In a PC/AT-based system it is approximately 1.19MHz.
GATE2	22	16	I	Gate 2: This signal is a gate input for the 8254's Counter 2. In a PC/AT-based system, Counter 2 is used for tone generation for a speaker. It is driven by bit 0 of I/O Port 61h.
OUT1	20	14	O	Out 1: This is an output of the 8254's Counter 1. In a PC/AT-based system, Counter 1 is programmed as a rate generator to produce a 15µs period signal for DRAM refresh.
OUT2	19	13	O	Out 2: This is an output of the 8254's Counter 2. In a PC/AT-based system, Counter 2 is used for tone generation for a speaker.
AS	71	74	I	Address Strobe: An active high input which is pulsed by the System Controller when the CPU accesses the real-time clock or CMOS RAM of the 82C206. The falling edge of this pulse latches the address from the XD bus.
PSRSTB#	15	7	I	Power Supply Strobe: An active low input used to establish the condition of the control registers of the real-time clock when power is applied to the device. In a PC/AT-based system, it should be tied to the battery back-up circuit. When PSRSTB# and TEST are both low, the 82C206 is not accessible and the following bits in the RTC are cleared to zero: • PIE (Periodic Interrupt Enable) • AIE (Alarm Interrupt Enable) • UIE (Update ended Interrupt Enable) • UF (Update ended Interrupt Flag) • IRQF (Interrupt Request status Flag) • PF (Periodic Interrupt Flag) • AF (Alarm interrupt Flag) • Square Wave output enable.
PWRGD	14	6	I	Power Good: An active high input that is connected to the power good of the power supply in a PC/AT-based system. It must be high for bus cycles in which the CPU accesses the RTC. When low, all address, data, data strobe, and R/W pins are disconnected from the processor.
3.2.2 Miscellaneous Signals				
NC		4, 11, 12, 27, 28, 29, 40, 51, 53, 66, 67, 68, 88, 89, 90		No Connect: These pins are not connected.

Signal Name	Pin No. 84 PLCC	Pin No. 100 PQFP	Signal Type	Signal Description
TEST	17	9	I	Test: This active high input used in production testing of the 82C206. It should be tied low for normal operation.
3.2.3 Power & Ground Pins				
VCC	32, 75	26, 78	I	Power Connection
GND	12, 53, 74	3, 52, 77	I	Ground Connection

4.0 Functional Description

The following sub-sections will give detailed operational information about the 82C206 Integrated Peripheral Controller (IPC).

4.1 Top Level Decoder & Configuration Register

The 82C206's top level decoder provides eight separate enables to various internal subsystems. Table 4-1 contains a truth table for the top level decoder.

Enabling of the 82C206's XD[7:0] output buffers is also controlled by this section. The output buffers are enabled whenever an enable is generated to an internal subsystem and the XIOR# signal is also asserted. The top level decoder is enabled by the ACK# and XA[9:8] signals. To enable any internal subsystem, ACK# must be 1 and XA[9:8] must be 00.

Table 4-2 gives the format of the 82C206's Configuration Register.

Table 4-1 82C206 Internal Decode

ACK#	XA9	XA8	XA7	XA6	XA5	XA4	XA3	XA2	XA1	XA0	Address Range	Selected Device
1	0	0	0	0	0	0	X	X	X	X	000h-00Fh	DMA8
1	0	0	0	0	1	0	0	0	0	X	020h-021h	INTC1
1	0	0	0	0	1	0	0	0	1	X	022h-023h	CONFIG
1	0	0	0	1	0	0	0	0	X	X	040h-043h	CTC
1	0	0	0	1	1	1	0	0	0	1	071h	RTC
1	0	0	1	0	0	0	X	X	X	X	080h-08Fh	DMAPAGE
1	0	0	1	0	1	0	0	0	0	X	0A0h-0A1h	INTC2
1	0	0	1	1	0	X	X	X	X	X	0C0h-0DFh	DMA16
0	X	X	X	X	X	X	X	X	X	X		Disabled
X	1	X	X	X	X	X	X	X	X	X		Disabled
X	X	1	X	X	X	X	X	X	X	X		Disabled

Note: **Note:**X = Don't care

Table 4-2 Configuration Register (Index Port 22h, Data Port 23h) - Index: 01h

Bit(s)	Type	Default	Function
7:6	R/W	11	These bits control the number of wait states inserted when the CPU accesses the registers of the 82C206. Wait states are counted as SYSCLK cycles and are not affected by the DMA clock selection. **7 6 R/W Wait States** 0 0 1 0 1 2 1 0 3 1 1 4 (Default)

Bit(s)	Type	Default	Function
5:4	R/W	00	These bits control the number of wait states inserted in 16-bit DMA cycles. Further control of the DMA cycle length is available through the use of the 82C206's IOCHRDY pin. During DMA cycles, this pin is used as an input to the wait state generation logic to extend the cycle if necessary. **5 4 16-Bit DMA Wait States** 0 0 1 (Default) 0 1 2 1 0 3 1 1 4
3:2	R/W	00	These bits control the number of wait states inserted in 8-bit DMA cycles. Further control of the DMA cycle length is available through the use of the 82C206's IOCHRDY pin. During DMA cycles, this pin is used as an input to the wait state generation logic to extend the cycle if necessary. **3 2 8-Bit DMA Wait States** 0 0 1 (Default) 0 1 2 1 0 3 1 1 4
1	R/W	0	This bit enables the early DMAMEMR# function. In a PC/AT-based system, DMAMEMR# is delayed one clock cycle later than XMEMR#. If set to 1, it will start DMAMEMR# at the time as XMEMR#. If set to 0, it will start DMAMEMR#.
0	R/W	0	If this bit is set to 0, the SYSCLK input is divided by two and is used to drive both 8- and 16-bit DMA subsystems. If this bit is set to 1, SYSCLK will directly drive the DMA subsystems. Whenever the state of this bit is changed, an internal synchronizer controls the actual switching of the clock to prevent a short clock pulse from causing a DMA malfunction.

4.2 DMA Subsystem

The 82C206 contains two 8237 DMA controllers. Each controller is a four channel DMA device which will generate the memory address and control signals necessary to transfer data between a peripheral device and memory directly. The two DMA controllers are internally cascaded to provide four DMA channels for transfers to 8-bit peripherals (DMA8) and three channels for transfers to 16-bit peripherals (DMA16). Channel 0 of DMA16 provides the cascade interconnection of the two DMA controllers, hence maintaining PC/AT compatibility. Hereafter, the description of the DMA subsystem pertains to both DMA8 and DMA16 unless otherwise noted.

Table 4-3 gives the I/O address map of the 82C206's DMA subsystem. The mapping is fully PC/AT compatible.

Table 4-3 DMA I/O Address Map

Address		Operation			
DMA8	DMA16	XIOR#	XIOW#	Byte Pointer	Register Function
000h	0C0h	0	1	0	Read Channel 0's current address low byte
		0	1	1	Read Channel 0's current address high byte
		1	0	0	Write Channel 0's base and current address low byte
		1	0	1	Write Channel 0's base and current address high byte
001h	0C2h	0	1	0	Read Channel 0's current word count low byte
		0	1	1	Read Channel 0's current word count high byte
		1	0	0	Write Channel 0's base and current word count low byte
		1	0	1	Write Channel 0's base and current word count high byte

Address		Operation		Byte Pointer	Register Function
DMA8	DMA16	XIOR#	XIOW#		
002h	0C4h	0	1	0	Read Channel 1's current address low byte
		0	1	1	Read Channel 1's current word count high byte
		1	0	0	Write Channel 1's base and current address low byte
		1	0	1	Write Channel 1's base and current address high byte
003h	0C6h	0	1	0	Read Channel 1's current word count low byte
		0	1	1	Read Channel 1's current word count high byte
		1	0	0	Write Channel 1's base and current word count low byte
		1	0	1	Write Channel 1's base and current word count high byte
004h	0C3h	0	1	0	Read Channel 2's current address low byte
		0	1	1	Read Channel 2's current address high byte
		1	0	0	Write Channel 2's base and current address low byte
		1	0	1	Write Channel 2's base and current address high byte
005h	0CAh	0	1	0	Read Channel 2's current word count low byte
		0	1	1	Read Channel 2's current word count high byte
		1	0	0	Write Channel 2's base and current word count low byte
		1	0	1	Write Channel 2's base and current word count high byte
006h	0CCh	0	1	0	Read Channel 3's current address low byte
		0	1	1	Read Channel 3's current address high byte
		1	0	0	Write Channel 3's base and current address low byte
		1	0	1	Write Channel 3's base and current address high byte
007h	0CEh	0	1	0	Read Channel 3's current word count low byte
		0	1	1	Read Channel 3's current word count high byte
		1	0	0	Write Channel 3's base and current word count low byte
		1	0	1	Write Channel 3's base and current word count high byte
008h	0D0h	0	1	X	Read Status Register
		1	0	X	Write Command Register
009h	0D2h	0	1	X	Read DMA Request Register
		1	0	x	Write DMA Request Register
00Ah	0D4h	0	1	X	Read Command Register
		1	0	X	Write single bit DMA Request Mask Register
00Bh	0D6h	0	1	X	Read Mode Register
		1	0	X	Write Mode Register
00Ch	0D8h	0	1	X	Set byte pointer flip-flop
		1	0	X	Clear byte pointer flip-flop
00Dh	0DAh	0	1	X	Read Temporary Register
		1	0	X	Master clear
00Eh	0DCh	0	1	X	Clear Mode Register counter
		1	0	X	Clear all DMA Request Mask Register bits
00Fh	0DEh	0	1	X	Read all DMA Request Mask Register bits
		1	0	X	Write all DMA Request Mask Register bits

4.2.1 DMA Operation

During normal operation, the DMA subsystem of the 82C206 will be in one of three modes: the Idle Mode, Program Mode, or the Active Mode. When the DMA controller is in the Idle Mode, it only executes the S1 idle state cycles. The DMA controller will remain in the Idle Mode unless it has been initialized to work and one of the DMA request pins has been asserted. In this case, the DMA controller will exit the Idle

Mode and enter the Active Mode. The DMA controller will also exit the Idle Mode and enter the Program Mode when the CPU attempts to access its internal registers.

4.2.1.1 Idle Mode

If no peripheral requests service, the DMA subsystem will enter the Idle Mode and perform only S1 idle states. During this time, the 82C206 will sample the DREQ input pins every clock cycle to determine if any peripheral is requesting a DMA service. The internal select from the top level decoder and HLDA1 input pin will also sample at the same time to determine if the CPU is attempting to access the internal registers. With either of the above conditions, the DMA subsystem will exit the Idle Mode and enter either the Program Mode or Active Mode. Note that the Program Mode has priority over the Active Mode since a CPU cycle has already started before the DMA was granted use of the bus.

4.2.1.2 Program Mode

The DMA subsystem will enter the Program Mode whenever HLDA1 is inactive and an internal select from the top level decoder is active. During this time, the address lines XA[3:0] become inputs if DMA8 is selected or XA[4:1] become inputs if DMA16 is selected. These address inputs are used to decode which registers in the DMA controller are to be accessed. The XIOR# and XIOW# signals are used to select and time the CPU reads or writes. When DMA16 is selected, XA0 is not used to decode and is ignored. Due to the large number and size of the internal registers of the DMA controller, an internal byte pointer flip-flop is used to supplement the addressing of the 16-bit word and count address registers. This byte pointer is used to determine the upper or lower byte of word count and address registers and is cleared by a hardware reset or a master clear command. It may also be set or cleared by the CPU's set byte pointer flip-flop or clear byte pointer flip-flop commands.

The DMA subsystem supports some special commands when in the Program Mode. These commands do not use the data bus, but are derived from a set of address, the internal select, and XIOR# or XIOW#. These commands are listed at the end of Table 4-3. Erratic operation of the 82C206 can occur if a request for service occurs on an unmasked DMA channel which is being programmed. The channel should be masked or the DMA should be disabled to prevent the 82C206 from attempting to service a peripheral with a channel which is only partially programmed.

4.2.1.3 Active Mode

The DMA subsystem will enter the Active Mode whenever a software request occurs or a DMA request occurs on an unmasked channel which has already been programmed. When a DREQ occurs and the corresponding mask bit is clear, or a software DMA request is made, the DMA subsystem issues HRQ to the System Controller. After the CPU releases the system bus, the System Controller then issues a HLDA1 back to the 82C206 if the DMA has been permitted to control the system bus. After being granted control of the bus, the DMA subsystem will then begin a DMA transfer cycle. An example of this would be a DMA read cycle. After receiving a DREQ, the 82C206 will issue an HRQ to the System Controller. Until an HLDA1 is returned, the DMA subsystem will remain in an idle state. On the next clock cycle, the DMA will exit the idle state and enter an S0 state. During S0, the DMA will resolve priority and issue DACK on the highest priority channel which is requesting service. The DMA then enters the S1 state where the multiplexed addresses are output and latched. Next, the DMA enters the S2 state where the 82C206 asserts the DMAMEMR# command. Then the DMA will enter the S3 state where the 82C206 asserts the XIOW# command. The DMA will then remain in the S3 state until the wait state counter has expired and IOCHRDY is high. Note that at least one additional S3 will occur unless compressed timing is programmed. Once a ready condition is detected, the DMA will enter S4 where DMAMEMR# and XIOW# are deasserted.

In the Compressed and Demand Modes, subsequent transfers will begin in S2 unless the intermediate addresses require updating. In these subsequent transfers, the lower addresses are changed in S2.

4.2.2 DMA Transfer Modes

There are four transfer modes supported by the DMA subsystem: Single, Block, Demand, and Cascade. The DMA subsystem can be programmed on a channel-by-channel basis to operate in one of these four modes.

4.2.2.1 Single Transfer Mode

In the Single Transfer Mode, the DMA will execute only one cycle at a time. DREQ must be held active until DACK becomes active in order to be recognized. If DREQ is held active throughout the Single Transfer, the 82C206 will deassert HRQ and release the bus to the system once the transfer is complete. After HLDA has gone inactive, the 82C206 will again assert HRQ and execute another transfer on the same channel unless a request from a higher priority channel has been received.

During the Single Transfer Mode, the CPU is ensured of at least one full machine cycle execution between DMA transfers. Following each transfer, the Word Count Register is decreased and the Address Register is increased or decreased (depending on the DEC bit of the Mode Register). When the word count decrements from 0000h to FFFFh, the terminal count bit in the Status Register is set and a pulse is output to the TC pin. If auto-initialization is selected, the channel will reinitialize itself for the next service - otherwise, the DMA will set the corresponding DMA request bit mask and suspend transferring on that channel.

4.2.2.2 Block Transfer Mode

In the Block Transfer Mode, the DMA will begin transfers in response to either a DREQ or a software reset. If DREQ starts the transfer, it needs to be held active until DACK becomes active. The transfers will continue until the word count decrements from 0000h to FFFFh, at which time the TC pin is pulsed and the terminal count bit in the Status Register is set. Once more, an auto-initialization will occur at the end of the last service if the channel has been programmed to do so.

4.2.2.3 Demand Transfer Mode

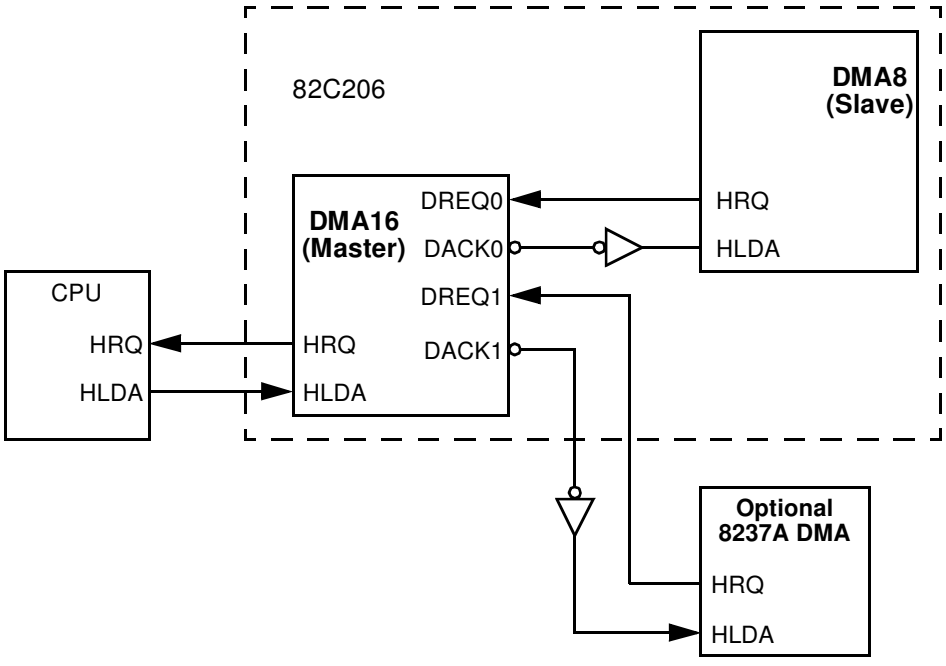
In the Demand Transfer Mode, the DMA will begin transfers in response to the assertion of DREQ and will continue until either the terminal count is reached or DREQ becomes active. The Demand Transfer Mode is normally used for peripherals which have limited buffering capacity. The peripheral can initiate a transfer and continue until its buffer capacity is exhausted. The peripheral may then re-establish service by again asserting DREQ. During idle states between transfers, the CPU is released to operate and can monitor the operation by reading intermediate values from the Address and Word Count Registers. Once DREQ is deasserted, higher priority channels are allowed to intervene. Reaching the terminal count will result in the generation of a pulse on the TC pin, the setting of the terminal count bit in the Status Register, and auto-initialization if programmed to do so.

4.2.2.4 Cascade Mode

The Cascade Mode is used to interconnect more than one DMA controller to extend the number of DMA channels while preserving the priority chain. While in this mode, the master DMA controller does not generate address or control signals. The DREQ and DACK signals of the master are used to interface the HRQ and HLDA1 signals of the slave DMA devices. Once the master has received an HLDA1 from the CPU in response to a DREQ caused by the HRQ from a slave DMA controller, the master DMA controller will ignore all inputs except HLDA1 from the CPU and DREQ on the active channel. This prevents conflicts between the DMA devices.

Figure 4-1 shows the cascade interconnection for two levels of DMA devices. Note that Channel 0 of DMA16 is internally connected for the Cascade Mode to DMA8. Additional devices can be cascaded to the available channels in either DMA8 or DMA16 since the Cascade Mode is not limited to two levels of DMA controllers.

Figure 4-1 Cascade Mode Interconnect



When programming cascaded controllers, begin with the device which is actually generating HRQ to the system (first level device) and then proceed to the second level devices. RESET causes the DACK outputs to become active low and are placed in the inactive state. To allow the internal cascade between DMA8 and DMA16 to operate correctly, the active low state of DACK should not be modified. The first level device's DMA request mask bits will prevent the second level

cascaded devices from generating unwanted hold requests during the initialization process.

4.2.3 Transfer Types

There are three types of transfers:

- Read Transfers
- Write Transfers
- Verify Transfers

The Single, Block, and Demand Transfer Modes can perform any of the three transfer types.

Read Transfers move data from memory to an I/O peripheral by generating the memory address and asserting DMA-MEMR# and XIOW# during the same transfer cycle.

Write Transfers move data from an I/O peripheral to memory by generating the memory address and asserting DMA-MEMW# and XIOR# during the same transfer cycle.

Verify Transfers are pseudo transfers. In this type of transfer, the DMA will operate as in Read or Write Transfers by generating HRQ, DACK, memory addresses and respond to the terminal count, but it does not activate the memory or I/O command signals. Since no transfer actually takes place, IOCHRDY is also ignored during Verify Transfers.

In addition to the three transfer types mentioned above, there is also a memory-to-memory transfer which can only be used on DMA Channels 0 and 1. The memory-to-memory transfer is used to move a block of memory from one location in memory to another. DMA Channels 0 and 1 may be programmed to operate as memory-to-memory channels by setting a bit in the DMA Command Register. Once programmed, the transfer can be started by generating either a software or an external request to Channel 0. During the transfer, Channel 0 provides the address for the source block during the memory write portion of the same transfer. During the read portion of the transfer, a byte of data is latched in the internal Temporary Register of the DMA. The contents of this register are then output on the XD[7:0] output pins during the write portion of the transfer and subsequently written to the memory location. Channel 0 may be programmed to maintain the same source address on every transfer. This allows the CPU to initialize large blocks of memory with the same value. The DMA subsystem will continue performing transfers until Channel 1 reaches the terminal count.

4.2.4 Auto-initialization

The Mode Register of each DMA channel contains a bit which will cause the channel to reinitialize after reaching the terminal count. During auto-initialization, the Base Address and Base Word Count Registers (which were originally programmed by the CPU) are reloaded into the Current Address and Current Word Count Registers. The Base Registers remain unchanged during DMA active cycles and can only be changed by the CPU. If the channel has been programmed to auto-initialize, the request mask bit will remain cleared upon reaching the terminal count. This allows the DMA to continue operation without CPU intervention. In memory-to-memory transfers, the Word Count Registers of Channels 0 and 1 must be programmed with the same starting value for full auto-initialization.

4.2.5 DREQ Priority

The 82C206 supports two types of software programmable priority schemes: fixed and rotating. Fixed priority assigns priority based on channel position. With this method, Channel 0 is assigned the highest priority and Channel 3 is the lowest. After the recognition of any one channel for service, the other channels are prevented from interfering with that service until it is completed.

In the rotating priority scheme, the ordering of priority from Channel 0 to Channel 3 is maintained, but the actual assignment of priority changes. The channel most recently serviced will be assigned the lowest priority and since the order of priority assignment remains fixed, the remaining three channels rotate accordingly. Table 4-4 shows the rotating priority scheme. In cases where multiple requests occur at the same time, the 82C206 will issue HRQ but will not freeze the priority logic until HLDA1 is returned. After HLDA1 becomes active, the priority logic is frozen and DACK is asserted on the highest requesting channel. Priority will not be reevaluated until HLDA1 has been deactivated.

Table 4-4 Rotating Priority Scheme

Priority	First Arbitration	Second Arbitration	Third Arbitration
Highest	Channel 0	Channel 2 - Cycle Grant	Channel 3 - Cycle Grant
	Channel 1 - Cycle Grant	Channel 3	Channel 0
	Channel 2	Channel 0	Channel 1
Lowest	Channel 3	Channel 1	Channel 2

Channel X = Requested Channel

4.2.6 Address Generation

During active cycles of the DMA, eight intermediate bits of the address are multiplexed onto the data lines. This reduces the number of pins required by the DMA subsystem. During an S1 state, the intermediate addresses are output on data lines XD[7:0]. These addresses should be externally latched and used to drive the system address bus. Since DMA8 is used for 8-bit transfers and DMA16 is used for 16-bit transfers, a one bit skew occurs in the intermediate address fields. DMA8 will therefore output address on A[15:8] on the data bus at this time whereas DMA16 will output A[16:9]. A separate set of latch and enable signals are provided for both DMA8 and DMA16 to accommodate the address skew.

During 8-bit DMA transfers in which DMA8 is active, the 82C206 will output the lower eight bits of address on XA[7:0]. The intermediate eight bits of address will be output on XD[7:0] and ADSTB8 will be asserted for one DMA clock cycle. The falling edge of ADSTB8 is used to latch the intermediate addresses A[15:8]. An enable signal, AEN8, is issued to control the output drivers of the external latch. A[23:16] are also generated at this time from a DMA page register in the 82C206. Note that A16 is output on the XA16 pin of the device.

During 16-bit DMA transfers in which DMA16 is active, the 82C206 will output the lower eight bits of address on XA[8:1]. The intermediate eight bits of address A[16:9]will be output on XD[7:0] and ADSTB16 will be asserted for one DMA clock cycle. The falling edge of ADSTB16 is used to latch the intermediate addresses A[16:9]. An enable signal, AEN16, is issued to control the output drivers of the external latch. A[23:17] are also generated at this time from a DMA page register in the 82C206. Note that XA0 and XA16 remain tristated during 16-bit DMA transfers.

The DMA page registers are a set of 16 8-bit registers in the 82C206 which are used to generate the high order addresses during DMA cycles. Only eight of the registers are actually used, but all 16 were included to maintain PC/AT compatibility. Each DMA channel has a page register associated with it except Channel 0 of DMA16 which is used for cascading to DMA8. Assignment of each of these registers is shown in Table 4-5 along with its CPU I/O read/write address.

During Demand and Block Transfers, the 82C206 generates multiple sequential transfers. For most of these transfers, the information in the external address latches will remain the same, thus eliminating the need to be relatched. Since the need to update the latches occurs only when a carry or borrow from the lower eight bits of the address counter exists, the 82C206 will only update the latch contents when necessary. The 82C206 will there only execute an S1 state when necessary and improve the overall system throughput.

Table 4-5 DMA Page Register I/O Address Map

I/O Addr	Type	Register Function
080h	R/W	Unused
081h	R/W	DMA8 Channel 2 (DACK2)
082h	R/W	DMA8 Channel 3 (DACK3)
083h	R/W	DMA8 Channel 1 (DACK1)
084h	R/W	Unused
085h	R/W	Unused
086h	R/W	Unused
087h	R/W	DMA8 Channel 0 (DACK0)
088h	R/W	Unused
089h	R/W	DMA16 Channel 2 (DACK6)
08Ah	R/W	DMA16 Channel 3 (DACK7)
08Bh	R/W	DMA16 Channel 1 (DACK5)
08Ch	R/W	Unused
08Dh	R/W	Unused
08Eh	R/W	Unused
08Fh	R/W	DRAM Refresh Cycle

4.2.7 Compressed Timing

The DMA subsystem in the 82C206 can be programmed to transfer a word in as few as two DMA clock cycles. Normal transfers require four DMA clock cycles since S3 is executed twice (due to the one wait state insertion). In systems capable of supporting higher throughput, the 82C206 can be programmed to omit one S3 and assert both commands in S2. S2 begins the cycle by generating the address and asserting both commands. One S3 cycle is executed and the cycle terminates in S4. If compressed timing is selected, TC will be output in S2 and S1 cycles which will be executed as necessary to update the address latch. Note that compressed timing is not allowed for memory-to-memory transfers.

4.2.8 DMA Register Descriptions

4.2.8.1 Current Address Register

Each DMA channel has a 16-bit Current Address Register which holds the address used during transfers. Each channel can be programmed to increment or decrement this register whenever a transfer is completed. This register can be read or written by the CPU in consecutive 8-bit bytes. If auto-initialization is selected, this register will be reloaded from the Base Address Register upon reaching the terminal count in the Current Word Count Register. Channel 0 can be prevented from incrementing or decrementing by setting the address hold bit in the Command Register.

4.2.8.2 Current Word Count Register

Each channel has a Current Word Count Register which determines the number of transfers. The actual number of transfers performed will be one greater than the value programmed into the register. The register is decremented after each transfer until it goes from 0 to FFFFh. When this rollover occurs, the 82C206 will generate TC and either suspend the operation on that channel and set the appropriate request mask bit, or auto-initialize and continue.

4.2.8.3 Base Address Register

Associated with each Current Address Register is a Base Address Register. This is a write-only register which is loaded by the CPU when writing to the Current Address Register. The purpose of this register is to store the initial value of the Current Address Register for auto-initialization. The contents of this register are loaded into the Current Address Register whenever the terminal count is reached and the auto-initialize bit is set.

4.2.8.4 Base Word Count Register

This register preserves the initial value of the Current Word Count Register. It too is a write-only register which is loaded by writing to the Current Word Count Register. The Base Word Count Register is loaded into the Current Word Count Register during auto-initialization.

4.2.8.5 Command Register

The Command Register controls the overall operation of the DMA subsystem. This register can be read or written by the CPU and is cleared by either a reset or master clear command. Table 4-6 gives the Command Register's format.

4.2.8.6 Mode Register

Each DMA channel has a Mode Register associated with it. All four Mode Registers reside at the same I/O address. Bits 0 and 1 of the Write Mode Register command determine which channel Mode Register gets written. The remaining six bits control the mode of the selected channel. Each channel Mode Register can be read by sequentially reading the Mode Register location. A Clear Mode Register Counter command is provided to allow the CPU to restart the mode read process at a known point. During mode read operations, bit 0 and 1 will both equal 1. Table 4-7 gives the format for the Mode Register.

Table 4-6 Command Register

Bit(s)	Type	Default	Function
7	R/W		The setting of this bit determines if the DACK output pin will be active low or active high. 0 = Active low 1 = Active high
6	R/W		The setting of this bit determines if the DREQ input pin will be active low or active high. 0 = Active low 1 = Active high
5	R/W		Extended Write: The extended write feature is controlled by this bit. When enabled, it causes the write command to be asserted one DMA cycle earlier during a transfer. Thus, read and write commands both begin in the S2 state. 0 = Disabled 1 = Enabled
4	R/W	0	Rotating Priority: This bit selects wether the priority scheme is fixed or rotating. 0 = Fixed (Default) 1 = Rotating
3	R/W	1	Compressed Timing: This bit enables the Compressed Timing feature. 0 = Compressed Timing 1 = Normal Timing (Default)
2	R/W		Controller Disable: Setting this bit to 1 disables the DMA subsystem (DMA8 or DMA16). This function is normally used whenever the CPU needs to reprogram one of the channels to prevent DMA cycles from occurring. 0 = Enable 1 = Disable
1	R/W		Address Hold: Setting this bit to 1 enables the address hold feature in Channel 0 when performing memory-to-memory transfers. 0 = Disable 1 = Enable

Bit(s)	Type	Default	Function
0	R/W		Memory-to-Memory: This bit enables Channel 0 and 1 to be used for memory-to-memory transfers. 0 = Disable 1 = Enable

Table 4-7 Mode Register

Bit(s)	Type	Default	Function
7:6	R/W		Mode Select bits 1 and 0: These bits are used to select the mode for each channel. **7 6 Mode Select** 0 0 Demand Mode 0 1 Single Cycle Mode 1 0 Block Mode 1 1 Cascade Mode
5	R/W		Decrement: Writing a 1 to this bit decrements the address after each transfer.
4	R/W		Auto-initialization: Writing a 1 to this bit enables the auto-initialization function.
3:2	R/W		Transfer Type bits 1 and 0: These bits control the type of transfer to performed. **3 2 Transfer Type Select** 0 0 Verify 0 1 Write Transfer 1 0 Read Transfer 1 1 Illegal
1:0	R/W		Channel Selection bits 1 and 0: These bits determine which channel's Mode Register will be written. Read back of a Mode Register will cause these bits to both be 1. **1 0 Channel Selection** 0 0 Select Channel 0 0 1 Select Channel 1 1 0 Select Channel 2 1 1 Select Channel 3

4.2.8.7 Request Register

This 4-bit register is used to generate software requests (DMA service can be requested either externally or under software control). Request Register bits can be set or reset independently by the CPU. The register mask has no effect on software generated request. All four bits are read in one operation and appear in the lower four bits of the byte. Bits 7 through 4 are read as 1s. All four request bits are cleared to 0 by a reset. Table 4-8 gives the write format of the Request Register and Table 4-9 gives the read format.

Table 4-8 Request Register Write Format

Bit(s)	Type	Default	Function
7:3	W	XXXXX	Don't Care
2	W		Request Bit: Writing a 1 to this bit sets the request bit.
1:0	W		Request Select bits 1 and 0: These bits determine which channel's request bit will be set. 1 0 Channel Selection 0 0 Select Channel 0 0 1 Select Channel 1 1 0 Select Channel 2 1 1 Select Channel 3

Table 4-9 Request Register Read Format

Bit(s)	Type	Default	Function
7:4	R	11111	Reserved: Always reads 1.
3:0	R		Request Channel bits 3 through 0: These bits contain the state of the request bit associated with each request channel. The bit position corresponds to the channel number.

4.2.8.8 Request Mask Register

The Request Mask Register is a set of four bits which are used to inhibit external DMA requests from generating transfer cycles. This register can be programmed in two ways. Each channel can be independently masked by writing to the Write Single Mask bit location. The data format for this operation is shown in Table 4-10.

Alternatively, all four mask bits can be programmed in one operation by writing to the write all mask bits address. The data format for this and the read all mask bits function is shown in Table 4-11.

All four mask bits are set following a reset or a Master Clear command. Individual channel mask bits will be set as a result of the terminal count being reached, if auto-initialize is disabled. The entire register can be cleared, enabling all four channels by performing a Clear Mask Register operation.

Table 4-10 Request Mask Register Set/Reset Format

Bit(s)	Type	Default	Function
7:3		XXXXX	Don't Care
2			Mask Bit: Writing a 1 to this bit sets the request mask bit and inhibits external requests.
1:0			Mask Select bits 1 and 0: These bits determine which channel's request bit will be set. **1 0 Channel Selection** 0 0 Select Channel 0 0 1 Select Channel 1 1 0 Select Channel 2 1 1 Select Channel 3

Table 4-11 Request Mask Register Read/Write Format

Bit(s)	Type	Default	Function
7:4	R/W	1	Reserved: Always reads 1.
3:0	R/W		Mask Bits 3 through 0: These bits contain the state of the request mask bit associated with each request channel. The bit position corresponds to the channel number.

4.2.8.9 Status Register

The status of all four channels can be determined by reading the Status Register. Information is available to determine if a channel has reached the terminal count and whether an external service request is pending. Table 4-12 gives the format for the Status Register.

4.2.8.10 Temporary Register

The Temporary Register is used as a temporary holding register for data during memory-to-memory transfers. The register is loaded during the first cycle of a memory-to-memory transfer from XD[7:0]. During the second cycle of the transfer, the data in the Temporary Register is output on the XD[7:0] pins. Data from the last memory-to-memory; transfer will remain in the register.

4.2.9 Special Commands

Five Special Commands are provided to make the task of programming the 82C206 easier. These commands are activated as a result of a specific address and assertion of either XIOR# or XIOW#. For these Special Commands, the data bus is ignored by the 82C206 whenever an XIOW# activated command is issued. Data returned on XIOR# activated commands is undefined.

- Clear Byte Pointer Flip-Flop: This command is normally executed prior to reading or writing to the Address or Word Count Registers. This initializes the flip-flop to point to the

low byte of the register and allows the CPU to read or write the register bytes in correct sequence.

- Set Byte Pointer Flip-Flop: Setting the byte pointer flip-flop allows the CPU to adjust the pointer to the high byte of an Address or Word Count Register.
- Master Clear: This command has the same effect as a hardware reset. The Command Register, Status Register, Request Register, Temporary Register, Mode Register counter, and byte pointer flip-flop are cleared and the Request Mask Register is set. Immediately following a Master Clear or reset, the DMA will be in the Idle Mode.
- Clear Request Mask Register: This command enables all four DMA channels to accept requests by clearing the mask bits in the register.
- Clear Mode Register Counter: In order to allow access to the four Mode Registers while only using one address, an internal counter is used. After clearing the counter, all four Mode Registers may be read by successive reads to the Mode Register. The order in which the registers are read is Channel 0 first and Channel 3 last.

Table 4-12 Status Register

Bit(s)	Type	Default	Function
7:4	R		Data Request bits 3 through 0: These bits show the status of each channel request and are not affected by the state of the Mask Register bits. Reading a 1 means “request” occurs and bits 7 through 4 represent Channels 3 through 0, respectively. These bits can be cleared by a reset, Master Clear of the pending request being deasserted.
3:0	R		Terminal Count bits 3 through 0: These bits indicate which channel has reached the terminal count reading 1. These bits can be cleared by a reset, Master Clear, or each time a status read takes place. The channel number corresponds to the bit position.

4.3 Interrupt Controller Subsystem

The programmable interrupt controllers in the 82C206 serve as a system wide interrupt manager in an X86 system. They accept requests from peripherals, resolve priority on pending interrupts and interrupts in service, issue an interrupt request to the CPU, and provide a vector which is used as an index by the CPU to determine which interrupt service routine to execute.

A variety of priority assignment modes are provided which can be reconfigured at any time during system operation. This allows the complete subsystem to be restructured based on the system environment.

4.3.1 Interrupt Controller Subsystem Overview

There are two interrupt controllers, INTC1 and INTC2, included in the 82C206. Each of the interrupt controllers is equivalent to an 8259A device operating in X86 mode. The two devices are interconnected and must be programmed to operate in the Cascade Mode for all 16 interrupt channels to operate properly. Figure 4-2 shows the internal Cascade interconnection.

INTC1 is located at addresses 020h-021h and is configured for master operation in the Cascade Mode. INTC2 is a slave device and is located at 0A0h-0A1h. The interrupt request output signal (INT) from INTC2 is internally connected to the interrupt request input Channel 2 (IR2) of INTC1. The address decoding and cascade interconnection matches that of the PC/AT.

Two additional interconnections are made to the interrupt request inputs of the interrupt controllers. The output of Timer 0 in the counter/timer subsystem is connected to Channel 0 (IR0) of INTC1. Interrupt request from the real-time clock is connected to Channel 0 (IR0) of INTC2. Table 4-13 lists the 16 interrupt channels and their interrupt request sources.

Description of the interrupt subsystem will pertain to both INCT1 and INCT2 unless otherwise noted. Wherever register addresses are used, the address for the INTC1 Register will be listed first and the address for the INTC2 Register will follow in parenthesis. Example: 02h (0A0h).

Figure 4-2 Internal Cascade Interconnect

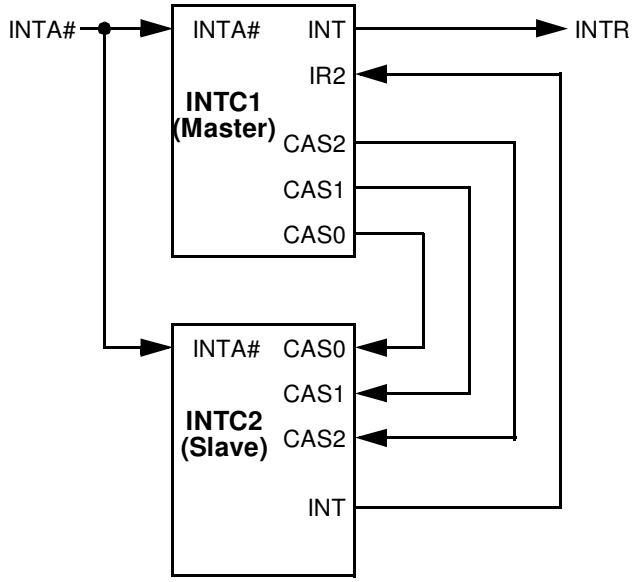


Table 4-13 Interrupt Request Source

Controller Number	Channel Name	Interrupt Request Source
INTC1	IR0	Counter/Timer OUT0
INTC1	IR1	IRQ1 Input Pin
INTC1	IR2	INTC2 Cascade Interrupt
INTC1	IR3	IRQ3 Input Pin
INTC1	IR4	IRQ4 Input Pin
INTC1	IR5	IRQ5 Input Pin
INTC1	IR6	IRQ6 Input Pin
INTC1	IR7	IRQ7 Input Pin
INTC2	IR0	Real-time Clock IRQ
INTC2	IR1	IRQ9 Input Pin
INTC2	IR2	IRQ10 Input Pin
INTC2	IR3	IRQ11 Input Pin
INTC2	IR4	IRQ12 Input Pin
INTC2	IR5	IRQ13 Input Pin
INTC2	IR6	IRQ14 Input Pin
INTC2	IR7	IRQ15 Input Pin

4.3.2 Interrupt Controller Operation

Figure 4-3 is a block diagram of the major components in the interrupt controller subsystem. The Interrupt Request Register (IRR) is used to store requests from all of the channels which are requesting service. The IRR's bits are labeled using the channel name IR[7:0]. The In-Service Register (ISR) contains all the channels which are currently being serviced (more than one channel can be in service at a time). The ISR's bits are labeled IS[7:0] and correspond to IR[7:0]. The Interrupt Mask Register (IMR) allows the CPU to disable any or all of the interrupt channels. The Priority Resolver evaluates inputs from the IRR, ISR, and IMR, issues an interrupt request, and latches the corresponding bit into the ISR. During interrupt acknowledge cycles, a master controller outputs a code to the slave device which is compared in the Cascade Buffer/Comparator with a 3-bit ID code previously written. If a match occurs in the slave controller, it will generate an interrupt vector. The contents of the Vector Register are used to provide the CPU with an interrupt vector during interrupt acknowledge (INTA) cycles.

4.3.3 Interrupt Sequence

The 82C206 allows the CPU to perform an indirect jump to a service routine in response to a request for service in response to a request for service from as peripheral device. The indirect jump is based on a vector which is provided by the 82C206 on the second of two CPU generated INTA cycles (the first INTA cycle is used for resolving priority and the second is for transferring the vector to the CPU (see Figure 4-4). The events which occur during an interrupt sequence are as follows:

1. One or more of the interrupt requests (IR[7:0]) becomes active, setting the corresponding IRR bit(s).
2. The interrupt controller resolves priority based on the state of the IRR, IMR, and ISR and asserts the INTR output if needed.
3. The CPU accepts the interrupt and responds with an INTA cycle.
4. During the first INTA cycle, the highest priority ISR bit is set and the corresponding IRR bit is reset. The internal cascade address is generated.
5. The CPU will execute a second INTA cycle, during which the 82C206 will drive an 8-bit vector onto the data pins XD[7:0], which is read by the CPU. The format of this vector is shown in Table 4-14. Note that V[7:3] in Table 4-14 are programmable by writing to ICW2 (Initialization Command Word 2).
6. At the end of the second INTA cycle, the ISR bit will be cleared if the Automatic End of Interrupt Mode is selected (see below). Otherwise, the ISR bit must be cleared by an End of Interrupt (EOI) command from the

CPU at the end of the interrupt service routine to allow further interrupts. If no interrupt request is present at the beginning of the first INTA cycle (i.e., a spurious interrupt), INCT1 will issue an interrupt level 7 vector during the second INTA cycle.

Figure 4-3 Interrupt Controller Block Diagram

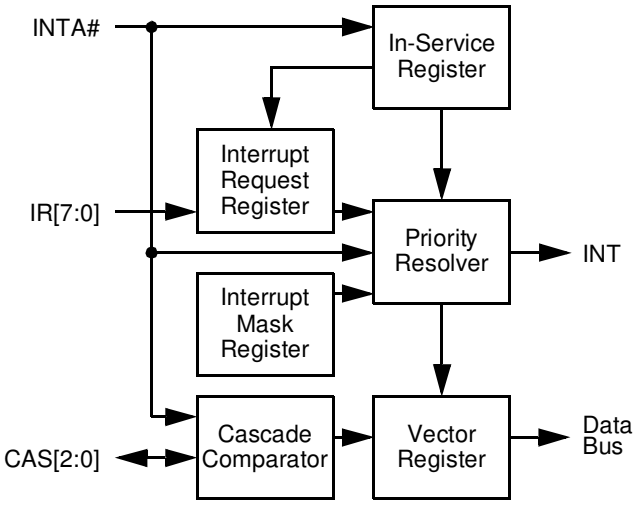


Figure 4-4 Interrupt Sequence

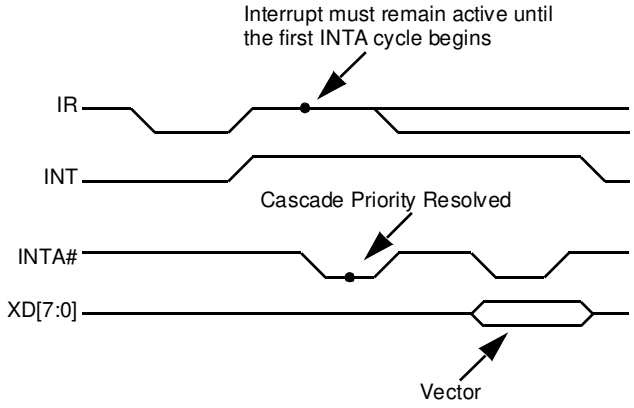


Table 4-14 Interrupt Vector Type

Interrupt	D7	D6	D5	D4	D3	D2	D1	D0
IR7	V7	V6	V5	V4	V3	1	1	1
IR6	V7	V6	V5	V4	V3	1	1	0
IR5	V7	V6	V5	V4	V3	1	0	1
IR4	V7	V6	V5	V4	V3	1	0	0
IR3	V7	V6	V5	V4	V3	0	1	1
IR2	V7	V6	V5	V4	V3	0	1	0
IR1	V7	V6	V5	V4	V3	0	0	1
IR0	V7	V6	V5	V4	V3	0	0	0

4.3.3.1 End of Interrupt (EOI)

EOI is defined as the condition which causes an ISR bit to be reset. Determination of which ISR bit is to be reset can be done by a CPU command (specific EOI) or the Priority Resolver can be instructed to clear the highest priority ISR bit (non-specific EOI). The 82C206 can determine the correct ISR bit to reset when operated in modes which do not alter the fully nested structure since the current highest priority ISR bit is the last level acknowledged and serviced. In conditions where the fully nested structure is not preserved, a specific EOI must be generated at the end of the interrupt service routine. An ISR bit that is masked, in the Special Mask Mode by an IMR bit, will not be cleared by a non-specific EIO command. The interrupt controller can optionally generate an Automatic End of Interrupt (AEOI) on the trailing edge of the second INTA cycle.

4.3.3.2 Priority Assignment

Assignment of priority is based on an interrupt channel's position relative to the other channels in the interrupt controller. After the initialization sequence, IR0 has the highest priority, IR7 the lowest, and priority assignment is Fixed. Priority assignment can be rotated either manually (Specific Rotation Mode) or automatically (Automatic Rotation Mode) by programming Operational Command Word 2 (OCW2).

Fixed Priority Mode

This is the default condition which exists unless rotation (either manual or automatic) is enabled, or the controller is programmed for Polled Mode. In the Fixed Priority Mode, interrupts are fully nested with priority assigned as shown:

	Lowest							Highest
Priority Status	7	6	5	4	3	2	1	0

Nesting allows interrupts of a higher priority to generate interrupt requests prior to the completion of the interrupt service. When an interrupt is acknowledged, priority is resolved, the highest priority request's vector is placed on the bus, and the ISR bit for that channel is set. This bit remains set until an EIO (automatic or CPU generated) is issued to that channel. While the ISWR bit is set, all interrupts of equal or lower priority are inhibited. Note that a higher priority service routine will only be acknowledged if the CPU has internally re-enabled interrupts.

Specific Rotation Mode

Specific Rotation allows the system software to re-assign priority levels by issuing a command which redefines the highest priority channel. Before rotation:

	Lowest							Highest
Priority Status	7	6	5	4	3	2	1	0

(Specific Rotation command issued with Channel 5 specified.) After rotation:

	Lowest							Highest
Priority Status	5	4	3	2	1	0	7	6

Automatic Rotation Mode

In applications where a number of equal priority peripherals are requesting interrupts, Automatic Rotation may be used to equalize the priority assignment. In this mode, after a peripheral is serviced it is assigned the lowest priority. All peripherals connected to the controller will be serviced at least once in eight interrupt requests to the CPU from the controller. Automatic Rotation will occur, if enabled, due to the occurrence of an EOI (automatic or CPU generated).

Before rotation (IR3 is the highest priority request being serviced):

ISR Status Bit	IS7	IS6	IS5	IS4	IS3	IS2	IS1	IS0
	1	1	0	0	1	0	0	0
	Lowest							Highest
Priority Status	7	6	5	4	3	2	1	0

(Specific Rotation command issued with Channel 4 specified.) After rotation:

ISR Status Bit	IS7	IS6	IS5	IS4	IS3	IS2	IS1	IS0
	1	1	0	0	0	0	0	0
	Lowest							Highest
Priority Status	3	2	1	0	7	6	5	4

4.3.4 Programming the Interrupt Controller

Two types of commands are used to control the 82C206's interrupt controllers: Initialization Command Words (ICWs) and Operational Command Words (OCWs).

4.3.4.1 Initialization Command Words (ICWs)

The initialization process consists of writing a sequence of four bytes to each interrupt controller. The initialization sequence is started by writing the first Initialization Command Word (ICW1) to address 020h (0A0h) with a 1 on bit 4 of the data byte. The interrupt controller interprets this as the start of an initialization sequence and does the following:

1) The Initialization Command Word Counter is reset to 0.
2) ICW1 is latched into the device.
3) Fixed Priority Mode is selected.
4) IR0 is assigned the highest priority.
5) The Interrupt Mask Register is cleared.
6) The Slave Mode Address is set to 7.
7) Special Mask Mode is disabled.
8) IRR is selected for status read operations.

The next three I/O writes to address 021h (0A1h) will load ICW2 through ICW4. See Figure 4-5 for a flow chart of the initialization sequence. The initialization sequence can be terminated at any point (all four bytes must be written for the controller to be properly initialized) by writing to address 020h (0A0h) with a 0 in data bit 4. Note this will cause OCW2 or OCW3 to be written.

Figure 4-5 Initialization Sequence

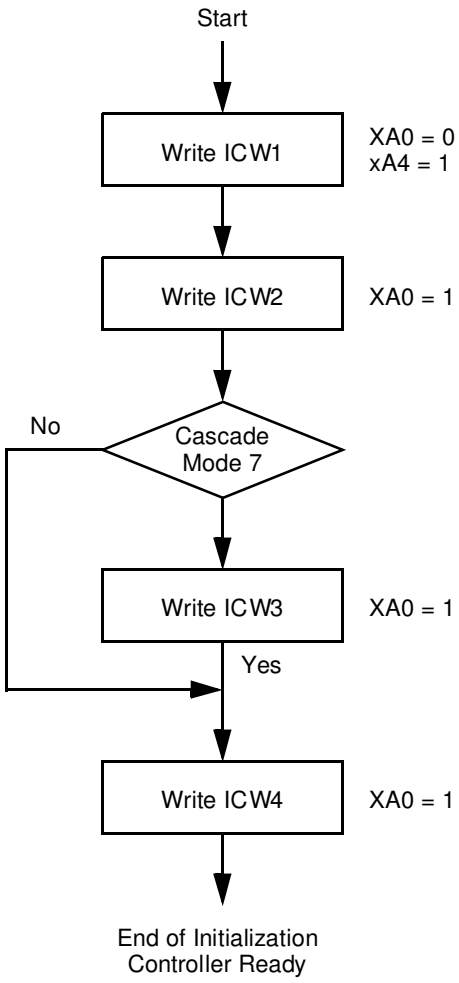


Table 4-15 ICW1 Register - Address: 020h (0A0h)

Bit(s)	Type	Default	Function
7:5	W	XXX	Don't Care
4	W	1	Must be set to 1 for ICW1 since ICW1, OCW2, and OCW3 share the same address, 020h (0A0h).
3	W		Level Trigger Mode: This bit selects either the Level Triggered Mode or Edge Triggered Mode input to the IR. If a 1 is written to LTM, a high level on the IR input will generate an interrupt request and the IR must be removed prior to EOI to prevent another interrupt. In the Edge Triggered Mode, a low-to-high will generate an interrupt request. In either mode, IR must be held high until the first INTA cycle is started in order to generate the proper vector. IR7 vector will be generated if the IR input is deasserted early.

Bit(s)	Type	Default	Function
2	W	X	Don't Care
1	W		Single Mode: This bit selects between the Single and Cascade Modes. The Single Mode is used whenever only one interrupt controller (INTC1) is used and is not recommended for this device. The Cascade Mode allows the two interrupt controllers to be connected through IR2 of INTC1. INTC1 will allow INTC2 to generate its own interrupt vectors if the Cascade Mode is selected and the highest priority IR pending is from an INTC2 input. INTC1 and INTC2 must be programmed for the Cascade Mode.
0	W	X	Don't Care

Table 4-16 ICW2 Register - Address: 021h (0A1h)

Bit(s)	Type	Default	Function
7:3	W		Vector bits 5 through 0: These bits are the upper five bits of the interrupt vector and are programmable by the CPU. INTC1 and INTC2 need not be programmed with the same value in ICW2. Usually INTC1 is programmed with 08h and INTC2 with 70h.
2:0	W	1	Vector bits 2 through 0: The lower three bits of the vector are generated by the Priority Resolver during INTA (see Table 4-14).

Table 4-17 ICW3 Register - Format for INTC1 - Address: 021h

Bit(s)	Type	Default	Function
7:0	W		Slave Mode bits 7 through 0: These bits select which IR inputs have Slave Mode controller connected. ICW3 in INTC1 must be written with 04h (IRQ2) for INTC2 to function correctly.

Table 4-18 ICW3 Register - Format for INCT1 - Address: 0A1h

Bit(s)	Type	Default	Function
7:3	W	XXXXX	Don't Care
2:0	W		Identify bits 2 through 0: Determines the Slave Mode address the controller will respond to during the cascade INTA sequence. ICW3 in INTC2 should be written with a 02h (IRQ2 of INTC1) for operation in the Cascade Mode.

Table 4-19 ICW4 Register - Address 021h (0A1h)

Bit(s)	Type	Default	Function
7:5	W	XXX	Don't Care
4	W	1	Enable Multiple Interrupts: This bit will enable multiple interrupts from the same channel in the Fixed Priority Mode. This allows INTC2 to fully nest interrupts when the Cascade and Fixed Priority Mode are both selected, without being blocked by INTC1. Correct handling in this type of mode requires the CPU to issue a non-specific EOI command to zero when exiting an interrupt service routine. If zero, a non-specific EOI command should be sent to INTC1. If non-zero, no command is issued.
3:2	W	XX	Don't Care

Bit(s)	Type	Default	Function
1	W		Auto End of Interrupt: An AEOI is enabled when this bit is 1. The interrupt controller will perform a non-specific EOI on the trailing edge of the second INTA cycle. Note this function should not be used in a device with fully nested interrupts unless the device is a cascade master type.
0	W	X	Don't Care

4.3.4.2 Operational Command Words (OCWs)

Operational Command Words (OCWs) allow the 82C206's interrupt controllers to be controlled or reconfigured at any time while operating. Each interrupt has three OCWs which can be programmed to affect the proper operating configuration and a status register to monitor controller operation.

OCW1 is located at address 021h (0A1h) and may be written any time the controller is not in the Initialization Mode. OCW2 and OCW3 are located at address 020h (0A0h). Writing to address 020h (0A0h) with a 0 in bit 4 will place the controller in the operating mode and load OCW2 (if data bit 3 = 0) or OCW3 (if data bit 3 = 1).

Table 4-20 OCW1 Register - Address: 021h (0A1h)

Bit(s)	Type	Default	Function
7:0	R/W		Mask bits 7 through 0: These bits control the state of the Interrupt Mask Register. Each Interrupt Register can be masked by writing a 1 in the appropriate bit position (M0 controls IR0, etc.). Setting an IMR bit has no affect on lower priority requests. All IMR bits are cleared by writing ICW1.

Table 4-21 OCW2 Register - Address: 020h (0A0h)

Bit(s)	Type	Default	Function
7:5	W		These bits are used to select various operating functions. Writing a 1 in bit 7 causes one of the rotate functions to be selected. Writing a 1 in bit 6 causes a specific or immediate function to occur. All specific commands require L[2:0] to be valid except no operation. Writing a 1 in bit 5 causes a function related to EOI to occur. **7 6 5 Function** 0 0 0 Clear Rotate in Auto-EOI mode 0 0 1 Non-specific EOI Command 0 1 1 No Operation 0 1 1 Specific EOI Command* 1 0 0 Set Rotate in Auto-EOI Mode 1 0 1 Rotate on Non-specific EOI Command 1 1 0 Set Priority Command* 1 1 1 Rotate on specific EOI Command *L[2:0] are used by these commands.
4:3	W		These bits must be set to 0 to indicate that OCW2 is selected, because ICW1, OCW2, and OCW3 share the same address. 020h (0A0h).
2:0	W		These three bits are internally decoded to select which interrupt channel is to be affected by the Specific command. L[2:0] must be valid during three of the four specific cycles.

Table 4-22 OCW3 Register - Address 020h (0A0h)

Bit(s)	Type	Default	Function
7	W	0	Reserved: This bits must be set to 0.
6:5	W		Enable Special Mask Mode: Writing a 1 in bit 5 enables the set/reset Special Mask Mode function. ESMM allows the other functions in OCW3 to be accessed and manipulated without affecting the Special Mask Mode (SMM) state. During SMM, writing a 1 to any bit position of OCW1 inhibits interrupts and a 0 enables interrupts on the associated channel by causing the Priority Resolver to ignore the condition of the ISR. **6 5 Function** 0 X No operation 1 0 Reset Special Mask Mode to Normal Mask Mode 1 1 Set Special Mask Mode
4:3	W	00	These bits must be set to 0 to indicate that OCW3 is selected because ICW1, OCW2, and OCW3 share the same address, 020h (0A0h).
2	W		Polled Mode: Writing a 1 to this bit of OCW3 enables the Polled Mode. Writing OCW3 with the Polled Mode acts like the first INTA cycle, freezing all interrupt request lines and resolving priority. The next read operation to the controller acts like a second INTA cycle and polled vector is output to the data bus. The format of polled vector is described later.
1:0	W		Read Register: A 1 to this bit enables the contents of IRR or ISR (determined by RIS) to be placed on XD[7:0] when reading the Status Port at address 020h (0A0h). Asserting PM forces RR to reset. **1 0 Function** 0 X No Operation 1 0 Read IRR on the next read 1 1 Read ISR on the next read

4.3.5 IRR, ISR, & Poll Vector

IRR, ISR, and Poll Vector are the same address, 020h (0A0h). The selection of the registers depends on the programming of ITC. If the latest OCW3 issued the poll command (PM = 1), the poll vector is selected for the next read. Before another poll command is issued, subsequent reads to the address will select IRR or ISR depending on the latest OCW3, if RR = 1 and RIS = 0, ISR is selected. Note that the poll command is cleared after the first read to the ITC. After initialization (ICW1 or reset), IRR is selected.

Table 4-23 through Table 4-25 give each of these registers' formats.

Many registers share the same I/O address of INTC. Table 4-26 summarizes the address of each register.

Table 4-23 IRR Register - Address: 020h (0A0h)

Bit(s)	Type	Default	Function
7:0			Interrupt Request bits 7 through 0: These bits correspond to the interrupt request bits of the Interrupt Request Register. A 1 on these bits indicate that an interrupt request is pending on the corresponding line.

Table 4-24 ISR Register - Address: 020h (0A0h)

Bit(s)	Type	Default	Function
7:0			Interrupt Service bits 7 through 0: These bits correspond to the interrupt service bits of the Interrupt Service Register. A 1 on these bits indicate that an interrupt is being serviced on the corresponding IS bits of the ISR.

Table 4-25 Poll Vector - Address: 020h (0A0h)

Bit(s)	Type	Default	Function
7			Interrupt: A 1 on this bit indicates that a pending interrupt is polled. If there is no pending interrupt request or the request is removed before the poll command, this bit is 0.
6:3		XXX	Don't Care
2:0			Vector bits 2 through 0: These bits are the binary encoding of the highest priority level pending interrupt request being polled. If no pending interrupt has been polled, all three bits are equal to 1.

Table 4-26 Register Summary

Write			
A0	**XD4**	**XD3**	**Register Selected**
0	0	0	OCW2
0	0	1	OCW3
0	1	X	ICW1
1	X	X	ICW2, ICW3, ICW4 during initialization sequence
			OCW1 (Mask Register) after initialization sequence
Read			
A0			**Register Selected**
0			IRR, ISR, or Poll Vector
1			OCW1 (Mask Register)

4.4 Counter/Timer Subsystem

The 82C206 contains an 8254 compatible counter/timer. The counter/timer can be used to generate accurate time delays under software control. It contains three 16-bit counters (Counters 2 through 0) which can be programmed to count in binary or binary-coded decimal (BCD). Each counter operates independently of the other and can be programmed for operation as a timer or a counter.

All counters in this subsystem are controlled by a common control logic as shown in Figure 4-6. The control logic decodes and generates the necessary commands to load, read, configure, and control each counter. Counter 0 and Counter 1 can be programmed for all six modes, but Mode 1 and Mode 5 have limited usefulness because their gate is hard-wired to GND internally. Counter 2 can be programmed to operate in any of the six modes:

- Mode 0 - Interrupt on terminal count
- Mode 1 - Hardware retriggerable one-shot
- Mode 2 - Rate generator
- Mode 3 - Square wave generator
- Mode 4 - Software triggered strobe
- Mode 5 - Hardware retriggerable strobe

All three counters are driven from a common clock input pin (TMRCLK) which is different from other clock inputs to the 82C206. Counter 0's output (OUT0) is internally connected to IR0 of INTC1 and may be used as an interrupt to the system for time keeping and task switching. Counter 1 may be programmed to generate pulses or square waves for external

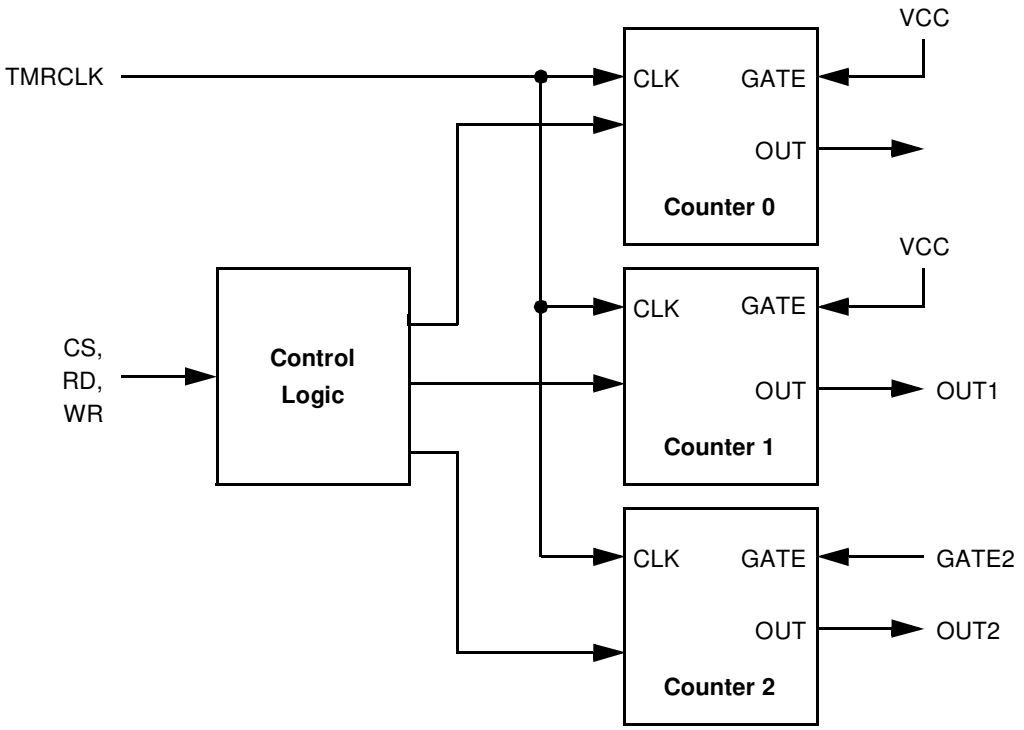
devices. Counter 2 is a full function counter/timer. It can be used as an interval timer, a counter, or as a gated rate/pulse generator. In a PC/AT compatible design, Counter 0 is used as a system timer, Counter 1 is used as a DRAM refresh rate generator, and Counter 2 is used for speaker sound generation.

4.4.1 Counter Description

Each counter in this subsystem contains a control register, a status register, a 16-bit counting component, a pair of 8-bit counter input latches, and a pair of 8-bit counter output latches. Each counter shares the same clock input (TMRCLK). GATE0, GATE1, and OUT0 are not externally accessible. This is fully compatible with a PC/AT-based design. Output of OUT0 is dependent on the counter mode.

The control register stores the mode and command information used to control the counter. It may be loaded by writing a byte to the write control word at Port 043h. The status register allows the software to monitor counter conditions and read back the contents of the control register.

Figure 4-6 Counter/Timer Block Diagram



The 16-bit counting component is a loadable synchronous down counter. It is loaded or decremented on the falling edge of TMRCLK. The counting component contains a maximum count when a 0 is loaded, which is equivalent to 65536 in binary operation or 1000 in BCD. The counting component does not stop when it reaches 0. In Modes 2 and 3, the counting component will be reloaded and in all other modes it will wrap around to 0FFFFh in binary operation or 9999 in BCD.

The counting component is indirectly loaded by writing one or two bytes (optional) to the counter input latches, which are in turn loaded into the counting component. Thus, the counting component can be loaded or reloaded in one TMRCLK cycle. The counting component is also read indirectly by reading the contents of the counter output latches. The counter output latches are transparent latches which can be read while transparent or latched (see Latch Counter Command).

4.4.2 Programming the Counter/Timer

After a system reset, the contents of the control registers, counter registers, counting components, and the output of all counters are undefined. Each counter must be programmed before it can be used. Each counter is programmed by writing its control register with a control word and then giving an initial count to its counting component. Table 4-27 lists the I/O address map used by the counter/timer subsystem.

Table 4-27 Counter/Timer I/O Address Map

Address	Function
040h	Counter 0 Read/Write
041h	Counter 1 Read/Write
042h	Counter 2 Read/Write
043h	Control Register Write Only

4.4.2.1 Read/Write Counter Command

Each counter has a write only control register. This control register is written with a control word to the I/O address 043h. The control word format is described in Table 4-28.

Table 4-28 Control Word Format (Write Only)

Bit(s)	Type	Default	Function
7:6	W		Select Counter bits 1 and 0: These bits select which counter this control word is written to. **7 6 Function** 0 0 Select Counter 0 0 1 Select Counter 1 1 0 Select Counter 2 1 1 Reserved for read-back command
5:4	W		Read/Write bits 1 and 0: These bits determine the counter read/write word size. **5 4 Function** 0 0 Reserved for counter latch command 0 1 Read/write LSB only 1 0 Read/write MSB only 1 1 Read/write LSB first, then MSB
3:1	W		Mode Select bits 2 through 0: These bits select the counter operating mode. **3 2 1 Function** 0 0 0 Select Mode 0 0 0 1 Select Mode 1 X 1 0 Select Mode 2 X 1 1 Select Mode 3 1 0 0 Select Mode 4 1 0 1 Select Mode 5
0	W		Binary Coded Decimal: During read/write counter commands control word writing, a 1 selects binary coded decimal count format. A 0 selects binary counting format. During read-back command word writing, this bit must be 0.

When programming to a counter, the following steps must sequentially occur:

1) Each counter's control register must be written with a control word before the initial count is written.
2) Writing the initial count must follow the format specified in the control word (least significant bit only, most significant bit only, or least significant bit and then most significant bit.

A new initial count can be written into the counter at any time after programming without rewriting the control word.

4.4.2.2 Counter Latch Command

When a counter latch command is issued, the counter's output latches latch the current state of the counting component. The counter's output latches remain latched until read by the CPU or the counter is reprogrammed. After that, the output latches then returns to a "transparent" condition. Counter latch commands may be issued to more than one counter before reading the first counter to which this command was issued. Also, multiple counter latch commands issued to the same counter without reading the counter will cause all but the first command to be ignored. Table 4-29 describes the counter latch command format.

4.4.2.3 Read-Back Command

The read-back command allows the user to check the count value, mode, and state of the OUT signal and null count flag of the selected counter(s). The format of the read-back command is given in Table 4-30.

Each counter's latches remain latched until either the latch is read or the counter is reprogrammed. If both LSTATUS and LCOUNT are 0, the status will be returned on the next read from the counter. The next one or two reads (depending on whether the counter is programmed to transfer one or two bytes) from the counter result in the count being returned. Multiple read-back commands issued to the same counter without reading the counter will cause all but the first command to be ignored. The status read from each counter is described in Table 4-31.

Table 4-29 Counter Latch Command Format (Write Only)

Bit(s)	Type	Default	Function
7:6	W		Select Counter bits 1 and 0: These bits select which counter is being latched. **7** **6** **Function** 0 0 Select Counter 0 0 1 Select Counter 1 1 0 Select Counter 2 1 1 Reserved for read-back command
5:4	W	00	These bits must be 0 for the counter latch command.
3:0	W	XXXX	Don't Care

Table 4-30 Read-Back Command Format (Write Only)

Bit(s)	Type	Default	Function
7:6	W	11	These bits must be 1 for the read-back command
5	W		Latch Count: A 0 in this bit will latch the count of the counting component of the selected counter(s);
4	W		Latch Status: A 0 in this bit will latch the status information of the selected counter(s).
3:1	W		Counter Select Bits 2 through 0: These bits select which counter(s) the read-back command is applied to. **3** **2** **1** **Function** 0 X X Select Counter 2 X 0 X Select Counter 1 X X 0 Select Counter 0
0	W	0	Reserved: Write as 0.

Table 4-31 Status Format

Bit(s)	Type	Default	Function
7	R		Out: This bit contains the state of the OUT signal of the counter.
6	R		Null Count: This bit contains the condition of the null count flag. This flag is used to indicate that the contents of the counting element are valid. It will be set to 1 during a write to the control register or the counter. It is cleared to a 0 whenever the counter is loaded from the counter input register.
5:4	R		Read/Write Word bits 1 and 0: These bits indicate the counter read/write word size. This information is useful in determining where the high byte, the low byte, or both must be must be transferred during counter read/write operations.

Bit(s)	Type	Default	Function
3:1	R		Mode bits 2 through 0: These bits reflect the operating mode of the counter and are interpreted in the same manner as in the write control word format.
0	R		Binary Coded Decimal: This bit indicates the counting element is operating in binary format or BCD format.

4.4.3 Counter Operation

Since Counter 1 and 0 have limitations in some of their operation modes, Counter 2 will be used to describe the various counter operating modes. However, the description of Modes 0, 2, 3, and 4 are suitable for all counters. The following terms are defined for describing the counter/timer operation.

- TMRCLK pulse - A rising edge followed by a falling edge of the 82C206's TMRCLK input.
- Trigger - The rising edge of the GATE2 input.
- Counter Load - the transfer of the 16-bit value in counter input latches to the counting element.
- Initialized - A control word written and the counter input latches loaded.
- Counter 2 can operate in one of the following modes:
  - Mode 0 - Interrupt on terminal count
  - Mode 1 - Hardware retriggerable one-shot
  - Mode 2 - Rate generator
  - Mode 3 - Square wave generator
  - Mode 4 - Software triggered strobe
  - Mode 5 - Hardware triggered strobe

4.4.3.1 Mode 0 - Interrupt on Terminal Count

Mode 0 is usually used for event counting. After a counter is written with the control word, OUT2 of that counter goes low and remains low until the counting element reaches 0, at which time it goes back high and remains high until a new count or control word is written. Counting is enabled when GATE2 = 1 and disabled when GATE2 = 0. GATE2 has no effect on OUT2.

The counting component is loaded at the first TMRCLK pulse after the control word and initial count are loaded. When both initial count bytes are required, the counting component is loaded after the high byte is written. This TMRCLK pulse does not decrement the count, so for an initial count of N, OUT2 does not go high until N + 1 TMRCLK pulses after initialization. Writing a new initial count to the counter reloads the counting element on the next TMRCLK pulse and counting continues from the new count. If an initial count is written with GATE2 = 0, it will still be loaded on the next TMRCLK pulse. But counting does not progress until GATE2 = 1. When GATE2 goes high, OUT2 will go high after N TMRCLK pulses later.

4.4.3.2 Mode 1 - Hardware Retriggerable One-Shot

Writing the control word causes OUT2 to go high initially. Once initialized, the counter is armed and a trigger causes OUT2 to go low on the next TMRCLK pulse. OUT2 then remains low until the counter reaches 0. An initial count of N results in a one-shot pulse N TMRCLK cycles long. Any subsequent triggers while OUT2 is low cause the counting component to be reloaded, extending the length of the pulse. Writing a new count to the counter input latches will not affect the current one-shot pulse unless the counter is retriggered. In the latter case, the counting component is loaded with the new count and the one-shot pulse continues until the new count expires.

4.4.3.3 Mode 2 - Rate Generator

This mode functions as a divide-by-N counter. After writing the control word during initialization, the counter's OUT2 is set to high. When the initial count is decremented to 1, OUT2 goes low on the next TMRCLK pulse. The following TMRCLK pulse returns OUT2 high, reloads the CE, and the process is repeated. In Mode 2, the counter continues counting (if GATE2 = 1) and will generate an OUT2 pulse every N TMRCLK cycles. Note that a count of 1 is illegal in Mode 2.

GATE2 = 0 disables counting and forces OUT2 high immediately. A trigger reloads the CE on the next TMRCLK pulse. Thus, GATE 2 can be used to synchronize the counter to external events.

Writing a new count while counting does not affect current operation unless a trigger is received. Otherwise, the new count will be loaded at the end of the current counting cycle.

4.4.3.4 Mode 3 - Square wave Generator

Mode 3 is similar to Mode 2 in every respect except for the duty cycle of OUT2. OUT2 is set high initially and remains high for the first half of the count. When the first half of the initial count expires, OUT2 goes low for the remainder of the count.

If the counter is loaded with an even count, the duty cycle of OUT2 will be 50% (high = low = N/2). For odd count values, OUT2 is high one TMRCLK cycle longer than it is low. Therefore, high = (N + 1)/2 and low = (N - 1)/2.

4.4.3.5 Mode 4 - Software Triggered Strobe

Writing the Control Word causes OUT2 To go high initially. Expiration of the initial count causes OUT2 to go low for one

TMRCLK cycle. GATE2 = 0 disables counting but has no effect on OUT2. Also, a trigger will not reload CE.

The counting sequence is started by writing the initial count. The CE is loaded on the TMRCLK pulse after initialization. The CE begins decrementing one TMRCLK pulse later, OUT2 will go low for one TMRCLK cycle, (N + 1) cycles after the initial count is written.

If a new initial count is written during a counting sequence, it is loaded into the CE on the next TMRCLK pulse and the sequence continues from the new count. This allows the sequence to be "retriggerable" by software.

4.4.3.6 Mode 5 - Hardware Triggered Strobe

Writing the Control Word causes OUT2 to go high initially. Counting is started by a trigger. The expiration of the initial count causes OUT2 to go low for one TMRCLK cycle. GATE2 = 0 disables counting.

The CE is loaded on the TMRCLK pulse after a trigger. Since loading the CE inhibits decrementing, OUT2 will go low for one TMRCLK cycle, (N + 1) TMRCLK cycles after the trigger.

If a new count is loaded during counting, the current counting sequence will not be affected unless a trigger occurs. A trigger causes the counter to be reloaded from CIL and CIH making the counter "retriggerable".

4.4.3.7 GATE2

In Modes 0, 2, 3, and 4 GATE2 is level-edge sensitive and is sampled on the rising edge of TMRCLK. In Modes 1, 2, 3, and 5 the GATE2 input is rising-edge sensitive. This rising edge sets an internal flip-flop whose output is sampled on the rising edge of TMRCLK. The flip-flop resets immediately after being sampled. Note that in Modes 2 and 3, the GATE2 input is both edge and level sensitive. Table 4-32 details this operation.

Table 4-32 GATE2 Pin Function

Mode	GATE2		
	Low	Rising	High
0	Disables counting		Enables counting
		A) Initiates counting B) Reset OUT2 pin	
2	A) Disables counting B) Forces OUT2 pin high	Initiates counting	Enables counting
3	A) Disables counting B) Forces OUT2 pin high	Initiates counting	Enables counting
4	Disables counting		Enables counting
5		Initiates counting	

4.5 Real-time Clock Subsystem

This subsystem of the 82C206 integrates a complete time-of-day real-time clock with alarm, one hundred year calendar, a programmable periodic interrupt, and 114 bytes of CMOS static RAM. The 82C206 is designed to operate in a low-power (battery powered) mode and protects the contents of both the CMOS static RAM and clock from change during system power-up/down.

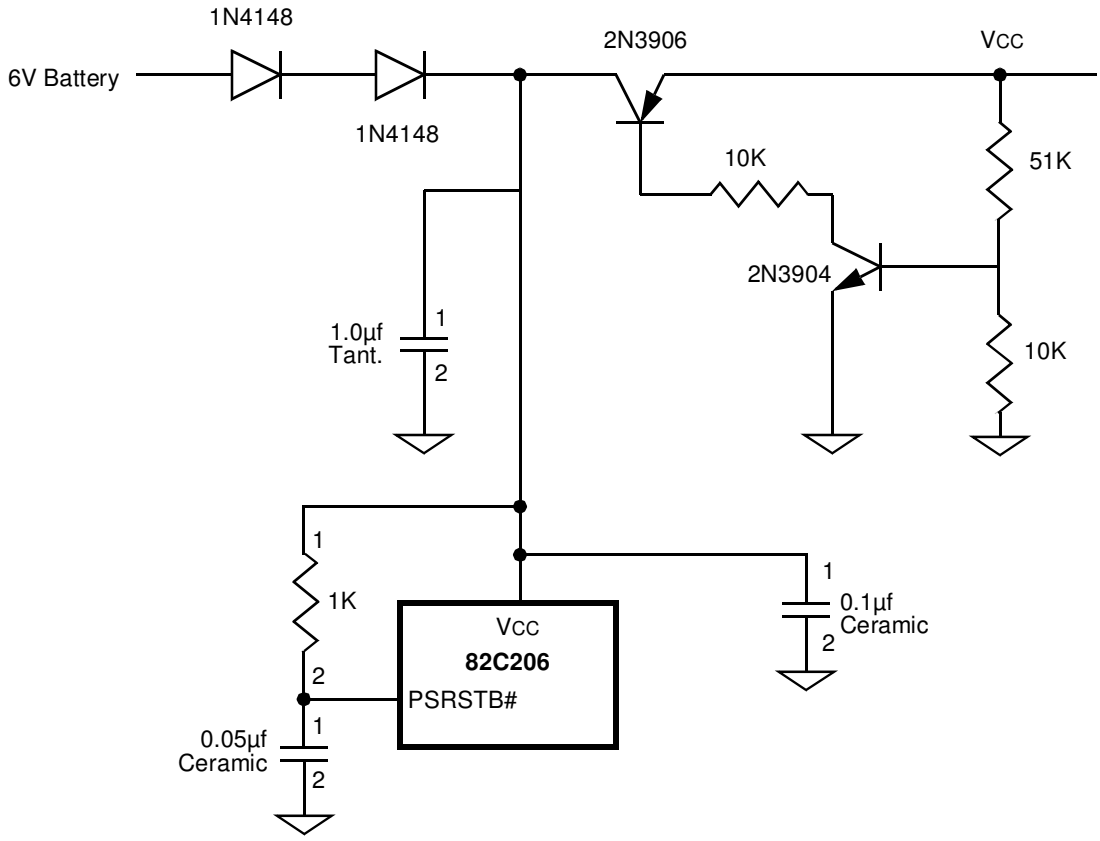
4.5.1 Power-Up/Down

Most applications will require the real-time clock to remain active whenever the system power is turned off. To accomplish this, the user must provide an alternate source of power to the 82C206. This alternate source of power is normally provided by connecting a battery to the V_{CC} pin to switch from the system power supply to the battery. A circuit implementing such a function is shown in Figure 4-7. It is used to eliminate the power drain on the batter when the entire 82C206 is active. It will also make a clean and reliable transition between system and battery power without drawing too much battery power.

The PWRGD pin is provided on the device to protect the contents of RAM and the real-time clock. It is also used to reduce power consumption whenever the system is powered down. This pin should be low whenever the system power supply is not within specifications for proper operation of the system. This pin may be driven by circuitry in either the power supply or on the system board. When the PWRGD input is low, it will disable all unnecessary inputs and outputs. In this way, it will prevent noise on the inactive pins and reduce leakage current when the system is powered down. This pin must therefore be at a high level for the remainder of the device to operate properly when system power is applied.

The PSRSTB# pin is provided to initialize the device whenever power is applied to the 82C206. This pin will not alter the RAM or real-time clock contents but it will initialize the necessary control register bits. A low on the PSRSTB# pin disables the generation of interrupts and sets a flag indicating that the contents of the device may not be valid. A recommended circuit for controlling the PSRSTB# input is shown in Figure 4-7.

Figure 4-7 Power Conversion & Reset Circuitry



4.5.2 Register Access

Reading and writing to the 128 locations in this subsystem is accomplished by first placing the index address of the location you wish to access on the data input pins XD[6:0] and then strobing the AS input pin. The address will then be latched into the Index Address Register on the falling edge of AS. The Index Address Register is then used as a pointer to the specific byte in this subsystem, which may be read or written by asserting XIOR# or XIOW# with an address of 071h on the XA[9:0] input pins.

In PC/AT-compatible designs, AS is generated by an I/O write operation to Port 070h. To avoid an unintentional change of the contents of the real-time clock and CMOS RAM, it is recommended that an address of 070h be applied to the XA[9:0] inputs of the 82C206 during the time AS is asserted.

4.5.2.1 Address Map

Table 4-33 illustrates the internal register/RAM organization of the real-time clock subsystem of the 82C206. The 128 addressable locations in this subsystem are divided into ten bytes which normally contain the time, calendar, alarm setting, four control and status bytes, and 114 general purpose CMOS RAM bytes. All 128 bytes can be read by the CPU. The CPU may also write to all locations except registers 0Ch, 0Dh, bit 7 of Register 0Ah and bit 7 of the Register 00h, which is always 0.

4.5.2.2 Time, Calendar, & Alarm Bytes

The CPU can obtain the time and calendar information by reading the appropriate locations in the real-time clock. Initialization of the time, calendar, and alarm information is accomplished by writing to these bytes. Data stored in these locations are in binary-coded decimal (BCD) format.

Before initialization of the internal registers of the real-time clock, the SET bit in Register 0Bh should be set to 1 to prevent the real-time clock from updating. The CPU then initializes the first ten locations in BCD format. The SET bit should then be cleared to allow updating. After initialized and enabled, the real-time clock will perform clock/calendar updates at a 1.024kHz rate in PC/AT-compatible designs.

The alarm bytes can be programmed to generate an interrupt at a specific time or they can be programmed to generate a periodic interrupt. To generate an interrupt at a specific time, the user need only program the time that the interrupt is to occur into the three alarm bytes. Alternately, a periodic interrupt can be generated by setting the high order two bits in an alarm register to a 1, which turns that byte into a "don't care". For instance, an interrupt can be generated once a second by programming the same value into all three alarm registers.

Table 4-34 shows the format for the ten clock, calendar, and alarm registers. The 24/12 bit in Register 0Bh determines whether the hour locations will be updated using a 1-12 or 0-23 format. After initialization, the 24/12 bit cannot be changed without reinitializing the hour locations. In a 12-hour format, bit 7 of the hours byte in both the time and alarm bytes will indicate PM when it is a 1.

Table 4-33 Real-Time Clock Address Map

Index	Function	Index	Function
00h	Seconds	0Ah	Register A
01h	Seconds Alarm	0Bh	Register B
02h	Minutes	0Ch	Register C
03h	Minutes Alarm	0Dh	Register D
04h	Hours	0Eh	User RAM
05h	Hours Alarm	0Fh	User RAM
06h	Day of Week	:	
07h	Day of Month	:	
08h	Month	7Eh	User RAM
09h	Year	7Fh	User RAM

Table 4-34 Time, Calendar, Alarm Data Format

Index	Function	BCD Range
00h	Seconds	00-59
01h	Seconds Alarm	00-59
02h	Minutes	00-59
03h	Minutes Alarm	00-59
04h	Hours, 12-Hour Mode Hours, 24-Hour Mode	01-12 (AM) 81-92 (PM) 00-23
05h	Hours Alarm, 12-Hour Mode Hours Alarm, 24-Hour Mode	01-12 (AM) 81-92 (PM) 00-23
06h	Day of Week	01-07
07h	Day of Month	01-31
08h	Month	01-12
09h	Year	00-99

4.5.2.3 Update Cycle

During normal operation, the real-time clock will perform an update cycle assuming one of the proper time bases is chose, the divider bits DV[2:0] are not reset and the SET bit in Register 0Bh is cleared. The function of the update cycle is to increment the clock/calendar registers and compare them to the alarm registers. If a match or don't care condition occurs between the two sets of registers, an alarm is issued and interrupt control bits are enabled.

During an update cycle, the lower ten registers are not accessible by the CPU. This way it can prevent the possible corruption of data in the real-time clock registers or the reading of incorrect data. To avoid contention between the real-time clock and the CPU, a flag is provided in Register 0Ah to alert the user of an update-in-progress cycle. The update-in-progress bit (UIP) is asserted 244µs before the actual start of the cycle and is maintained until the cycle is complete. Once the cycle is complete, the UIP bit will be cleared and the update flag (UF) in Register 0Ch will be set. Figure 4-8 illustrates the update cycle. CPU access is always allowed to registers 0Ah-0Dh during update cycles.

Two recommended methods can be used for reading and writing to the real-time clock in a PC/AT-compatible design. Both of them will allow the user to avoid contention between the CPU and the real-time clock for access to the time and date data.

The first method is to read Register 0Ah, determine the state of the UIP bit and if it is 0, perform the read or write operation. For this method to work successfully, the entire read or write operation (including any interrupt service routines which might occur) must not require longer than 244µs to complete from the beginning of the read of Register 0Ah to the completion of the last read or write operation to the clock calendar registers.

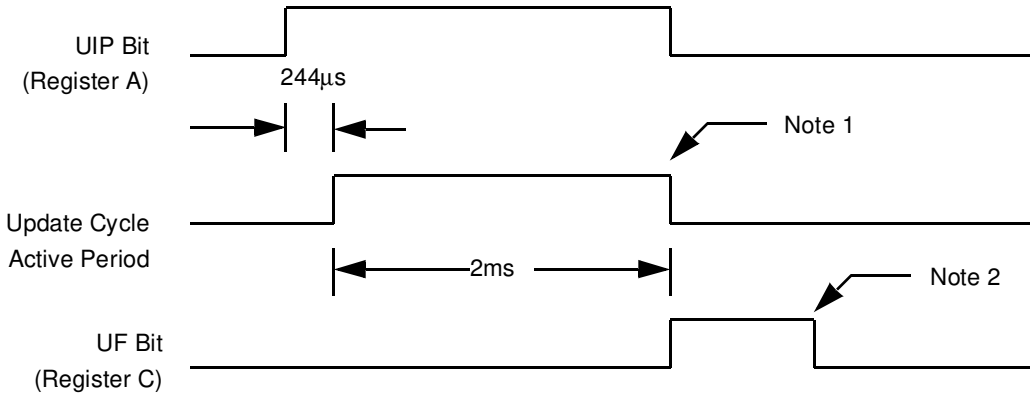
The second method of accessing the lower ten registers is to read Register 0Ch once and disregard the contents. Then, subsequently continue reading this register until the UF bit is a 1. This bit will become true immediately after an update

cycle has been completed. The user then has to complete a read or write operation before the next update cycle.

4.5.2.4 Control and Status Registers

The 82C206 contains four registers used to control the operation and monitor the status of the real-time clock. The CPU can access these registers at any time with the Index Address at 0Ah-0Dh.

Figure 4-8 Update Cycle



Notes: 1. Registers 0-9 are unavailable for read or write this time.

2. UF bit cleared by reading Register C.

Table 4-35 Register 0Ah, (Index Port 70h, Data Port 71h)

Bit(s)	Type	Default	Function
7	R		Update-In-Progress: The update-in-progress flag is a status bit used to indicate when an update cycle is about to take place. A 1 indicates that an update cycle is taking place or is imminent. UIP will go active (high) 244µs prior to start of an update cycle and will remain active for an additional 2ms while the update is occurring. The UIP bit is a read only bit and is not affected by a reset. Writing a 1 to the SET bit in Register 0Bh will clear the UIP status bit.
6:4	R/W		Divider bits 2 through 0: These bits are used to control the divider/prescaler on the real-time clock. While the 82C206 can operate at frequencies higher than 32.768kHz, this is not recommended for battery powered operation due to the increased power consumption at these higher frequencies. **6 5 4 Divider Options** 0 0 0 4.194304MHz 0 0 1 1.048576MHz 0 1 0 32.768kHz 1 1 X Reset Divider

Bit(s)	Type	Default	Function
3:0	R/W		Rate Select: These bits control the periodic interrupt rate. The periodic interrupt is derived from the divider/prescaler in the real-time clock and is separated from the alarm interrupt. Both the alarm and periodic interrupts do, however, use the same interrupt channel in the interrupt controller. Use of the periodic interrupt allows the generation of interrupts at rates higher than once per second. Below are the interrupt rates for which the real-time clock can be programmed.

3	**2**	**1**	**0**	**4.194304MHz & 1.048576MHz**	**32.768kHz**
0	0	0	0	None	None
0	0	0	1	30.517µs	3.90625ms
0	0	1	0	61.035µs	7.8125 ms
0	0	1	1	122.070µs	122.070µs
0	1	0	0	244.141µs	244.141µs
0	1	0	1	488.281µs	488.281µs
0	1	1	0	976.562µs	976.562µs
0	1	1	1	1.953125ms	1.953125ms
1	0	0	0	3.90625ms	3.90625ms
1	0	0	1	7.8125ms	7.8125ms
1	0	1	0	15.625ms	15.625ms
1	0	1	X	31.25ms	31.25ms
1	1	X	1	62.5ms	62.5ms
1	1	0	1	125ms	125ms
1	1	1	0	250ms	250ms
1	1	1	1	500ms	500ms

Table 4-36 Register 0Bh (Index Port 70h, Data Port 71h)

Bit(s)	Type	Default	Function
7	R/W		Set: Writing a 0 to this bit enables the update cycle and allows the real-time clock to function normally. When set to a 1, the update cycle is inhibited and any cycle in progress is aborted. The SET bit is not affected by the RESET input pin.
6	R/W		Periodic Interrupt Enable: This bit controls the generation of interrupts based on the value programmed into the RS[3:0] bits of Register 0Ah. This allows the user to disable this function without affecting the programmed rate. Writing a 1 to this bit enables the generation of periodic interrupts. It is cleared to 0 by a reset.
5	R/W		Alarm Interrupt Enable: Setting this bit to 1 enables the generation of alarm interrupts. Once enabled, the real-time clock will generate an alarm whenever a match occurs between the programmed alarm and clock information. If the “don’t care” condition is programmed into one or more of the alarm registers, this will enable the generation of periodic interrupts at rates of one second or greater. This bit is cleared by a reset.
4	R/W		Update Ended Interrupt Enable: This bit is used to enable the update end flag (UF) bit in Register 0Ch to generate an interrupt. A 1 in this bit enables the interrupt generating. A 0 disables it. This bit is cleared by a reset.It is also cleared when the SET bit goes high.
3	R/W	0	Square Wave: This bit is always fixed to 0. It disables the square wave generation.
2	R/W	0	Data Mode: This bit is always fixed to 0. It always selects the BCD format for the real-time clock.
1	R/W		24/12 Hour Mode: This bit controls the format of both the hour and hour alarm bytes. If 1, the real-time clock will interpret and update the information in these two bytes using the 24 Hour Mode. This bit can be read or written by the CPU and is not affected by a reset.

Bit(s)	Type	Default	Function
0	R/W		Daylight Savings Enable: The real-time clock can be instructed to handle daylight savings time changes by setting this bit to 1. This enables two exceptions to the normal time keeping sequence to occur. On the first Sunday in April the time increments from 1:59:59 AM to 3:00:00 AM. On the last Sunday in October when the time first reaches 1:59:59 AM it changes to 1:00:00 AM. Setting this bit to 0 disables the execution of these two exceptions. A reset has no effect on this bit.

Table 4-37 Register 0Ch (Index Port 70h, Data Port 71h)

Bit(s)	Type	Default	Function
7	R		Interrupt Request Flag: When set to 1, any of the conditions which can cause an interrupt is true and the interrupt enable for that condition is true. The condition which causes this bit to be set also generates an interrupt. The logic expression for this flag: IRQF = PF & PIE + AF & AIE + UF & UIE This bit and all other active bits in this register are cleared by reading the register or by activating the PSRSTB# input pin. Writing to this register has no effect on the contents.
6	R		Periodic Interrupt Flag: This bit is set to a 1 when a transition (which is selected by the RS[3:0] bits of Register 0Ah) occurs in the divider chain. This bit will become active independent of the condition of the PIE control bit. The PF bit will then generate an interrupt and set IRQF if PIE is a 1.
5	R		Alarm Flag: A 1 appears in this bit whenever a match has occurred between the time registers and alarm registers during an update cycle. This flag is also independent of its enable (AIE) and will generate an interrupt if AIE is true.
4	R		Update Ended Flag: A 1 appears in this bit whenever an update cycle is ended. This flag is also independent of its enable (UIE) and will generate an interrupt if UIE is true.
3:0	R	0000	Reserved, Not Used: All unused bits will be 0 when read and can not be written to.

Table 4-38 Register 0Dh (Index Port 70h, Data Port 71h)

Bit(s)	Type	Default	Function
7	R		Valid RAM and Time: This bit indicates the condition of the contents of the RAM and real-time clock. It is cleared to 0 whenever the PSRSTB# input pin is low. This is normally derived from the power supply which supplies power to the device and will allow the user to determine whether the registers have been initialized since power was applied to the device. A reset has no effect on this bit and it can only be set by reading Register 0Dh.
6:0	R	0000 000	Reserved, Not Used: All unused bits will be 0 when read and can not be written to.

4.5.2.5 CMOS Static RAM

The 114 bytes of RAM from index address 0Eh to 7Fh are not affected by the real-time clock. They are accessible during the update cycle and may be used for whatever the designer wishes. Typical applications will use these as non-volatile storage for system configuration parameters. They are normally battery power when the system is turned off.

5.0 Maximum Ratings

Stresses above those listed in the following tables may cause permanent damage to the device. These are stress ratings only and functional operation of the device at these or any other conditions above those indicated in the operational sections of this specification are not implied.

5.1 Absolute Maximum Ratings

Symbol	Parameter	Min	Max	Unit
Vcc	Supply Voltage	-0.3	+6.7	V
Vi	Input Voltage	-0.3	+6.7	V
Vo	Output Voltage	-0.3	Vcc + 0.3	V
Top	Operating Temperature	-20	+70	°C
Tstg	Storage Temperature	-55	+125	°C

5.2 DC Characteristics (Vcc = 4.75V to 5.25V, TA = -20°C to +70°C)

Symbol	Parameter	Min	Max	Unit
Vil	Input Low Voltage	-0.3	+0.8	V
Vih	Input High Voltage	+2.2	Vcc + 0.3	V
Vol	Output Low Voltage		0.5	V
Voh	Output High Voltage	3.5		V
Iil	Input Leakage Current		1.0	µA
Iozi	Output High-Z Leakage Current		1.0	µA
Iccsb	Standby Power Supply Current		10.0	µA

5.3 AC Characteristics (Vcc = 4.75V to 5.25V, TA = -20°C to +70°C)

Symbol	Parameter	Min	Max	Unit
t1	Address Setup to Command Active	25		ns
t2	Command Active Period		200	ns
t3	Address Hold Time from Command Inactive	0		ns
t4	Data Valid Delay	160		ns
t5	Data Hold Time from XIOR# Inactive	10		ns
t6	XD[7:0] Active form XIOR#	5	40	ns
t7	Data Setup to XIOW# Inactive	160		ns
t8	Data Hold Time from XIOW# Inactive	0		ns
t9	Command Recovery Time	120		ns
t10	Interrupt Request Low Width	100		ns
t11	Interrupt Request High Width	200		ns
t12	INTR Output Delay		300	ns

Symbol	Parameter	Min	Max	Unit
t20	Real-Time Clock Cycle Time		500	ns
t21	AS Pulse Width	160		ns
t22	Data Valid Setup to AS Inactive	160		ns
t23	Data Hold Time from AS Inactive	0		ns
t24	OSCI Period	500		ns
t25	OSCI High Time	200		ns
t26	OSCI Low Time	200		ns
t27	PSRSTB# High Delay from VCC	5		µs
t28	PSRSTB# Low Time	5		µs
t29	VRT Bit Valid Delay		2	ns
t40	TMRCLK Period	125		ns
t41	TMRCLK Low Time	50		ns
t42	TMRCLK High Time	50		ns
t43	GATE2 Setup to TMRCLK	50		ns
t44	GATE2 Hold Time from TMRCLK	50		ns
t45	GATE2 Low Time	50		ns
t46	GATE2 High Time	50		ns
t47	OUT2 Delay from TMRCLK		120	ns
t48	OUT2 Delay from GATE2		120	ns
t50	SYSCLK Period	125		ns
t50a	SYSCLK Period	62		ns
t51	SYSCLK Low Time	43		ns
t51a	SYSCLK Low Time	22		ns
t52	SYSCLK High Time	55		ns
t52a	SYSCLK High Time	27		ns
t53	DREQ# Setup to SYSCLK	0		ns
t54	HRQ valid from SYSCLK		75	ns
t55	HLDA1 Setup to SYSCLK	45		ns
t56	AEN# Valid Delay from SYSCLK		105	ns
t57	AEN# Invalid Delay from SYSCLK		90	ns
t58	ADSTB# Valid Delay from SYSCLK		50	ns
t59	ADSTB# Invalid Delay from SYSCLK		120	ns
t60	XD[7:0] Active Delay from SYSCLK		60	ns
t61	XD[7:0] Valid Setup to ADSTB# Low	64		ns
t62	XD[7:0] Hold Time from ADSTB# Low	25		ns
t63	XD[7:0] Tristate Delay from SYSCLK		135	ns
t64	Address Valid Delay from SYSCLK		60	ns

Symbol	Parameter	Min	Max	Unit
t65	Address Hold Time from DMAMEMR# High	50		ns
t66	Address Tristate Delay from SYSCLK		55	ns
t67	DACK# Delay From SYSCLK		105	ns
t68	Command Enable Delay From SYSCLK		90	ns
t69	Command Active Delay from SYSCLK		120	ns
t70	Write Command Inactive Delay from SYSCLK		80	ns
t71	Address Command	75		ns
t72	Command Tristate Delay from SYSCLK		75	ns
t73	Read Command Inactive Delay from SYSCLK TC Delay from SYSCLK		115	ns
t74	TC Delay from SYSCLK		60	ns
t75	XD[7:0] Setup to read Command Inactive	90		ns
t76	XD[7:0] Hold Time from Read Command Inactive	0		ns
t77	XD[7:0] Hold from Write Command Inactive		120	ns
t78	IOCHRDY Input Setup to SYSCLK	15		ns
t79	IOCHRDY Input Setup to SYSCLK	35		ns
t80	IOCHRDY Input Hold Time from SYSCLK	20		ns

5.4 Samsung KS83C206 Data Sheet

Figure 5-1 AC Characteristics Measurement Waveform
TBD

Figure 5-2 Peripheral Read/Data Cycle, Write Cycle
TBD

Figure 5-3 Peripheral Read/INTA Cycle
TBD

Figure 5-4 Peripheral Write Cycle
TBD

Figure 5-5 Peripheral Read/Write Cycle
TBD

Figure 5-6 Command Recovery
TBD

Figure 5-7 INTA Sequence
TBD

Figure 5-8 Real Time Clock Access Cycle
TBD

Figure 5-9 Real Time Clock Power-up Sequence
TBD

Figure 5-10 Counter/Timer Parameters
TBD

Figure 5-11 IOCHRDY Output
TBD

Figure 5-12 DMA Reset
TBD

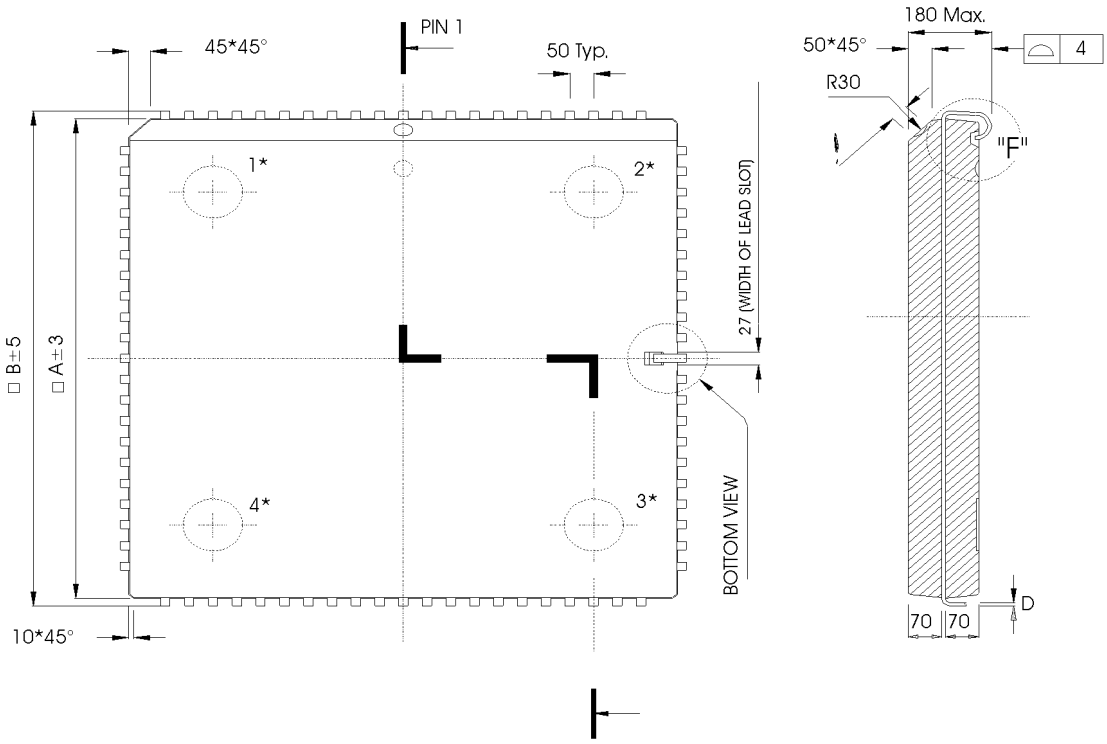
Figure 5-13 DMA Transfer Timing
TBD

Figure 5-14 Memory-to-Memory Transfer
TBD

Figure 5-15 Compressed Transfer
TBD

6.0 Mechanical Package Outline

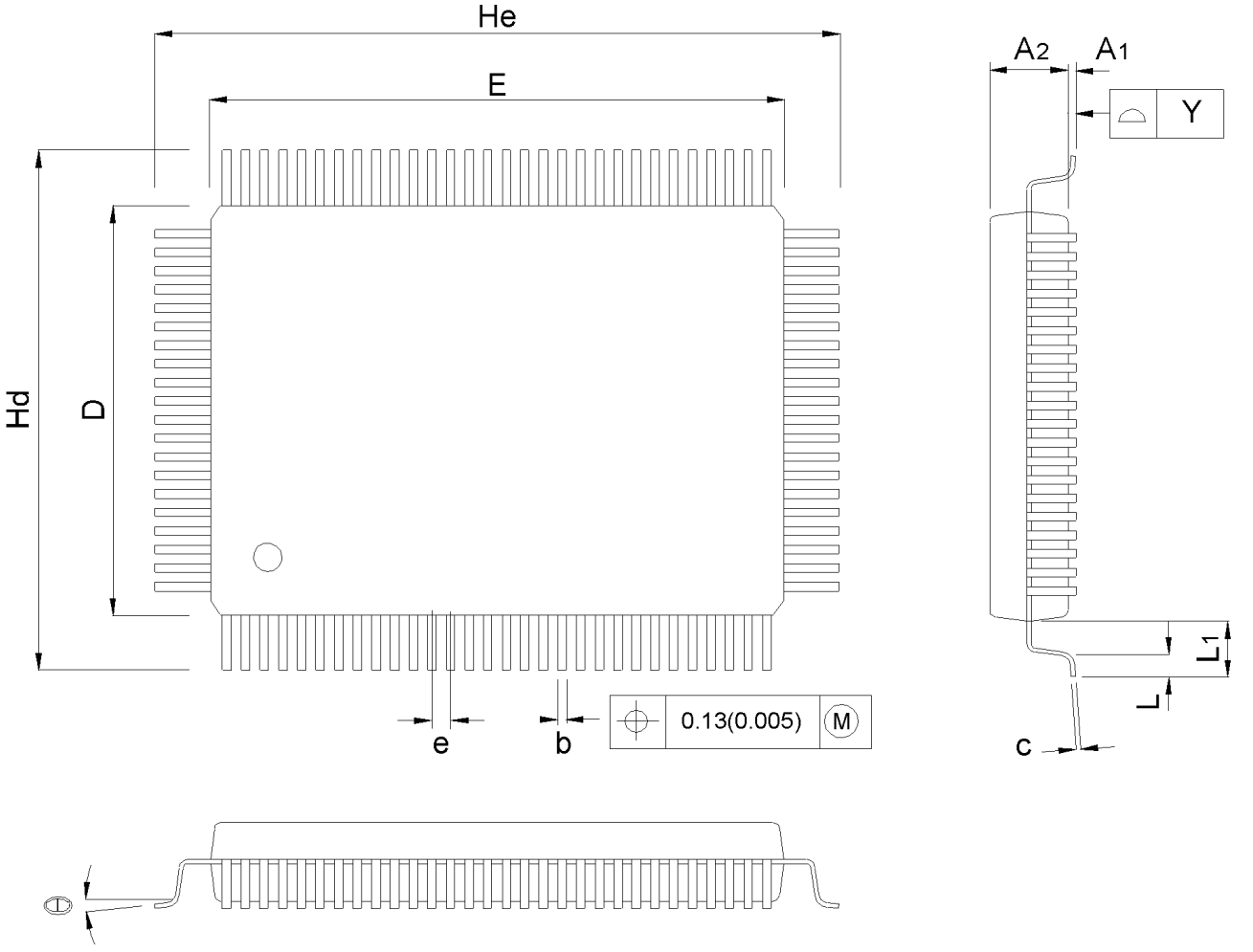
Figure 6-1 84-Pin PLCC (Plastic Leadless Chip Carrier)



LEAD COUNT	A	B	D	E	BOTTOM EJECT PIN LOCATION
20	353	390	10	310	ONLY ONE PIECE AT 3
28	453	490	10	410	ONLY ONE PIECE AT 3
44	653	690	10	610	1,3
52	753	790	10	710	1,2,3,4
68	953	990	8	910	1,2,3,4
84	1153	1190	8	1110	1,2,3,4

Dwg. No.:	ASPLCC-001		
Dwg. Rev.:	0	Unit:	MM

Figure 6-2 100-Pin QFP (Quad Flat Pack)



SYMBOL	MILLIMETER			INCH		
	MIN.	NOM.	MAX.	MIN.	NOM.	MAX.
A_1	0.25	0.35	0.45	0.010	0.014	0.018
A_2	2.57	2.72	2.87	0.101	0.107	0.113
b	0.20	0.30	0.40	0.008	0.012	0.016
c	0.10	0.15	0.20	0.004	0.006	0.008
D	13.90	14.00	14.10	0.547	0.551	0.555
E	19.90	20.00	20.10	0.783	0.787	0.791
e		0.65			0.026	
Hd	17.00	17.20	17.40	0.669	0.677	0.685
He	23.00	23.20	23.40	0.905	0.913	0.921
L	0.65	0.80	0.95	0.025	0.031	0.037
L_1		1.60			0.063	
Y			0.08			0.003
θ	0		7	0		7

Dwg. No.:	AS100PQFP-001
Dwg. Rev.: A0	Unit: MM / INCH