INTELLIGENT FOUR-PORT SERIAL I/O Model SI04-DMA

FOR IEEE-696/S-100 COMPUTER SYSTEMS

USER'S MANUAL

DUAL SYSTEMS CORPORATION

system reliability / system integrity

2530 San Pablo Avenue • Berkeley • CA 94702 • (415) 549-3854 • 172029 SPX

Table of contents

Section 1 General information

1.1 Introduction

The SIO4/DMA is an intelligent, high performance S100 four port serial input/output board. It conforms to the IEEE-696 specification current at the time of design but is also designed to work with older S100 processors. The board is intended to be used with asynchronous terminals or printers in a multi-user system.

The SIO4/DMA is the first S100 serial interface board designed specifically to reduce operating system overhead in multi-user systems. Most of the tasks normally performed by the operating system to handle serial input and output are greatly assisted by the SIO4/DMA.

All characters sent to the SIO4/DMA can be buffered in a 256 byte FIFO. This removes the requirement for fast response time from the operating system needed with conventional boards to avoid the possibility of lost characters.

On output the SIO4/DMA reads characters directly from the S100 machines memory by DMA and transfers them to the serial port. This can reduce the operating system overhead for output operations several hundred times. The board is capable of a full 24 bit address range and will increment over any address boundary.

Baud rate is independently selectable on each channel by program. Sixteen speeds are provided.

1.2 Features

The SIO4/DMA is a four port serial input/output board designed at Dual Systems for use with multi-user systems.

* 256 bytes of FIFO buffer for input characters.
* DMA transfers for output.
* 24 bits of address supported with no restrictions on boundaries.
* Can be connected to RS-232 terminals or modems.
* RS-232 drivers and receivers conform fully to specification.
* 8085A-2 local processor.
* 2716 program EPROM.
* 2K static RAM.
* Control-S/Control-Q protocol supported.
* Printers with 'Buffer full' signal lines supported.
* USARTs accessible directly for special functions.
* Limited synchronous mode supplied.
* Program selectable Baud rate.
* Occupies only 16 I/O addresses.
* Uses easily assembled cables.

1.3 Specifications

Bus compatibility: The SIO4/DMA is designed to be compatible with the IEEE-696 S100 specification as defined by January 1982, suitable for systems with up to 8MHz clocks. The SIO4/DMA is also intended to be compatible with many older S100 processors such as the North Star Horizon.

IEEE 696/S100 compliance: M24,VI,TM,F8,W1-16. Number of wait states depends on S100 clock rate and state of SIO4/DMA when accessed.

Address selection: I/O address selected by DIP switch. Eight bits of I/O address are decoded by the SIO4/DMA.

Operating power: The maximum power consumption of the SIO4/DMA is as follows:

+8 Volts - 1.3 Amps
+16 Volts - 0.1 Amps
-16 Volts - 0.1 Amps

RS-232 compliance: All RS-232 drivers and receivers conform to the RS-232 specification. The SIO4/DMA can simulate either a modem (DCE) or terminal (DTE) for connection to a terminal or modem respectively.

Mating Connectors: 3M 3339-0000 or T & B Ansley 609-2600M are suitable for connecting the SIO4/DMA to 26 conductor ribbon cable.

Baud rates: The following baud rates are supplied:

45.5	1200
50	1800
75	2000
110	2400
134.5	4800
150	9600
300	19200
600	38400

2.1 Unpacking the SIO4/DMA

The SIO4/DMA board is shipped in a carton containing the following:

1) Dual Systems SIO4/DMA board.

2) Ribbon cables and mounting hardware as ordered.

3) Additional set of headers for custom applications.

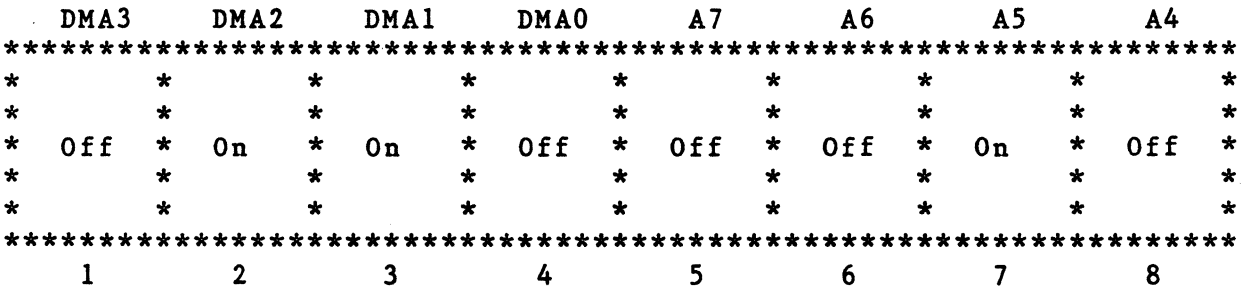
4) SIO4/DMA Manual.

The SIO4/DMA board comes configured for use as the first SIO4/DMA board in a Dual Systems multi-user UNIX system. It is configured for connection to four terminals. If this is not the configuration that you require, you should see the appropriate part of Section 3 or 4 for further information.

2.2 Switch settings for Dual Systems UNIX

The SIO4/DMA has one eight position DIP switch. Four positions of this switch are used to determine the address at which the SIO4/DMA resides. The remaining four positions are used to determine the SIO4/DMA's DMA priority. It is essential that each board that performs DMA operations in the S100 system, have a different priority.

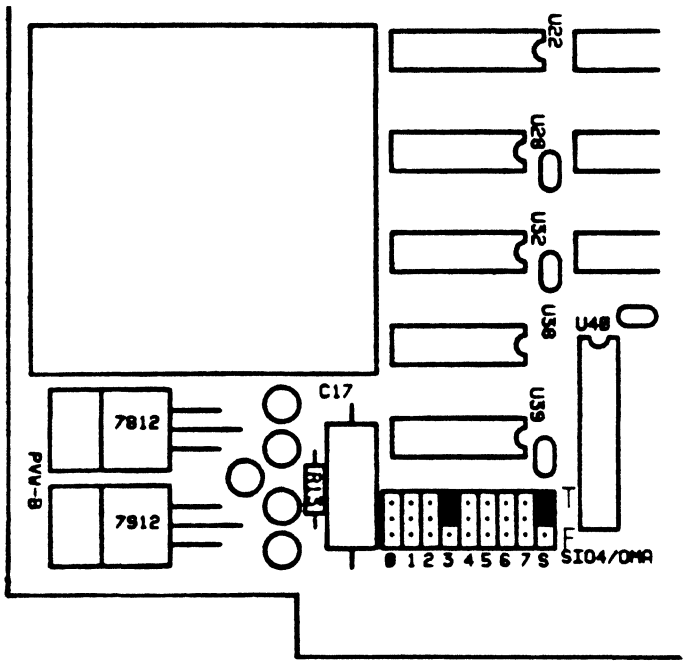
When the board is viewed from the front with the edge connector at the bottom, the switch will be found in the top center of the board. The following figure indentifies the setting of each switch for the first SIO4/DMA in a Dual Systems' UNIX system. This gives an I/O address of 20H and a DMA priority of 9.



Note that some DIP switches may be marked with 'Open' instead of 'Off'.

2.3 Interrupt jumper settings for Dual Systems UNIX.

The interrupt jumpers should be installed as indicated in the following diagram.



Section 3 Connecting serial devices to the SIO-4/DMA

The RS-232 standard describes a connection between a terminal or printer and a Modem. Consequently, a device such as the SIO-4/DMA has to be configured as a modem to communicate with a terminal and has to be configured as a terminal to communicate with a modem.

3.1 Connection to a terminal or printer.

The SIO-4/DMA is shipped configured for connection to most terminals or printers. If the original configuration has been changed and you wish to return the board to its original condition, perform the following:

Remove the configuration header, if present, from its socket on the line you wish to change.

Wire a new header with the following connections:

pin 1 to pin 16
pin 2 to pin 15
pin 3 to pin 14
pin 4 to pin 13
pin 5 to pin 12
pin 6 to pin 11
pin 7 to pin 10
pin 8 to pin 9

Insert it into the configuration header socket for the line you require.

3.2 Connection to a Modem

If the SIO-4/DMA is to communicate with a Modem or possibly another computer configured as a Modem, it will be necessary to perform the following:

Remove the configuration header, if present, from the socket on the line you wish to change.

Wire a new header with the following connections:

pin 1 to pin 2
pin 3 to pin 4
pin 5 to pin 6
pin 7 to pin 8
pin 9 to pin 10
pin 11 to pin 12
pin 13 to pin 14
pin 15 to pin 16

Insert it into the configuration header socket for the line you require.

3.3 "Three wire" connections.

In many cases, it is considered useful to use the minimum number of conductors to connect two RS-232 devices together. The minimum possible number of wires in this case is three. However the SIO-4/DMA does use some of the control lines specified in RS-232 and it is necessary to bias these to a high state for the SIO-4/DMA to send any characters to the terminal. Consequently the wiring of a three conductor cable for connecting the SIO-4/DMA to a terminal should be as follows:

Pin 2 to pin 2 of distant connector (TxD)
Pin 3 to pin 3 of distant connector (RxD)
Pin 7 to pin 7 of distant connector (Gnd)

Pin 4 to pin 6 to pin 20 of local connector
Pin 4 to pin 6 to pin 20 of distant connector
(RTS, DSR and DTR)

3.4 Buffer full signals

An increasing number of printers have the ability to accept data into a buffer at a faster rate than they can print it. When the buffer is nearly full, the printer must indicate this to the sending device so that it can stop sending characters to the printer. There are currently a number of ways for the printer to do this.

1) When the printer buffer is nearly full it sends a Control-S back to the SIO-4/DMA. When it is ready for more information, it sends a Control-Q to the SIO-4/DMA. This method is supported by the SIO-4/DMA and is selected by software. Control-S and Control-Q can also be sent from the printer's keyboard, if it has one, to permit an operator to stop and start output.

2) A block of characters of an agreed size, depending on the variety of printer used, is sent to the printer followed by an ETX character. When the printer is ready for more information, it sends an ACK character. This method can be supported by software in the S100 computer and requires no special action from the SIO-4/DMA.

3) When the printer buffer is nearly full, a control line from the printer changes state. Although there is no formal standardization of this method, the most common variant appears to be to have pin 20 go low to indicate inability to accept characters. This is supported by the SIO-4/DMA and can, by changing the configuration header, be wired to different pins. There is no provision for changing the polarity of this signal.

3.5 Synchronous operation of the SIO-4/DMA

Synchronous operation is used primarily for communication with devices which are incapable of asynchronous operation such as certain high speed modems.

Synchronous operation requires that a clock signal be provided with a data signal to indicate when the data should be sampled. The SIO-4/DMA transmitter generates a clock along with transmitted data, which is used by the receiver to examine the incoming data. When used in this way, the transmitting device determines the Baud rate of transmission.

There is a related communications mode known as iso-synchronous mode. In this mode data is transmitted with a clock for reference as in the synchronous case, but the data itself is framed by start and stop bits like asynchronous data. This mode can be very valuble for communicating between computers at high speed as pauses during transmission do not require any special action or protocol.

The SIO-4/DMA can receive and transmit synchronous information and requires no hardware changes to do so. The USART does have to be set up in synchronous mode by the S100 machine however. Details of how to do this can be found in the 2661 data sheet.

The USARTs used in the SIO-4/DMA are capable of operating synchronously or iso-synchronously at speeds of up to 800k Baud. However, the SIO-4/DMA's processor may not be capable of receiving data at this speed, particularly if all four channels are active at the same time.

3.6 RS-232 signals supported by the SIO-4/DMA

Pin	Signal name	RS-232 name
2	Transmitted Data	BA
3	Received Data	BB
4	Request to Send	CA
5	Clear to Send	CB
6	Data Set Ready	CC
7	Signal Ground	AB
8	Received Line Signal Detector *	CF
17	Receiver Signal Element Timing	DD
20	Data Terminal Ready	CD
24	Transmit Signal Element Timing.	DA

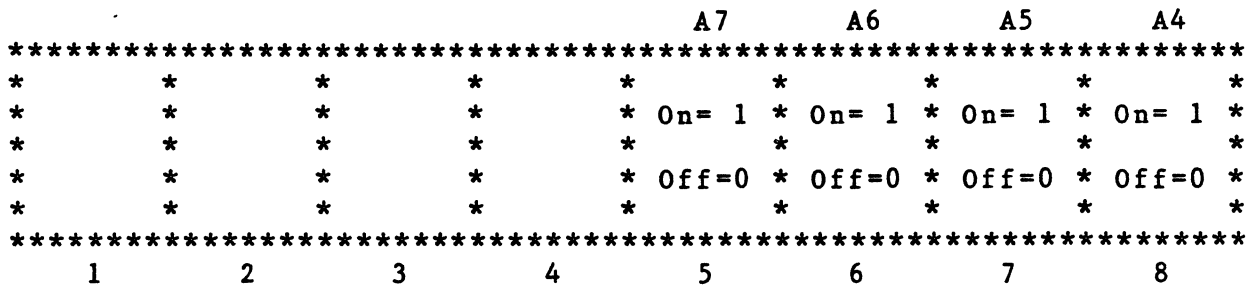
* Pin 8, the Received Line Signal Detector, is permanently asserted high by a driver meeting RS-232 specifications.

Section 4 SIO4/DMA configuration for the S100 bus

The SIO4/DMA has one eight position DIP switch. Four positions of this switch are used to determine the address at which the SIO4/DMA resides. The remaining four positions are used to determine the SIO4/DMA's DMA priority. This switch will be found in the centre of the board opposite the S100 edge connector. Interrupt selection is provided by a jumper area at the regulator end of the S100 edge connector.

4.1 I/O address selection

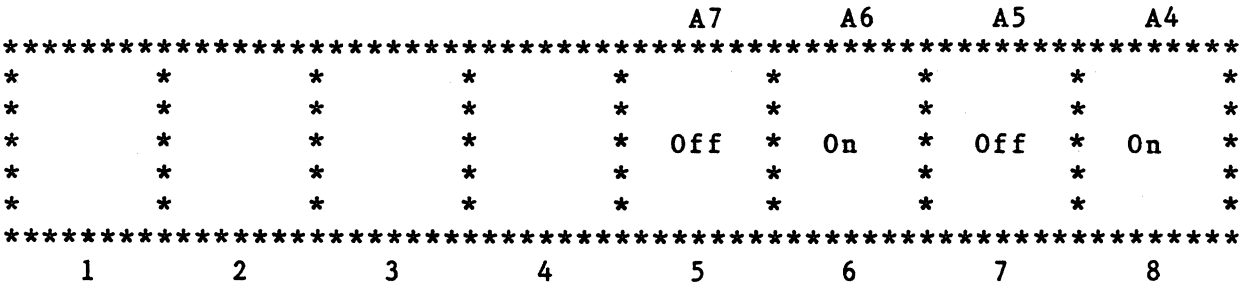
The SIO4/DMA occupies 16 consecutive I/O addresses starting at X0H and running to XFH, where 'X' represents a hexadecimal digit set by switches five through eight on the board. To find the switch setting that you require, consult the following table:



Starting Address	Switch setting			
000H	Off	Off	Off	Off
010H	Off	Off	Off	On
020H	Off	Off	On	Off
030H	Off	Off	On	On
040H	Off	On	Off	Off
050H	Off	On	Off	On
060H	Off	On	On	Off
070H	Off	On	On	On
080H	On	Off	Off	Off
090H	On	Off	Off	On
0A0H	On	Off	On	Off
0B0H	On	Off	On	On
0C0H	On	On	Off	Off
0D0H	On	On	Off	On
0E0H	On	On	On	Off
0F0H	On	On	On	On

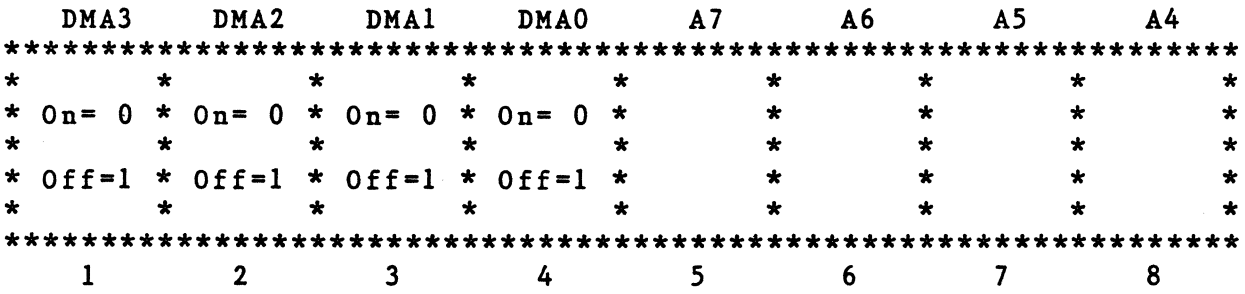
Some DIP switches may be marked with 'Open' instead of 'Off'.

For example for an SIO4/DMA residing at 50H would have its I/O switches set in the following way.



4.2 DMA priority selection

Sixteen DMA priorities are available on the IEEE S100 bus. The priority of the SIO4/DMA is set by the four left hand switches. It is essential that each board that performs DMA operations in the S100 system, have a different priority. As the SIO4/DMA performs DMA fairly infrequently compared to a disk for example, it can be given almost any priority without adversely affecting either system or SIO4/DMA performance.



Switch settings				DMA Priority
On	On	On	On	0
On	On	On	Off	1
On	On	Off	On	2
On	On	Off	Off	3
On	Off	On	On	4
On	Off	On	Off	5
On	Off	Off	On	6
On	Off	Off	Off	7
Off	On	On	On	8
Off	On	On	Off	9
Off	On	Off	On	10
Off	On	Off	Off	11
Off	Off	On	On	12
Off	Off	On	Off	13
Off	Off	Off	On	14
Off	Off	Off	Off	15

Some DIP switches may be marked with 'Open' instead of 'Off'.

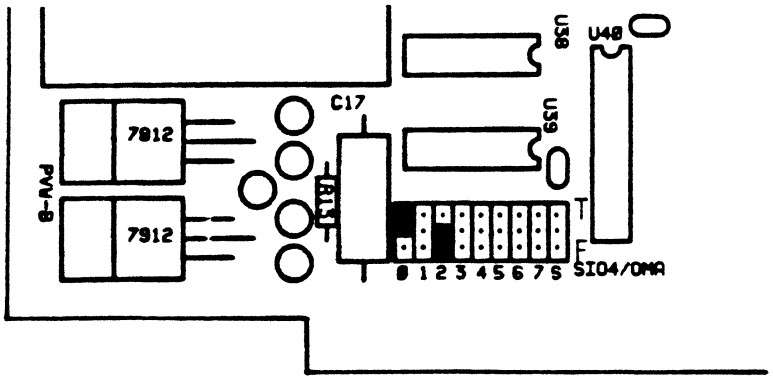
For example for an SIO4/DMA residing at 20H and having DMA priority 9 would have its switches set in the following way.

DMA3	DMA2	DMA1	DMA0	A7	A6	A5	A4
Off	On	On	Off	Off	Off	On	Off
1	2	3	4	5	6	7	8

4.3 Setting interrupt levels.

The SIO4/DMA is capable of causing an interrupt, if required, in two circumstances. If a character is ready in the SIO4/DMA's FIFO and the appropriate interrupt enable is set, then an interrupt can be caused. If a DMA transfer has completed and has the appropriate interrupt enable bits set, this can also cause an interrupt. The interrupt can be caused on any of the S100 interrupts VIO through VI7.

The interrupt required must be selected on the interrupt jumper area of the SIO4/DMA. This jumper area is near the left end of the S100 edge connector. The centre row of pins are connected to the VI interrupt lines of the S100 bus as indicated by the numbers below the pins. The top row of pins are connected to the DMA done interrupt source. The bottom row of pins are connected to the FIFO ready interrupt source. To connect the source of interrupt to the 'VI' line required, install a jumper between the 'VI' pin desired, and the pin above or below it, depending on the source required. For example to connect DMA done interrupts to VIO and character ready interrupts to VI2, the jumpers would be installed as shown in the diagram at the bottom of the page. If both sources of interrupt are to generate the same interrupt, insert one jumper as usual and insert the other jumper between the top and centre rows at the position marked 'S'.



4.4 Operating with non-IEEE S100 processors.

Many early S100 processors do not disable pSTVAL as required by the IEEE-696 S100 specification. In this case, the trace under the STVL jumper should be cut to avoid bus contention. Operation with non-IEEE machines is also dependent on memory, in particular, some older dynamic memories may expect different timing relationships between signals. No provision has been made to alter the timing provided. Static memories are less likely to cause difficulties.

Section 5 Programmer's guide to the SIO4/DMA

5.1 Programmer's overview

The SIO4/DMA has an on board processor to supervise the various data transfers required. To permit communication between the S100 processor and the SIO4/DMA's processor, certain devices and areas of memory attached to the SIO4/DMA's processor, can be accessed by either the SIO4/DMA or the S100 machine. With the exception of S100 interrupts, all communication between the SIO4/DMA and the S100 machine takes place through this shared area. It should be noted that it is not possible for both the SIO4/DMA and the S100 machine to access any of these shared areas at the same time. The S100 processor always waits for the SIO4/DMA's processor to finish.

To co-ordinate DMA output, the SIO4/DMA has four sets of eight registers, one for each channel. These are used for a 24 bit address, a 16 bit byte count and two control registers.

During an output operation the DMA addresses are incremented and the byte count is decremented. On completion, the byte count is zero and the address is that of the last byte read. An initial byte count of zero will transfer zero characters and will complete in the usual manner as soon as the USART is ready to transmit another character. The byte count and address registers can be read while output is in progress but the results are unlikely to be useful.

All characters received can be loaded into a FIFO along with the line number and some status information. The FIFO has a maximum capacity of 256 characters shared between the four lines.

5.2 Address space usage.

Address offset	R/W functions on selected line
00	DMA LS address
01	DMA IS address
02	DMA MS address
03	Receiver control
04	Byte count LS
05	Byte count MS
06	Output start/stop enable
07	DMA control and status
08	USART data (See 2661 data sheet)
09	USART status (See 2661 data sheet)
10	USART mode (See 2661 data sheet)
11	USART command (See 2661 data sheet)

The remaining functions are not dependent on selected line number

	Read function	Write function
12	FIFO status	Diagnostic use only
13	FIFO data	Diagnostic use only
14	Global DMA status	Global interrupt masks
15	Diagnostic use only	Select line number

Addresses 00-02 - DMA addresses R/W

These three locations must be loaded with the starting address from which DMA transfers will occur when output is started. After each character is output, the 24 bit address contained in these three locations is incremented by one.

Address 03 - Receiver control R/W

The receiver control byte is used to turn the receiver on a particular line on or off. On reset the receiver is disabled. Note that no data is read from the USART by the SIO4/DMA when the receiver is disabled by this means, so that data can be read directly from the USART by the S100 processor if required.

The format is as follows:

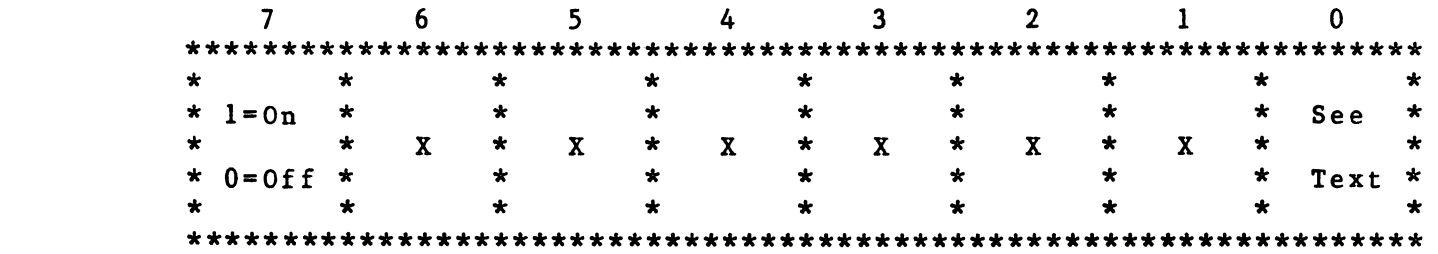
7	6	5	4	3	2	1	0
X	X	X	X	X	X	1=On 0=Off	X

This byte is never changed by the SIO4/DMA. Bits marked X may be used to store information of value to the host machine if so desired. The initial value of this byte is zero.

Address 06 - Output start/stop

The output start/stop feature permits the output to a terminal to be stopped by receiving a Ctrl-S and started by a Ctrl-Q. These characters will be passed into the FIFO whether the feature is enabled or not. On power up this feature is disabled. The feature should only be turned off when the receiver is disabled. It may be enabled at any time. The byte may be read at any time.

The format is as follows:

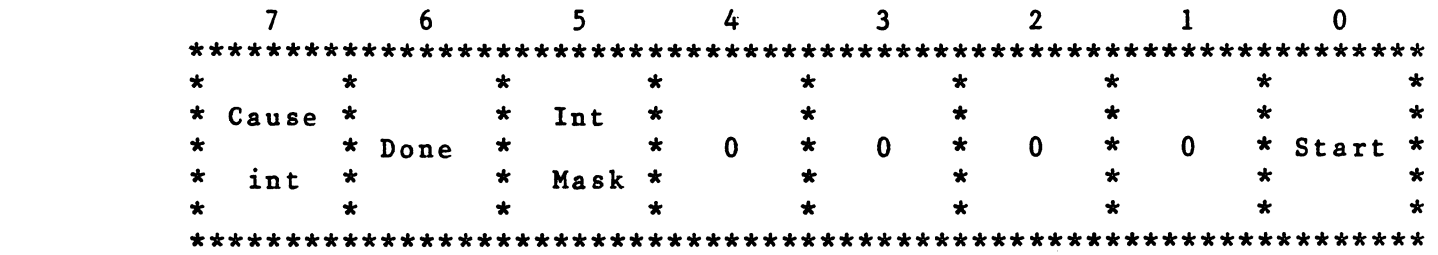


On read, a one in bit zero indicates that output has been halted. A zero indicates that output can proceed. The data in bits one to six may be changed by the SIO4/DMA during output and should not be used. If the feature is disabled, bit zero can be set to stop output and cleared to continue output. This would be useful if characters other than Control-S and Control-Q were required to stop and start output. On reset the byte is set to zero.

Address 07 - DMA control

The DMA control byte permits starting of DMA transfers, setting and clearing of interrupt masks and setting and clearing of interrupts. This byte should normally only be changed when output is not in progress. This is indicated by the 'Done' bit being high.

The format is as follows:



Bit 0, the "start" bit is set to a one to start a DMA transfer after the DMA address, byte count and output control registers have been set. On completion of the transfer, this bit will be cleared.

Bit 5, the interrupt enable bit. If this bit is set to a one and the global DMA interrupt is enabled, an interrupt will occur when the DMA transfer has been completed. Note that clearing this bit will not clear an interrupt. This must be done by clearing bit 7, the "Cause interrupt" bit.

Bit 6, the done bit, indicates that the DMA transfer has been completed independently of the state of any interrupt enable bits.

Bit 7, the cause interrupt bit. If the global DMA interrupt enable is set, then setting this bit will cause an interrupt. When set by the SIO4/DMA, this bit is the AND of bits 5 and 6 at the time of setting. The interrupt service routine must clear this bit to clear the interrupt.

At power up, the byte is set to 5EH. After an output operation on the relevant channel, bits one to four will always be set to zero by the SIO4/DMA.

Address 8-11 - USART data and control

These addresses allow direct access to the selected USART as described in the USART data sheet. The most common use for this is for changing baud rate, number of stop bits and so on.

Address 12 - FIFO status

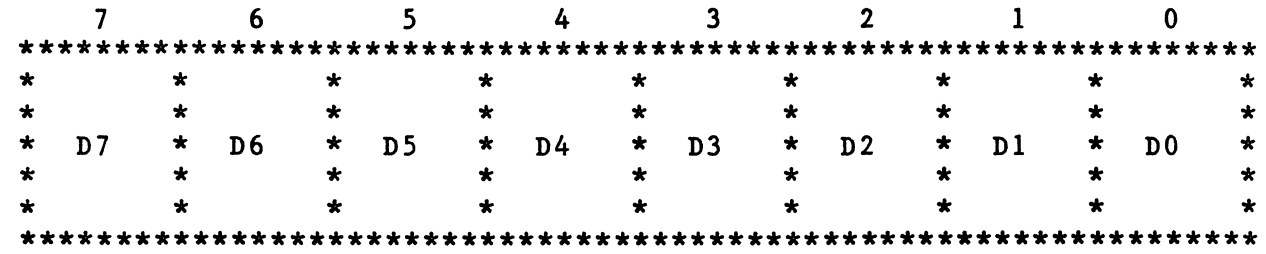
7	6	5	4	3	2	1	0
DMA int	FIFO byte ready	RFU	RFU	LN1	LN0	FE	OVR

Bit six indicates the presence of data in the FIFO. Bits three and two give the number of the line from which the status and data came. Bit one set indicates a framing error. This can be caused by the use of an incorrect baud rate or by a terminal user sending "Break". Bit zero is the overrun bit. If set it indicates that a character was lost because it was not read from the USART in time. This will normally only occur when the FIFO has been filled and no space is left. The status may be read as many times as desired. Reading the FIFO data will produce the next entry in the FIFO, if any. Bit seven is covered in the section on address 14. This byte should not be written to except by diagnostic routines. "RFU" means reserved for future use, on read the value of these bits cannot be depended on.

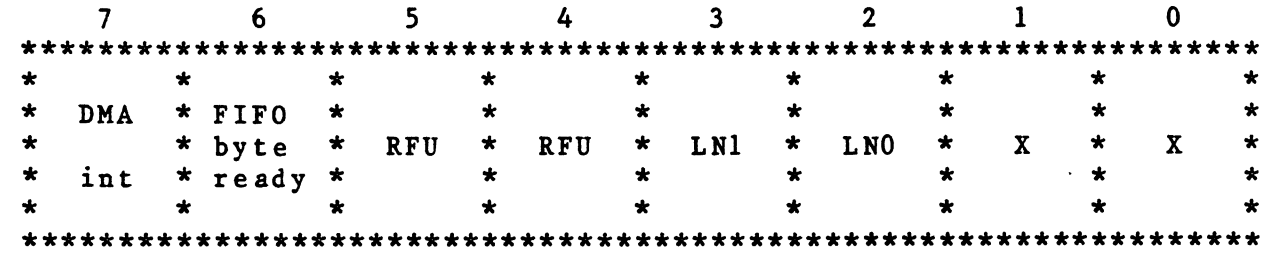
Address 13 - FIFO Data

This byte contains the data read from the USART. Reading this byte will clear the FIFO interrupt if set. It also causes the next entry in the FIFO to be displayed in any subsequent status or data reads. Because of this, it is desirable that the FIFO data be read as soon as possible in any interrupt service routine to allow the maximum time for the next entry to be produced. This byte should not be written to except by diagnostic routines.

The format is as follows:



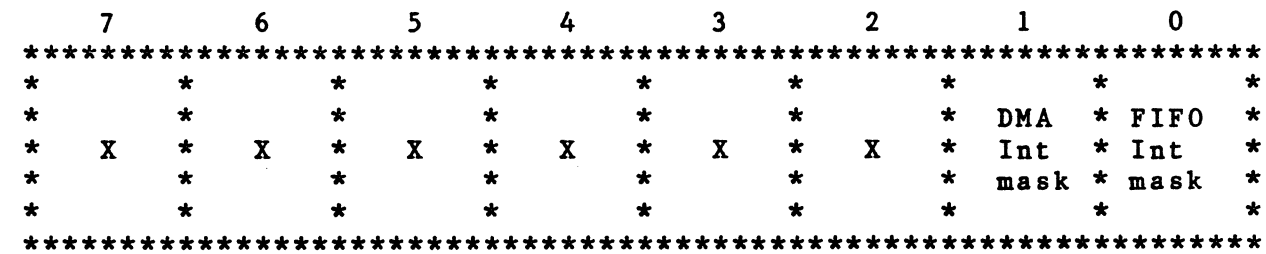
Address 14 - Read DMA interrupt status
==========



Bit seven indicates the completion of a DMA operation in which the interrupt mask bit was set. If the global DMA interrupt enable bit is set, then this will indicate the presence of an interrupt. Bit six is described in the section on address 12. Bits two and three indicate the highest line number that has completed its DMA operation and had its interrupt mask bit set at the time. If this has occurred on more than one channel, then the highest numbered channel that has completed will be given. When the 'Cause interrupt' bit is cleared on that channel, the next highest interrupting channel number will be given on the next status read. The interrupt will remain until all interrupting channels have been correctly serviced.

Address 14 - Write global interrupt masks
==========

The format is as follows:



Bit 1 is the global DMA interrupt mask bit. If this bit is set then any DMA operation which completed and had bit seven set at the time of completion, will cause an interrupt. The channel causing the interrupt can be determined from the DMA status byte, address 13.

Bit zero is the FIFO interrupt mask bit. When any data becomes available in the FIFO, an interrupt will be caused if this bit is set.

Address 15 - Write - Select line number.
==========

The format is as follows:

7	6	5	4	3	2	1	0
X	X	X	X	RFU	RFU	LN1	LN0

Bits zero and one are the selected line number. Addresses zero to eleven all perform operations on the selected line. Writing to this location permits the selected line to be changed. For example, to set up USART number two, it would first be neccessary to output 02H to this address, and then issue commands to the USART.

"RFU" means reserved for future use, these bits should be set to zero on writes to maintain compatibility with possible future products.

The on-board processor is reset by RESET, SLVCLR or POC. The S100 DMA circuitry is reset by POC.

Section 6 Programming examples for the SIO4/DMA

6.1 Changing Baud rate.

```
0000                ; Sample program to illustrate changing Baud rate.
0000                ;
0010                XSIO    EQU     10H             ; Base address of SIO-4/DMA
0000                ;
0018                EPCI    EQU     XSIO+8          ; Address of USART
001A                EPMO    EQU     EPCI+2          ; USART mode address
001B                EPCM    EQU     EPCI+3          ; USART command address
0000                ;
001F                LNSEL   EQU     XSIO+15         ; Address for line selection.
0000                ;
0000                ;
0000                        ORG     100H
0100                ;
0100                ; Assume that the SIO-4/DMA is already set up
0100                ; for desired output characteristics, such
0100                ; as parity, number of stop bits etc.
0100                ;
0100                ; Set Baud rate of line 2 to 4800 Baud
0100                ;
0100  3E02                  MVI     A,2             ; Line two
0102  060C                  MVI     B,0CH           ; Baud rate of 4800 Baud
0104  CD0801                CALL    BAUD
0107  C9                    RET     ;               ; Return to operating system.
0108                ;
0108                ; Set Baud rate routine.
0108                ; Line number supplied in A
0108                ; Baud rate supplied in B
0108                ;
0108  D31F          BAUD    OUT     LNSEL           ; Set line number
010A  DB1B                  IN      EPCM            ; Ensure Mode register one
010C  DB1A                  IN      EPMO            ; is next and read it.
010E  3E0F                  MVI     A,0FH
0110  A0                    ANA     B               ; Get Baud rate only.
0111  47                    MOV     B,A             ; Save Baud rate.
0112  DB1A                  IN      EPMO            ; Read mode register 2
0114  E6F0                  ANI     0F0H            ; Remove old Baud rate.
0116  B0                    ORA     B               ; OR in new Baud rate
0117  47                    MOV     B,A             ; Save in B
0118  DB1A                  IN      EPMO            ; Read mode register 1 again
011A  78                    MOV     A,B             ; Get new data for mode
011B  D31A                  OUT     EPMO            ; register 2 and write to it.
011D  C9                    RET
011E                ;
011E                ; The 2661 has two mode registers at the 'Mode register'
011E                ; address.  These are accessed alternately.  Performing
011E                ; a read from the command register ensures that the next
011E                ; mode read or write affects mode register 1.  The next
011E                ; read or write after that affects mode register 2.  As
011E                ; the Baud rate is set by writing to register 2, a read
011E                ; from register 1 has to be performed before the Baud
```

```
011E                ; rate can be set.  Mode register 2 contains information
011E                ; in addition to the Baud rate.  The program example above
011E                ; reads this and replaces it along with the new Baud rate.
011E                ;
011E                ; For full explanation see 2661 data sheet
011E                ;
011E                      END
SYMBOL TABLE

BAUD  0108 01  EPCI  0018 00  EPCM  001B 00  EPMO  001A 00  LNSEL 001F 00  XSIO  0010 00
```

6.2 Outputting a block of data.

```
0000                ; Sample program to illustrate SIO-4/DMA output
0000                ;
0010                XSIO    EQU     10H             ; Base address of SIO-4/DMA
0010                DADR    EQU     XSIO+0          ; Address for DMA address
0014                DBYC    EQU     XSIO+4          ; Address for DMA byte count.
0017                DCTL    EQU     XSIO+7          ; Address for DMA control.
001F                LNSEL   EQU     XSIO+15         ; Address for line selection.
0000                ;
0001                STRT    EQU     1               ; Bit to start transfer using DTCL
0040                DONE    EQU     40H             ; Bit to indicate completion in DCTL
0000                ;
000A                LF      EQU     10              ; Line feed
000D                CR      EQU     13              ; Carriage return
0000                ;
0000                        ORG     100H
0100                ;
0100                ; Output sample message using OUTP routine.
0100                ;
0100  212D01                LXI     H,DTADR         ; Get pointer to 24 bit address.
0103  112C00                LXI     D,MEND-MESS     ; Get length in DE.
0106  CD1001                CALL    OUTP
0109                ;
0109  110000                LXI     D,0
010C  CD1001                CALL    OUTP            ; Ensure output complete before
010F  C9                    RET     ;               ; returning to operating system.
0110                ;
0110                ; Assume that the board is already set up
0110                ; for desired output characteristics, such
0110                ; as Baud rate, number of stop bits etc.
0110                ;
0110                ; Output routine.  Requires
0110                ; pointer to address in HL and
0110                ; byte count in DE
0110                ;
0110  DB17          OUTP    IN      DCTL            ; Read DMA control to see
0112  E640                  ANI     DONE            ; if previous output done.
0114  C21001                JNZ     OUTP            ; Wait till done.
0117  7E                    MOV     A,M             ; Get LS 8 bits
0118  D310                  OUT     DADR            ; Put in SIO-4/DMA
011A  23                    INX     H               ; Point to next 8 bits.
011B  7E                    MOV     A,M             ; Get next 8 bits
011C  D311                  OUT     DADR+1          ; Put in SIO-4/DMA.
011E  23                    INX     H
011F  7E                    MOV     A,M             ; Get most significant
0120  D312                  OUT     DADR+2          ; 8 bits to SIO-4/DMA.
0122                ;
0122  7B                    MOV     A,E             ; Get LS 8 bits of count
0123  D314                  OUT     DBYC            ; Put in SIO-4/DMA
0125  7A                    MOV     A,D             ; Get MS 8 bits of count
0126  D315                  OUT     DBYC+1          ; Put in SIO-4/DMA.
0128                ;
0128  3E01                  MVI     A,STRT          ; Get value to start DMA
012A  D317                  OUT     DCTL            ; Start DMA transfer.
```

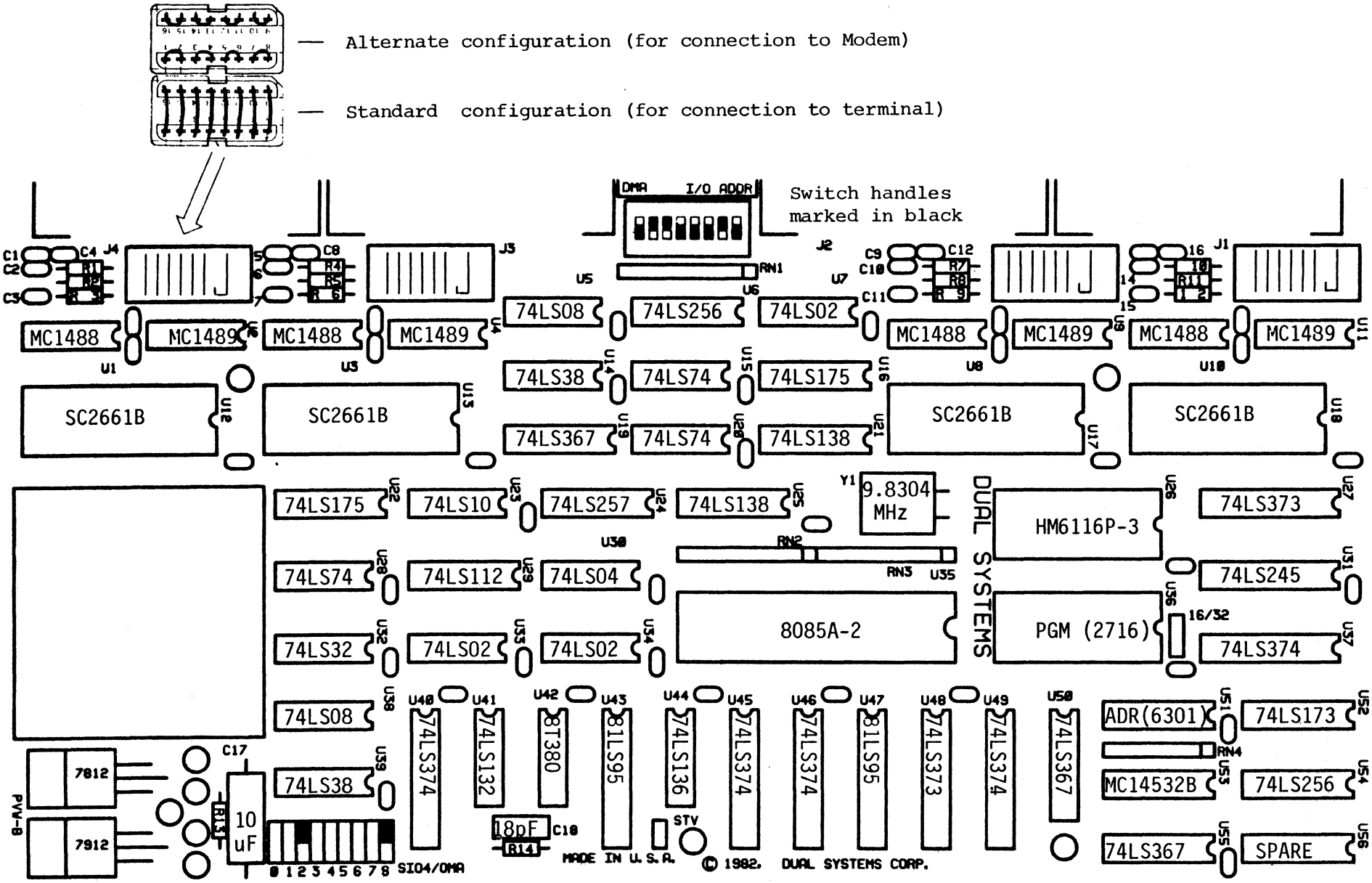
```
012C                ;
012C  C9                    RET
012D                ;
012D                ;
012D  3001          DTADR   DW      MESS
012F  00                    DB      0               ; Three bytes of data address.
0130                ;
0130  48656C6C      MESS    DB      'Hello and good afternoon '
      6F20616E
      6420676F
      6F642061
      66746572
      6E6F6F6E
      20
0149  66726F6D              DB      'from Dual Systems'
      20447561
      6C205379
      7374656D
      73
015A  0D0A                  DB      CR,LF
015C                MEND    EQU     $
015C                ;
015C                        END
SYMBOL TABLE

CR     000D 00  DADR   0010 00  DBYC   0014 00  DCTL   0017 00  DONE   0040 00  DTADR  012D 01
LF     000A 00  LNSEL  001F 00  MEND   015C 01  MESS   0130 01  OUTP   0110 01  STRT   0001 00
XSIO   0010 00
```

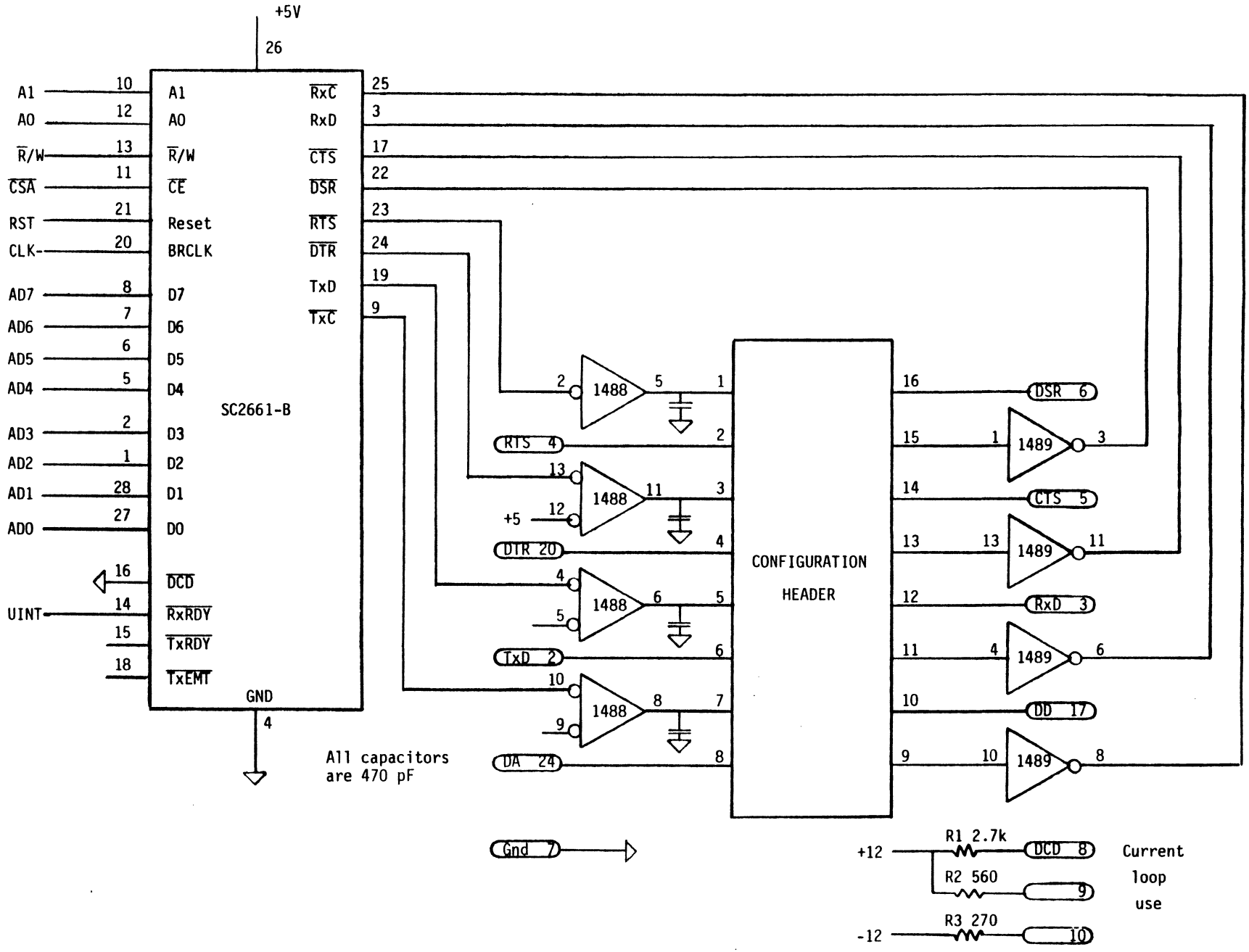
6.3 Inputting characters.

```
0000                ; Sample program to illustrate SIO-4/DMA input
0000                ;
0010                XSIO    EQU     10H             ; Base address of SIO-4/DMA
0010                DADR    EQU     XSIO+0          ; Address for DMA address
0014                DBYC    EQU     XSIO+4          ; Address for DMA byte count.
0017                DCTL    EQU     XSIO+7          ; Address for DMA control.
001C                IPST    EQU     XSIO+12         ; Address for FIFO status
001D                IPDT    EQU     XSIO+13         ; Address for FIFO data
001F                LNSEL   EQU     XSIO+15         ; Address for line selection.
0000                ;
0040                CHRDY   EQU     40H             ; Bit to indicate character ready.
0000                ;
000A                LF      EQU     10              ; Line feed
000D                CR      EQU     13              ; Carriage return
0000                ;
0000                        ORG     100H
0100                ;
0100                ; Assume that the board is already set up
0100                ; for desired input characteristics, such
0100                ; as Baud rate, number of stop bits etc.
0100                ;
0100                ;
0100                ; Sample routine to get a character in A with
0100                ; line number in B.
0100                ;
0100  E3            INPW    XTHL    ;               ; Waste some time
0101  E3                    XTHL
0102  DB1C          INPT    IN      IPST            ; Check if character ready
0104  47                    MOV     B,A             ; Save status
0105  E640                  ANI     CHRDY
0107  CA0001                JZ      INPW            ; Loop till ready
010A  3E0C                  MVI     A,0CH           ; Line number in bits 2 and 3
010C  A0                    ANA     B               ; Remove everything else
010D  47                    MOV     B,A             ; Line number now in B
010E  DB1D                  IN      IPDT            ; Read data from FIFO
0110  C9                    RET     ;               ; Return to operating system.
0111                ;
0111                        END
SYMBOL TABLE

CHRDY 0040 00  CR    000D 00  DADR  0010 00  DBYC  0014 00  DCTL  0017 00  INPT  0102 01
INPW  0100 01  IPDT  001D 00  IPST  001C 00  LF    000A 00  LNSEL 001F 00  XSIO  0010 00
```


— Alternate configuration (for connection to Modem)
— Standard configuration (for connection to terminal)
Switch handles marked in black
DMA
I/O ADDR
MC1488
MC1489
SC2661B
74LS08
74LS256
74LS02
74LS38
74LS74
74LS175
74LS367
74LS138
74LS10
74LS257
74LS112
74LS04
74LS32
74LS373
74LS245
74LS374
8085A-2
PGM (2716)
HM6116P-3
9.8304 MHz
DUAL SYSTEMS
7812
7912
10 uF
18pF
8T380
81LS95
74LS132
74LS136
ADR(6301)
MC14532B
SPARE
SIO4/DMA
MADE IN U.S.A.
© 1982, DUAL SYSTEMS CORP.

60027


+5V
26
SC2661-B
A1
A0
R/W
CSA
RST
CLK
AD7
AD6
AD5
AD4
AD3
AD2
AD1
AD0
D7
D6
D5
D4
D3
D2
D1
D0
Reset
BRCLK
DCD
RxRDY
TxRDY
TxEMT
UINT
GND
RxC
RxD
CTS
DSR
RTS
DTR
TxD
TxC
1488
1489
CONFIGURATION
HEADER
DSR 6
CTS 5
RxD 3
DD 17
DA 24
DTR 20
RTS 4
Gnd 7
All capacitors
are 470 pF
R1 2.7k
R2 560
R3 270
+12
-12
DCD 8
Current
loop
use

60028

7.3 USART data sheet

The USART data sheet describes all possible USART connections. Not all of these connections are supported by the SIO4/DMA. In particular, the TxC/XSYNC pin can only be used as an output and the RxC/BKDET pin can only be used as an input. This means that external character synchronization cannot be performed and that the break detect line cannot be examined directly. Both of these functions, however, can be performed under program control.

The manufacturers of the 2661-B USART choose to call it an EPCI. This is an acronym for Enhanced Programmable Communications Interface.

MOTOROLA
SEMICONDUCTORS
3501 ED BLUESTEIN BLVD., AUSTIN, TEXAS 78721

MC2661A/MC68661A
(Baud Rate Set A)
MC2661B/MC68661B
(Baud Rate Set B)
MC2661C/MC68661C
(Baud Rate Set C)

Advance Information

MOS
(N-CHANNEL, SILICON-GATE)

ENHANCED PROGRAMMABLE COMMUNICATIONS INTERFACE (EPCI)

ENHANCED PROGRAMMABLE COMMUNICATIONS INTERFACE (EPCI)

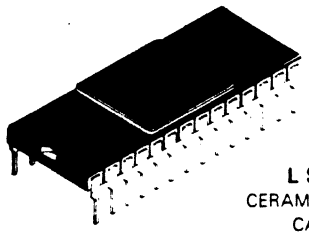
The MC2661/MC68661, Enhanced Programmable Communications Interface (EPCI), is a universal synchronous/asychronous data communications controller chip that is an enhanced version of the Signetics 2651. The EPCI directly interfaces to most 8-bit MPUs and easily to the MC68000 MPU and other 16-bit MPUs. It may be used in either a polled or interrupt driven system. Programmed instructions can be accepted from the host MPU while supporting many synchronous or asynchronous serial-data communication protocols in a full or half-duplex mode. Special support for BISYNC is provided.

The EPCI converts parallel data characters, accepted from the microprocessor data bus, into transmit-serial data. Simultaneously, the EPCI can convert receive-serial data to parallel data characters for input to the microprocessor.

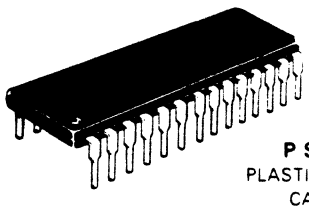
A baud rate generator in the EPCI can be programmed to either accept an external clock, or to generate internal transmit or receive clocks. Sixteen different baud rates can be selected under program control when operating in the internal clock mode. Each version of the EPCI (A, B, C) has a different set of baud rates.

FEATURES

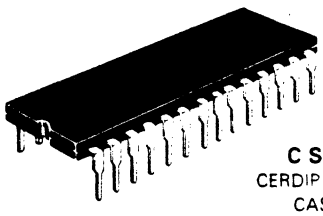
- **Synchronous Operation**
  - Single or Double SYN Operation
  - Internal or External Character Synchronization
  - Transparent or Non-transparent Mode
  - Transparent Mode DLE Stuffing (Tx) and Detection (Rx)
  - Automatic SYN or DLE-SYN Insertion
  - SYN, DLE, and DLE-SYN Stripping
  - Baud Rate: dc to 1M bps (1X Clock)
- **Asynchronous Operation**
  - 1, 1½, or 2 Stop Bits Transmitted
  - Parity, Overrun, and Framing Error Detection
  - Line Break Detection and Generation
  - False Start Bit Detection
  - Automatic Serial Echo Mode (Echoplex)
  - Baud Rate: dc 1M bps (1X Clock)
    dc to 62.5k bps (16X Clock)
    dc to 15.625k bps (64X Clock)
- **Common Features**
  - Internal or External Baud Rate Clock; No System Clock Required
  - 3 Baud Rate Sets (A, B, C); 16 Internal Rates for Each Set
  - 5- to 8-Bit Characters plus parity; Odd, Even, or No Parity
  - Double Buffered Transmitter and Receiver
  - Dynamic Character Length Switching
  - Full- or Half-Duplex Operation
  - Local or Remote Maintenance Loop-Back Mode
  - TTL-Compatible Inputs and Outputs
  - RxC and TxC Pins and Short Circuit Protected
  - 3 Open-Drain MOS Outputs can be Wire ORed
  - Single 5 V Power Supply
- **Applications**
  - Intelligent Terminals
  - Network Processors
  - Front End Processors
  - Remote Data Concentrators
  - Computer-to-Computer Links
  - Serial Peripherals
  - BISYNC Adaptors



L SUFFIX
CERAMIC PACKAGE
CASE 719

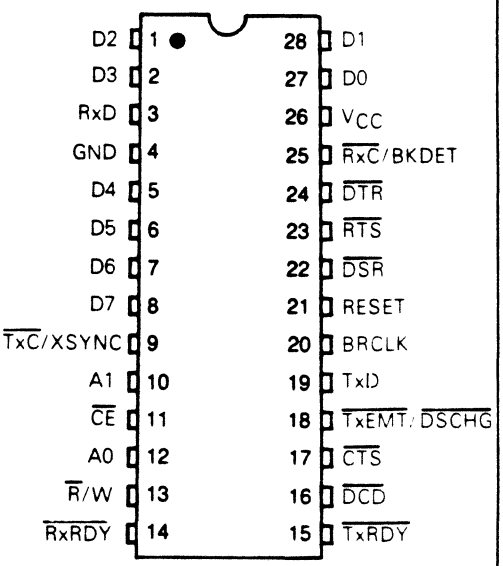


P SUFFIX
PLASTIC PACKAGE
CASE 710



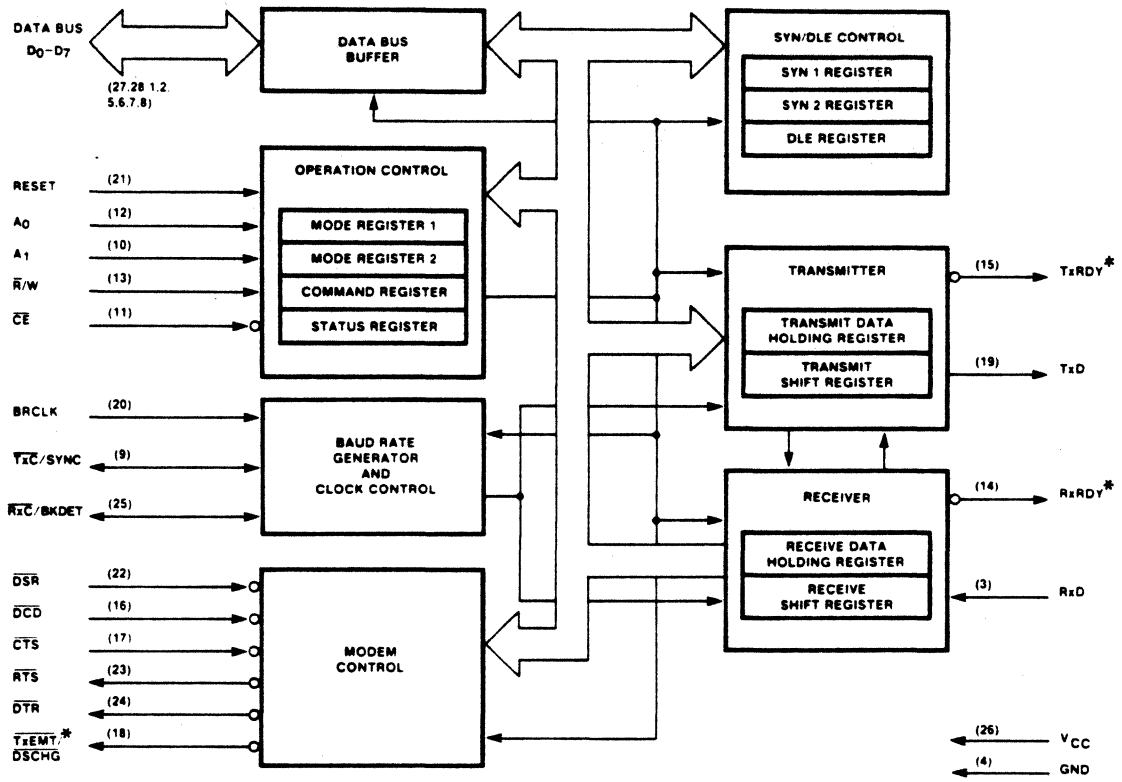
C SUFFIX
CERDIP PACKAGE
CASE 733

PIN ASSIGNMENT



This document contains information on a new product. Specifications and information herein are subject to change without notice.

BLOCK DIAGRAM



NOTE
* Open drain output pin.

BLOCK DIAGRAM

The EPCI consists of six major sections. These are the transmitter, receiver, timing, operation control, modem control and SYN/DLE control. These sections communicate with each other via an internal data bus and an internal control bus. The internal data bus interfaces to the microprocessor data bus via a data bus buffer.

Operation Control

This functional block stores configuration and operation commands from the CPU and generates appropriate signals to various internal sections to control the overall device operation. It contains read and write circuits to permit communications with the microprocessor via the data bus and contains mode registers 1 and 2, the command register, and the status register. Details of register addressing and protocol are presented in the EPCI programming section of this data sheet.

Table 1 BAUD RATE GENERATOR CHARACTERISTICS
Set B (BRCLK = 4.9152MHz)

MR23-20	BAUD RATE	ACTUAL FREQUENCY 16X CLOCK	PERCENT ERROR	DIVISOR
0000	45.5	0.7279kHz	0.005	6752
0001	50	0.8	-	6144
0010	75	1.2	-	4096
0011	110	1.7598	−0.01	2793
0100	134.5	2.152	-	2284
0101	150	2.4	-	2048
0110	300	4.8	-	1024
0111	600	9.6	-	512
1000	1200	19.2	-	256
1001	1800	28.7438	−0.19	171
1010	2000	31.9168	−0.26	154
1011	2400	38.4	-	128
1100	4800	76.8	-	64
1101	9600	153.6	-	32
1110	19200	307.2	-	16
1111	38400	614.4	-	8

Timing
The EPCI contains a baud rate generator (BRG) which is programmable to accept external transmit or receive clocks or to divide an external clock to perform data communications. The unit can generate 16 commonly used baud rates, any one of which can be selected for full duplex operation. See table 1.

Receiver
The receiver accepts serial data on the RxD pin, converts this serial input to parallel format, checks for bits or characters that are unique to the communication technique and sends an "assembled" character to the CPU.

Transmitter
The transmitter accepts parallel data from the CPU, converts it to a serial bit stream, inserts the appropriate characters or bits (based on the communication technique) and outputs a composite serial stream of data on the TxD output pin.

Modem Control
The modem control section provides interfacing for three input signals and three output signals used for "handshaking" and status indication between the CPU and a modem.

SYN/DLE Control
This section contains control circuitry and three 8-bit registers storing the SYN1, SYN2, and DLE characters provided by the CPU. These registers are used in the synchronous mode of operation to provide the characters required for synchronization, idle fill and data transparency.

Table 2 CPU-RELATED SIGNALS

PIN NAME	PIN NO.	INPUT/ OUTPUT	FUNCTION
V_{CC}	26	I	+5V supply input
GND	4	I	Ground
RESET	21	I	A high on this input performs a master reset on the 2661. This signal asynchronously terminates any device activity and clears the mode, command and status registers. The device assumes the idle state and remains there until initialized with the appropriate control words.
A_1-A_0	10,12	I	Address lines used to select internal EPCI registers.
$\overline{R}/W$	13	I	Read command when low, write command when high.
$\overline{CE}$	11	I	Chip enable command. When low, indicates that control and data lines to the EPCI are valid and that the operation specified by the $\overline{R}/W$, A_1 and A_0 inputs should be performed. When high, places the D_0-D_7 lines in the three-state condition.
D_7-D_0	8,7,6,5, 2,1,28,17	I/O	8-bit, three-state data bus used to transfer commands, data and status between EPCI and the CPU. D_0 is the least significant bit; D_7 the most significant bit.
$\overline{TxRDY}$	15	O	This output is the complement of status register bit SR0. When low, it indicates that the transmit data holding register (THR) is ready to accept a data character from the CPU. It goes high when the data character is loaded. This output is valid only when the transmitter is enabled. It is an open drain output which can be used as an interrupt to the CPU.
$\overline{RxRDY}$	14	O	This output is the complement of status register bit SR1. When low, it indicates that the receive data holding register (RHR) has a character ready for input to the CPU. It goes high when the RHR is read by the CPU, and also when the receiver is disabled. It is an open drain output which can be used as an interrupt to the CPU.
$\overline{TxEMT}$/ $\overline{DSCHG}$	18	O	This output is the complement of status register bit SR2. When low, it indicates that the transmitter has completed serialization of the last character loaded by the CPU, or that a change of state of the $\overline{DSR}$ or $\overline{DCD}$ inputs has occurred. This output goes high when the status register is read by the CPU, if the TxEMT condition does not exist. Otherwise, the THR must be loaded by the CPU for this line to go high. It is an open drain output which can be used as an interrupt to the CPU.

OPERATION

The functional operation of the 2661 is programmed by a set of control words supplied by the CPU. These control words specify items such as synchronous or asynchronous mode, baud rate, number of bits per character, etc. The programming procedure is described in the EPCI programming section of the data sheet.

After programming, the EPCI is ready to perform the desired communications functions. The receiver performs serial to parallel conversion of data received from a modem or equivalent device. The transmitter converts parallel data received from the CPU to a serial bit stream. These actions are accomplished within the framework specified by the control words.

Receiver

The 2661 is conditioned to receive data when the $\overline{DCD}$ input is low and the RxEN bit in the command register is true. In the asynchronous mode, the receiver looks for a high to low (mark to space) transition of the start bit on the RxD input line. If a transition is detected, the state of the RxD line is sampled again after a delay of one-half of a bit time. If RxD is now high, the search for a valid start bit is begun again. If RxD is still low, a valid start bit is assumed and the receiver continues to sample the input line at one bit time intervals until the proper number of data bits, the parity bit, and one stop bit have been assembled. The data are then transferred to the receive data holding register, the RxRDY bit in the status register is set, and the $\overline{RxRDY}$ output is asserted. If the character length is less than 8 bits, the high order unused bits in the holding register are set to zero. The parity error, framing error, and overrun error status bits are strobed into the status register on the positive going edge of $\overline{RxC}$ corresponding to the received character boundary. If the stop bit is present, the receiver will immediately begin its search for the next start bit. If the stop bit is absent (framing error), the receiver will interpret a space as a start bit if it persists into the next bit time interval. If a break condition is detected (RxD is low for the entire character as well as the stop bit), only one character consisting of all zeros (with the FE status bit SR5 set) will be transferred to the holding register. The RxD input must return to a high condition before a search for the next start bit begins.

Pin 25 can be programmed to be a break detect output by appropriate setting of MR27-MR24. If so, a detected break will cause that pin to go high. When RxD returns to mark for one RxC time, pin 25 will go low. Refer to the break detection timing diagram.

Table 3 DEVICE-RELATED SIGNALS

PIN NAME	PIN NO.	INPUT / OUTPUT	FUNCTION
BRCLK	20	I	Clock input to the internal baud rate generator (see table 1). Not required if external receiver and transmitter clocks are used
*$\overline{RxC}$ / BKDET	25	I/O	Receiver clock. If external receiver clock is programmed, this input controls the rate at which the character is to be received. Its frequency is 1X, 16X or 64X the baud rate, as programmed by mode register 1. Data are sampled on the rising edge of the clock. If internal receiver clock is programmed, this pin can be a 1X / 16X clock or a break detect output pin.
*$\overline{TxC}$ / XSYNC	9	I/O	Transmitter clock. If external transmitter clock is programmed, this input controls the rate at which the character is transmitted. Its frequency is 1X, 16X or 64X the baud rate, as programmed by mode register 1. The transmitted data changes on the falling edge of the clock. If internal transmitter clock is programmed, this pin can be a 1X / 16X clock output or an external jam synchronization input.
RxD	3	I	Serial data input to the receiver. "Mark" is high, "space" is low.
TxD	19	O	Serial data output from the transmitter. "Mark" is high, "space" is low. Held in mark condition when the transmitter is disabled.
$\overline{DSR}$	22	I	General purpose input which can be used for data set ready or ring indicator condition. Its complement appears as status register bit SR7. Causes a low output on $\overline{TxEMT}$ / $\overline{DSCHG}$ when its state changes if CR2 or CR0 = 1.
$\overline{DCD}$	16	I	Data carrier detect input. Must be low in order for the receiver to operate. Its complement appears as status register bit SR6. Causes a low output on $\overline{TxEMT}$ / $\overline{DSCHG}$ when its state changes if CR2 or CR0 = 1. If $\overline{DCD}$ goes high while receiving, the RxC is internally inhibited.
$\overline{CTS}$	17	I	Clear to send input. Must be low in order for the transmitter to operate. If it goes high during transmission, the character in the transmit shift register will be transmitted before termination.
$\overline{DTR}$	24	O	General purpose output which is the complement of command register bit CR1. Normally used to indicate data terminal ready.
$\overline{RTS}$	23	O	General purpose output which is the complement of command register bit CR5. Normally used to indicate request to send. If the transmit shift register is not empty when CR5 is reset (1 to 0), then $\overline{RTS}$ will go high one TxC time after the last serial bit is transmitted.

NOTE
*$\overline{RxC}$ and $\overline{TxC}$ outputs have short circuit protection max C_L = 100pF Outputs become open circuited upon detection of a zero pulled high or a one pulled low

When the EPCI is initialized into the synchronous mode, the receiver first enters the hunt mode on a 0 to 1 transition of RxEN(CR2). In this mode, as data are shifted into the receiver shift register a bit at a time, the contents of the register are compared to the contents of the SYN1 register. If the two are not equal, the next bit is shifted in and the comparison is repeated. When the two registers match, the hunt mode is terminated and character assembly mode begins. If single SYN operation is programmed, the SYN DETECT status bit is set. If double SYN operation is programmed, the first character assembled after SYN1 must be SYN2 in order for the SYN DETECT bit to be set. Otherwise, the EPCI returns to the hunt mode. (Note that the sequence SYN1-SYN1-SYN2 will not achieve synchronization.) When synchronization has been achieved, the EPCI continues to assemble characters and transfer them to the holding register, setting the RxRDY status bit and asserting the $\overline{RxRDY}$ output each time a character is transferred. The PE and OE status bits are set as appropriate. Further receipt of the appropriate SYN sequence sets the SYN DETECT status bit. If the SYN stripping mode is commanded, SYN characters are not transferred to the holding register. Note that the SYN characters used to establish initial synchronization are not transferred to the holding register in any case.

External jam synchronization can be achieved via pin 9 by appropriate setting of MR27-MR24. When pin 9 is an XSYNC input, the internal SYN1, SYN1-SYN2, and DLE-SYN1 detection is disabled. Each positive going signal on XSYNC will cause the receiver to establish synchronization on the rising edge of the next RxC pulse. Character assembly will start with the RxD input at this edge. XSYNC may be lowered on the next rising edge of RxC. This external synchronization will cause the SYN DETECT status bit to be set until the status register is read. Refer to XSYNC timing diagram.

Transmitter

The EPCI is conditioned to transmit data when the $\overline{CTS}$ input is low and the TxEN command register bit is set. The 2661 indicates to the CPU that it can accept a character for transmission by setting the TxRDY status bit and asserting the $\overline{TxRDY}$ output. When the CPU writes a character into the transmit data holding register, these conditions are negated. Data are transferred from the holding register to the transmit shift register when it is idle or has completed transmission of the previous character. The TxRDY conditions are then asserted again. Thus, one full character time of buffering is provided.

In the asynchronous mode, the transmitter automatically sends a start bit followed by the programmed number of data bits, the least significant bit being sent first. It then appends an optional odd or even parity bit and the programmed number of stop bits. If, following transmission of the data bits, a new character is not available in the transmit holding register, the TxD output remains in the marking (high) condition and the $\overline{TxEMT}/\overline{DSCHG}$ output and its corresponding status bit are asserted. Transmission resumes when the CPU loads a new character into the holding register. The transmitter can be forced to output a continuous low (BREAK) condition by setting the send break command bit (CR3) high.

In the synchronous mode, when the 2661 is initially conditioned to transmit, the TxD output remains high and the TxRDY condition is asserted until the first character to be transmitted (usually a SYN character) is loaded by the CPU. Subsequent to this, a continuous stream of characters is transmitted. No extra bits (other than parity, if commanded) are generated by the EPCI unless the CPU fails to send a new character to the EPCI by the time the transmitter has completed sending the previous character. Since synchronous communication does not allow gaps between characters, the EPCI asserts TxEMT and automatically "fills" the gap by transmitting SYN1s, SYN1-SYN2 doublets, or DLE-SYN1 doublets, depending on the state of MR16 and MR17. Normal transmission of the message resumes when a new character is available in the transmit data holding register. If the SEND DLE bit in the command register is true, the DLE character is automatically transmitted prior to transmission of the message character in the THR.

EPCI PROGRAMMING

Prior to initiating data communications, the 2661 operational mode must be programmed by performing write operations to the mode and command registers. In addition, if synchronous operation is programmed, the appropriate SYN/DLE registers must be loaded. The EPCI can be reconfigured at any time during program execution. A flowchart of the intialization process appears in figure 1.

The internal registers of the EPCI are accessed by applying specific signals to the $\overline{CE}$, $\overline{R}/W$, A_1 and A_0 inputs. The conditions necessary to address each register are shown in table 4.

The SYN1, SYN2, and DLE registers are accessed by performing write operations with the conditions $A_1 = 0$, $A_0 = 1$, and

Table 4 MC2661/MC68661 REGISTER ADDRESSING

$\overline{CE}$	A_1	A_0	$\overline{R}/W$	FUNCTION
1	X	X	X	Three-state data bus
0	0	0	0	Read receive holding register
0	0	0	1	Write transmit holding register
0	0	1	0	Read status register
0	0	1	1	Write SYN1/SYN2/DLE registers
0	1	0	0	Read mode registers ½
0	1	0	1	Write mode registers ½
0	1	1	0	Read command register
0	1	1	1	Write command register

NOTE
See AC characteristics section for timing requirements

MC2661/MC68661 INITIALIZATION FLOW CHART

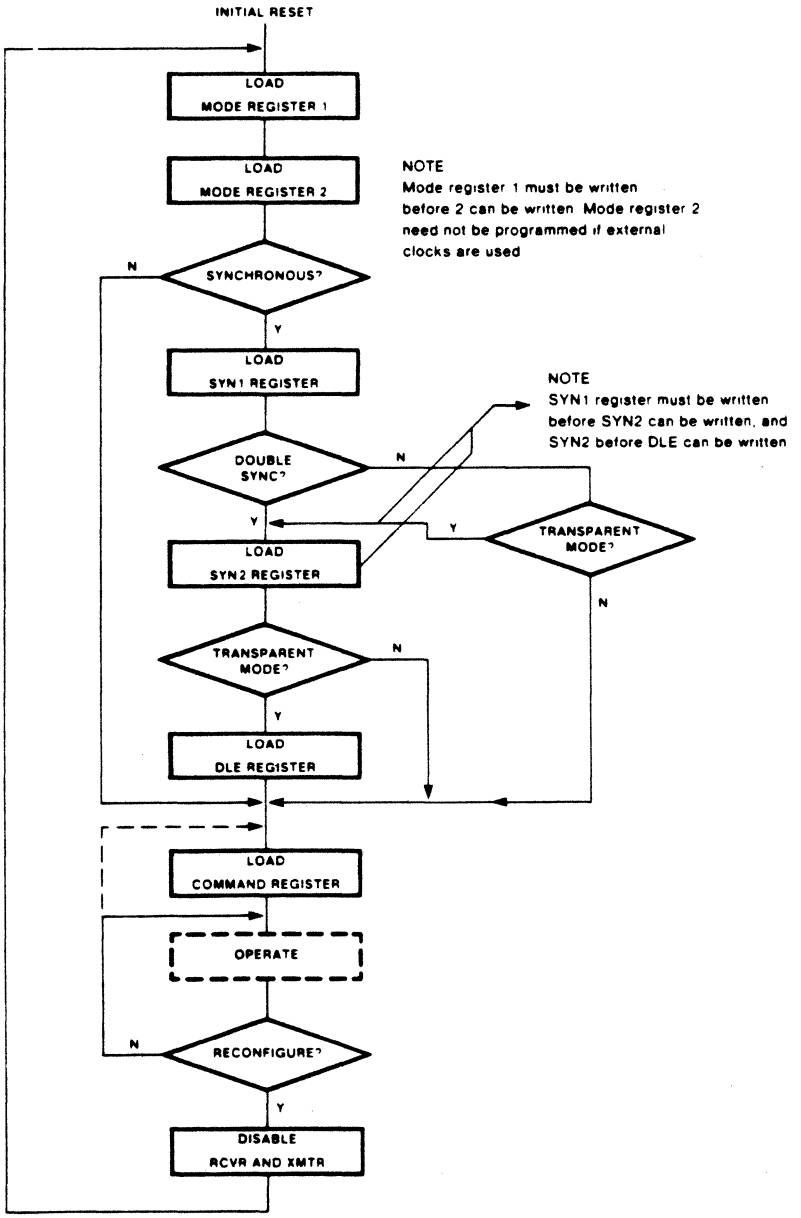


Figure 1

$\overline{R}/W$ = 1. The first operation loads the SYN1 register. The next loads the SYN2 register, and the third loads the DLE register. Reading or loading the mode registers is done in a similar manner. The first write (or read) operation addresses mode register 1, and a subsequent operation addresses mode register 2. If more than the required number of accesses are made, the internal sequencer recycles to point at the first register. The pointers are reset to SYN1 register and mode register 1 by a RESET input or by performing a read command register operation, but are unaffected by any other read or write operation.

The 2661 register formats are summarized in tables 5, 6, 7 and 8. Mode registers 1 and 2 define the general operational characteristics of the EPCI, while the command register controls the operation within this basic framework. The EPCI indicates its status in the status register. These registers are cleared when a RESET input is applied.

Mode Register 1 (MR1)

Table 5 illustrates Mode Register 1. Bits MR11 and MR10 select the communication format and baud rate multiplier. 00 specifies synchronous mode and 1X multiplier. 1X, 16X, and 64X multipliers are programmable for asynchronous format. However, the multiplier in asynchronous format applies only if the external clock input option is selected MR24 or MR25

MR13 and MR12 select a character length of 5, 6, 7 or 8 bits. The character length does not include the parity bit, if programmed, and does not include the start and stop bits in asynchronous mode.

MR14 controls parity generation. If enabled, a parity bit is added to the transmitted character and the receiver performs a parity check on incoming data. MR15 selects odd or even parity when parity is enabled by MR14.

In asynchronous mode, MR17 and MR16 select character framing of 1, 1.5, or 2 stop bits. (If 1X baud rate is programmed, 1.5 stop bits defaults to 1 stop bits on transmit.) In synchronous mode, MR17 controls the number of SYN characters used to establish synchronization and for character fill when the transmitter is idle. SYN1 alone is used if MR17 = 1, and SYN1-SYN2 is used when MR17 = 0. If the transparent mode is specified by MR16, DLE-SYN1 is used for character fill and SYN detect, but the normal synchronization sequence is used to establish character sync. When transmitting, a DLE character in the transmit holding register will cause a second DLE character to be transmitted. This DLE stuffing eliminates the software DLE compare and stuff on each transparent mode data character. If the send DLE command (CR3) is active when a DLE is loaded into THR, only one additional DLE will be transmitted. Also, DLE stripping and DLE detect (with MR14 = 0) are enabled.

The bits in the mode register affecting character assembly and disassembly (MR12-MR16) can be changed dynamically (during active receive/transmit operation). The character mode register affects both the transmitter and receiver; therefore in synchronous mode, changes should be made only in half duplex mode (RxEN = 1 or TxEN = 1, but not both simultaneously = 1). In asynchronous mode, character changes should be made when RxEN and TxEN=0 or when TxEN = 1 and the transmitter is marking in half duplex mode (RxEN = 0).

To effect assembly/disassembly of the next received/transmitted character, MR12-15 must be changed within n bit times of the active going state of $\overline{RxRDY}/\overline{TxRDY}$. Transparent and non-transparent mode changes (MR16) must occur within n-1 bit times of the character to be affected when the receiver or transmitter is active. (n = smaller of the new and old character lengths.)

Mode Register 2 (MR2)

Table 6 illustrates mode register 2. MR23, MR22, MR21 and MR20 control the frequency of the internal baud rate generator (BRG). Sixteen rates are selectable for each EPCI version (A, B, C). Versions A and B specify a 4.9152 MHz TTL input at BRCLK (pin 20); version C specifies a 5.0688 MHz input which is identical to the Signetics 2651. MR23-20 are don't cares if external clocks are selected (MR25-MR24 = 0). The individual rates are given in table 1.

MR24-MR27 select the receive and transmit clock source (either the BRG or an external input) and the function at pins 9 and 25. Refer to table 6.

Command Register (CR)

Table 7 illustrates the command register. Bits CR0 (TxEN) and CR2 (RxEN) enable or disable the transmitter and receiver respectively. A 0 to 1 transition of CR2 forces start bit search (async mode) or hunt mode (sync mode) on the second $\overline{RxC}$ rising edge. Disabling the receiver causes $\overline{RxRDY}$ to go high (inactive). If the transmitter is disabled, it will complete the transmission of the character in the transmit shift register (if any) prior to terminating operation. The TxD output will then remain in the marking state

Table 5 MODE REGISTER 1 (MR 1)

<table>
<tr><th colspan="2">MR17 MR16</th><th>MR15</th><th>MR14</th><th>MR13 MR12</th><th>MR11 MR10</th></tr>
<tr><td colspan="2">Sync/Async</td><td>Parity Type</td><td>Parity Control</td><td>Character Length</td><td>Mode and Baud Rate Factor</td></tr>
<tr><td colspan="2">Async: Stop Bit Length
00 = Invalid
01 = 1 stop bit
10 = 1½ stop bits
11 = 2 stop bits</td><td rowspan="2">0 = Odd
1 = Even</td><td rowspan="2">0 = Disabled
1 = Enabled</td><td rowspan="2">00 = 5 bits
01 = 6 bits
10 = 7 bits
11 = 8 bits</td><td rowspan="2">00 = Synchronous 1X rate
01 = Asynchronous 1X rate
10 = Asynchronous 16X rate
11 = Asynchronous 64X rate</td></tr>
<tr><td>Sync:
Number of SYN char
0 = Double SYN
1 = Single SYN</td><td>Sync:
Transparency Control
0 = Normal
1 = Transparent</td></tr>
</table>

NOTE

Baud rate factor in asynchronous applies only if external clock is selected. Factor is 16X if external clock is selected. Mode must be selected (MR11, MR10) in any case

Table 6 MODE REGISTER 2 (MR2)

MR27–MR24											MR23–MR20
	TxC	RxC	Pin 9	Pin 25		TxC	RxC	Pin 9	Pin 25	Mode	Baud Rate Selection
0000	E	E	TxC	RxC	1000	E	E	XSYNC[1]	RxC/TxC	sync	
0001	E	I	TxC	1X	1001	E	I	TxC	BKDET	async	
0010	I	E	1X	RxC	1010	I	E	XSYNC[1]	RxC	sync	
0011	I	I	1X	1X	1011	I	I	1X	BKDET	async	See baud rates in table 1
0100	E	E	TxC	RxC	1100	E	E	XSYNC[1]	RxC/TxC	sync	
0101	E	I	TxC	16X	1101	E	I	TxC	BKDET	async	
0110	I	E	16X	RxC	1110	I	E	XSYNC[1]	RxC	sync	
0111	I	I	16X	16X	1111	I	I	16X	BKDET	async	

NOTES
1. When pin 9 is programmed as XSYNC input, SYN1, SYN1-SYN2, and DLE-SYN1 detection is disabled

E = External clock
I = Internal clock (BRG)
1X and 16X are clock outputs

Table 7 COMMAND REGISTER (CR)

CR7 CR6	CR5	CR4	CR3	CR2	CR1	CR0
Operating Mode	**Request To Send**	**Reset Error**	**Sync/Async**	**Receive Control (RxEN)**	**Data Terminal Ready**	**Transmit Control (TxEN)**
00 = Normal operation 01 = Async: Automatic echo mode Sync: SYN and/or DLE stripping mode 10 = Local loop back 11 = Remote loop back	0 = Force $\overline{RTS}$ output high one clock time after TxSR serialization 1 = Force $\overline{RTS}$ output low	0 = Normal 1 = Reset error flags in status register (FE, OE, PE/DLE detect)	**Async:** **Force break** 0 = Normal 1 = Force break	0 = Disable 1 = Enable	0 = Force $\overline{DTR}$ output high 1 = Force $\overline{DTR}$ output low	0 = Disable 1 = Enable
			Sync: **Send DLE** 0 = Normal 1 = Send DLE			

Table 8 STATUS REGISTER (SR)

SR7	SR6	SR5	SR4	SR3	SR2	SR1	SR0
Data Set Ready	**Data Carrier Detect**	**FE SYN Detect**	**Overrun**	**PE DLE Detect**	**TxEMT DSCHG**	**RxRDY**	**TxRDY**
0 = $\overline{DSR}$ input is high 1 = $\overline{DSR}$ input is low	0 = $\overline{DCD}$ input is high 1 = $\overline{DCD}$ input is low	**Async:** 0 = Normal 1 = Framing Error	0 = Normal 1 = Overrun Error	**Async:** 0 = Normal 1 = Parity error	0 = Normal 1 = Change in $\overline{DSR}$, or $\overline{DCD}$, or transmit shift register is empty	0 = Receive holding register empty 1 = Receive holding register has data	0 = Transmit holding register busy 1 = Transmit holding register empty
		Sync: 0 = Normal 1 = SYN detected		**Sync:** 0 = Normal 1 = Parity error or DLE received			

(high) while $\overline{TxRDY}$ and $\overline{TxEMT}$ will go high (inactive). If the receiver is disabled, it will terminate operation immediately. Any character being assembled will be neglected. A 0 to 1 transition of CR2 will initiate start bit search (async) or hunt mode (sync).

Bits CR1 (DTR) and CR5 (RTS) control the $\overline{DTR}$ and $\overline{RTS}$ outputs. Data at the outputs are the logical complement of the register data.

In asynchronous mode, setting CR3 will force and hold the TxD output low (spacing condition) at the end of the current transmitted character. Normal operation resumes when CR3 is cleared. The TxD line will go high for at least one bit time before beginning transmission of the next character in the transmit data holding register. In synchronous mode, setting CR3 causes the transmission of the DLE register contents prior to sending the character in the transmit data holding register. Since this is a one time command, CR3 does not have to be reset by software. CR3 should be set when entering and exiting transparent mode and for all DLE—non-DLE character sequences.

Setting CR4 causes the error flags in the status register (SR3, SR4, and SR5) to be cleared. This is a one time command. There is no internal latch for this bit.

Table 9 MC2661/MC68661 EPCI vs SIGNETICS 2651 PCI

FEATURE	EPCI	PCI
1. MR2 Bit 6, 7	Control pin 9, 25	Not used
2. DLE detect-SR3	SR3 = 0 for DLE-DLE, DLE-SYNC1	SR3 = 1 for DLE-DLE, DLE-SYNC1
3. Reset of SR3, DLE detect	Second character after DLE, or receiver disable, or CR4 = 1	Receiver disable, or CR4 = 1
4. Send DLE-CR3	One time command	Reset via CR3 on next $\overline{\text{TxRDY}}$
5. DLE stuffing in transparent mode	Automatic DLE stuffing when DLE is loaded except if CR3 = 1	None
6. SYNC1 stripping in double sync non-transparent mode	All SYNC1	First SYNC1 of pair
7. Baud rate versions	Three	One
8. Terminate ASYNC transmission (drop RTS)	Reset CR5 in response to $\overline{\text{TxRDY}}$ changing from 0 to 1	Reset CR0 when $\overline{\text{TxEMT}}$ goes from 1 to 0. Then reset CR5 when $\overline{\text{TxEMT}}$ goes from 0 to 1
9. Break detect	Pin 25	FE and null character
10. Stop bit searched	One	Two
11. External jam sync	Pin 9	No
12. Data bus timing	Improved over 2651	—
13. Data bus drivers	Sink 2.2mA	Sink 1.6mA
	Source 400μA	Source 100μA

NOTES
1. Internal BRG used for RxC
2. Internal BRG used for TxC

When CR5 (RTS) is set, the $\overline{\text{RTS}}$ pin is forced low and the transmit serial logic is enabled. A 1 to 0 transition of CR5 will cause $\overline{\text{RTS}}$ to go high (inactive) one TxC time after the last serial bit has been transmitted (if the transmit shift register was not empty).

The EPCI can operate in one of four sub-modes within each major mode (synchronous or asynchronous). The operational sub-mode is determined by CR7 and CR6. CR7-CR6 = 00 is the normal mode, with the transmitter and receiver operating independently in accordance with the mode and status register instructions.

In asynchronous mode, CR7-CR6 = 01 places the EPCI in the automatic echo mode. Clocked, regenerated received data are automatically directed to the TxD line while normal receiver operation continues. The receiver must be enabled (CR2 = 1), but the transmitter need not be enabled. CPU to receiver communications continues normally, but the CPU to transmitter link is disabled. Only the first character of a break condition is echoed. The TxD output will go high until the next valid start is detected. The following conditions are true while in automatic echo mode:

1. Data assembled by the receiver are automatically placed in the transmit holding register and retransmitted by the transmitter on the TxD output.
2. The transmitter is clocked by the receive clock.
3. $\overline{\text{TxRDY}}$ output = 1.
4. The $\overline{\text{TxEMT}}$/$\overline{\text{DSCHG}}$ pin will reflect only the data set change condition.
5. The TxEN command (CR0) is ignored.

In synchronous mode, CR7-CR6 = 01 places the EPCI in the automatic SYN/DLE stripping mode. The exact action taken depends on the setting of bits MR17 and MR16:

1. In the non-transparent, single SYN mode (MR17-MR16 = 10), characters in the data stream matching SYN1 are not transferred to the receive data holding register (RHR).
2. In the non-transparent, double SYN mode (MR17-MR16 = 00), characters in the data stream matching SYN1, or SYN2 if immediately preceded by SYN1, are not transferred to the RHR.
3. In transparent mode (MR16 = 1), characters in the data stream matching DLE, or SYN1 if immediately preceded by DLE, are not transferred to the RHR. However, only the first DLE of a DLE-DLE pair is stripped.

Note that automatic stripping mode does not affect the setting of the DLE detect and SYN detect status bits (SR3 and SR5).

Two diagnostic sub-modes can also be configured. In local loop back mode (CR7-CR6 = 10), the following loops are connected internally:

1. The transmitter output is connected to the receiver input.
2. $\overline{\text{DTR}}$ is connected to $\overline{\text{DCD}}$ and $\overline{\text{RTS}}$ is connected to $\overline{\text{CTS}}$.
3. The receiver is clocked by the transmit clock.
4. The $\overline{\text{DTR}}$, $\overline{\text{RTS}}$ and $\overline{\text{TxD}}$ outputs are held high.
5. The $\overline{\text{CTS}}$, $\overline{\text{DCD}}$, $\overline{\text{DSR}}$ and RxD inputs are ignored.

Additional requirements to operate in the local loop back mode are that CR0 (TxEN), CR1 (DTR), and CR5 (RTS) must be set to 1. CR2 (RxEN) is ignored by the EPCI.

The second diagnostic mode is the remote loop back mode (CR7-CR6 = 11). In this mode:

1. Data assembled by the receiver are automatically placed in the transmit holding register and retransmitted by the transmitter on the TxD output.
2. The transmitter is clocked by the receive clock.
3. No data are sent to the local CPU, but the error status conditions (PE, OE, FE) are set.
4. The $\overline{\text{RxRDY}}$, $\overline{\text{TxRDY}}$, and $\overline{\text{TxEMT}}$/$\overline{\text{DSCHG}}$ outputs are held high.
5. CR1 (TxEN) is ignored.
6. All other signals operate normally.

Status Register

The data contained in the status register (as shown in table 8) indicate receiver and transmitter conditions and modem/data set status.

SR0 is the transmitter ready (TxRDY) status bit. It, and its corresponding output, are valid only when the transmitter is enabled. If equal to 0, it indicates that the transmit data holding register has been loaded by the CPU and the data has not been transferred to the transmit shift register. If set equal to 1, it indicates that the holding register is ready to accept data from the CPU. This bit is initially set when the transmitter is enabled by CR0, unless a character has previously been loaded into the holding register. It is not set when the automatic echo or remote loopback modes are programmed. When this bit is set, the $\overline{\text{TxRDY}}$ output pin is low. In

the automatic echo and remote loop back modes, the output is held high.

SR1, the receiver ready (RxRDY) status bit, indicates the condition of the receive data holding register. If set, it indicates that a character has been loaded into the holding register from the receive shift register and is ready to be read by the CPU. If equal to zero, there is no new character in the holding register. This bit is cleared when the CPU reads the receive data holding register or when the receiver is disabled by CR2. When set, the $\overline{\text{RxRDY}}$ output is low.

The TxEMT / DSCHG bit, SR2, when set, indicates either a change of state of the $\overline{\text{DSR}}$ or $\overline{\text{DCD}}$ inputs (when CR2 or CR0 = 1) or that the transmit shift register has completed transmission of a character and no new character has been loaded into the transmit data holding register. Note that in synchronous mode this bit will be set even though the appropriate "fill" character is transmitted. TxEMT will not go active until at least one character has been transmitted. It is cleared by loading the transmit data holding register. The DSCHG condition is enabled when TxEN = 1 or RxEN = 1. It is cleared when the status register is read by the CPU. If the status register is read twice and SR2 = 1 while SR6 and SR7 remain unchanged, then a TxEMT condition exists. When SR2 is set, the $\overline{\text{TxEMT}}$ / $\overline{\text{DSCHG}}$ output is low.

SR3, when set, indicates a received parity error when parity is enabled by MR14. In synchronous transparent mode (MR16 = 1), with parity disabled, it indicates that a character matching DLE register was received and the present character is neither SYN1 nor DLE. This bit is cleared when the next character following the above sequence is loaded into RHR, when the receiver is disabled, or by a reset error command, CR4.

The overrun error status bit, SR4, indicates that the previous character loaded into the receive holding register was not read by the CPU at the time a new received character was transferred into it. This bit is cleared when the receiver is disabled or by the reset error command, CR4.

In asynchronous mode, bit SR5 signifies that the received character was not framed by a stop bit, i.e., only the first stop bit is checked. If RHR = 0 when SR5 = 1, a break condition is present. In synchronous non-transparent mode (MR16 = 0), it indicates receipt of the SYN1 character in single SYN mode or the SYN1-SYN2 pair in double SYN mode. In synchronous transparent mode (MR16 = 1), this bit is set upon detection of the initial synchronizing characters (SYN1 or SYN1-SYN2) and, after synchronization has been achieved, when a DLE-SYN1 pair is received. The bit is reset when the receiver is disabled, when the reset error command is given in asynchronous mode, or when the status register is read by the CPU in the synchronous mode.

SR6 and SR7 reflect the conditions of the $\overline{\text{DCD}}$ and $\overline{\text{DSR}}$ inputs respectively. A low input sets its corresponding status bit, and a high input clears it.

WARRANTY

Dual Systems Corporation warrants the equipment covered hereby to be free from defects in material and workmanship for twelve (12) months from date of original shipment to purchaser. During this warranty period Dual Systems will repair or replace defective equipment FOB its place of business without charge to purchaser.

This warranty applies to defects arising out of normal use and service of the equipment as specified by Dual Systems. This warranty does not cover abnormal operation of the equipment, accident, alteration, negligence, misuse and repairs or service performed by other than Dual Systems' authorized representatives. Purchaser shall upon request by Dual Systems furnish reasonable evidence that the defect arose from causes placing a liability on Dual Systems.

The obligation of Dual Systems under this warranty is limited to repair or replacement of the defective equipment and is the only warranty applicable to the equipment. Dual Systems shall not be liable for any injury, loss or damage, direct or consequential, arising out of the use or inability to use the product. No changes in the warranty shall be effective without the prior approval in writing of both parties. This warranty and obligations and liabilities thereunder shall replace all warranties or guarantees express or implied including the implied warranty of merchantability.

Dual Systems Corporation
2530 San Pablo Avenue
Berkeley, California 94702

(415) 549-3854