TECHNICAL MANUAL

AM-310

COMMUNICATIONS CONTROLLER BOARD

DWM-00310-00

REV. A00

alpha micro

FIRST EDITION

JANUARY, 1982

REVISIONS INCORPORATED:	
EDITION	DATE

TABLE OF CONTENTS

SECTION III

PROGRAMMING (con't)

SECTION IV

FUNCTIONAL THEORY OF OPERATION

SECTION IV

FUNCTIONAL THEORY OF OPERATION (con't)

SECTION V

MAINTENANCE AND TROUBLESHOOTING

SECTION VI

PARTS LIST AND SCHEMATIC

LIST OF ILLUSTRATIONS

LIST OF TABLES

SECTION I

GENERAL DESCRIPTION

1.0 INTRODUCTION

This manual provides operating and maintenance instructions for the AM-310 Communications Controller circuit board manufactured by Alpha Microsystems Inc., located in Irvine, California. Circuit board description, operating and usage instructions, programming, theory of operation, and maintenance instructions are included to provide the user with the information necessary to use this circuit board to its full capability.

1.1 CIRCUIT BOARD DESCRIPTION.

The AM-310 Communications Controller circuit board provides communications interface between serial interface devices and the Alpha Micro computer system or any standard S-100 Bus computer. The AM-310 contains four independent, programmable communications ports which provide the data conversion and processing necessary for the requirements of RS-232C compatible devices. The AM-310 can generate multilevel interrupts.

A CPU microprocessor, a DMA controller and a Random Access Memory (RAM) with associated control logic provide the data processing necessary for this sophisticated communications interface. Four Programmable Communications Interface (PCI) modules, one for each port, provide for the programmable baud rates and direct interface with the RS-232C peripheral devices.

A simplified block diagram of the circuit board is shown in Figure 1-1. For a complete detailed description of circuit board operation, see Section IV of this manual. For programming requirements, see Section III of this manual.

1.2 APPLICATION.

The AM-310 provides communications capability with peripheral devices that conform to the requirements of EIA Standard RS-232C.

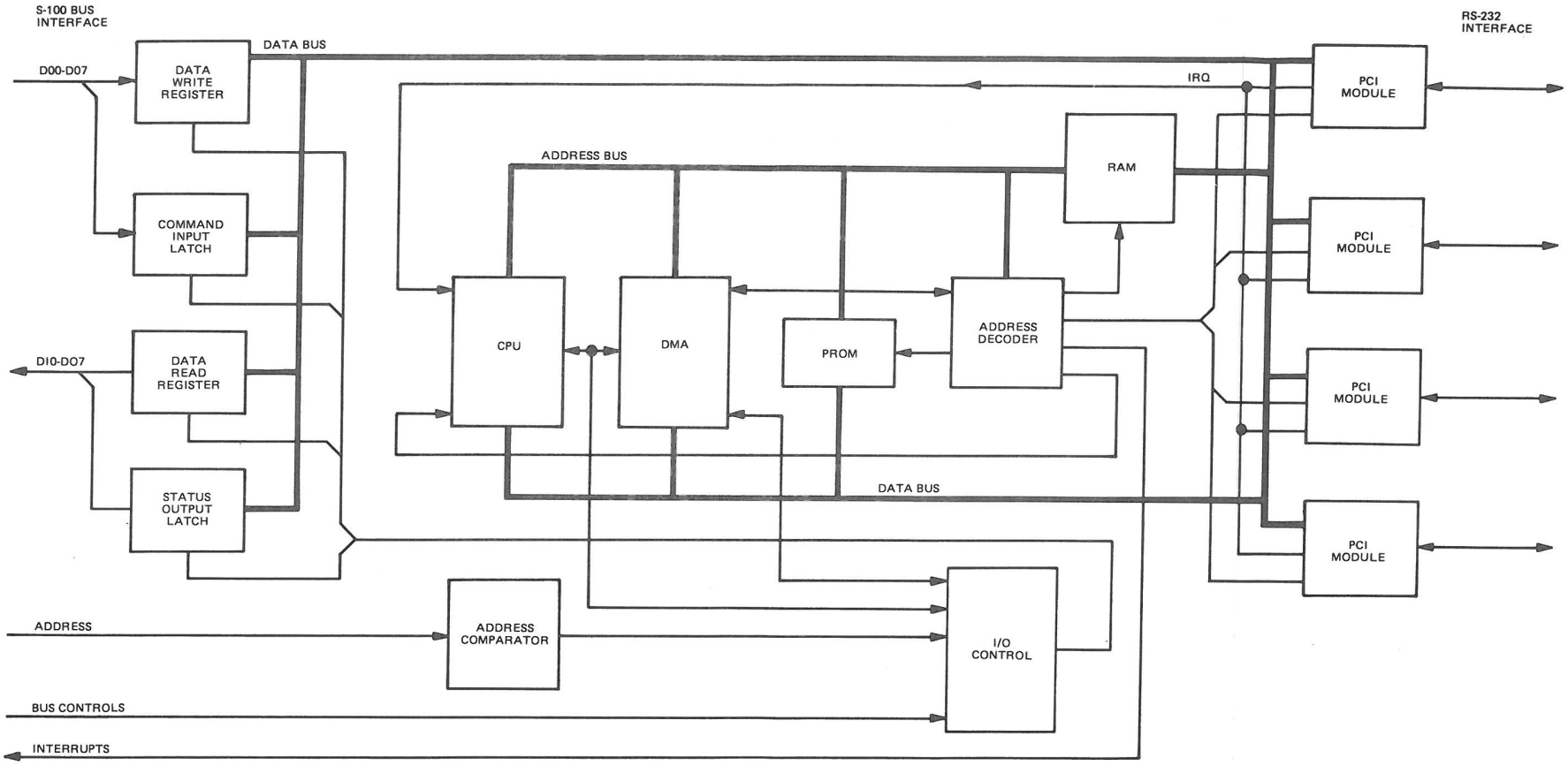


Figure 1-1. AM-310 Simplified Block Diagram

SECTION II
OPERATING DATA

2.0 INTRODUCTION

This section contains information on the use of the AM-310 Communications Controller circuit board. Capabilities, specifications, interface wiring and user options are described for the successful integration of the board into the user's system.

2.1 CAPABILITIES AND SPECIFICATIONS.

This circuit board operates from the standard S-100 Bus structure and the Alpha Micro 16-bit Bus to provide communications capability for up to four peripheral devices. The AM-310 provides an intelligent interface that allows the CPU to communicate with terminals, modems, and other peripheral equipment via asynchronous and synchronous transmission. Specifications for the AM-310 circuit board are contained in Table 2-1.

Table 2-1. AM-310 Specifications

PARAMETER	SPECIFICATION
CPU Interface	Alpha Micro and standard S-100 Bus.
Interrupt Capability	Multi-level interrupt driven.
I/O Interface	EIA RS-232C or other synchronous/asynchronous protocol devices conforming to RS-232 levels.
Number of Ports	Four
Transmit/Receive Modes	Fully programmable, 16 software selectable baud rates, synchronous/asynchronous, jumper programmable external or internal baud rate clock.
Buffer Organization Asynchronous	Input: Double Buffered, 96 Bytes/channel. Output: Single Buffered, 256 Bytes/channel.
Baud Rates	Up to 19.2K Baud: 50, 75, 134.5, 150, 300, 600, 1200, 1800, 2000, 2400, 3600, 4800, 7200, 9600, 19,200.
Data Transfer Mode	Block transfer, programmed I/O, and interrupt driven.

Table 2-1 (Con't.). AM-310 Specifications

PARAMETER	SPECIFICATION
On-Board RAM	2048 Bytes.
Interconnections	One mounting slot of an S-100 Bus chassis. One 26 pin connector for each I/O port.
Dimensions	5 1/4" x 10" (13.3 cm x 25.4 cm).
Input Power (Nominal)	7.5 Volts DC @ 1.25A +16 Volts DC @ 100mA -16 Volts DC @ 100mA
Environment (operating) Temperature Humidity	 60° - 90° F (16° -32° C). 10%-80% (non-condensing).

2.2 INTERFACE DESCRIPTION AND WIRING.

The AM-310 Communications Controller provides communications capability between S-100 Bus systems and RS-232 peripheral devices.

2.2.1 S-100 BUS INTERFACE.

The AM-310 circuit board is fully S-100 Bus compatible. The board and its associated peripherals are addressed through the standard address lines and data is transferred through the standard data in and data out lines. The S-100 Bus connections are made via the bottom edge connector and are listed in Table 2-2. For a complete description of these signals and their operation in the AM-310, see Paragraph 4.1 of this manual.

Table 2-2. Alpha Micro Bus Interface Signals List

MNEMONIC	NAME	PIN NO.
+7.5V	+ 7.5vdc Power	1
+16V	+ 16vdc Power	2
$\overline{\text{VI8}}$	Vectored Interrupt 8	3
$\overline{\text{VI0}}$	Vectored Interrupt 0	4
$\overline{\text{VI1}}$	Vectored Interrupt 1	5
$\overline{\text{VI2}}$	Vectored Interrupt 2	6
$\overline{\text{VI3}}$	Vectored Interrupt 3	7
$\overline{\text{VI4}}$	Vectored Interrupt 4	8
$\overline{\text{VI5}}$	Vectored Interrupt 5	9
$\overline{\text{VI6}}$	Vectored Interrupt 6	10
$\overline{\text{VI7}}$	Vectored Interrupt 7	11
RTC	Real Time Clock, 50Hz or 60Hz	12
POWFAIL	AC Power Failure Status	13
$\overline{\text{VI9}}$	Vectored Interrupt 9	14
A18	Address 18	15
A16	Address 16	16
A17	Address 17	17
$\overline{\text{STATDSB}}$	Status Disable	18
$\overline{\text{C/CDSB}}$	Command/Control Disable	19
GND	Ground	20

Table 2-2.(con't) Alpha Micro Bus Interface Signals List

MNEMONIC	NAME	PIN NO.
$\overline{\text{IODIS}}$	I/O Disable	21
$\overline{\text{ADDDSB}}$	Address Disable	22
$\overline{\text{DODSB}}$	Data Bus Disable	23
Ø2	Phase 2 Clock	24
$\overline{\text{STVAL}}$	Status and Address Valid	25
PHLDA	DMA Request Acknowledge	26
PWAIT	Processor Wait	27
N/U	Not Used	28
A5 A4 A3 A15 A12 A9	Address 5 Address 4 Address 3 Address 15 Address 12 Address 9	29 30 31 32 33 34
DOUT 1/D1 DOUT 0/D0	Data Bus Bit 1 Data Bus Bit 0	35 36
A10	Address 10	37

Table 2-2.(con't) Alpha Micro Bus Interface Signals List

MNEMONIC	NAME	PIN NO.
DOUT 4/D4	Data Bus Bit 4	38
DOUT 5/D5	Data Bus Bit 5	39
DOUT 6/D6	Data Bus Bit 6	40
DIN 2/D10	Data Bus Bit 10	41
DIN 3/D11	Data Bus Bit 11	42
DIN 7/D15	Data Bus Bit 15	43
SMI	Bus Master OP Code Fetch	44
SOUT	I/O Output Cycle	45
SINP	I/O Input Cycle	46
SMEMR	Memory Read Cycle	47
SHLTA	HLT Acknowledge	48
$\overline{\text{PERR}}$	Parity Error Pulse	49
GND	Ground	50
+7.5V	+7.5vdc Power	51
-16V	-16vdc Power	52
GND	Ground	53
$\overline{\text{SLAVECLR}}$	Reset Signal To All I/O Devices	54

Table 2-2.(con't) Alpha Micro Bus Interface Signals List

MNEMONIC	NAME	PIN NO.
$\overline{\text{DMA0}}$	DMA Controller Arbitration	55
$\overline{\text{DMA1}}$	Lines For Use With Standard	56
$\overline{\text{DMA2}}$	S-100 Bus DMA System	57
$\overline{\text{SXTRQ}}$	16 Bit Cycle	58
A19	Address 19	59
N/U	Not Used	60
A20 A21 A22 A23	Address 20 Address 21 Address 22 Address 23	61 62 63 64
$\overline{\text{ADVAL}}$	Address Valid On Data Bus	65
$\overline{\text{WRDIS}}$	Write Disable	66
$\overline{\text{PHANTOM}}$	ROM Memory Enable	67
N/U N/U	Not Used Not Used	68 69
Gnd	Ground	70
N/U	Not Used	71
PRDY	Processor Ready	72

Table 2-2.(con't) Alpha Micro Bus Interface Signals List

MNEMONIC	NAME	PIN NO.
N/U	Not Used	73
$\overline{\text{PHOLD}}$	DMA Request	74
$\overline{\text{PRESET}}$	Preset	75
PSYNC	Processor Sync, Start of Bus Cycle	76
$\overline{\text{PWR}}$	Write Strobe	77
PDBIN	Data Bus Input Command	78
A0	Address 0	79
A1	Address 1	80
A2	Address 2	81
A6	Address 6	82
A7	Address 7	83
A8	Address 8	84
A13	Address 13	85
A14	Address 14	86
A11	Address 11	87
DOUT 2/D2	Data Bus Bit 2	88
DOUT 3/D3	Data Bus Bit 3	89
DOUT 7/D7	Data Bus Bit 7	90
DIN 4/D12	Data Bus Bit 7	91
DIN 5/D13	Data Bus Bit 13	92
DIN 6/D14	Data Bus Bit 14	93
DIN 1/D9	Data Bus Bit 9	94
DIN 0/D8	Data Bus Bit 8	95

Table 2-2.(con't) Alpha Micro Bus Interface Signals List

MNEMONIC	NAME	PIN NO.
SINTA	Interrupt Acknowledge	96
$\overline{\text{SWO}}$	Bus Master Output	97
$\overline{\text{ERROR}}$	Memory Error Interrupt	98
$\overline{\text{BERR}}$	Bus Error	99
GND	Ground	100

2.2.2 RS-232 INTERFACE.

The AM-310 circuit board contains four programmable I/O ports that are compatible with most standard RS-232 peripherals. Connections are made via the four connectors on the top of the circuit board and are listed in Table 2-3. For a complete description of these signals and their operation in the AM-310, see Paragraph 4.1 of this manual.

Table 2-3. RS-232 Interface Signals List

SIGNAL	NAME	J2, J3, J4 J5 PIN NO.
CTS	Clear to Send	4
DCD	Data Carrier Detect	8
DSR	Data Set Ready	20
DTR	Data Terminal Ready	6
RTS	Request to Send	5
RXC	Receiver Clock X1	17
RXD	Serial Input Data	2
TXC	Transmitter Clock X1	15
TXD	Serial Output Data	3
GND	Logic Ground	7
CHASSIS	Chassis Ground	1

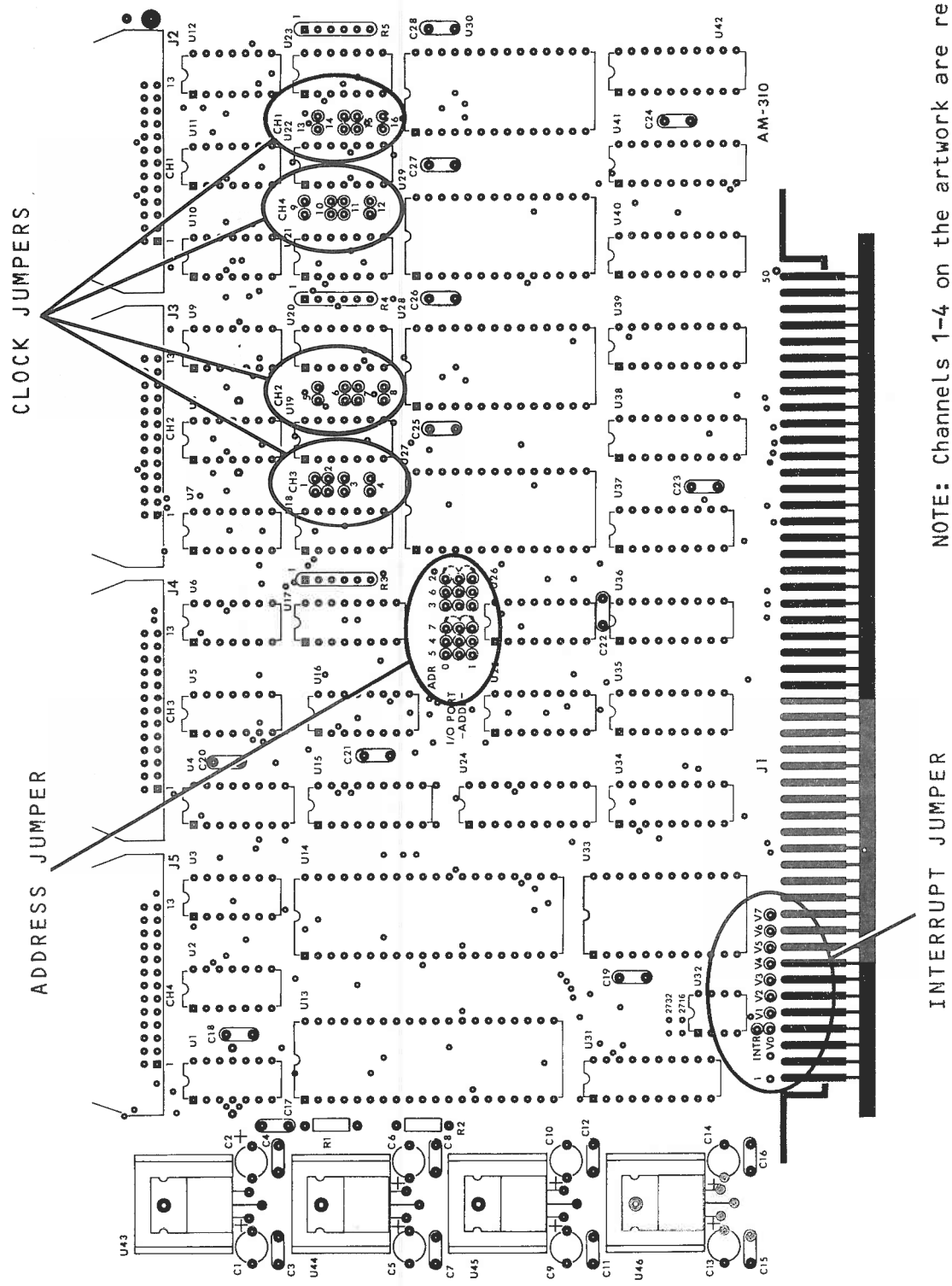


Figure 2-1. AM-310 JUMPER OPTIONS

2.3 USER OPTIONS

Several options are available on the AM-310 that the user may select according to system requirements. These options are described in the following paragraphs.

2.3.1 I/O ADDRESS SELECTION.

The standard I/O address block for the AM-310 is B0-B1. This address is strapped in etch on the circuit board. To change the I/O address block, cut the appropriate etch on the component side of the board and jumper the board for the desired address. See Figure 2-1 for the location of the address jumpers and sheet 2 of the schematic for the address logic. Note that the I/O address block may be set in increments of 4 ports only, even though only two I/O ports are used by the board.

2.3.2 INTERRUPTS.

The interrupt signal from the AM-310 may be strapped to any of the vectored interrupt lines VI0-VI7. The standard configuration connects the AM-310 to VI1. Interrupt jumper location is shown in Figure 2-1.

2.3.3 INSTALLATION INSTRUCTIONS.

Installation of the AM-310 into an Alpha Micro system requires a minimal number of system changes. Before installing the board, take the time to look over these instructions to help eliminate the chances of encountering difficulties.

First, as the AM-310 requires the use of vectored interrupts VI1, you must make sure the jumper on the AM-100 CPU is installed for interrupt level 1. Systems using the AM-100/T do not require installation of an interrupt jumper; they are effectively installed already. Included with the board are cables with 26 pin flat cable connectors on one end and DB-25S (RS-232) connectors on the other. The cables are usually installed by screwing the RS-232 connector into the mainframe of the processor such that port numbers 0 through 3 appear in an orderly fashion when viewed from the back. The flat cables should then be installed in the AM-310 such that the arrows marking pin 1 on the connectors line up correctly. As the board is viewed looking at the component side, port 0 is the far right connector, and the port number increments towards the left, the left-most connector being numbered port 3.

Jumpering has been provided to support synchronous devices' clock signals, TxC and RxC. These signals operate differently for various devices. In the case of asynchronous terminals and modems, clocks are not necessary. However, synchronous modems require that the device attached to them use the clocks the modem produces for synchronization. This being the case, it is necessary also to generate clocks whenever a synchronous terminal is attached to the Alpha Micro. To determine the correct clock jumpering you will need, consult the table on the following page.

RS-232 PORT Numbers	Asynchronous Devices	Synchronous Terminals	Synchronous Modems
0	All Off	13, 15	14, 16
1	All Off	5, 7	6, 8
2	All Off	1, 3	2, 4
3	All Off	9, 11	10, 12

EIA RS-232 specifications indicate two ground lines - logic ground and chassis ground - are to be present in any standard RS-232 connection. As such, logic ground is supported by the AM-310 in standard configuration, but chassis ground is not. Although the great majority of applications (and for that matter, cables) do not support chassis ground, provision for this signal is located in the upper right corner of the AM-310. To install chassis ground, simply connect the ground pad in the corner to your mainframe with a nut and bolt. Finally, the software must be set up. AM310.IDV is to reside in DSK0:[1,6] for correct operation. The SYSTEM.INI file must then be set up for the AM-310. A typical TRMDEF line will look like this:

```
TRMDEF TERM1,AM310=0,SOROC,100,100,100
```

This will set up an asynchronous SOROC terminal running at 19,200 baud, the default value, on RS-232 port 0 of the AM-310 board. To set up other baud rates on synchronous/asynchronous terminals, the long form must be used:

```
TRMDEF TERM1,AM310=0:037716,SOROC,100,100,100
TRMDEF TERM2,AM310=1:033716,SOROC,100,100,100
```

The 16 bit parameter consists of two bytes. The upper byte defines the baud rate and is passed directly to PCI mode register 2. The lower byte controls synchronous/asynchronous operations and other parameters and is passed to PCI mode register 1. Therefore, the first example is simply the long form of setting up a 19,200 baud async terminal, while the second line specifies a baud rate of 1200 on port 1. Detailed operation of both the PCI chips and the overall architecture of the AM-310 is presented so that the inventive user will not have to proceed uninformed. See Section III for more information.

2.3.4 MULTIPLE 310 BOARD CONFIGURATION.

Multiple AM-310's (up to four boards total) may be run by implementing the following procedure.

All AM-310's share the same interrupt line and the software driver AX310.IDV in area DSK0:[1,6] takes care of polling each board to find which has generated the interrupt. The boards must be addressed in a sequential manner, B0 hex, B4 hex, B8 hex, and BC hex. There must be no gaps in this order or the software will not function.

To re-address the boards, refer to Figure 2-1 and do the following:

Directly above U26 is a series of 18 jumper pads arranged in 3 rows of 6 pads. The standard port address of B0 is in etch at these pads. The upper row is used to select a logic 0 and the lower row to select a logic 1. The center row will be jumpered either to the upper or lower row depending on the port. Standard is 5, 4, 7 jumpered to a logic 1 and 3, 6, 2 jumpered to a logic 0. Cut the traces and add jumpers to obtain the port desired.

The following table will show how it is done. An X in the column denotes a jumper.

ADDRESS	ADD 3		ADD 2	
	0	1	0	1
B0	X		X	
B4	X			X
B8		X	X	
BC		X		X

The TRMDEF line for an AM-310 using the AX310.IDV is the same as for the AM310.IDV.

```
TRMDEF TERM1,AX310=0:37716,SOROC,100,100,100
```

SECTION III
PROGRAMMING

3.0 INTRODUCTION.

This section describes the programming requirements for the AM-310 circuit board. Circuit board addressing, command and status formats are described for complete system compatibility.

3.1 ADDRESSING.

The AM-310 and associated peripherals are addressed through the S-100 Bus address lines. The standard address is B0-B1 (Hex) and this address is etched in the I/O address block. This address can be changed by cutting the etch on the component side of the board and adding jumpers to generate the desired address as described in Section II.

3.2 I/O PORTS.

Two I/O ports are used by the AM-310 Communications Controller. These ports are the Command/Status register and the Data Read/Write register. The CPU issues commands to the Command/Status register and reads status outputs at the same address. Commands are given by the handshake procedure as described in Paragraph 3.4. Transmission data is read and written via the Data Read/Write register. All operations on Data Read/Write registers must be set up by commands in the Command/Status register.

3.3 AM-310 COMMANDS.

The AM-310 has twelve basic commands that control operation of the board. These commands, with their associated command numbers, are as follows:

1. Initialize
2. Transmit
3. Generate Transmit Interrupt
4. Poll RS-232 Port Status
5. Force Input Buffer Read
6. Enter Self Test Mode

7. Interrupt Response
8. Set DTR On
9. Set DTR Off
10. Set Download Address
11. Download
12. Call Download Subroutine

3.4 COMMAND SUMMARY

The set of twelve commands available to the AM-310 user are designed to require a minimum of overhead time from the host processor, leaving the semantics of the various commands to be executed by the on-board Z80 microprocessor.

3.4.1 COMMAND 1: INITIALIZE

This command is used to initialize one RS-232 port of the AM-310 and requires that a number of parameters be passed to the board. The parameters required and the order in which they are to be written is as follows:

INTERRUPT DEFINITION BYTE
PCI MODE REGISTER ONE
PCI MODE REGISTER TWO
PCI SYN REGISTER ONE
PCI SYN REGISTER TWO
PCI DLE REGISTER

The interrupt definition byte is available to program the AM-310 for host processor interrupt generation. If bit zero is set, the AM-310 will generate an interrupt when each character is received. If bit zero is reset, the AM-310 will continue to buffer the incoming data stream without interrupting the host processor. In a similar manner, if bit one is set, the host processor will be interrupted when the transmit data buffer is emptied and upon a transmit interrupt request. No interrupt will be generated from this source if bit one is reset.

Bit two is used to enable generation of an interrupt to a change in DCD in a manner similar to the two previous examples. Standard configuration is to allow all interrupts, and as such, the standard interrupt definition byte is 00000111. The next two bytes are used to program for the selected RS-232 port. As these parameters are passed on directly to the PCI, the user must determine the correct values by consulting the PCI device description attached. The final two parameters specify the PCI SYN register values. If the first parameter is zero, the SYN registers will not be programmed, as in the case of asynchronous devices, and the second parameter may be deleted. If non-zero, the data will be programmed, in order, into SYN registers one and two. After the parameters have been written, the EXECUTE command must be issued.

3.4.2 COMMAND 2: TRANSMIT

Data may be transmitted in blocks of 1 to 256 characters. First, the transmit command is issued. When the status port reads :FF the byte count is written into the data port, followed by the data to be transmitted. The handshake is then completed. A count of zero is used to specify 256 bytes.

3.4.3 COMMAND 3: GENERATE TRANSMIT INTERRUPT

In systems such as the Alpha Micro, transmissions are most easily initiated by setting up an output buffer which is emptied by an interrupt service routine. Command 3 is issued to generate the first interrupt for a given RS-232 port; normal transmit interrupt service routines are used from then on to empty the buffer.

3.4.4 COMMAND 4: POLL RS-232 PORT STATUS

By programming an interrupt definition byte of zero, the AM-310 may be set up in such a way that it will in no case interrupt the processor due to incoming characters. The user must then be able to determine the number of characters in the input buffer on demand. Non-interrupt based schemes require that both input and output buffer's status be read. This data is provided via the POLL RS-232 PORT STATUS command.

When the status register is cleared, the following data may be read from the data port (after reading the first byte, which is meaningless);

INPUT CHARACTER COUNT
OUTPUT CHARACTER COUNT
DSR-DCD STATUS

If DSR is valid, bit 7 of byte 3 is set. If DCD is valid, bit 6 of byte 3 is set.

3.4.5 COMMAND 5: FORCE INPUT BUFFER READ

In non-interrupt systems (or in certain instances in interrupt based setups) the user will determine that input data is to be read though no interrupt has been generated. This is accomplished via the FORCE READ command. When the status register has been cleared, the input character count and input data will be available at the data port as in the normal interrupt read.

3.4.6 COMMAND 6: ENTER SELF TEST MODE

With proper interfacing, the AM-310 self test will check the on-board RAM for any errors and insure the command status port is functioning correctly. First, the SELF TEST command is issued. The test program will wait for the status register to be cleared, at which point the failure address can be read, low byte first, from the data port. If the RAM has no errors, the failure address will be zero; otherwise, the address will be in the range of hex 4000-47FF. After reading the failure address, the program must write hex AA into the command port. When hex AA is read at the status port, write hex 55 into the command port and read hex 55. When 55 has been read, the command port shall be cleared. When the status register is cleared, the board has been returned to the POWER ON RESET state and must be completely re-initialized to allow further communication.

3.4.7 COMMAND 7: INTERRUPT RESPONSE

When the host processor receives an interrupt generated by the AM-310 board, it is necessary to find out the source of the interrupt. This can be ascertained by issuing command 7 to the AM-310. Note that this command is to be issued without regard to the RS-232 port number contained in the upper nibble, which may be zero for convenience. The interrupt response command will cause the AM-310 to reply with one or two possible formats. If the board being polled has not generated an interrupt, the board will reply with :FF status and command handshaking should be terminated. If the AM-310 has indeed generated the interrupt, the reply byte will have bit 7 reset, and the remaining bits will reflect the interrupt status. For an explicit definition of these bits, refer to the appended programming information.

3.4.8 COMMAND 8: SET DTR ON

Upon initialization, DTR is set to the active condition. If the user wishes to modify the status of DTR, he may set it on or off via commands 8 and 9. The status has been set upon completion of the handshaking sequence.

3.4.9 COMMAND 9: SET DTR OFF

See above.

3.4.10 COMMAND 10: SET DOWNLOAD ADDRESS

In some cases, the user may wish to have some specialized code executed by the Z80 on-board the AM-310. This may be accomplished through commands 10, 11, and 12. Command 10 is used to set the address to which the download code is to be written. This must, of course, be in the range of on-board RAM, addressed at :4000-47FF. The download code is to be in the form of a subroutine, and it is CALLED by the executive on command of the user. To avoid overlaying key status bytes for the executive, no download address should proceed :4030. The address is to be written low byte first.

3.4.11 COMMAND 11: DOWNLOAD

This commands the Z80 to set up the DMA chip for a write to the address provided via command 10. Data is to be written within the handshake procedure.

3.4.12 COMMAND 12: CALL DOWNLOAD SUBROUTINE

Upon receipt of this command, the handshaking sequence will be completed, then the downloaded subroutine will be called at the address specified in command 10.

3.5 OPERATING ROUTINE REQUIREMENTS.

The AM-310 has two ports available for access by the CPU. The first is the Command/Status port at the base address of the board. The second port is the Data Write/Read port at base address plus one. The following paragraphs describe the routines necessary for specific operations.

These commands are invoked by a simple handshake procedure. The command number is loaded into the right nibble of the command byte, and the RS-232 port number is loaded into the left nibble. FORMAT:

00ccxxxx

in which cc represents RS-232 port number and xxx is the command number. The handshake to the AM-310 is performed as follows:

WRITE to	:B0	00100011	(:23)
READ from	:B0	11111111	(:FF)
WRITE to	:B0	00000000	(:00)
READ from	:B0	00000000	(:00)

In this example, the GENERATE TRANSMIT INTERRUPT command has been issued properly. In the handshake procedure, the host processor must wait to read :FF from the status port before clearing the command register. The host must also wait to receive :00 from the status port, as various processing routines set up data and are not valid until the status register is cleared.

3.5.1 BOARD INITIALIZATION.

On system reset, the command register must be zeroed; the status register will be cleared and processing may continue.

3.5.2 DATA RS-232 PORT INITIALIZATION.

The following sequence of commands are used to initialize a communications RS-232 port.

OPERATION	ADDRESS	DATA	
WRITE	:B0	00cc0001	:c1 [INITIALIZE]
READ	:B0	11111111	:FF
WRITE	:B1	00000111	:07 INT DEFINITION
* WRITE	:B1	11001010	:CA PCI MR 1
** WRITE	:B1	00111111	:3F PCI MR 2
*** WRITE	:B1	xxxxxxxx	:xx SYN REG 1
**** WRITE	:B1	yyyyyyyy	:yy SYN REG 2
***** WRITE	:B1	zzzzzzzz	:zz DLE REG
WRITE	:B0	00000000	:00
READ	:B0	00000000	:00

* Write to PCI mode register one: Asynchronous terminals

** Write to PCI mode register two: Asynchronous terminals at 19,200 baud

*** Syn register one character (zero for asynchronous communication)

**** Syn register two character (not necessary if SYN 1 is zero)

***** DLE register character

cc Enter the RS-232 port number 00-11 (0-3) you wish to initialize

3.5.3 TRANSMITTING DATA.

TYPE	ADDRESS	DATA
WRITE	:B0	00cc0010 :c2 [TRANSMIT]
READ	:B0	11111111 :FF
WRITE	:B1	WORD COUNT
WRITE	:B1	TRANSMIT DATA
WRITE	:B0	00000000 :00
READ	:B0	00000000 :00

The AM-310 will generate an interrupt when all characters have been transmitted. A count of zero is used to specify 256 bytes.

3.5.4 FORCE AN AM-100 TRANSMIT INTERRUPT.

TYPE	ADDRESS	DATA
WRITE	:B0	00cc0011 :c3 [FORCE INTERRUPT]
READ	:B0	11111111 :FF
WRITE	:B0	00000000 :00
READ	:B0	00000000 :00

This will cause the AM-310 to generate a transmit interrupt for the AM-100 immediately upon conclusion; of the command sequence.

3.5.5 POLL TRANSMISSION STATUS.

TYPE	ADDRESS	DATA
WRITE	:B0	00cc0100 :c4 [POLL STATUS]
READ	:B0	11111111 :FF
READ	:B1	unspecified
READ	:B1	INPUT CHARACTER COUNT
READ	:B1	OUTPUT CHARACTER COUNT
READ	:B1	RS-232 PORT ON-LINE STATUS
WRITE	:B0	00000000 :00
READ	:B0	00000000 :00

The RS-232 port on-line status byte reflects the state of the RS-232C signals DSR and DCD. If bit 7 is set, DSR is valid. If bit 6 is set, DCD is valid.

3.5.6 FORCE THE AM-310 TO SET UP FOR RECEIVE TRANSFER TO AM-100.

TYPE	ADDRESS	DATA
WRITE	:B0	00cc0101 :c5 [FORCE READ]
READ	:B0	11111111 :FF
READ	:B1	unspecified
READ	:B1	INPUT CHARACTER COUNT
READ	:B1	RECEIVE DATA
WRITE	:B0	00000000 :00
READ	:B0	00000000 :00

3.5.7 DIAGNOSTIC OPERATION.

The AM-310 has been provided with full on-board diagnostic capabilities to help analyze partially functional boards. These diagnostics do a full RAM test and test for correct operation of the commands register. The sequence of commands requires little from host processor; most of the commands are used to invoke Z80 action on the AM-310. The following sequence is to be used to test the board:

TYPE	ADDRESS	DATA	
WRITE	:B0	00000110	:06 [TEST MODE]
READ	:B0	11111111	:FF
WRITE	:B0	00000000	:00
READ	:B0	00000000	:00
READ	:B1	unspecified	
READ	:B1	FAILURE ADDRESS LOW	
READ	:B1	FAILURE ADDRESS HIGH	
WRITE	:B0	10101010	:AA
READ	:B0	10101010	:AA
WRITE	:B0	01010101	:55
READ	:B0	01010101	:55
WRITE	:B0	00000000	:00
READ	:B0	00000000	:00

At this point the board will be reset and must be reinitialized.

3.5.8 RECEIVING DATA.

The communications board will accept and buffer incoming data. Buffering is 128 bytes per buffer, two buffers per communication channel. The host processor will be interrupted when the board has received a block of data. This block is terminated upon reception of any character although, if interrupt response time is lengthy, any further characters will be buffered, to a maximum of 128 characters.

TYPE	ADDRESS	DATA
WRITE	:B0	00000111 :07 [INTERRUPT POLL]
READ	:B0	00010cc0 :0c *
READ	:B1	unspecified
READ	:B1	INPUT CHARACTER COUNT
READ	:B1	RECEIVE DATA
WRITE	:B0	000000000 :00
READ	:B0	000000000 :00

* If the AM-310 board has no interrupt pending, response will be :FF to the :07 interrupt poll command. This will be useful in multiple AM-310 board applications in which all AM-310 boards share the same interrupt vector.

3.5.9 PROCESSING TRANSMITTER INTERRUPTS.

The AM-310 will interrupt the host processor after all characters in a given block have been transmitted. The following sequence of events shall occur to process the interrupt:

TYPE	ADDRESS	DATA
WRITE	:B0	00000111 :07 [INTERRUPT POLL]
READ	:B0	00100cc0 ...
WRITE	:B0	00000000 :00
READ	:B0	00000000 :00

3.5.10 INTERRUPTS GENERATED BY A CHANGE IN RS-232 PORT ON-LINE STATUS.

If the device connected to one of the AM-310 ports is a modem or other type device which supplies the RS-232 signals DCD and DSR, these lines will reflect the on-line status of the device. This status is driven valid on the AM-310 for devices which do not provide these signals. Should a device drive the DCD signal invalid (if a user hangs up on the modem), this change will be reported to the host processor in the following manner: (First the host processor will be interrupted, as in the case of a transmit or receive interrupt, and polling will take place as normal.)

TYPE	ADDRESS	DATA	
WRITE	:B0	00000111	:07 [INTERRUPT POLL]
READ	:B0	01s00cc0	...
WRITE	:B0	00000000	:00
READ	:B0	00000000	:00

s: New status of RS-232 port: 0 = off line 1 = on-line

cc: RS-232 port which this applies

3.5.11 SET DTR ON.

TYPE	ADDRESS	DATA	
WRITE	:B0	00cc1000	:c8
READ	:B0	11111111	:FF
WRITE	:B0	00000000	:00
READ	:B0	00000000	:00

3.5.12 SET DTR OFF.

TYPE	ADDRESS	DATA
WRITE	:B0	00cc1001 :c9
READ	:B0	11111111 :FF
WRITE	:B0	00000000 :00
READ	:B0	00000000 :00

3.5.13 SET DOWNLOAD ADDRESS.

TYPE	ADDRESS	DATA
WRITE	:B0	00001010 :0A
READ	:B0	11111111 :FF
WRITE	:B1	DOWNLOAD ADDRESS [LO BYTE]
WRITE	:B1	DOWNLOAD ADDRESS [HI BYTE]
WRITE	:B0	00000000 :00
READ	:B0	00000000 :00

3.5.14 DOWNLOAD.

TYPE	ADDRESS	DATA
WRITE	:B0	00001011 :0B
READ	:B0	11111111 :FF
WRITE	:B1	DOWNLOAD PROGRAM
WRITE	:B0	00000000 :00
READ	:B0	00000000 :00

3.5.15 CALL DOWNLOAD SUBROUTINE.

TYPE	ADDRESS	DATA
WRITE	:B0	00001100 :0C
READ	:B0	11111111 :FF
WRITE	:B0	00000000 :00
READ	:B0	00000000 :00

4.0 INTRODUCTION.

The AM-310 Communications Controller circuit board contains integrated circuit elements for the data processing necessary for the performance of the functions described in Sections I, II and III of this manual. This section describes the functional theory of operation of the circuit board and also provides information for each of the integrated circuit elements.

4.1 CIRCUIT BOARD OPERATION.

This circuit board provides communications interface between RS-232C interface devices and the S-100 Bus system. The functional block diagram of the circuit board is shown in Figure 4-1. The circuit board schematic, parts list, and component cross-reference list are contained in Section VI of this manual. Table 4-1 contains a list of the signals used in this circuit board with descriptions of their functions. For S-100 bus signals, see Table 4-2; for RS-232 interface signals, see Table 4-3.

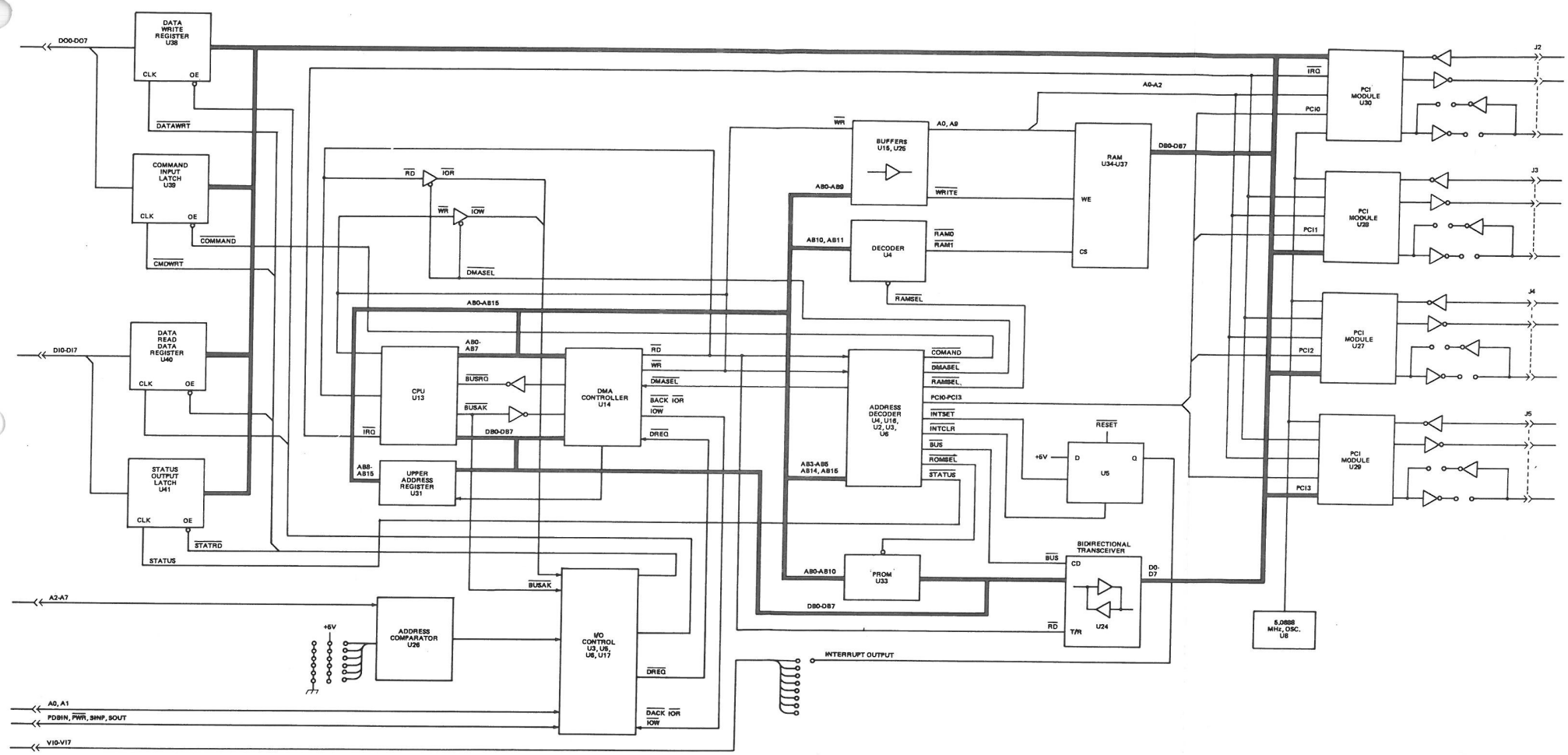


Figure 4-1. AM-310 Functional Block Diagram

Table 4-1. AM-310 Signal Descriptions

SIGNAL	NAME	SCHEM PAGE OF SOURCE	DESCRIPTION
AB0-AB15	Address Bus	1	16 bit tri-state address bus. Provides addressing for memory and data exchanges.
BRCLK	Baud Rate Generator Clock	2	5.0688 MHz clock that drives the PCI modules.
$\overline{\text{BUS}}$	Bus Control	1	Controls the bi-directional transceiver and, when asserted, puts the device in a tri-state condition.
$\overline{\text{BUSAK}}$	Bus Acknowledge	1	CPU module output indicating that the CPU module address bus, data bus, and tri-state control bus signals have been set to their high impedance state.
CLK	Clock	1	Output of the 4 MHz oscillator to supply the CPU module and the DMA controller.

Table 4-1 (Con't.). AM-310 Signal Descriptions

SIGNAL	NAME	SCHEM PAGE OF SOURCE	DESCRIPTION
$\overline{\text{CMDWRT}}$	Command Write	1	Output of the decoder in the I/O control logic to clock the command data from S-100 Bus data output lines into the Command Input Latch.
$\overline{\text{COMMAND}}$	Command	1	Output of the Address decoder that transfers the command data from the Command Input Latch to the internal data bus.
$\overline{\text{DACK}}$	DMA Acknowledge	1	DMA cycle output from the DMA controller to generate, with IOR, a signal to transfer the data from the Data Write Register to the internal data bus.
$\overline{\text{DATARD}}$	Data Read Enable	1	Output of the decoder in the I/O control logic to generate DREQ and transfer the contents of the Data Read Register to the S-100 Bus data input lines.

Table 4-1 (Con't.). AM-310 Signal Descriptions

SIGNAL	NAME	SCHEM PAGE OF SOURCE	DESCRIPTION
$\overline{\text{DATAWRT}}$	Data Write Enable	1	Output of the I/O +DREQ control logic to clear DREQ and enable data from the S-100 Bus to AM-310.
$\overline{\text{DATRD}}$	Data Read Clock	1	Output of the I/O +DREQ control logic to clear DREQ and clock data from the AM-310 to the Data Read register.
$\overline{\text{DMAOP}}$	DMA Operation	1	Clocks the DREQ flip-flop to assert DREQ and generate $\overline{\text{BUS}}$.
$\overline{\text{DMASEL}}$	DMA Select	1	Output of the Address Decoder during a read or write cycle that selects DMA operations.
DREQ	Data Request	1	Asserted during data read or write cycles.
D0-D7	Data Bus	1	Internal 8-bit tri-state data bus.
$\overline{\text{INTCLR}}$	Interrupt Clear	1	Output of the address decoder to clear the selected interrupt.

Table 4-1 (Con't.). AM-310 Signal Descriptions

SIGNAL	NAME	SCHEM PAGE OF SOURCE	DESCRIPTION
$\overline{\text{INTSET}}$	Interrupt Set	1	Output of the address decoder to set the selected interrupt.
$\overline{\text{IOR}}$	I/O Read	1	Bi-directional tri-state line to read the internal control registers or to access data during a DMA transfer.
$\overline{\text{IOW}}$	I/O Write	1	Bi-directional tri-state line to load information into the DMA module or to load data during a DMA transfer.
$\overline{\text{PCI0}}$-$\overline{\text{PCI3}}$	PCI Select 0-3	1	Output of the address decoder to enable the selected PCI module.
$\overline{\text{RAMSEL}}$	RAM Select	1	Output of the Address Decoder during a read or write cycle that enables the RAM select decoder U4.
$\overline{\text{RAM0}}$ $\overline{\text{RAM1}}$	RAM Select 0 RAM Select 1	2	Selects RAM U34, U37 or RAM U35, U36 from internal address bits AB10, AB11.

Table 4-1 (Con't.). AM-310 Signal Descriptions

SIGNAL	NAME	SCHEM PAGE OF SOURCE	DESCRIPTION
$\overline{RD}$	Read Select	1	Output of the DMA module to select a DMA read operation.
$\overline{RESET}$	Reset	2	Reset signal driven from S-100 Bus $\overline{PRESET}$.
$\overline{ROMSEL}$	ROM Select	1	Output of the address decoder during a read or write cycle that enables the PROM U33.
$\overline{STATRD}$	Status Read	1	Output of the decoder in the I/O control logic to transfer the contents of the Status Output Latch to the S-100 Bus data input lines.
$\overline{STATUS}$	Status Transfer	1	Output of the address decoder that clocks the status data from the internal data bus into the Status Output Latch.
$\overline{WR}$	Write Select	1	Output of the DMA module to select a DMA write operation.
$\overline{WRITE}$	Write	2	Signal $\overline{WR}$ through a buffer.

Table 4-2. S-100 Bus Interface Signals Descriptions

SIGNAL	NAME	J1 PIN NO.	DESCRIPTION
AD0	Address 0	79	Eight bits of tri-state addressing.
AD1	Address 1	80	
AD2	Address 2	81	
AD3	Address 3	31	
AD4	Address 4	30	
AD5	Address 5	29	
AD6	Address 6	82	
AD7	Address 7	83	
DI0	Data In Bus Bit 0	95	Data input port. Eight bit tri-state data bus from bus slave to bus master.
DI1	Data In Bus Bit 1	94	
DI2	Data In Bus Bit 2	41	
DI3	Data In Bus Bit 3	42	
DI4	Data In Bus Bit 4	91	
DI5	Data In Bus Bit 5	92	
DI6	Data In Bus Bit 6	93	
DI7	Data In Bus Bit 7	43	

Table 4-2 (Con't.). S-100 Bus Interface Signals Description

SIGNAL	NAME	J1 PIN NO.	DESCRIPTION
D00	Data Out Bus Bit 0	36	Data output port. Eight bit tri-state data bus from bus master to bus slave.
D01	Data Out Bus Bit 1	35	
D02	Data Out Bus Bit 2	88	
D03	Data Out Bus Bit 3	89	
D04	Data Out Bus Bit 4	38	
D05	Data Out Bus Bit 5	39	
D06	Data Out Bus Bit 6	40	
D07	Data Out Bus Bit 7	90	
PDBIN	Data Bus Input Command	8	Read enable. Used by bus master to request addressed slave to place data on the data bus.
$\overline{\text{PRESET}}$	Preset	75	Reset signal normally originating from front panel reset pushbutton.

Table 4-2 (Con't.). S-100 Bus Interface Signals Description

SIGNAL	NAME	J1 PIN NO.	DESCRIPTION
$\overline{\text{PWR}}$	Write Strobe	77	When asserted, is a command from the bus master for the addressed slave to accept the data on the data bus.
SINP	I/O Input Cycle	46	When asserted, indicates that the current bus cycle is a bus master input from an I/O address.
SOUT	I/O Output Cycle	45	When asserted, indicates that the current bus cycle is a bus master output to an I/O address.

Table 4-2 (Con't.). S-100 Bus Interface Signals Description

SIGNAL	NAME	J1 PIN NO.	DESCRIPTION
$\overline{VI0}$	Vectored Interrupt 0	4	Vectored interrupt lines for both interrupt requests and DMA results.
$\overline{VI1}$	Vectored Interrupt 1	5	
$\overline{VI2}$	Vectored Interrupt 2	6	
$\overline{VI3}$	Vectored Interrupt 3	7	
$\overline{VI4}$	Vectored Interrupt 4	8	
$\overline{VI5}$	Vectored Interrupt 5	9	
$\overline{VI6}$	Vectored Interrupt 6	10	
$\overline{VI7}$	Vectored Interrupt 7	11	
+16V	Power and Ground	2	Circuit board power and ground.
-16V		52	
+ 8V		1, 51	
GND		50, 100	

Table 4-3. RS-232 Interface Signal Descriptions

SIGNAL	NAME	PIN	INPUT/ OUTPUT	DESCRIPTION
CTS	Clear to Send	4	I	Must be high in order for the transmitter to operate.
DCD	Data Carrier Detect	8	I	Must be high in order for the receiver to operate.
DSR	Data Set Ready	20	I	Indicates the status of the terminal.
DTR	Data Terminal Ready	6	O	AM-310 output to indicate data terminal ready.
RTS	Request to Send	5	O	AM-310 output to indicate request to send. High for every character transmission.
RXC	Receiver Clock	17	I/O	If external receiver clock is programmed, this input controls the rate at which the character is to be received. If internal receiver clock is programmed, this becomes an output at 1X the baud rate.

Table 4-3 (Con't.). RS-232 Interface Signal Descriptions

SIGNAL	NAME	PIN	INPUT/ OUTPUT	DESCRIPTION
$\overline{\text{RXD}}$	Serial Input Data	2	I	Serial input data to the receiver. Mark is low, space is high.
TXC	Transmitter Clock	15	I/O	If external transmitter clock is programmed, this input controls the rate at which the character is transmitted. If internal transmitter clock is programmed, this becomes an output at 1X the programmed baud rate.
$\overline{\text{TXD}}$	Serial Output Data	3	O	Serial output data from the transmitter. Mark is low, space is high.
GND	Ground	1,7		Data transmission system ground. (Signal ground.)

4.1.1 ADDRESSING.

Address data is received from the S-100 bus on address lines AD0-AD7 for direct addressing of the circuit board. Lines AD2-AD7 are wired directly to comparator U26. The other inputs to this comparator come from circuit board etch or jumper wires to either +5V or ground to produce the desired address. The output of U26 is asserted when the data from the address lines compare with the address of the AM-310. This enables decoder U17 in the I/O control logic to generate read and write signals. The address in etch is B0 (Hex) and other addresses can be selected as described in Section II.

4.1.2 SYSTEM CPU OUTPUT.

System CPU output data is transmitted to the AM-310 for circuit board control and data transfer to peripherals by the S-100 bus data and control lines. Data is received by the AM-310 by the Data Write register U38 and commands are received by the Command Input Latch U39. The input to these registers comes from the S-100 bus data lines D00-D07; and the output, when enabled, goes to internal data bus D0-D7.

4.1.2.1 COMMAND DATA.

Data from the S-100 data lines is clocked into the Command Input Latch U39 by Command Write signal $\overline{\text{CMDWRT}}$ from the decoder in the I/O control logic. This takes place when system address lines AD0 and AD1 are zero and I/O output signal SOUT from the S-100 bus is asserted.

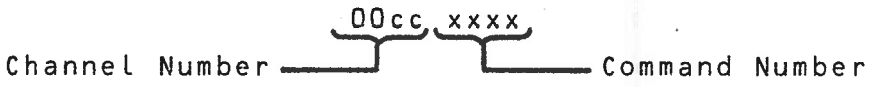
When the CPU microprocessor issues a write command ($\overline{\text{WR}}$) and address lines AB14 and AB15 from the internal address bus are both one, decoder U16 is enabled. Internal address lines AB3-AB5 generate the $\overline{\text{COMMAND}}$ signal when AB3 is one and AB4 and AB5 are both zero. This transfers the contents of the Command Input Latch to the internal data bus.

4.1.2.2 COMMAND TYPES.

The AM-310 has twelve commands that control operation of the board. These commands, with their associated command numbers, are as follows:

Command 1: INITIALIZE
Command 2: TRANSMIT
Command 3: GENERATE TRANSMIT INTERRUPT
Command 4: POLL CHANNEL STATUS
Command 5: FORCE INPUT BUFFER READ
Command 6: ENTER SELF TEST MODE
Command 7: INTERRUPT RESPONSE
Command 8: SET DTR ON
Command 9: SET DTR OFF
Command 10: SET DOWNLOAD ADDRESS
Command 11: DOWNLOAD
Command 12: CALL DOWNLOAD SUBROUTINE

A detailed description of these commands and their function is contained in Paragraph 3.3. The command format is shown below:



4.1.2.3 OUTPUT DATA.

The output data from the CPU to the AM-310 transfers from the S-100 bus data lines to the internal data bus through Data Write Register U38. The input to the Data Write Register comes from the S-100 bus data lines D00-D07; and the output, when enabled, goes to the internal AM-310 data bus D0-D7.

Data from the S-100 bus data lines is clocked into the Data Write Register by signal $\overline{\text{DATAWRT}}$. This is generated by the decoder U17 in the I/O +DREQ control logic when the board is addressed and address lines AD0=1 and AD1=0 and I/O output signal SOUT from the S-100 bus are asserted. A low on $\overline{\text{IOR}}$ from the CPU module and a low on $\overline{\text{DACK}}$ from the DMA controller transfer the data to the internal data bus.

4.1.3 CPU INPUT.

CPU input data and status information are transmitted from the AM-310 by the S-100 bus data and control lines. It is transmitted from the AM-310 by the Read Data Register U40 and Status Output Latch U41.

4.1.3.1 STATUS DATA.

The input to the status output latch comes from the internal data bus D0-D7 and the output, when enabled, goes to the S-100 bus data lines DI0-DI7. The status output latch always contains 00 or FF (Hex) except for interrupt response.

Data from the internal data bus is clocked into the Status Output Latch by signal $\overline{\text{STATUS}}$ from the address decoder U16. This occurs when $\overline{\text{RD}}$ output from the CPU module is asserted and internal address bits AB3-AB5 are zero and AB14 and AB15 are both one. The contents of the Status Output Latch are then transferred to the internal data bus by signal $\overline{\text{STATRD}}$ from the decoder in the I/O control logic. This signal is asserted when AD0=0, PDBIN is true (high) and $\overline{\text{PWR}}$ is false (high).

4.1.3.2 INPUT DATA.

The Data Read Register (U40) controls the transfer of data from the internal data bus to the S-100 bus data lines. The inputs to this register come from internal data bus D0-D7 and the output, when enabled, goes to the S-100 bus data lines DI0-DI7.

Data is clocked into the register when $\overline{\text{IOW}}$ and $\overline{\text{BUSAK}}$ from the CPU module are both asserted. Data is placed on the S-100 bus data lines by $\overline{\text{DATARD}}$ from the decoder U17 in the I/O +DREQ control logic when the board is addressed and AD0=1, AD1=0, Read Enable PDBIN is true (high) and write strobe $\overline{\text{PWR}}$ is false (high).

4.1.4 DATA TRANSFERS.

Data transfers between the system CPU and the AM-310 are accomplished by DMA transfers with the AM-310 CPU module memory. The AM-310 CPU module formats and directs the data through the four PCI modules for data transfers between the AM-310 and the associated peripheral devices.

Commands are issued to the AM-310 CPU module by a handshaking process. A non-zero command is written into the Command Input Latch (U39) by the system CPU. The AM-310 CPU module recognizes this command and writes :FF into the Status Output Latch (U41). The system CPU zeros the Command Input Latch and waits for a zero response in the Status Output Latch. This completes the cycle and another command may be issued or response data read.

All commands that set up the AM-310 for a DMA read, including the receive interrupt response, have the DMA chip (U14) set up for the transfer when the status register is zeroed. In all cases, the first read of the data register is an invalid byte. The SECOND byte read is the first valid data byte of the block to be transferred.

Once the DMA chip has been programmed by the AM-310 CPU module, transfers take place as follows: A read or write to the data register sets DREQ through U6 and U5 while reading or writing the data register. The DMA chip then acquires the internal bus as soon as the AM-310 CPU module relinquishes control. The DMA write and read cycle differs from this point.

The DMA write cycle generates a low on IOR with the low present on $\overline{\text{DACK}}$, giving a low on U3B-11 to present the data byte to the bus. The byte is written in the RAM under control of the DMA module.

With a DMA read, the system CPU will have read the value in U40. If this was the first DMA read for a given block of data, this byte is meaningless and should be ignored. As the DMA module responds to DREQ, it accesses the memory and pulls $\overline{IOW}$ low. BUSAK is low at this point, causing the byte from memory to be written into U40. This byte is available to the AM-100 on the next data register read cycle.

Either DMA request is terminated as U6 clocks a low into U5, pulling DREQ low and terminating the transfer.

The AM-310 CPU module is interrupt driven in this application. All commands to the AM-310 CPU module from the AM-100 are buffered; if the command calls for transmit, the AM-310 CPU module sets up the PCI on that channel and starts the transmit by causing an initial interrupt. Each interrupt requests another character, which is given to it by the AM-310 CPU module interrupt request processing routine until the transmit buffer is emptied. The AM-310 CPU module will set RTS true with every character transmitted. The transmitter is then disabled.

Receiving characters are also interrupt driven. If both DSR and DCD are valid, the receiver is enabled. On receipt of an input character, the AM-310 CPU module reads the character, enters it into the current buffer (each channel has two 96 byte input buffers) and sets a flag to the executive routine. The non-interrupt based executive determines if the buffer has been filled, the end being indicated by receipt of the number of characters specified in the block character count.

If this condition is satisfied, the AM-310 CPU module then switches input buffers, allowing the system CPU to read the current buffer and the AM-310 CPU module to enter incoming characters into the new buffer.

The character count is programmed during channel initialization and may be altered at any time. If the character count is non-zero, buffering will stop and data will be transferred when the input character count is met.

These transfers generate an interrupt to the system CPU. If the user wishes to have the AM-310 operate on a non-interrupt driven basis, the character count should be set to zero. Buffering will continue indefinitely and the status of each input buffer may be checked with the POLL command. When the desired byte count has been met, the characters may be transferred with the FORCE READ command.

4.2 CIRCUIT MODULE DESCRIPTION.

This section describes the operation of the individual circuit packages (DIPS) contained on the AM-310 circuit board. Most of the data processing is handled by the AM-310 CPU and DMA controllers and the programmable communications interface modules, so these are described in detail. The control logic and interface modules are also described with logic and connection diagrams for each one.

4.2.1 CPU MICROPROCESSOR (U13).

The CPU microprocessor is a single DIP module that handles the data processing of the AM-310 circuit board.

Figure 4-2 is a block diagram of the CPU, and Figure 4-3 details the internal register configuration which contains 208 bits of Read/ Write memory that are accessible to the programmer. The registers include two sets of six general purpose registers that may be used individually as 8-bit registers or as 16-bit register pairs. There are also two sets of accumulator and flag registers. The programmer has access to either set of main or alternate registers through a group of exchange instructions. This alternate set allows foreground/background mode of operation or may be reserved for very fast interrupt response. Each CPU also contains a 16-bit stack pointer which permits simple implementation of multiple level interrupts, unlimited subroutine nesting and simplification of many types of data handling.

The two 16-bit index registers allow tabular data manipulation and easy implementation of relocatable code. The Refresh register provides for automatic, totally transparent refresh of external dynamic memories. The I register is used in a powerful interrupt response mode to form the upper 8 bits of a pointer to an interrupt service address table, while the interrupting device supplies the lower 8 bits of the pointer. An indirect call is then made to this service address.

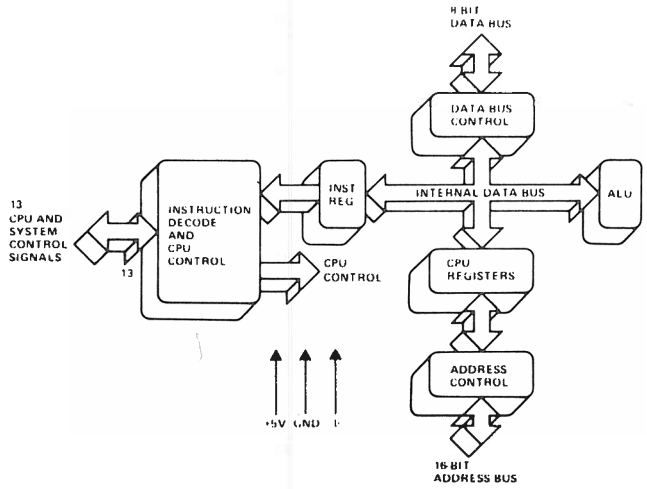


Figure 4-2. CPU Block Diagram

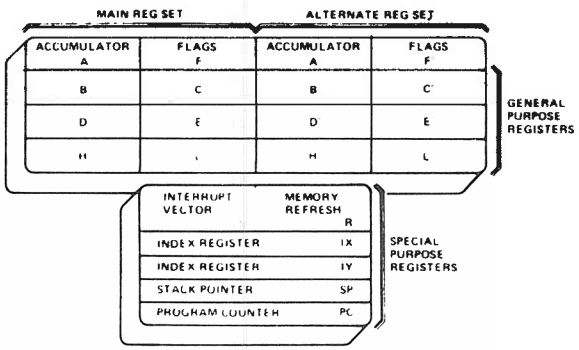
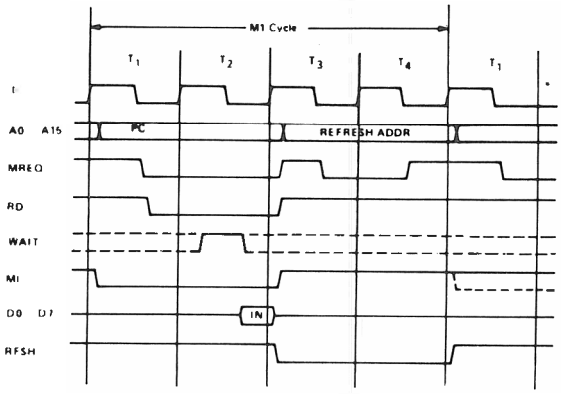


Figure 4-3. CPU Registers

4.2.1.1 INSTRUCTION OP-CODE FETCH.

The program counter content (PC) is placed on the address bus immediately at the start of the cycle. One-half clock time later $\overline{\text{MREQ}}$ goes active. The falling edge of $\overline{\text{MREQ}}$ can be used directly as a chip enable to dynamic memories. $\overline{\text{RD}}$, when active, indicates that the memory data should be enabled onto the CPU data bus. The CPU samples data with the rising edge of the clock state T3. Clock states T3 and T4 of a fetch cycle are used to refresh dynamic memories while the CPU is internally decoding and executing the instruction. The refresh control signal RFSH indicates that a refresh read of all dynamic memories should be accomplished.



4.2.1.2 MEMORY READ OR WRITE CYCLES.

Illustrated here is the timing of memory read or write cycles other than an OP code fetch (M1 cycle). The $\overline{MREQ}$ and $\overline{RD}$ signals are used exactly as in the fetch cycle. In the case of a memory write cycle, the $\overline{MREQ}$ also becomes active when the address bus is stable so that it can be used directly as a chip enable for dynamic memories. The $\overline{WR}$ line is active when data on the data bus is stable so that it can be used directly as an R/W pulse to virtually any type of semiconductor memory.

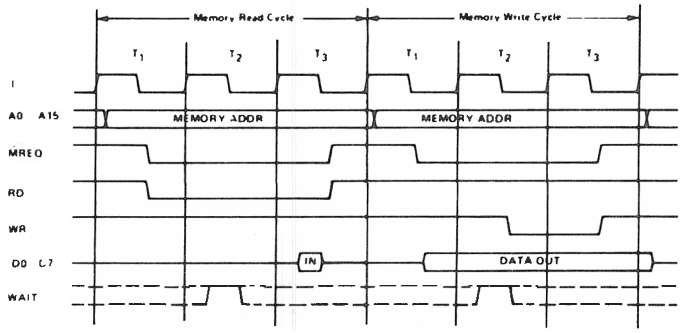


Figure 4-4 shows the CPU pin configuration and table 4-4 contains a list of CPU signals.

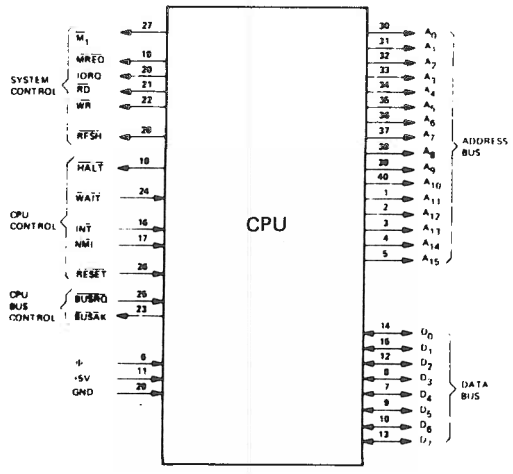


Figure 4-4. CPU Pin Configuration

Table 4-4. CPU Signal List

SIGNAL	PIN	FUNCTION
A0-A15 ADDRESS BUS		Tri-state output, active high. A0-A15 constitute a 16-bit address bus. The address bus provides the address for memory (up to 64k bytes) data exchanges and for I/O device data exchanges.
D0-D7 DATA BUS		Tri-state input/output, active high. D0-D7 constitute an 8-bit bi-directional data bus. The data bus is used for data exchanges and I/O devices.
$\overline{M1}$ MACHINE CYCLE ONE	27	Output, active low. $\overline{M1}$ indicates that the current machine cycle is the OP code fetch cycle of an instruction execution.
$\overline{MREQ}$ MEMORY REQUEST	19	Tri-state output, active low. The memory request signal indicates that the address bus holds a valid address for a memory read or memory write operation.

Table 4-4 (Con't.). CPU Signal List

SIGNAL	PIN	FUNCTION
$\overline{\text{IORQ}}$ INPUT/OUTPUT REQUEST	20	Tri-state output, active low. The $\overline{\text{IORQ}}$ signal indicates that the lower half of the address bus holds a valid I/O address for an I/O read or write operation. An IORQ signal is also generated when an interrupt is being acknowledged to indicate that an interrupt response vector can be placed on the data bus.
$\overline{\text{RD}}$ MEMORY READ	21	Tri-state output, active low. $\overline{\text{RD}}$ indicates that the CPU wants to read data from memory or an I/O device. The addressed I/O device or memory should use this signal to gate onto the CPU data bus.
$\overline{\text{WR}}$ MEMORY WRITE	22	Tri-state output, active low. $\overline{\text{WR}}$ indicates that the CPU data bus holds valid data to be stored in the addressed memory I/O device.

Table 4-4 (Con't.). CPU Signal List

SIGNAL	PIN	FUNCTION
$\overline{\text{RFSH}}$ REFRESH	28	Output, active low. $\overline{\text{RFSH}}$ indicates that the lower 7 bits of the address bus contain a refresh address for dynamic memories and the current $\overline{\text{MREQ}}$ signal should be used to do a refresh read to all dynamic memories.
$\overline{\text{HALT}}$ HALT STATE	18	Output, active low. $\overline{\text{HALT}}$ indicates that the CPU has executed a $\overline{\text{HALT}}$ software instruction and is awaiting either a nonmaskable or a maskable interrupt (with the mask enabled) before operation can resume. While halted, the CPU executes NOP's to maintain memory refresh activity.
$\overline{\text{WAIT}}$ WAIT	24	Input, active low. $\overline{\text{WAIT}}$ indicates to the CPU that the addressed memory or I/O devices are not ready for a data transfer. The CPU continues to enter wait states for as long as this signal is active.

Table 4-4 (Con't.). CPU Signal List

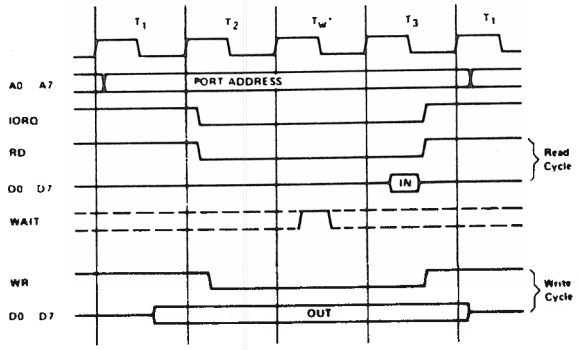
SIGNAL	PIN	FUNCTION
$\overline{\text{INT}}$ INTERRUPT REQUEST	16	Input, active low. The interrupt request signal is generated by I/O devices. A request will be honored at the end of the current instruction if the internal software controlled interrupt enabled flip-flop (IFF) is enabled.
$\overline{\text{NMI}}$ NON MASKABLE INTERRUPT	17	Input, active low. The nonmaskable interrupt request line has a higher priority than $\overline{\text{INT}}$ and is always recognized at the end of the current instruction, independent of the status of the interrupt enable flip-flop. $\overline{\text{NMI}}$ automatically forces the CPU to location 0066H.
$\overline{\text{RESET}}$	26	Input, active low. $\overline{\text{RESET}}$ initializes the CPU as follows: Reset interrupt enable flip-flop, clear PC and registers I and R and set interrupt I and R and set interrupt to 8080A mode. During reset time, the address and data bus go to a high impedance state and all control output signals go to the inactive state.

Table 4-4 (Con't.). CPU Signal List

SIGNAL	PIN	FUNCTION
$\overline{\text{BUSRQ}}$ BUS REQUEST	25	Input, active low. The bus request signal has a higher priority than $\overline{\text{NMI}}$ and is always recognized at the end of the current machine cycle and is used to request the CPU address bus, data bus and tri-state output control signals to go to a high impedance state so that other devices can control these busses.
$\overline{\text{BUSAK}}$ BUS ACKNOWLEDGE	23	Output, active low. Bus acknowledges are used to indicate to the requesting device that the CPU address bus, data bus and tri-state control bus signals have been set to their high impedance state and the external device can now control these signals.

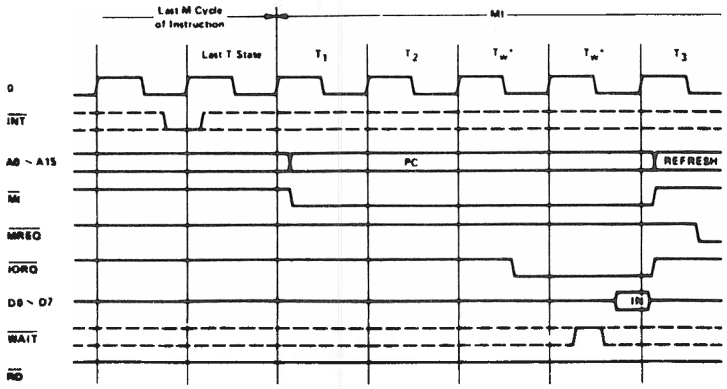
4.2.1.3 INPUT OR OUTPUT CYCLES.

Illustrated here is the timing for an I/O read or I/O write operation. Notice that during I/O operations a single wait state is automatically inserted (Tw*). The reason for this is that during I/O operations this extra state allows sufficient time for an I/O port to decode its address and activate the WAIT line if a wait is required.



4.2.1.4 INTERRUPT REQUEST/ACKNOWLEDGE CYCLE.

The interrupt signal is sampled by the CPU with the rising edge of the last clock at the end of any instruction. When an interrupt is accepted, a special M1 cycle is generated. During this M1 cycle, the IORQ signal becomes active (instead of MREQ) to indicate that the interrupting device can place an 8-bit vector on the data bus. Two wait states (Tw*) are automatically added to this cycle.



4.2.1.5 CPU INSTRUCTION SET

The following is a summary of the CPU instruction set showing the assembly language mnemonic and the symbolic operation performed by the instruction. The instructions are divided into the following categories:

- 8-bit Loads
- 16-bit Loads
- Exchanges
- Memory Block Moves
- Memory Block Searches
- 8-bit Arithmetic and Logic
- 16-bit Arithmetic
- General Purpose Accumulator & Flag Operations
- Miscellaneous Group
- Rotates and Shifts
- Bit Set, Reset and Test
- Input and Output
- Jumps
- Calls
- Restarts
- Returns

In Table 4-5 the following terminology is used:

b ≡ a bit number in any 8-bit register or memory location

cc ≡ flag condition code
- NZ ≡ non zero
- Z ≡ zero
- NC ≡ non carry
- C ≡ carry
- PO ≡ Parity odd or no over flow
- PE ≡ Parity even or over flow
- P ≡ Positive
- M ≡ Negative (minus)

d ≡ any 8-bit destination register or memory location

dd ≡ any 16-bit destination register or memory location

e ≡ 8-bit signed 2's complement displacement used in relative jumps and indexed addressing

L ≡ 8 special call locations in page zero. In decimal notation these are 0, 8, 16, 24, 32, 40, 48 and 56

n ≡ any 8-bit binary number

nn ≡ any 16-bit binary number

r ≡ any 8-bit general purpose register (A, B, C, D, E, H, or L)

s ≡ any 8-bit source register or memory location

s_b ≡ a bit in a specific 8-bit register or memory location

ss ≡ any 16-bit source register or memory location

subscript "L" ≡ the low order 8 bits of a 16-bit register

subscript "H" ≡ the high order 8 bits of a 16-bit register

( ) ≡ the contents within the ( ) are to be used as a pointer to a memory location or I/O port number

8-bit registers are A, B, C, D, E, H, L, I and R

16-bit register pairs are AF, BC, DE and HL

16-bit registers are SP, PC, IX and IY

Addressing Modes implemented include combinations of the following:

Immediate	Indexed
Immediate extended	Register
Modified Page Zero	Implied
Relative	Register Indirect
Extended	Bit

Table 4-5. CPU Instruction Set

	Mnemonic	Symbolic Operation	Comments
8–BIT LOADS	LD r, s	r ← s	s ≡ r, n, (HL), (IX+e), (IY+e)
	LD d, r	d ← r	d ≡ (HL), r (IX+e), (IY+e)
	LD d, n	d ← n	d ≡ (HL), (IX+e), (IY+e)
	LD A, s	A ← s	s ≡ (BC), (DE), (nn), I, R
	LD d, A	d ← A	d ≡ (BC), (DE), (nn), I, R
16–BIT LOADS	LD dd, nn	dd ← nn	dd ≡ BC, DE, HL, SP, IX, IY
	LD dd, (nn)	dd ← (nn)	dd ≡ BC, DE, HL, SP, IX, IY
	LD (nn), ss	(nn) ← ss	ss ≡ BC, DE, HL, SP, IX, IY
	LD SP, ss	SP ← ss	ss = HL, IX, IY
	PUSH ss	$(SP-1) \leftarrow ss_H$; $(SP-2) \leftarrow ss_L$	ss = BC, DE, HL, AF, IX, IY
	POP dd	$dd_L \leftarrow (SP)$, $dd_H \leftarrow (SP+1)$	dd = BC, DE, HL, AF, IX, IY
EXCHANGES	EX DE, HL	DE ↔ HL	
	EX AF, AF'	AF ↔ AF'	
	EXX	$\begin{pmatrix} BC \\ DE \\ HL \end{pmatrix} \leftrightarrow \begin{pmatrix} BC' \\ DE' \\ HL' \end{pmatrix}$	
	EX (SP), ss	$(SP) \leftrightarrow ss_L$, $(SP+1) \leftrightarrow ss_H$	ss = HL, IX, IY
MEMORY BLOCK MOVES	LDI	(DE) ← (HL), DE ← DE+1 HL ← HL+1, BC ← BC-1	
	LDIR	(DE) ← (HL), DE ← DE+1 HL ← HL+1, BC ← BC-1 Repeat until BC = 0	
	LDD	(DE) ← (HL), DE ← DE-1 HL ← HL-1, BC ← BC-1	
	LDDR	(DE) ← (HL), DE ← DE-1 HL ← HL-1, BC ← BC-1 Repeat until BC = 0	

Table 4-5 (Cont.). CPU Instruction Set

	Mnemonic	Symbolic Operation	Comments
MEMORY BLOCK SEARCHES	CPI	A-(HL), HL ← HL+1 BC ← BC-1	
	CPIR	A-(HL), HL ← HL+1 BC ← BC-1, Repeat until BC = 0 or A = (HL)	A-(HL) sets the flags only. A is not affected
	CPD	A-(HL), HL ← HL-1 BC ← BC-1	
	CPDR	A-(HL), HL ← HL-1 BC ← BC-1, Repeat until BC = 0 or A = (HL)	
8-BIT ALU	ADD s	A ← A + s	CY is the carry flag s ≡ r, n, (HL) (IX+e), (IY+e)
	ADC s	A ← A + s + CY	
	SUB s	A ← A - s	
	SBC s	A ← A - s - CY	
	AND s	A ← A ∧ s	
	OR s	A ← A ∨ s	
	XOR s	A ← A ⊕ s	
	CP s	A - s	s = r, n (HL) (IX+e), (IY+e)
	INC d	d ← d + 1	d = r, (HL) (IX+e), (IY+e)
	DEC d	d ← d - 1	
16-BIT ARITHMETIC	ADD HL, ss	HL ← HL + ss	ss ≡ BC, DE HL, SP
	ADC HL, ss	HL ← HL + ss + CY	
	SBC HL, ss	HL ← HL - ss - CY	
	ADD IX, ss	IX ← IX + ss	ss ≡ BC, DE, IX, SP
	ADD IY, ss	IY ← IY + ss	ss ≡ BC, DE, IY, SP
	INC dd	dd ← dd + 1	dd ≡ BC, DE, HL, SP, IX, IY
	DEC dd	dd ← dd - 1	dd ≡ BC, DE, HL, SP, IX, IY
GP ACC. & FLAG	DAA	Converts A contents into packed BCD following add or subtract.	Operands must be in packed BCD format
	CPL	$A \leftarrow \overline{A}$	
	NEG	A ← 00 – A	
	CCF	$CY \leftarrow \overline{CY}$	
	SCF	CY ← 1	

Table 4-5 (Cont.). CPU Instruction Set

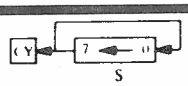
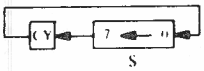
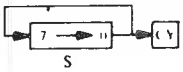
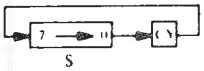
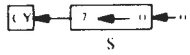
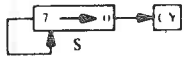
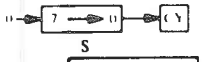
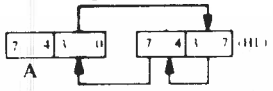
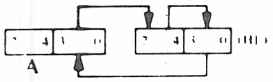
	Mnemonic	Symbolic Operation	Comments
MISCELLANEOUS	NOP	No operation	
	HALT	Halt CPU	
	DI	Disable Interrupts	
	EI	Enable Interrupts	
	IM 0	Set interrupt mode 0	8080A mode
	IM 1	Set interrupt mode 1	Call to 0038_H
	IM 2	Set interrupt mode 2	Indirect Call
ROTATES AND SHIFTS	RLC s		
	RL s		
	RRC s		
	RR s		
	SLA s		s ≡ r, (HL) (IX+e), (IY+e)
	SRA s		
	SRL s		
	RLD		
	RRD		
BIT S, R, & T	BIT b, s	$Z \leftarrow \overline{s_b}$	Z is zero flag
	SET b, s	$s_b \leftarrow 1$	s ≡ r, (HL) (IX+e), (IY+e)
	RES b, s	$s_b \leftarrow 0$	
INPUT AND OUTPUT	IN A, (n)	A ← (n)	
	IN r, (C)	r ← (C)	Set flags
	INI	(HL) ← (C), HL ← HL + 1 B ← B - 1	
	INIR	(HL) ← (C), HL ← HL + 1 B ← B - 1 Repeat until B = 0	
	IND	(HL) ← (C), HL ← HL - 1 B ← B - 1	
	INDR	(HL) ← (C), HL ← HL - 1 B ← B - 1 Repeat until B = 0	

Table 4-5 (Cont.). CPU Instruction Set

	Mnemonic	Symbolic Operation	Comments
	OUT(n), A	(n) ← A	
	OUT(C), r	(C) ← r	
	OUTI	(C) ← (HL), HL ← HL + 1 B ← B - 1	
	OTIR	(C) ← (HL), HL ← HL + 1 B ← B - 1 Repeat until B = 0	
	OUTD	(C) ← (HL), HL ← HL - 1 B ← B - 1	
	OTDR	(C) ← (HL), HL ← HL - 1 B ← B - 1 Repeat until B = 0	
JUMPS	JP nn	PC ← nn	
	JP cc, nn	If condition cc is true PC ← nn, else continue	cc { NZ PO, Z PE, NC P, C M
	JR e	PC ← PC + e	
	JR kk, e	If condition kk is true PC ← PC + e, else continue	kk { NZ NC, Z C
	JP (ss)	PC ← ss	ss = HL, IX, IY
	DJNZ e	B ← B - 1, if B = 0 continue, else PC ← PC + e	
CALLS	CALL nn	(SP-1) ← PC_H (SP-2) ← PC_L, PC ← nn	
	CALL cc, nn	If condition cc is false continue, else same as CALL nn	cc { NZ PO, Z PE, NC P, C M
RESTARTS	RST L	(SP-1) ← PC_H (SP-2) ← PC_L, PC_H ← 0 PC_L ← L	
RETURNS	RET	PC_L ← (SP), PC_H ← (SP+1)	
	RET cc	If condition cc is false continue, else same as RET	cc { NZ PO, Z PE, NC P, C M
	RETI	Return from interrupt, same as RET	
	RETN	Return from non- maskable interrupt	

4.2.2 DMA CONTROLLER (U14).

This device is a multimode Direct Memory Access (DMA) controller for microprocessor systems. It enhances system performance by allowing other devices to directly transfer information to or from memory or to transfer data from one memory to another. Figure 4-5 shows the DMA controller connections and Table 4-6 lists these signals with their functions.

The three basic transfer modes allow programmability of the types of DMA service by the user. Each channel can be individually programmed to Autoinitialize to its original condition following an End of Process ($\overline{EOP}$).

Each channel has a full 64k address and word count capability. An external $\overline{EOP}$ signal can terminate a DMA or memory-to-memory transfer. This is useful for block search or compare operations using external comparators or for intelligent peripherals to abort erroneous services.

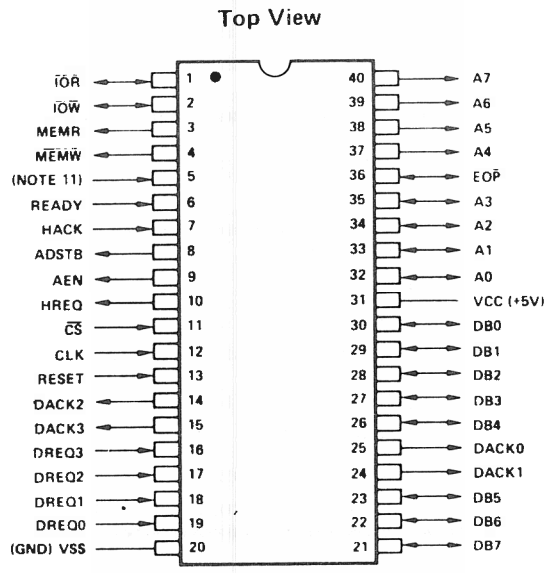


Figure 4-5. DMA Controller Connections

Table 4-6. DMA Controller Signal List

SIGNAL	PIN	FUNCTION
VCC	31	+5 Volt Supply
VSS	20	Ground
CLK CLOCK, INPUT	12	This input controls the internal operations of the DMA controller and its rate of data transfer. The input may be driven at up to 3 MHz the standard DMA controller and up to 4 MHz for the DMA controller.
$\overline{CS}$ CHIP SELECT, INPUT	11	Chip Select is an active low input used to select the DMA controller as an I/O device during the Idle cycle. This allows CPU communication on the data bus.
RESET RESET, INPUT	13	Reset is an asynchronous active high input which clears the Command, Status, Request and Temporary registers. It also clears the first/last flip-flop and sets the mask register. Following a reset, the device is in the Idle cycle.

Table 4-6 (Con't.). DMA Controller Signal List

SIGNAL	PIN	FUNCTION
READY READY, INPUT	6	Ready is an asynchronous input used to extend the memory read and write pulses from the DMA controller to accommodate slow memories or I/O peripheral devices.
HACK HOLD ACKNOWLEDGE, INPUT	7	The active high Hold Acknowledge from the CPU indicates control of the system busses has been relinquished.
DREQ0-DREQ3 DMA REQUEST, INPUT	19, 18, 17, 16	The DMA request lines are individual asynchronous channel request inputs used by peripheral circuits to obtain DMA service. In Fixed Priority, DREQ0 has the highest priority and DREQ3 has the lowest priority. A request is generated by activating the DREQ line of a channel. DACK will acknowledge the recognition of the DREQ signal. Polarity of DREQ is programmable. Reset initializes these lines to active high.

Table 4-6 (Con't.). DMA Controller Signal List

SIGNAL	PIN	FUNCTION
DB0-DB7 DATA BUS, INPUT/OUTPUT	30, 29, 28, 27, 26, 23, 22, 21	The Data Bus lines are bidirectional three-state signals connected to the system data bus. The outputs are enabled in the program condition during the I/O read to output the contents of an Address register, a Status register, the Temporary register, or a Word Count register to the CPU. The outputs are disabled and the inputs are read during an I/O write cycle when the CPU is programming the DMA controller control registers. During DMA cycles, the most significant eight bits of the address are output onto the data bus to be strobed into an external latch by ADSTB. In memory-to-memory operations, data from the memory comes into the DMA controller on the data bus during the read-from-memory transfer. In the write-to-memory transfer, the data bus outputs place the data into the new memory location.

Table 4-6 (Con't.). DMA Controller Signal List

SIGNAL	PIN	FUNCTION
DB0-DB7 DATA BUS, INPUT/OUTPUT	30, 29, 28, 27, 26, 23, 22, 21	The Data Bus lines are bidirectional three-state signals connected to the system data bus. The outputs are enabled in the program condition during the I/O read to output the contents of an Address register, a Status register, the Temporary register, or a Word Count register to the CPU. The outputs are disabled and the inputs are read during an I/O write cycle when the CPU is programming the DMA controller control registers. During DMA cycles, the most significant eight bits of the address are output onto the data bus to be strobed into an external latch by ADSTB. In memory-to-memory operations, data from the memory comes into the DMA controller on the data bus during the read-from-memory transfer. In the write-to-memory transfer, the data bus outputs place the data into the new memory location.

Table 4-6 (Con't.). DMA Controller Signal List

SIGNAL	PIN	FUNCTION
		also generates a pulse when the terminal count (TC) for any channel is reached. This generates an EOP signal which is output through the $\overline{\text{EOP}}$ line. The reception of $\overline{\text{EOP}}$, either internal or external, will cause the DMA controller to terminate the service, reset the request, and if Autoinitiaize is enabled, to write the base registers to the current registers of that channel. The mask bit and TC bit in the status word will be set for the currently active channel by $\overline{\text{EOP}}$ unless the channel is programmed for Autoinitialize. In that case, the mask bit remains clear. During memory-to-memory transfers, $\overline{\text{EOP}}$ will be output when the TC for channel 1 occurs.
A0-A3 ADDRESS, INPUT/OUTPUT	32, 33, 34, 35	The four least significant address lines are bidirectional, three-state signals. In the idle cycle, they are inputs and are used by the DMA controller to address the control register to be loaded or read. In the active cycle, they are outputs and provide the lower 4 bits of the output address.

Table 4-6 (Con't.). DMA Controller Signal List

SIGNAL	PIN	FUNCTION
A4-A7 ADDRESS, OUTPUT	37, 38, 39, 40	The four most significant address lines are three-state outputs and provide four bits of address. These lines are enabled only during the DMA service.
HREQ HOLD REQUEST, OUTPUT	10	This is the Hold Request to the CPU and is used to request control of the system bus. If the corresponding mask bit is clear, the presence of any valid DREQ causes the DMA controller to issue the HREQ.
DACK0-DACK3 DMA ACKNOWLEDGE, OUTPUT	25, 24, 17, 16	DMA Acknowledge is used to notify individual peripherals when one has been granted a DMA cycle. The sense of these lines is programmable. Reset initializes them to active low.

Table 4-6 (Con't.). DMA Controller Signal List

SIGNAL	PIN	FUNCTION
AEN ADDRESS ENABLE, OUTPUT	9	The Address Enable is an active high level used to enable the output of the external latch which holds the upper byte of address, and to disable the system bus during the DMA cycle. Note that during DMA transfers, HACK and AEN should be used to de-select all other I/O peripherals which may erroneously be accessed as programmed I/O during the DMA operation. The DMA controller automatically de-selects itself during DMA transfer.
ADSTB ADDRESS STROBE, OUTPUT	8	The active high Address Strobe is used to strobe the upper address byte into an external latch.
$\overline{\text{MEMR}}$ MEMORY READ, OUTPUT	3	The Memory Read signal is an active low, three-state output used to access data from the selected memory location during a DMA read or a memory-to-memory transfer.
$\overline{\text{MEMW}}$ MEMORY WRITE, OUTPUT	4	The Memory Write signal is an active low, three-state output used to write data to the selected memory location during a DMA write or a memory-to-memory transfer.

4.2.2.1 DMA CONTROLLER FUNCTIONAL DESCRIPTION.

The DMA Controller block diagram shown in Figure 4-6 includes the major logic blocks and all of the internal registers. The data interconnection paths are also shown. Not shown are the various cortrol signals between the blocks. The DMA Controller contains 344 bits of internal memory in the form of registers. Table 4-7 lists these registers by name and shows the size of each. A detailed description of the registers and their functions can be found under Register Description.

The DMA Controller contains three basic blocks of control logic. The Timing Control block generates internal timing and external control signals for the DMA Controller. The Program Command Control block decodes the various commands given to the DMA Controller by the microprocessor prior to servicing a DMA Request. It also decodes the Mode Control word used to select the type of DMA during the servicing. The Priority Encoder block resolves priority contention between DMA channels requesting service simultaneously.

The Timing Control block derives internal timing from the clock input.

4.2.2.2 DMA OPERATION.

The DMA Controller operates in two major cycles. These are called Idle and Active cycles. Each device cycle is made up of a number of states. The DMA Controller can assume seven separate states, each composed of one full clock period. State 1 (S1) is the inactive state. It is entered when the DMA Controller has no valid DMA requests pending. While in S1, the DMA Controller is inactive but may be in the Program Condition, being programmed by the processor. State 0 (S0) is the first state of a DMA service. The DMA Controller has requested a hold, but the processor has not yet returned an acknowledge. An acknowledge from the CPU signals that transfers may begin. S1, S2, S3 and S4 are working states of the DMA service. If more time is needed to complete a transfer than is available with normal timing, wait states (SW) can be inserted between S2 or S3 and S4 by the use of the Ready Line on the DMA Controller.

Table 4-7. DMA Controller Internal Registers

Name	Size	Number
Base Address Registers	16 bits	4
Base Word Count Registers	16 bits	4
Current Address Registers	16 bits	4
Current Word Count Registers	16 bits	4
Temporary Address Register	16 bits	1
Temporary Word Count Register	16 bits	1
Status Register	8 bits	1
Command Register	8 bits	1
Temporary Register	8 bits	1
Mode Registers	6 bits	4
Mask Register	4 bits	1
Request Register	4 bits	1

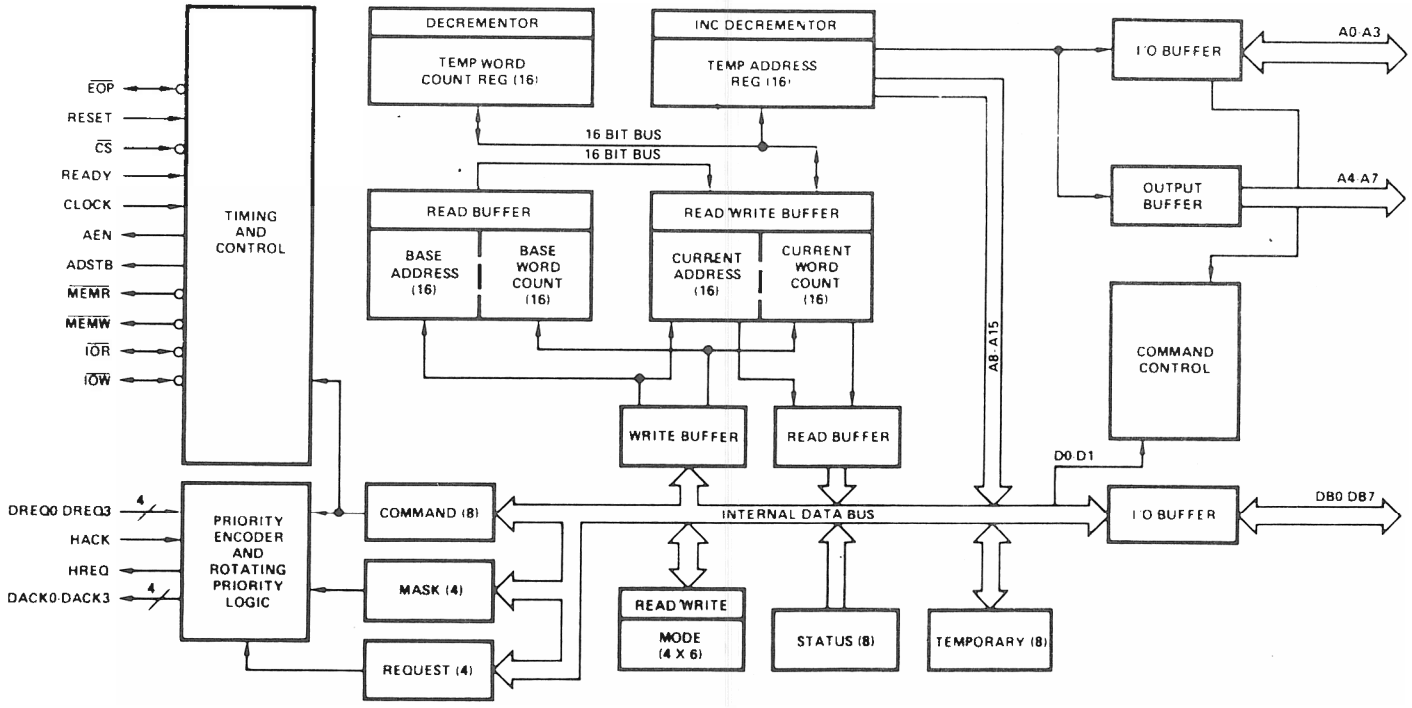


Figure 4-6. DMA Controller Block Diagram

Memory-to-memory transfers require a read-from and a write-to memory to complete each transfer. The states, which resemble the normal working states, use two digit numbers for indentification. Eight states are required for a single transfer. The first four states (S11, S12, S13, S14) are used for the read-from-memory half, and the last four states (S21, S22, S23 and S24) for the write-to-memory half of the transfer.

4.2.2.3 IDLE CYCLE.

When no channel is requesting service, the DMA Controller will enter the idle cycle and perform "SI" states. In this cycle, the DMA Controller samples the DREQ lines every clock cycle to determine if any channel is requesting a DMA service. The device also samples $\overline{CS}$, looking for an attempt by the microprocessor to write or read the internal registers of the DMA Controller. When $\overline{CS}$ is low and HREQ is low, the DMA Controller enters the Program Condition. The CPU can now establish, change, or inspect the internal definition of the part by reading from or writing to the internal registers. Address lines A0-A3 are inputs to the device and select which registers are to be read or written. The $\overline{IOR}$ and $\overline{IOW}$ lines are used to select and time reads or writes. Due to the number and size of the internal registers, an internal flip-flop is used to generate an additional bit of address. This bit is used to determine the upper or lower byte of the 16-bit Address and Word Count registers. The flip-flop is reset by Master Clear or Reset. A separate software command can also reset this flip-flop.

Special software commands can be executed by the DMA Controller in the Program Condition. These commands are decoded as sets of addresses with $\overline{CS}$ and $\overline{IOW}$. The commands do not make use of the data bus. Instructions include Clear First/Last Flip-Flop and Master Clear.

4.2.2.4. ACTIVE CYCLE.

When the DMA Controller is in the Idle cycle and a channel requests a DMA service, the device outputs an HREQ to the microprocessor and enters the Active cycle. It is in this cycle that the DMA service takes place in one of three modes:

Single Transfer Mode. In Single Transfer mode, the device is programmed to make one transfer only. The word count is decremented and the address decremented or incremented following each transfer. When the word count goes to zero, a Terminal Count (TC) causes an Autoinitialize if the channel has been programmed to do so.

DREQ must be held active until DACK becomes active in order to be recognized. If DREQ is held active throughout the single transfer, HREQ goes inactive and releases the bus to the system. It will again go active and upon receipt of a new HACK, another single transfer is performed. In 8080A/9080A systems, this will ensure one full machine cycle execution between DMA transfers.

Block Transfer Mode. In Block Transfer mode, the device is activated by DREQ to continue making transfers during the service until a TC, caused by the word count going to zero, or an external End of Process (EOP) is encountered. DREQ need only be held active until DACK becomes active. Again, an Autoinitialize occurs at the end of the service if the channel has been programmed for it.

Demand Transfer Mode. In Demand Transfer mode, the device is programmed to continue making transfers until a TC or external $\overline{EOP}$ is encountered or until DREQ goes inactive. Thus transfers may continue until the I/O device has exhausted its data capacity. After the I/O device has had a chance to catch up, the DMA service is re-established by means of a DREQ. During the time between services when the microprocessor is allowed to operate, the intermediate values of address and word count are stored in the DMA Controller Current Address and Current Word Count registers. Only an $\overline{EOP}$ can cause an Autoinitalize at the end of the service. $\overline{EOP}$ is generated either by TC or by an external signal.

4.2.2.5 TRANSFER TYPES.

Each of the three active transfer modes can perform three different types of transfers. These are Read, Write and Verify. Write transfers move data from an I/O device to the memory by activating $\overline{\text{MEMW}}$ and $\overline{\text{IOR}}$. Read transfers move data from memory to an I/O device by activating MEMR and IOW. Verify transfers are pseudo transfers. The DMA Controller operates as in Read or Write transfers generating addresses, and responding to $\overline{\text{EOP}}$. However, the memory and I/O control lines all remain inactive.

Memory-to-Memory. To perform block moves of data from one memory address space to another with a minimum of program effort and time, the DMA Controller includes memory-to-memory transfer features. Programming a bit in the Command register selects channels 0 and 1 to operate as memory-to-memory transfer channels. The transfer is initiated by setting the software DREQ for channel 0. The DMA Controller requests a DMA service in the normal manner. After HACK is true, the device, using eight-state transfers in Block Transfer mode, reads data from the memory. The channel 0 Current Address register is the source for the address used and is decremented or incremented in the normal manner. The data byte read from the memory is stored in the DMA Controller internal Temporary register. Channel 1 then writes the data from the Temporary register to memory using the address in its Current Address register and incrementing or decrmenting it in the normal manner. The channel 1 Current Word Count is decremented. When the word count of channel 1 goes to zero, a TC is generated causing an $\overline{\text{EOP}}$ output terminating the service.

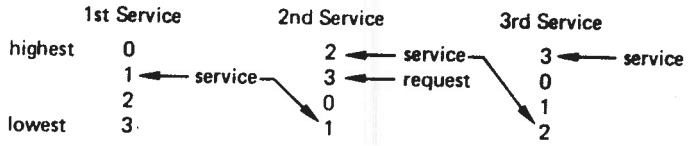
Channel 0 may be programmed to retain the same address for all transfers. This allows a single word to be written to a block of memory.

The DMA Controller responds to external $\overline{\text{EOP}}$ signals during memory-to-memory transfers. Data comparators in block search schemes may use this input to terminate the service when a match is found.

Autoinitialize. By programming a bit in the Mode register, a channel may be set up as an Autoinitialize channel. During Autoinitialization, the original values of the Current Address and Current Word Count registers are automatically restored from the Base Address and Base Word Count registers of that channel following $\overline{EOP}$. The base registers are loaded simultaneously with the current registers by the microprocessor and remain unchanged throughout the DMA service. The mask bit is not set when the channel is in Autoinitialize. Following Autoinitialize, the channel is ready to perform another service without CPU intervention.

Priority. The DMA Controller has two types of priority encoding available as software selectable options. The first is Fixed Priority which fixes the channels in priority order based upon the descending value of their number. The channel with the lowest priority is 3 followed by 2, 1 and 0 (the highest priority channel). After the recognition of any one channel for service, the other channels are prevented from interfering with that service until it is complete.

The second scheme is Rotating Priority. The last channel to get service becomes the lowest priority channel with the others rotating accordingly.



With Rotating Priority in a single chip DMA system, any device requesting service is guaranteed to be recognized after no more than three higher priority services have occurred. This prevents any one channel from monopolizing the system.

Compressed Timing. In order to achieve even greater throughput where system characteristics permit, the DMA Controller can compress the transfer time to two clock cycles. State S3 is used to extend the access time of the read pulse. By removing state S3, the read pulse width is made equal to the write pulse width, and a transfer consists only of state S2 to change the address and state S4 to perform the read/write. S1 states still occur when A8-A15 need updating (see Address Generation).

Address Generation. In order to reduce pin count, the DMA Controller multiplexes the eight higher order address bits on the data lines. State S1 is used to output the higher order address bits to an external latch from which they may be placed on the address bus. The falling edge of Address Strobe (ADSTB) is used to load these bits from the data lines to the latch. Address Enable (AEN) is used to enable the bits onto the address bus through a three-state enable. The lower order address bits are output directly. Lines A0-A7 should be connected to the address bus.

During Block and Demand Transfer mode services which include multiple transfers, the addresses generated are sequential. For many transfers, the data held in the external address latch remains the same. This data need only change when a carry or borrow from A7 to A8 takes place in the normal sequence of addresses. To save time and speed transfers, the DMA Controller executes S1 states only when updating of A8-A15 in the latch is necessary. This means for long services, S1 states may occur only once every 256 transfers, a savings of 255 clock cycles for each 256 transfers.

4.2.2.6 REGISTER DESCRIPTION.

Current Address Register. Each channel has a 16-bit Current Address register. This register holds the value of the address used during DMA transfers. The address is automatically incremented or decremented after each transfer, and the intermediate values of the address are stored in the Current Address register during the transfer. This register is written or read by the microprocessor in successive 8-bit bytes. It may also be re-initialized by an Autoinitialize back to its original value. Autoinitialization takes place only after an EOP.

Current Word Count Register. Each channel has a 16-bit Current Word Count register. This register holds the number of transfers to be performed. The word count is decremented after each transfer. The intermediate value of the word count is stored in the register during the transfer. When the value in the register goes to zero, a TC is generated. This register is loaded or read in successive 8-bit bytes by the microprocessor in the Program Condition. Following the end of a DMA service, it may also be re-initialized by an Autoinitialize back to its original value. Autoinitialize can occur only when an EOP occurs.

Base Address and Base Word Count Registers. Each channel has a pair of Base Address and Base Word Count registers. These 16-bit registers store the original values of their associated current registers. During Autoinitialize, these values are used to restore the current registers to their original values. The base registers are written simultaneously with their corresponding current register in 8-bit bytes in the Program Condition by the microprocessor. These registers cannot be read by the microprocessor.

Command Register. This 8-bit register controls the operation of the DMA Controller. It is programmed by the microprocessor in the Program Condition and is cleared by Reset. See Figure 4-7 for the function of the command bits and Table 4-8 for address coding.

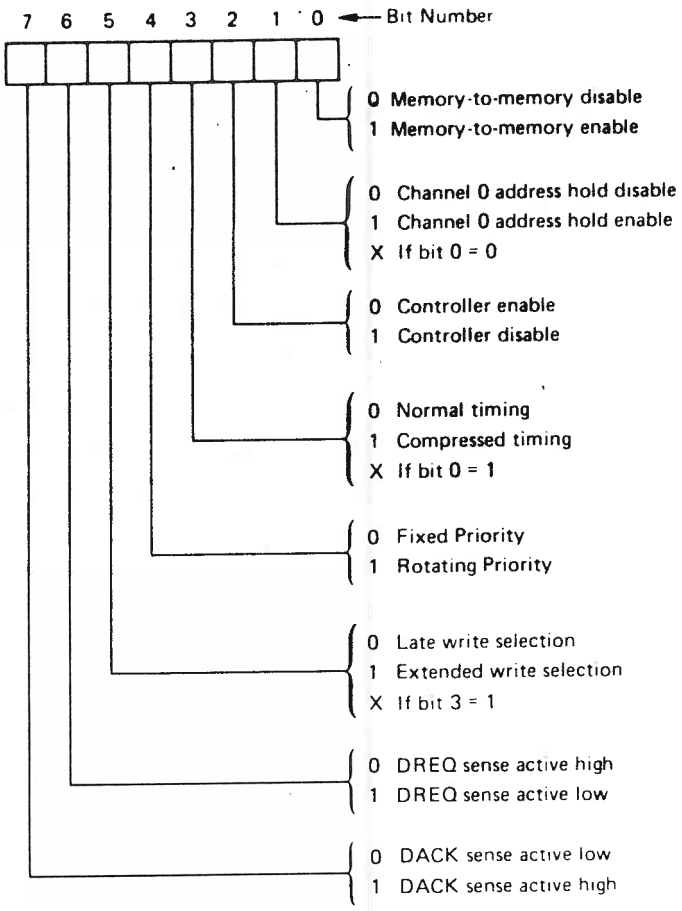


Figure 4-7. DMA Controller Command Bits

Mode Register. Each channel has a 6-bit Mode register associated with it. When the register is being written to by the microprocessor in the Program Condition, bits 0 and 1 determine which channel Mode register is to be written. See Figure 4-8 for the mode register bit configuration.

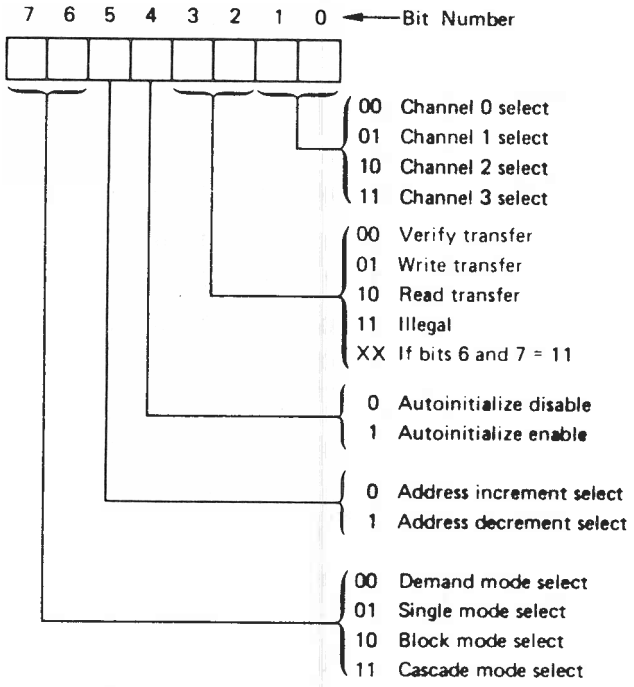


Figure 4-8. DMA Controller Mode Register Bits

Table 4-8. DMA Controller Word Count and Address Register Command Codes

Channel	Register	Operation	Signals							Internal Flip/Flop	Data Bus DB0–DB7
			$\overline{CS}$	$\overline{IOR}$	$\overline{IOW}$	A3	A2	A1	A0		
0	Base & Current Address	Write	0	1	0	0	0	0	0	0	A0-A7
			0	1	0	0	0	0	0	1	A8-A15
	Current Address	Read	0	0	1	0	0	0	0	0	A0-A7
			0	0	1	0	0	0	0	1	A8-A15
	Base & Current Word Count	Write	0	1	0	0	0	0	1	0	W0-W7
			0	1	0	0	0	0	1	1	W8-W15
	Current Word Count	Read	0	0	1	0	0	0	1	0	W0-W7
			0	0	1	0	0	0	1	1	W8-W15
1	Base & Current Address	Write	0	1	0	0	0	1	0	0	A0-A7
			0	1	0	0	0	1	0	1	A8-A15
	Current Address	Read	0	0	1	0	0	1	0	0	A0-A7
			0	0	1	0	0	1	0	1	A8-A15
	Base & Current Word Count	Write	0	1	0	0	0	1	1	0	W0-W7
			0	1	0	0	0	1	1	1	W8-W15
	Current Word Count	Read	0	0	1	0	0	1	1	0	W0-W7
			0	0	1	0	0	1	1	1	W8-W15
2	Base & Current Address	Write	0	1	0	0	1	0	0	0	A0-A7
			0	1	0	0	1	0	0	1	A8-A15
	Current Address	Read	0	0	1	0	1	0	0	0	A0-A7
			0	0	1	0	1	0	0	1	A8-A15
	Base & Current Word Count	Write	0	1	0	0	1	0	1	0	W0-W7
			0	1	0	0	1	0	1	1	W8-W15
	Current Word Count	Read	0	0	1	0	1	0	1	0	W0-W7
			0	0	1	0	1	0	1	1	W8-W15
3	Base & Current Address	Write	0	1	0	0	1	1	0	0	A0-A7
			0	1	0	0	1	1	0	1	A8-A15
	Current Address	Read	0	0	1	0	1	1	0	0	A0-A7
			0	0	1	0	1	1	0	1	A8-A15
	Base & Current Word Count	Write	0	1	0	0	1	1	1	0	W0-W7
			0	1	0	0	1	1	1	1	W8-W15
	Current Word Count	Read	0	0	1	0	1	1	1	0	W0-W7
			0	0	1	0	1	1	1	1	W8-W15

Request Register. The DMA Controller can respond to requests for DMA service which are initiated by software as well as by a DREQ. Each channel has a request bit associated with it in the four bit Request register. These are non-maskable and subject to prioritization by the Priority Encoder network. Each register bit is set or reset separately under software control or is cleared upon generation of a TC or external $\overline{EOP}$. The entire register is cleared by a Reset. To set or reset a bit, the software loads the proper form of the data word. See Figure 4-9 for the request register bit configuration and Table 4-11 for address coding.

Software requests will be serviced only if the channel is in Block mode. When initiating a memory-to-memory transfer, the software request for channel 0 should be set.

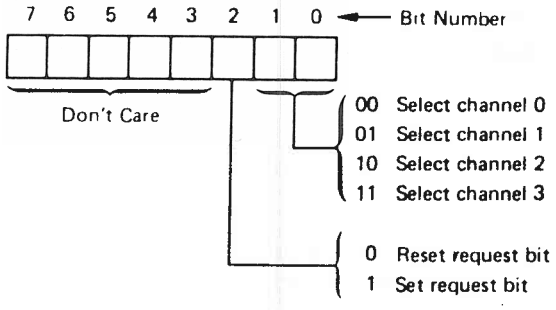


Figure 4-9. DMA Controller Request Register Bits

Mask Register. Each channel has associated with it a mask bit which can be set to disable the incoming DREQ. Each mask bit is set when its associated channel produces an $\overline{EOP}$ if the channel is not programmed for Autoinitialize. Each bit of the four bit Mask register may also be set or cleared separately under software control. The entire register is also set by a Reset. This disables all DMA requests until a clear Mask register instruction allows them to occur. The instruction to separately set or clear the mask bits is similar in form to that used with the Request register. See Figure 4-10 for the mask register bit configuration and Table 4-9 for Register instruction codes.

Table 4-9. DMA Controller Register Codes

Register	Operation	Signals						
		$\overline{CS}$	$\overline{IOR}$	$\overline{IOW}$	A3	A2	A1	A0
Command	Write	0	1	0	1	0	0	0
Mode	Write	0	1	0	1	0	1	1
Request	Write	0	1	0	1	0	0	1
Mask	Set/Reset	0	1	0	1	0	1	0
Mask	Write	0	1	0	1	1	1	1
Temporary	Read	0	0	1	1	1	0	1
Status	Read	0	0	1	1	0	0	0

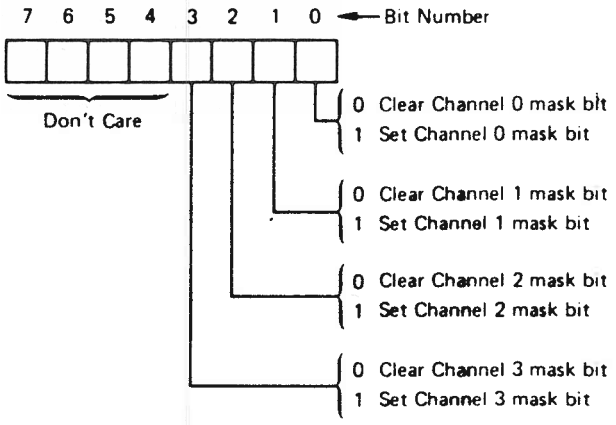


Figure 4-10. DMA Controller Mask Register Bits

Status Register. The Status register is available to read out of the DMA Controller by the microprocessor. It contains information about the status of the device at that point. This information includes which channels have reached a terminal count and which channels have pending DMA requests. Bits 0-3 are set every time a TC is reached by that channel. These bits are cleared upon Reset and on each Status Read. Bits 4-7 are set whenever their corresponding channel is requesting service. See Figure 4-11 for status register bit configuration.

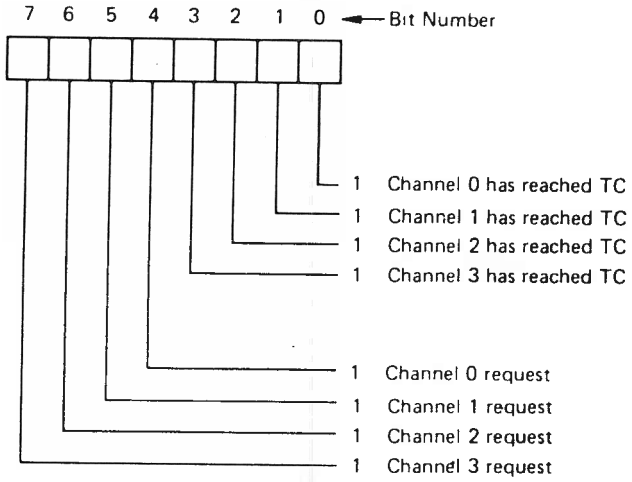


Figure 4-11. DMA Controller Status Register Bits

Temporary Register. The Temporary register is used to hold data during memory-to-memory transfers. Following the completion of the transfers, the last word moved can be read by the microprocessor in the Program Condition. The Temporary register always contains the last byte transferred in the previous memory-to-memory operation, unless cleared by a Reset.

Software Commands. These are additional special software commands which can be executed in the Program Condition. They do not depend on any specific bit pattern on the data bus. The two software commands are:

Clear First/Last Flip-Flop: This command is executed prior to writing or reading new address or word count information to the DMA Controller. This initializes the flip-flop to a known state so that subsequent accesses to register contents by the microprocessor will address upper and lower bytes in the correct sequence.

Master Clear: This software instruction has the same effect as the hardware Reset. The Command, Status, Request, Temporary, and Internal First/Last Flip-Flop registers are cleared and the Mask register is set. The DMA Controller enters the Idle cycle. Table 4-10 lists the address codes for the software commands.

Table 4-10. DMA Controller Software Command Codes

Operation	Registers Affected	Signals						
		$\overline{CS}$	$\overline{IOR}$	$\overline{IOW}$	A3	A2	A1	A0
Clear FF	Internal First/Last Flip/Flop	0	1	0	1	1	0	0
Master Clear	Clear: Command Status Request Temporary Internal First/Last Flip/Flop Set: Mask	0	1	0	1	1	0	1

4.2.3 PROGRAMMABLE COMMUNICATIONS INTERFACE MODULE (U27-U30).

The Programmable Communications Interface (PCI) module is a synchronous/asynchronous data communications controller chip for microcomputer systems in a polled or interrupt driven system environment. The PCI accepts programmed instructions from the microprocessor and supports many serial data communications disciplines, synchronous and asynchronous, in the full or half duplex mode.

The PCI serializes parallel data characters received from the microprocessor for transmission. Simultaneously, it can receive serial data and convert it into parallel data characters for input to the microcomputer.

The PCI contains a baud rate generator which 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. The pin configuration is shown in Figure 4-12 and the designations are shown in Table 4-11.

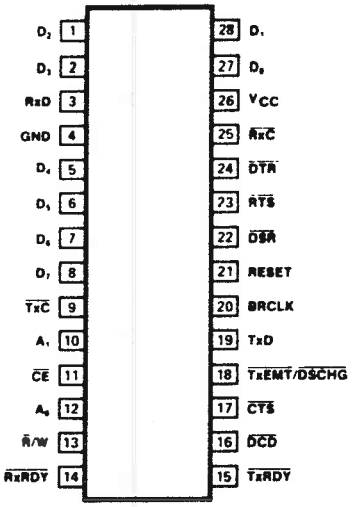


Figure 4-12. PCI Pin Configuration

Table 4-11. PCI Pin Designation

PIN NO.	SYMBOL	NAME AND FUNCTION	TYPE
27,28,1,2, 5-8	D_0-D_7	8-bit data bus	I/O
21	RESET	Reset	I
12,10	A_0-A_1	Internal register select lines	I
13	$\overline{R}/W$	Read or write command	I
11	$\overline{CE}$	Chip enable input	I
22	$\overline{DSR}$	Data set ready	I
24	$\overline{DTR}$	Data terminal ready	O
23	$\overline{RTS}$	Request to send	O
17	$\overline{CTS}$	Clear to send	I
16	$\overline{DCD}$	Data carrier detected	I
18	$\overline{TxEMT/DSCHG}$	Transmitter empty or data set change	O
9	$\overline{TxC}$	Transmitter clock	I/O
25	$\overline{RxC}$	Receiver clock	I/O
19	TxD	Transmitter data	O
3	RxD	Receiver data	I
15	$\overline{TxRDY}$	Transmitter ready	O
14	$\overline{RxRDY}$	Receiver ready	O
20	BRCLK	Baud rate generator clock	I
26	V_{CC}	+5V supply	I
4	GND	Ground	I

4.2.3.1 BLOCK DIAGRAM.

The PCI consists of six major sections. These are the operation control, timing, receiver, transmitter, 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. The PCI block diagram is shown in Figure 4-13.

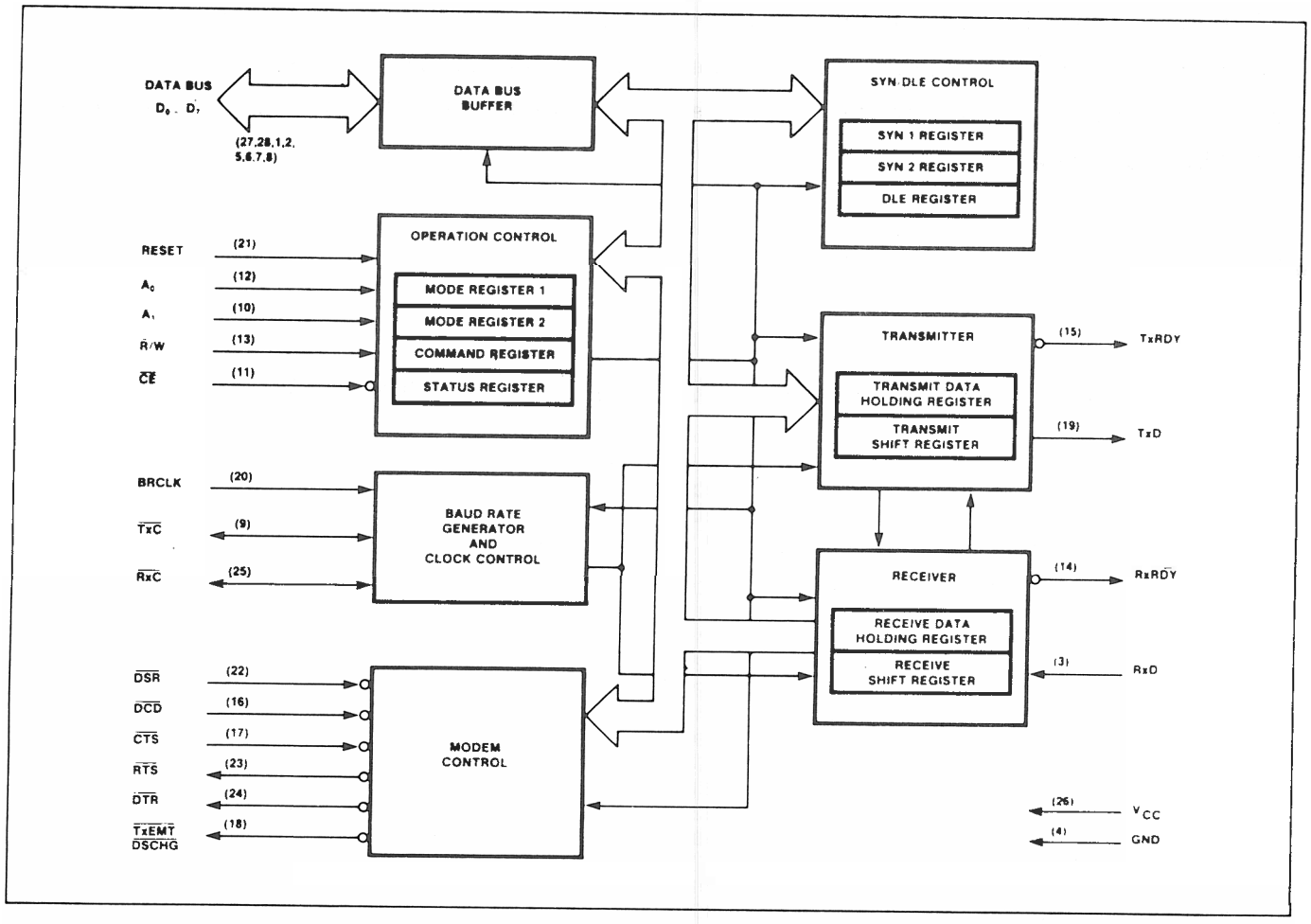


Figure 4-13. Programmable Communications Interface Block Diagram

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 Paragraph 4.2.2.6 of this manual.

Timing. The PCI 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 4-12 for baud rate generator characteristics.

Table 4-12. Baud Rate Generator Characteristics

BAUD RATE	THEORETICAL FREQUENCY 16X CLOCK	ACTUAL FREQUENCY 16X CLOCK	PERCENT ERROR	DIVISOR
50	0.8 KHz	0.8 KHz	--	6336
75	1.2	1.2	--	4224
110	1.76	1.76	--	2880
134.5	2.152	2.1523	0.016	2355
150	2.4	2.4	--	2112
300	4.8	4.8	--	1056
600	9.6	9.6	--	528
1200	19.2	19.2	--	264
1800	28.8	28.8	--	176
2000	32.0	32.081	0.253	158
2400	38.4	38.4	--	132
3600	57.6	57.6	--	88
4800	76.8	76.8	--	66
7200	115.2	115.2	--	44
9600	153.6	153.6	--	33
19200 *	307.2	316.8	3.125	16

NOTE
*Error at 19200 can be reduced to zero by using crystal frequency 4.9152MHz
16X clock is used in asynchronous mode. In synchronous mode, clock multiplier is 1X

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.

4.2.3.2 INTERFACE SIGNALS.

The PCI interface signals can be grouped into two types: The CPU-related signals shown in Table 4-13, which interface the PCI to the microprocessor system; and the device-related signals (shown in Table 4-14), which are used to interface to the communications device or system.

Table 4-13. 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 2651. 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 PCI 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 PCI 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 tri-state condition.
D_7-D_0	8,7,6,5, 2,1,28,27	I/O	8-bit, three-state data bus used to transfer commands, data and status between PCI 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.
$\overline{TxEMT/DSCHG}$	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.

Table 4-14. Device Related Signals

PIN NAME	PIN NO.	INPUT/OUTPUT	FUNCTION
BRCLK	20	I	5.0688MHz clock input to the internal baud rate generator. Not required if external receiver and transmitter clocks are used.
$\overline{RxC}$	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 is sampled on the rising edge of the clock. If internal receiver clock is programmed, this pin becomes an output at 1X the programmed baud rate.*
$\overline{TxC}$	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 becomes an output at 1X the programmed baud rate.*
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 $\overline{TxEMT/DSCHG}$ when its state changes.
$\overline{DCD}$	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 $\overline{TxEMT/DSCHG}$ when its state changes.
$\overline{CTS}$	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.

NOTE
*$\overline{RxC}$ and $\overline{TxC}$ outputs have short circuit protection max C_L 100pf

4.2.3.3 OPERATION.

The functional operation of the PCI 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. The programming procedure is described in the PCI Programming section of this manual.

After programming, the PCI is ready to perform the desired communications function. 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 PCI 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 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 the stop bit(s) have been assembled. The data is then transferred to the Receive Data Holding Register, the $\overline{RxRDY}$ bit in the status register is set, and the 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 the $\overline{RxC}$ corresponding to the received character boundary. If a break condition is detected (RxD is low for the entire character as well as the stop bit(s)), only one character consisting of all zeros (with the FE status bit 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.

When the PCI 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 is 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 PCI returns to the hunt mode. (Note that the sequence SYN1-SYN1-SYN2 will not achieve synchronization). When synchronization has been achieved, the PCI continues to assemble characters and transfer them to the Holding Register, setting the RxRDY status bit and asserting the $\overline{\text{RxRDY}}$ 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.

Transmitter. The PCI is conditioned to transmit data when the $\overline{\text{CTS}}$ input is low and the TxEN command register bit is set. The PCI indicates to the CPU that it can accept a character for transmission by setting the TxRDY status bit and asserting the $\overline{\text{TxRDY}}$ output. When the CPU writes a character into the Transmit Data Holding Register, these conditions are negated. Data is 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 $\overline{\text{TxEMT/DSCHG}}$ 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 high.

In the synchronous mode, when the PCI is initially conditioned to transmit, the TxD output remains high and the TxRDY condition is asserted until the first character to be transmitted (usually an 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 PCI by the time the transmitter has completed sending the previous character. Since synchronous communication does not allow gaps between characters, the PCI 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 SEN DLE bit in the command register is true, the DLE character is automatically transmitted prior to transmission of the message character in THR.

4.2.3.4 PCI PROGRAMMING.

Prior to initiating data communications, the PCI 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 PCI can be reconfigured at any time during program execution. However, if the change has an effect on the reception of a character, the receiver should be disabled. Alternatively if the change is made 1 1/2 RxC periods after RxRDY goes active, it will affect the next character assembly. A flowchart of the initialization process appears in Figure 4-14.

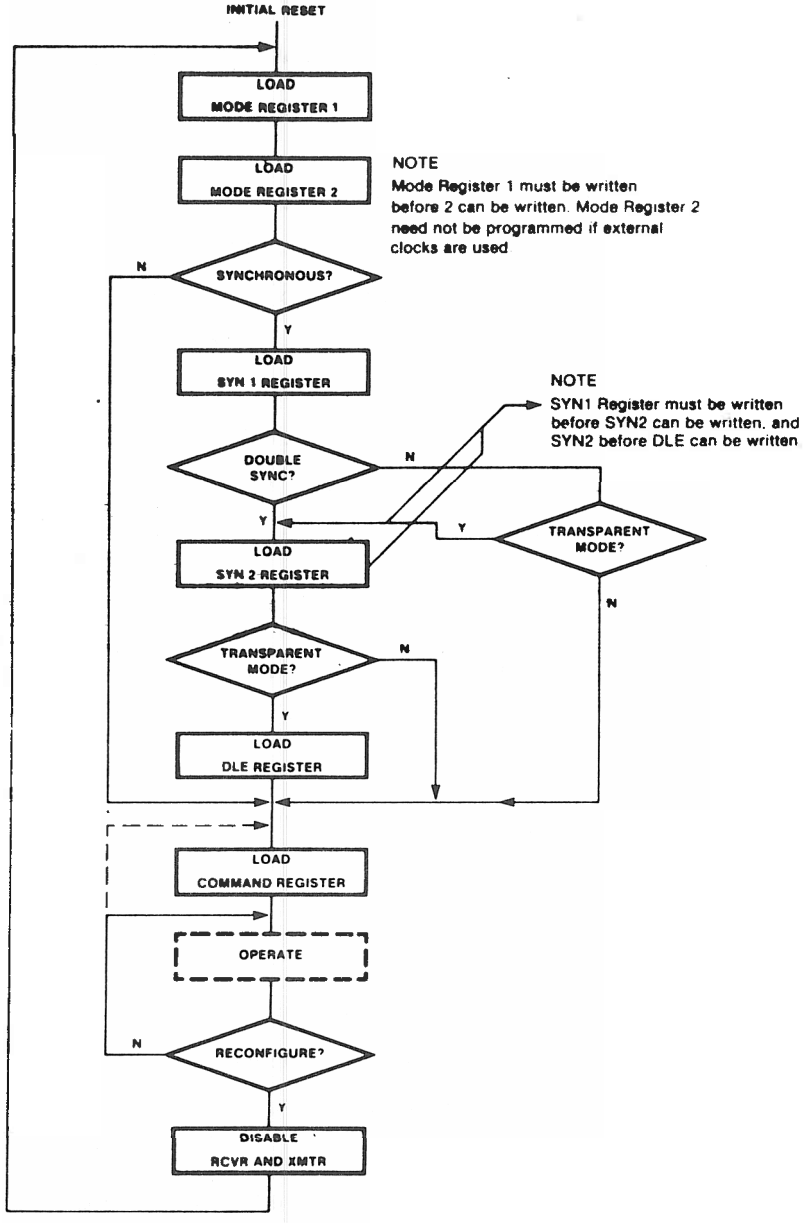


Figure 4-14. PCI Initialization Flow Chart

The internal registers of the PCI are accessed by applying specific signals to the $\overline{CE}$, $\overline{R/W}$, A1 and A0 inputs. The conditons necessary to address each register are shown in Table 4-15.

Table 4-15. PCI Register Addressing

$\overline{CE}$	A_1	A_0	$\overline{R}/W$	FUNCTION
1	X	X	X	Tri-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 1/2
0	1	0	1	Write mode registers 1/2
0	1	1	0	Read command register
0	1	1	1	Write command register

NOTE
See AC Characteristics section for timing requirements

The SYN1, SYN2, and DLE registers are accessed by performing write operations with the conditions A1=0, A0=1, and R/W=1. The first operation loads the SYN1 register. The next loads the ASYN2 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 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.

4.2.3.5 DESCRIPTION OF REGISTERS.

Mode Registers 1 and 2 define the general operational characteristics of the PCI, while the Command Register controls the operation within its basic framework. The PCI indicates its status in the Status Register. These registers are cleared when a RESET input is applied.

Mode Register 1 (MR1). Table 4-16 illustrates Mode Register 1. Bits MR11 and MR10 select the communication format and baud rate multiplier. 00 specifies synchronous mode and 1X amplifier. 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 by MR24 or MR25.

Table 4-16. Mode Register 1 (MR1)

<table>
<tr><th>MR17</th><th>MR16</th><th>MR15</th><th>MR14</th><th>MR13</th><th>MR12</th><th>MR11</th><th>MR10</th></tr>
<tr><td colspan="2"></td><td>Parity Type</td><td>Parity Control</td><td colspan="2">Character Length</td><td colspan="2">Mode and Baud Rate Factor</td></tr>
<tr><td colspan="2">ASYNCH: 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 colspan="2" rowspan="2">00 = 5 BITS
01 = 6 BITS
10 = 7 BITS
11 = 8 BITS</td><td colspan="2" rowspan="2">00 = SYNCHRONOUS 1X RATE
01 = ASYNCHRONOUS 1X RATE
10 = ASYNCHRONOUS 16X RATE
11 = ASYNCHRONOUS 64X RATE</td></tr>
<tr><td>SYNCH: NUMBER OF SYN CHAR
0 = DOUBLE SYN
1 = SINGLE SYN</td><td>SYNCH: TRANS-PARENCY 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 internal clock is selected. Mode must be selected (MR11, MR10) in any case

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 bit 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. Also DLE stripping and DLE detect (with MR14 = 0) are enabled.

Mode Register 2 (MR2). Table 4-17 illustrates Mode Register 2. MR23, MR22, MR21, and MR20 control the frequency of the internal baud rate generator (BRG). Sixteen rates are selectable. When driven by a 5.0688 MHz input at the BRCLK input (pin 20), the BRG output has zero error except at 134.5, 2000, and 19,200 baud respectively.

Table 4-17. Mode Register 2 (MR2)

MR27	MR26	MR25	MR24	MR23	MR22	MR21	MR20
		Transmitter Clock	Receiver Clock	Baud Rate Selection			
NOT USED		0 = EXTERNAL 1 = INTERNAL	0 = EXTERNAL 1 = INTERNAL	0000 = 50 BAUD 0001 = 75 0010 = 110 0011 = 134.5 0100 = 150 0101 = 300 0110 = 600 0111 = 1200		1000 = 1800 BAUD 1001 = 2000 1010 = 2400 1011 = 3600 1100 = 4800 1101 = 7200 1110 = 9600 1111 = 19,200	

MR25 and MR24 select either the BRG or the external inputs TxC and RxC as the clock source for the transmitter and receiver, respectively. If the BRG clock is selected, the baud rate factor in asynchronous mode is 16X regardless of the factor selected by MR11 and MR10. In addition, the corresponding clock pin provides an output at 1X the baud rate.

Command Register (CR). Table 4-18 illustrates the Command Register. Bits CR0 (TxEN) and CR2 (RxEN) enable or disable the transmitter and receiver respectively. Disabling the receiver causes $\overline{\text{RxRDY}}$ to go high (inactive). If the transmitter is disabled, it completes the transmission of the character in the Transmit Shift Register (if any) prior to terminating operation. The TxD output then remains in the marking state (high) while the $\overline{\text{TxRDY}}$ and $\overline{\text{TxEMT}}$ go high (inactive). If the receiver is disabled, it terminates operation immediately. Any character being assembled is neglected.

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

Table 4-18. Command Register (CR)

CR7	CR6	CR5	CR4	CR3	CR2	CR1	CR0
Operating Mode		**Request to Send**	**Reset Error**		**Receive Control (RxEN)**	**Data Terminal Ready**	**Transmit Control (TxEN)**
00 = NORMAL OPERATION 01 = ASYNCH: AUTOMATIC ECHO MODE SYNCH: SYN AND/OR DLE STRIPPING MODE 10 = LOCAL LOOP BACK 11 = REMOTE LOOP BACK		0 = FORCE $\overline{\text{RTS}}$ OUTPUT HIGH 1 = FORCE $\overline{\text{RTS}}$ OUTPUT LOW	0 = NORMAL 1 = RESET ERROR FLAG IN STATUS REG (FE, OE, PE/DLE DETECT)	**ASYNCH: FORCE BREAK** 0 = NORMAL 1 = FORCE BREAK **SYNCH: SEND DLE** 0 = NORMAL 1 = SEND DLE	0 = DISABLE 1 = ENABLE	0 = FORCE $\overline{\text{DTR}}$ OUTPUT HIGH 1 = FORCE $\overline{\text{DTR}}$ OUTPUT LOW	0 = DISABLE 1 = ENABLE

In asynchronous mode, setting CR3 forces and holds the TxD output low (spacing condition) at the end of the current transmitted character. Normal operation resumes when CR3 is cleared. The TxD line goes 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. CR3 should be reset in response to the next TxRDY.

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.

The PCI 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 PCI in the Automatic Echo mode. Clocked, regenerated, received data is automatically directed to the TxD line while normal receiver operation continues. The reciever 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 is automatically placed in the Transmit Holding register and retransmitted by the transmitter on the TxD output.

2. Transmit clock = receive clock.

3. $\overline{\text{TxRDY}}$ output = 1.

4. The $\overline{\text{TxEMT}}/\overline{\text{DSCHG}}$ pin reflects only the data set change condition.

5. The TxEN command (CR0) is ignored.

In synchronous mode, CR7-CR6 = 01 places the PCI 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 = 0), characters in the data stream matching SYN1 are not transferred to the Receive Data Holding Register (RHR).

2. In the non-transparent, single 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. However, only the first SYN1 of an SYN1-SYN1 pair is stripped.

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 CTS.

3. Receive clock ← transmit clock.

4. The $\overline{DTR}$, $\overline{RTS}$ and $\overline{TxD}$ outputs are held high.

5. The $\overline{CTS}$, $\overline{DCD}$, $\overline{DSR}$ and $\overline{RxD}$ 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 PCI.

The second diagnostic mode is the Remote Loop Back mode (CR7-CR6 = 11). In this mode:

1. Data assembled by the receiver is automatically placed in the Transmit Holding Register and retransmitted by the transmitter on the TxD output.

2. Transmit ← receive clock.

3. No data is sent to the local CPU, but the error status conditions (PE, OE, FE) are set.

4. The $\overline{RxRDY}$, $\overline{TxRDY}$, and $\overline{TxEMT/DSCHG}$ 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 4-19) indicate receiver and transmitter conditions and modem/data set status.

Table 4-19. Status Register (SR)

SR7	SR6	SR5	SR4	SR3	SR2	SR1	SR0
Data Set Ready	**Data Carrier Detect**	**FE/SYN Detect**	**Overrrun**	**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	**ASYNCH:** 0 = NORMAL 1 = FRAMING ERROR **SYNCH:** 0 = NORMAL 1 = SYN CHAR DETECTED	0 = NORMAL 1 = OVERRUN ERROR	**ASYNCH:** 0 = NORMAL 1 = PARITY ERROR **SYNCH:** 0 = NORMAL 1 = PARITY ERROR OR DLE CHAR RECEIVED	0 = NORMAL 1 = CHANGE IN $\overline{DSR}$ OR $\overline{DCD}$, OR TRANSMIT SHIFT REGISTER IS EMPTY	0 = RECEIVE HOLDING REG EMPTY 1 = RECEIVE HOLDING REG HAS DATA	0 = TRANSMIT HOLDING REG BUSY 1 = TRANSMIT HOLDING REG EMPTY

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 Loop Back modes are programmed. When this bit is set, the $\overline{TxRDY}$ 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, $\overline{\text{RxRDY}}$ output is low.

The TxEMB/DSCHG bit, SR2, when set, indicates either a change of state of $\overline{\text{DSR}}$ or $\overline{\text{DCD}}$ inputs 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. When SR2 is set, the TxEMT/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 the DLE Register has been received. However, only the first DLE of two successive DLEs will be SR3. This bit is cleared when the receiver is disabled and by the 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 and by the Reset Error command, CR4.

In asynchrounous mode, bit SR5 signifies that the received character was not framed by the programmed number of stop bits. (If 1.5 stop bits are programmed, 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 is 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 asyncrhonous mode, and 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.

Timing diagrams for the various PCI operations are shown in Figure 4-15.

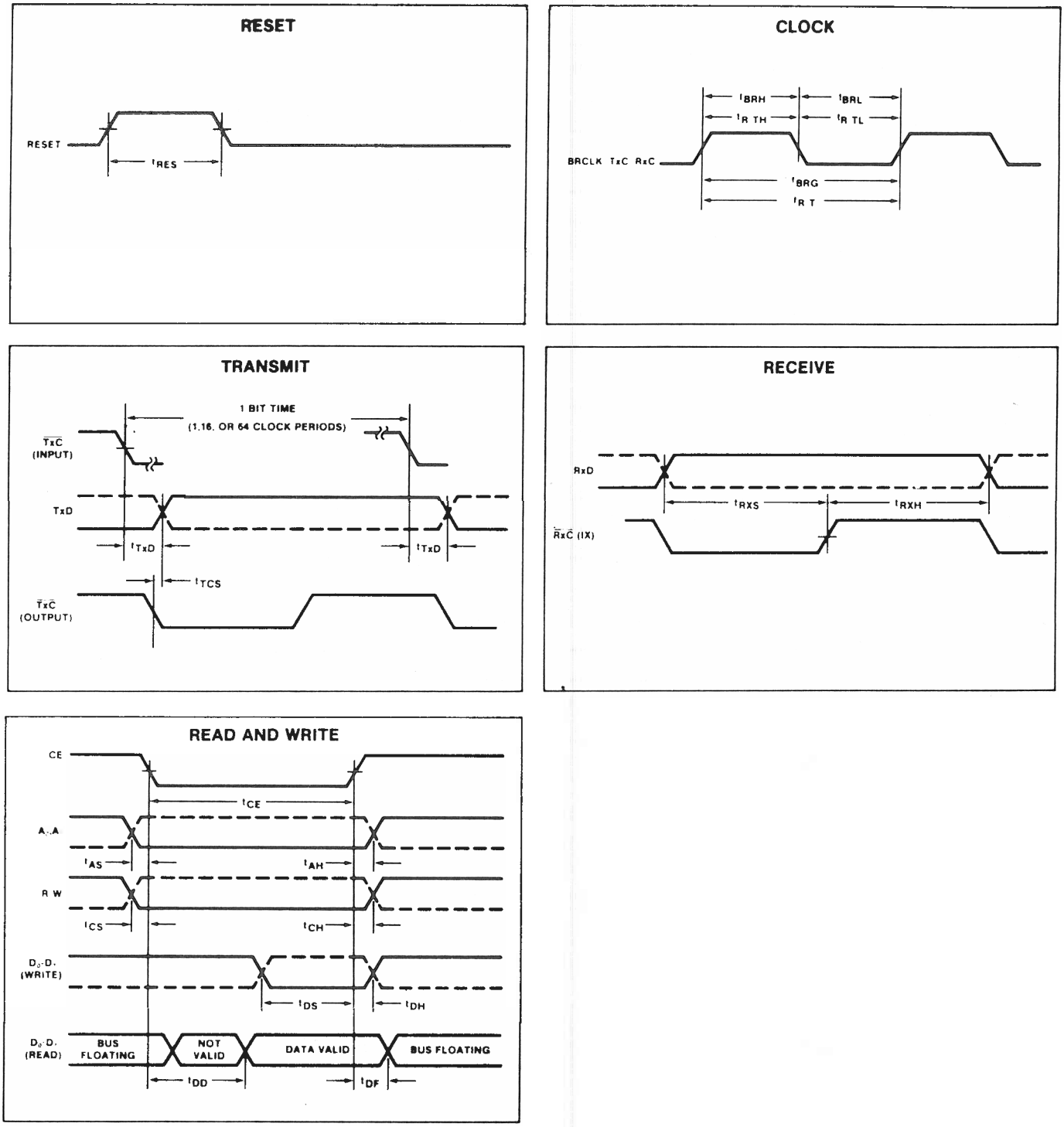


Figure 4-15. PCI Timing (Sheet 1 of 3)

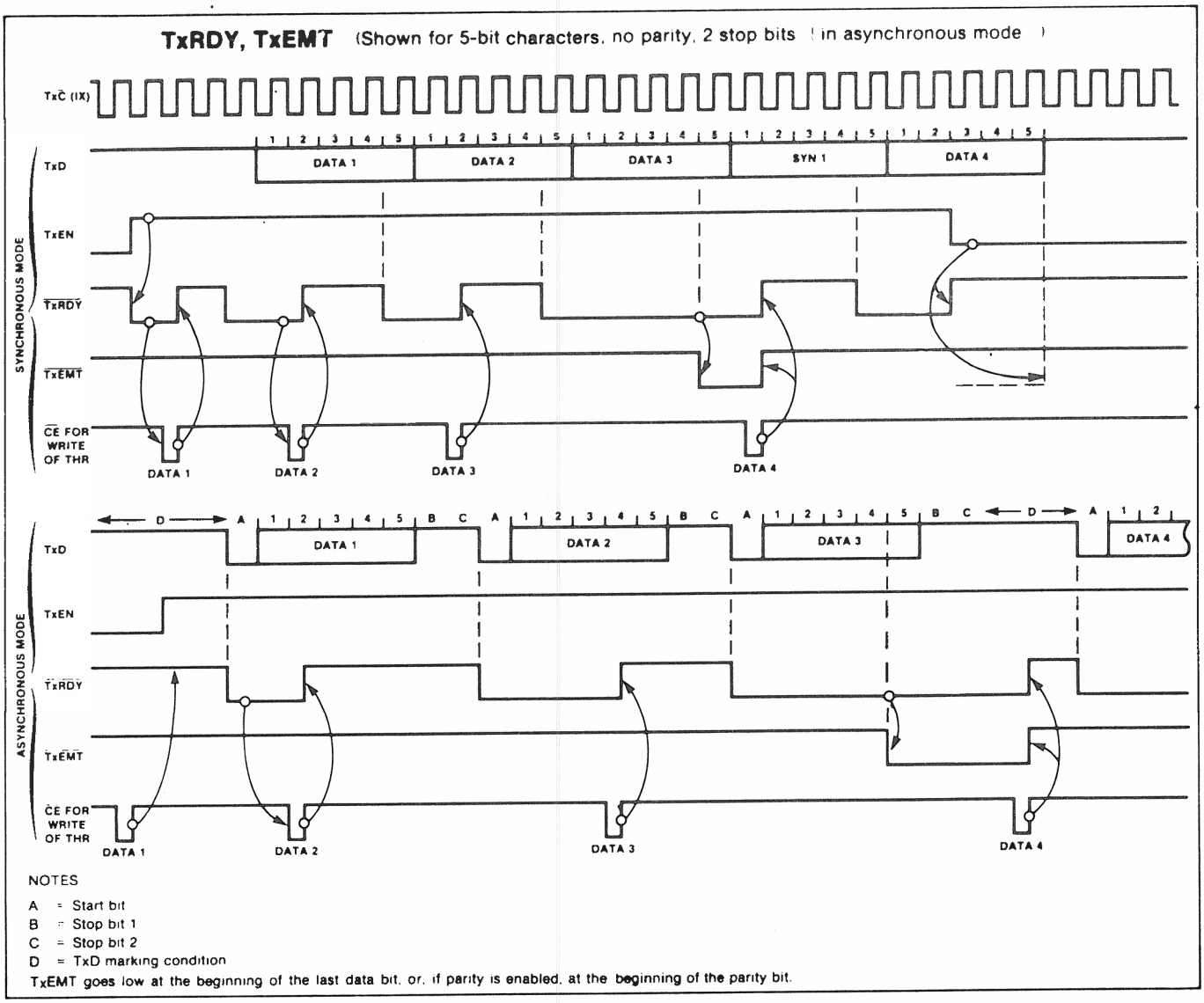


Figure 4-15. PCI Timing (Sheet 2 of 3)

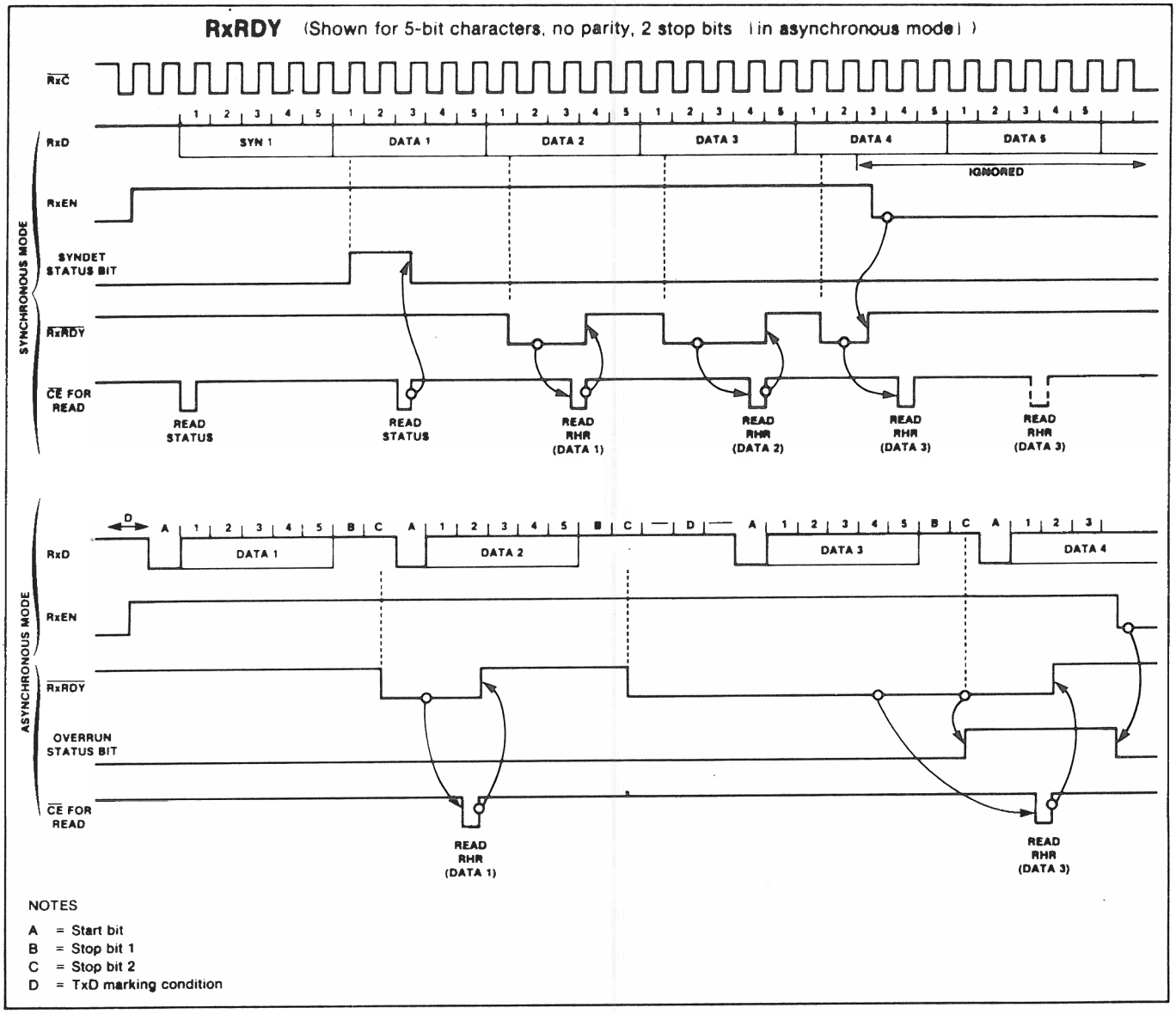


Figure 4-15. PCI Timing (Sheet 3 of 3)

4.2.4 16K (2K x 8) UV ERASABLE PROM (U33).

This device is a 16,384 bit ultraviolet erasable and electrically programmable read-only memory (EPROM). For logic and connections, see Figure 4-16.

Connection Diagram

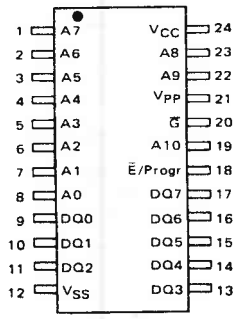


Figure 4-16. 16K UV Erasable Prom Connections

4.2.4.1 ERASURE CHARACTERISTICS.

The erasure characteristics of this device are such that erasure begins to occur when exposed to light with wavelengths shorter than approximately 4000 Angstroms. It should be noted that sunlight and certain types of fluorescent lamps have wavelengths in the 3000-4000A range. Data show that constant exposure to room level fluorescent lighting could erase the typical device in approximately 3 years, while it would take approximately 1 week to cause erasure when exposed to direct sunlight. If the device is to be exposed to these types of lighting conditions for extended periods of time, opaque labels are available which should be placed over the window to prevent unintentional erasure.

4.2.4.2 DEVICE OPERATION.

The five modes of operation of the device are listed in Table 4-20. It should be noted that all inputs for the five modes are at TTL levels. The power supplies required are a +5V VCC and a Vpp. The Vpp power supply must be at 25V during the three programming modes, and must be at 5V in the other two modes.

Table 4-20. 16K UV Erasable Prom Modes

Mode	PIN NUMBER					
	9–11, 13–17 DQ	12 V_{SS}	18 $\overline{E}$/Progr	20 $\overline{G}$	21 V_{PP}	24 V_{CC}
Read	Data out	V_{SS}	V_{IL}	V_{IL}	V_{CC}	V_{CC}
Output Disable	Hi Z	V_{SS}	Don't Care	V_{IH}	V_{CC}	V_{CC}
Standby	Hi Z	V_{SS}	V_{IH}	Don't Care	V_{CC}	V_{CC}
Program	Data in	V_{SS}	Pulsed V_{IL} to V_{IH}	V_{IH}	V_{IHP}	V_{CC}
Program Verify	Data out	V_{SS}	V_{IL}	V_{IL}	V_{IHP}	V_{CC}
Program Inhibit	Hi Z	V_{SS}	V_{IL}	V_{IH}	V_{IHP}	V_{CC}

Read Mode. The device has two control functions, both of which must be logically satisfied in order to obtain data at the outputs. Chip Enable (CE) is the power control and should be used for device selection. Assuming that addresses are stable, address access time (tACC) is equal to the delay from CE to output (tCE). Data is available at the outputs 120ns (tOE) after the falling edge of OE, assuming that CE has been low and addresses have been stable for at least tACC - tOE.

Standby Mode. The device has a standby mode which reduces the active power dissipation by 75 , from 525 mW to 132 mW. The device is placed in the standby mode by applying a TTL high signal to the CE input. When in standby mode, the outputs are in a high impedance state, independent of the OE input.

Output Deselection. The outputs of two or more devices may be OR-tied together on the same data bus. Only one should have its output selected ($\overline{OE}$ low) to prevent data bus contention between devices in this configuration. The outputs of the others should be deselected by raising the OE input to a TTL high level.

Programming. Initially, and after each erasure, all bits of the device are in the "1" state. Data is introduced by selectively programming "0's" into the desired bit locations.

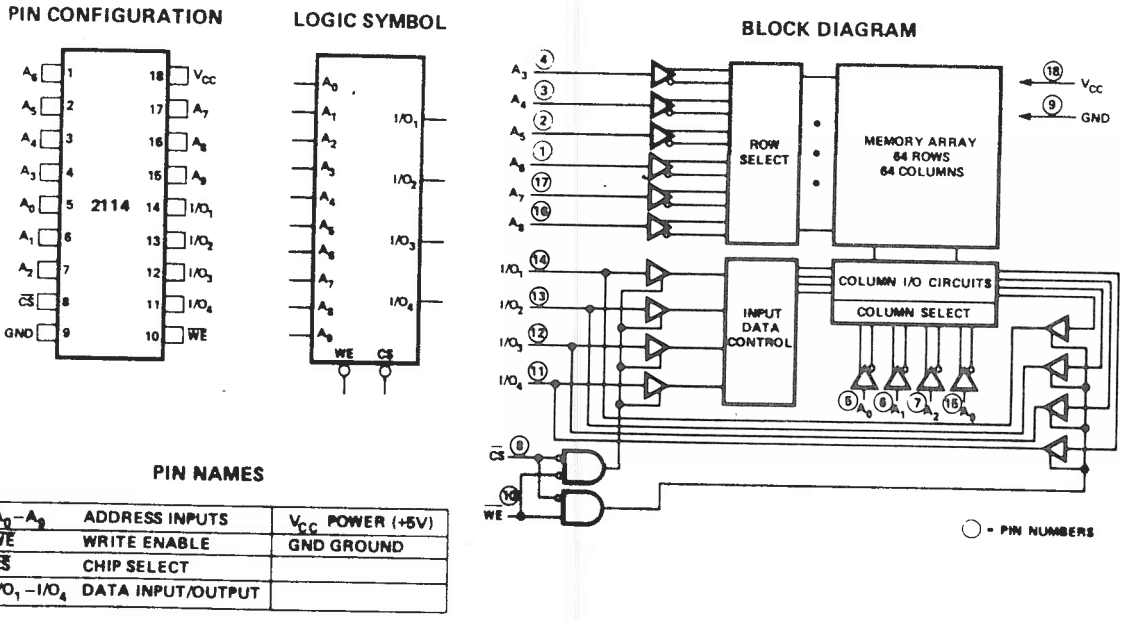
Although only "0's" will be programmed, both "1's" and "0's" can be presented in the data word. The only way to change a "0" to a "1" is by ultraviolet light erasure.

The device is in the programming mode when the Vpp power supply is at 25V and $\overline{OE}$ is at VIH. The data to be programmed is applied 8 bits in parallel to the data output pins. The level required for the address and data inputs are TTL.

When the address and data are stable, a 50 msec, active high, TTL program pulse is applied to the $\overline{CE}$/PGM input. A program pulse must be applied at each address location to be programmed. You can program any location at any time -- either individually, sequentially, or at random. The program pulse has a maximum width of 55 msec. The device must not be programmed with a DC signal applied to the $\overline{CE}$/PGM input.

4.2.5 1024 x 4 BIT STATIC RANDOM ACCESS MEMORY (U34-U37).

This device is a 4096 bit static Random Access Memory organized as 1024 words by 4 bits and requires no clocks or refreshing to operate. The data is read out non-destructively and has the same polarity as the input data. Common input/output pins are provided and a separate chip select ($\overline{CS}$) lead allows easy selection of an individual package. For logic and connections, see Figure 4-17.



PIN NAMES

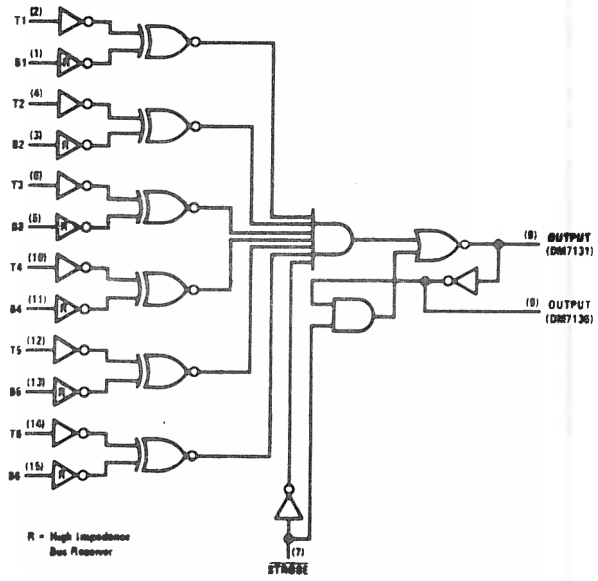
A_0-A_9	ADDRESS INPUTS	V_{CC}	POWER (+5V)
$\overline{WE}$	WRITE ENABLE	GND	GROUND
$\overline{CS}$	CHIP SELECT		
$I/O_1-I/O_4$	DATA INPUT/OUTPUT		

Figure 4-17. RAM Connections

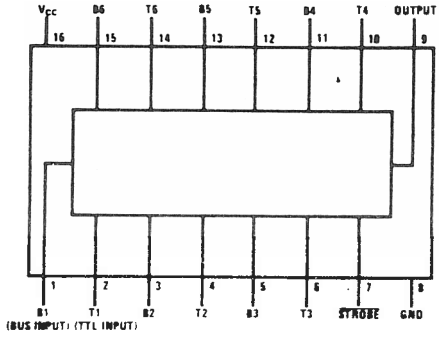
4.2.6 BUS COMPARATOR (U26)

This device compares two binary words of two to six bits in length and indicates matching (bit-for-bit) of the two words. Inputs for one word are TTL inputs, whereas inputs of the second word are high impedance receivers driven by a terminated data bus. The information to the output occurs when the $\overline{\text{STROBE}}$ input goes from a logical 1 to a logical 0 state. Inputs may be changed while the $\overline{\text{STROBE}}$ is at the logical 1 level, without affecting the state of the output. For logic and connections, see Figure 4-18.

Logic Diagram



Connection Diagram



Truth Table

CONDITION	$\overline{\text{STROBE}}$	OUTPUT	
		DM71/8131	DM71/8136
T = B, T ≠ B	H	Q_{N-1}*	Q_{N-1}*
T = B	L	L	H
T ≠ B	L	H	L

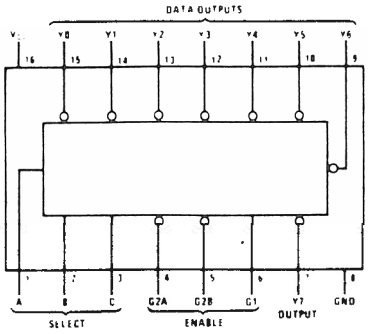
*Latched in previous state

Figure 4-18. Bus Comparator Connections

4.2.7 DECODER (U16, U17)

These are Schottky-clamped circuits designed for memory-decoding or data-routing applicatins requiring very short propagation delay times. This DIP decodes one of eight lines, based on the conditions at the three binary select inputs and the three enable inputs. For logic and connections, see Figure 4-19.

Logic and Connections



Truth Table

INPUTS					OUTPUTS							
ENABLE		SELECT										
G1	G2*	C	B	A	Y0	Y1	Y2	Y3	Y4	Y5	Y6	Y7
X	H	X	X	X	H	H	H	H	H	H	H	H
L	X	X	X	X	H	H	H	H	H	H	H	H
H	L	L	L	L	L	H	H	H	H	H	H	H
H	L	L	L	H	H	L	H	H	H	H	H	H
H	L	L	H	L	H	H	L	H	H	H	H	H
H	L	L	H	H	H	H	H	L	H	H	H	H
H	L	H	L	L	H	H	H	H	L	H	H	H
H	L	H	L	H	H	H	H	H	H	L	H	H
H	L	H	H	L	H	H	H	H	H	H	L	H
H	L	H	H	H	H	H	H	H	H	H	H	L

*G2 = G2A + G2B
H = High level, L = low level, X = don't care

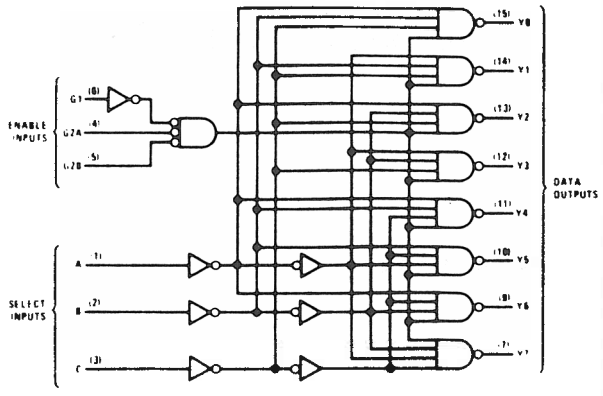
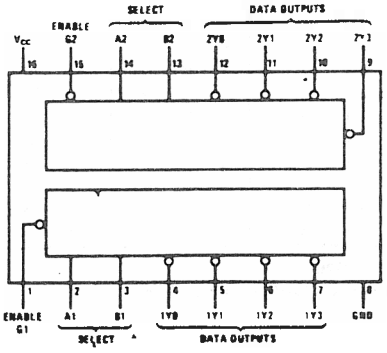


Figure 4-19. Decoder Connections

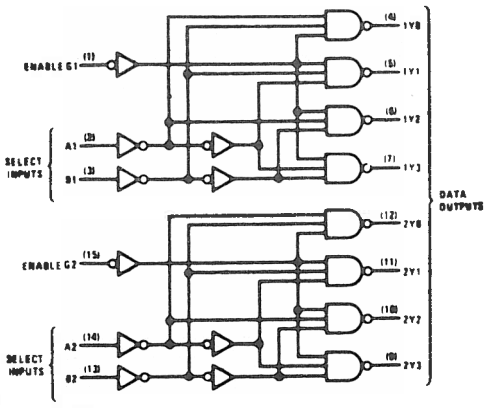
4.2.8 DECODER/DEMULTIPLEXER (U4)

These Schottky-clamped circuits are designed to be used in high performance memory decoding or data routing applications requiring very short propagation delay times. This device contains two separate two-line to four-line decoders in one package. The active-low enable input can be used as a data line in demultiplexing applications. The device features fully buffered inputs, presenting only one normalized load to its driving circuit. All inputs are clamped with high performance Schottky diodes to suppress line-ringing. For logic and connections, see Figure 4-20.

Connection Diagram



Logic Diagram



Truth Table

INPUTS			OUTPUTS			
ENABLE	SELECT					
G	B	A	Y0	Y1	Y2	Y3
H	X	X	H	H	H	H
L	L	L	L	H	H	H
L	L	H	H	L	H	H
L	H	L	H	H	L	H
L	H	H	H	H	H	L

Figure 4-20. Decoder/Demultiplexer

4.2.9 D POSITIVE-EDGE-TRIGGERED FLIP-FLOPS WITH PRESET AND CLEAR (U5)
For logic and connections see Figure 4-21.

Truth Table

INPUTS				OUTPUTS	
PR	CLR	CLK	D	Q	$\bar{Q}$
L	H	X	X	H	L
H	L	X	X	L	H
L	L	X	X	H*	H*
H	H	↑	H	H	L
H	H	↑	L	L	H
H	H	L	X	Q0	$\bar{Q}$0

Connection Diagram

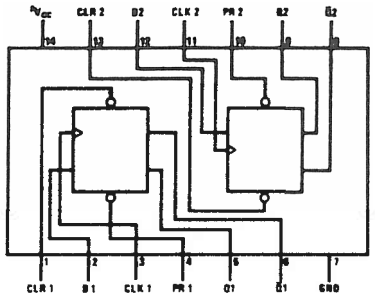
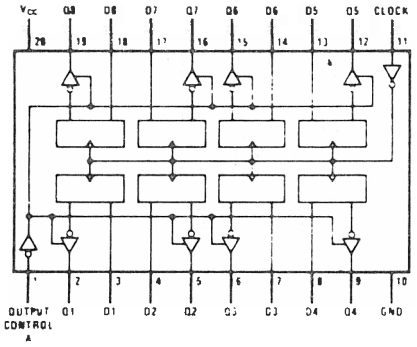


Figure 4-21. D Flip-Flop Connections

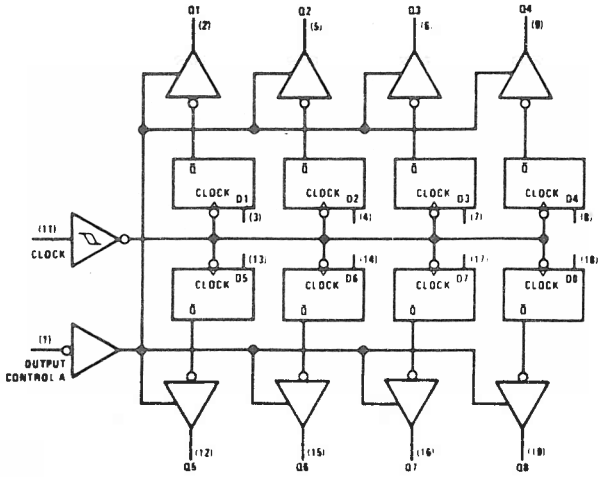
4.2.10 TRI-STATE D FLIP-FLOPS (U31, U38-U41).

These 8-bit registers contain D-type flip-flops with totem-pole tri-state outputs capable of driving highly capacitive or low impedance loads. When the output control is taken to a high logic level, the outputs go into the high impedance state. When a low logic level is applied to the output control, data at the D inputs are loaded into their respective flip-flops on the next positive-going transition of the clock. For logic and connections, see Figure 4-22.

Connection Diagram



Logic Diagram



Truth Table

OUTPUT CONTROL	CLOCK	D	OUTPUT
L	↑	H	H
L	↑	L	L
L	L	X	Q0
H	X	X	Z

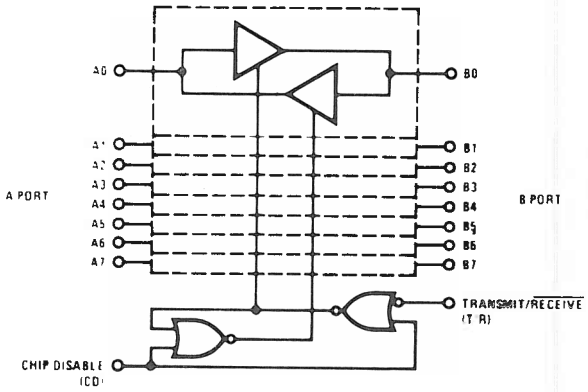
Figure 4-22. Tri-State D Flip-Flop Connections

4.2.11 OCTAL THREE-STATE BIDIRECTIONAL TRANSCEIVER (U24)

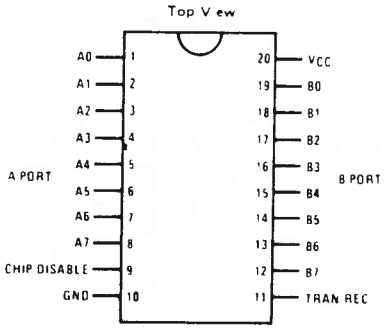
This device is an 8-bit, three-state Schottky transceiver that provides bidirectional drive for bus-oriented microprocessor and digital communications systems.

One input, Transmit/$\overline{\text{Receive}}$, determines the direction of logic signals through the bidirectional transceiver. The chip disable input disables both A and B ports by placing them in a three-state condition. For logic and connections, see Figure 4-23.

Logic Diagram



Connection Diagram



Logic Table

Inputs		Resulting Conditions	
Chip Disable	Transmit/$\overline{\text{Receive}}$	A Port	B Port
0	0	OUT	IN
0	1	IN	OUT
1	X	TRI STATE	TRI STATE

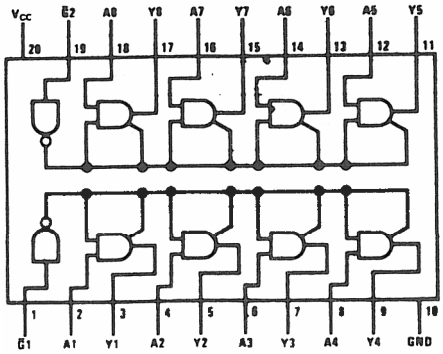
X Don't Care

Figure 4-23. Bidirectional Transceiver Connections

4.2.12 TRI-STATE OCTAL BUFFERS (U15)

This device provides eight 2-input buffers in each package and employs Schottky TTL Technology. One of the two inputs to each buffer is used as a control line to gate the output to the high-impedance state, while the other input passes the data through the buffer. The outputs are placed in the tri-state condition by applying a high logic level to the enable pins. For logic and connections, see Figure 4-24.

Logic and Connections



Truth Table

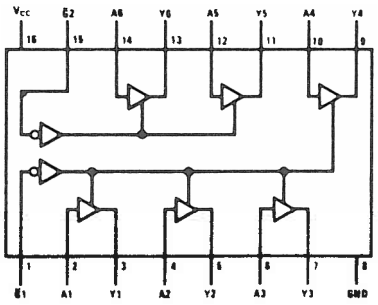
INPUTS		OUTPUT
$\overline{G}$	A	Y
H	X	Z
L	H	H
L	L	L

Figure 4-24. Tri-State Octal Buffer

4.2.13 TRI-STATE HEX BUFFERS (U25)

For logic and connections, see Figure 4-25.

Logic and Connections



Truth Table

INPUTS		OUTPUT
$\bar{G}$	A	Y
H	X	Z
L	H	H
L	L	L

Figure 4-25. Tri-State Hex Buffer Logic and Connections

SECTION V

MAINTENANCE AND TROUBLESHOOTING

5.0 INTRODUCTION.

The AM-310 circuit board performs to full capability with a minimum of maintenance. This section describes maintenance and troubleshooting procedures for handling warranty returns.

5.1 CIRCUIT BOARD CHECKOUT.

The AM-310 circuit board was fully tested before it left Alpha Microsystems and will operate satisfactorily in the system if the hardware and software requirements of Sections II and III of this manual are met. Should a problem arise after the circuit card has been in operation, use the following procedures to identify and locate the fault.

1. Check all cabling for proper seating of connectors.

2. Check the circuit board for proper seating in the slot.

3. Check all power connections for correct voltages.

4. Check jumper options to ensure correctness of application.

5. Verify that the fault is in the AM-310 and not either in the system or in the peripherals. This can best be accomplished with substitution of a known good circuit board.

6. Perform the diagnostic tests. These tests may be obtained from the Alpha Micro International Support Services Group.

5.2 WARRANTY PROCEDURES.

This circuit board is covered by warranty issued by Alpha Microsystems, Inc., Irvine, California. Complete details of the warranty are included with the circuit board. Should a problem arise with this circuit board, call your dealer or the Alpha Micro International Support Services Group for information.

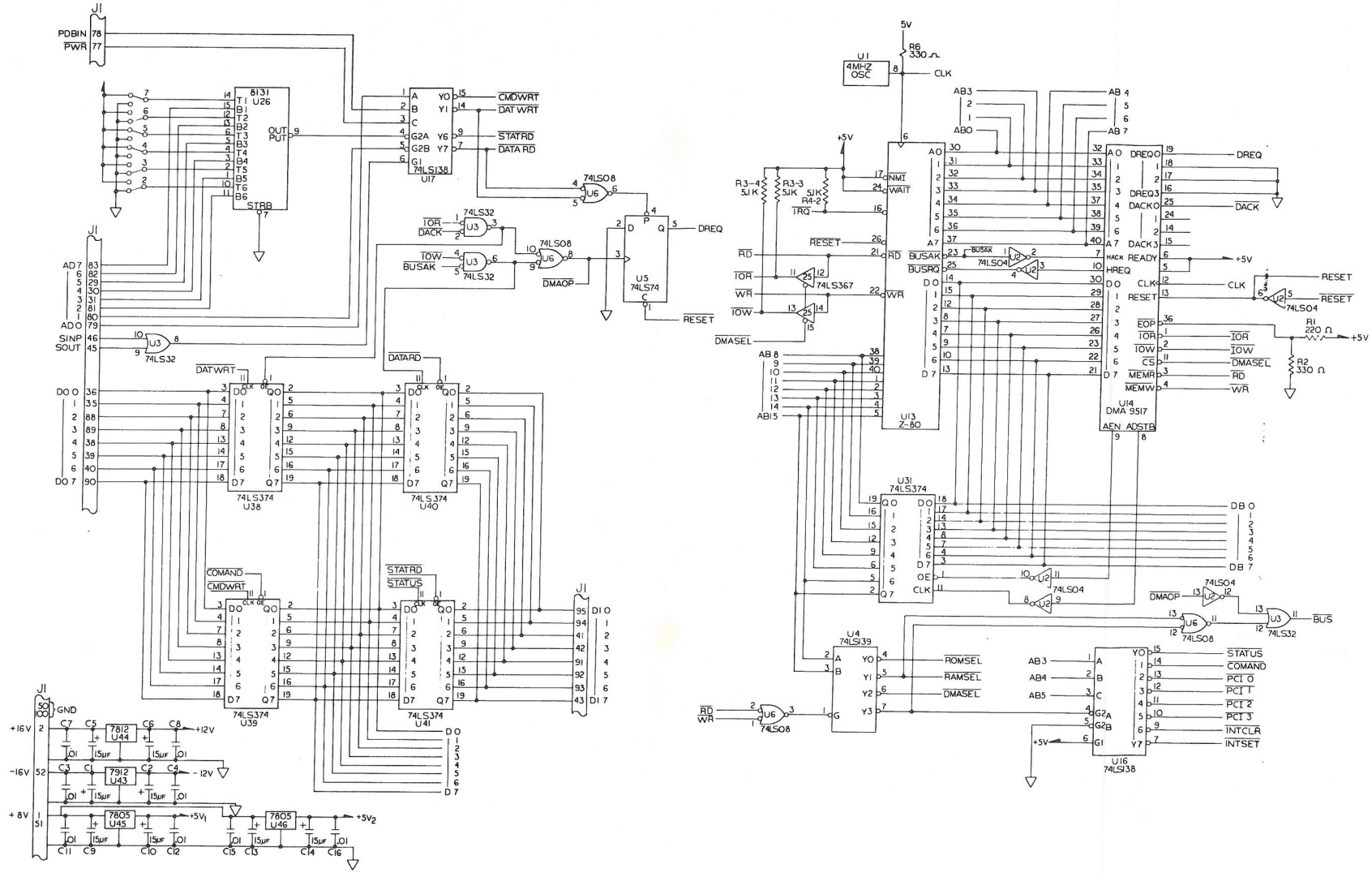
5.3 TROUBLESHOOTING PROCEDURES.

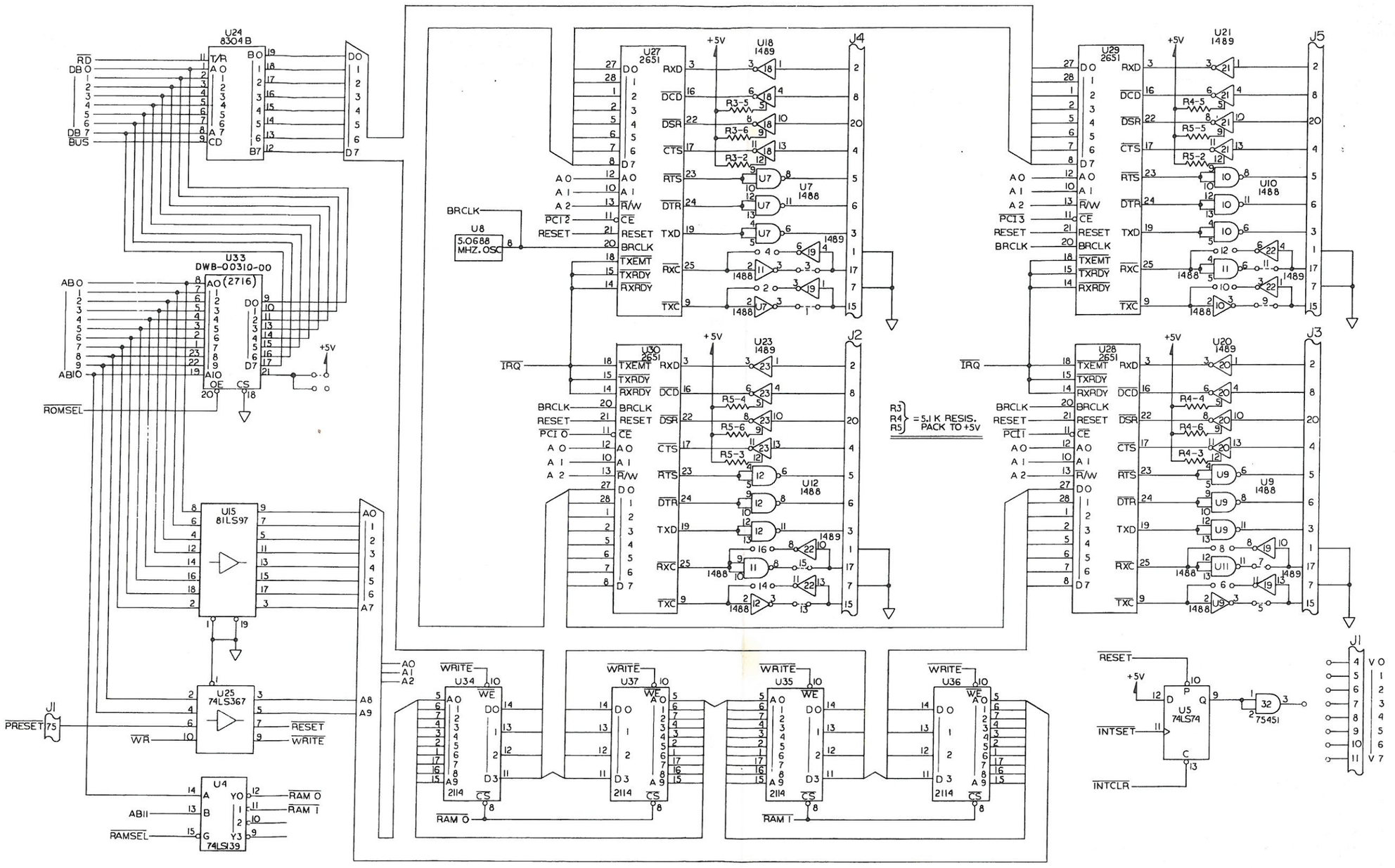
Diagnostic testing software should be used for troubleshooting and to verify proper operation of your of your AM-310. See Section III (Paragraph 3.5.7) for testing procedures. The diagnostic tests may be obtained from the Alpha Micro International Support/Services Group.

Table 6-1. AM-310 Component Cross-Reference List

REF DESIG	MFG TYPE NO.	PAR. NO.	REF DESIG	MFG TYPE NO.	PAR NO.
U1	4 MHz Osc.	-	U24	8304	4.2.11
U2	74LS04	-	U25	74LS367	4.2.13
U3	74LS32	-	U26	8131	4.2.6
U4	74LS139	4.2.8	U27	2651	4.2.3
U5	74LS74	4.2.9	U28	2651	4.2.3
U6	74LS08	-	U29	2651	4.2.3
U7	1488	-	U30	2651	4.2.3
U8	5.0688 MHz	-	U31	74LS374	4.2.3
U9	1488	-	U32	75451	-
U10	1488	-	U33	DWB-00311-01	4.2.4
U11	1488	-	U34	2114	4.2.5
U12	1488	-	U35	2114	4.2.5
U13	Z80	4.2.1	U36	2114	4.2.5
U14	95LS17	4.2.2	U37	2114	4.2.5
U15	81LS97	4.2.12	U38	74LS374	4.2.10
U16	74LS138	4.2.7	U39	74LS374	4.2.10
U17	74LS138	4.2.7	U40	74LS374	4.2.10
U18	1489	-	U41	74LS374	4.2.10
U19	1489	-			
U20	1489	-			
U21	1489	-			
U22	1489				
U23	1489				

.....ASSEMBLY..... NUMBER	DESCRIPTION	.....COMPONENT..... REC#	NUMBER	DESCRIPTION		ASSY. BIN#	QUANTITY
DWB0031000	ASSY 4 PORT COMM CONT	AM-310 C06 630					
		00001	DWF0031000	PCB 4 PORT COMM CONT	AM-310		1.000
		00002	ICS0265100	IC PROG COMMUNICATIONS I/F			4.000
		00003	DWB0031101	ASSY PROM COMM CONT	AM-310		1.000
		00004	ICI0148900	IC CONV/LINE DRIVER	RS-232		6.000
		00005	ICI0148800	IC CONV/LINE DRIVER	RS-232		5.000
		00006	IC10830400	IC TRANSCEIVER OCTAL			1.000
		00007	IC10819701	IC BUFFER OCTAL			1.000
		00008	ICM0211404	RAM 1K X 4 BIT STATIC			4.000
		00009	IC17436700	IC HEX BUFFER			1.000
		00010	IC17413801	IC DECODER 3 TO 8 LINE			2.000
		00011	IC10740401	IC HEX INVERTER			1.000
		00012	IC17437401	IC OCTAL D FLIPFLOP			5.000
		00013	IC17413901	IC DECODER 2 TO 4 LINE DUAL			1.000
		00014	IC10747401	IC DUAL D FLIPFLOP			1.000
		00015	IC10740801	IC QUAD 2 INPUT AND GATE			1.000
		00016	IC10743201	IC QUAD 2 INPUT OR GATE			1.000
		00017	ICI7545100	IC DUAL INTERFACE DRIVER NI OC			1.000
		00018	IC10813100	IC COMPARATOR 6 BIT			1.000
		00019	ICS0008000	IC MICROPROCESSOR Z-80			1.000
		00020	ICS0951701	IC CHIP DMA			1.000
		00021	ICS0000300	IC OSCILLATOR 4MHZ			1.000
		00022	ICL0780500	IC REGULATOR + 5V			2.000
		00023	ICL0781200	IC REGULATOR +12V			1.000
		00024	ICL0791200	IC REGULATOR -12V			1.000
		00025	CNF0000601	HEADER 26PIN W/O EJCT RT ANGLE UL			4.000
		00026	CNS0000800	SOCKET 8 PIN DIP			1.000
		00027	CNS0001400	SOCKET 14 PIN DIP			15.000
		00028	CNS0001600	SOCKET 16 PIN DIP			5.000
		00029	CNS0001800	SOCKET 18 PIN DIP			4.000
		00030	CNS0002000	SOCKET 20 PIN DIP			7.000
		00031	CNS0002400	SOCKET 24 PIN DIP			1.000
		00032	CNS0002800	SOCKET 28 PIN DIP			4.000
		00033	CNS0004000	SOCKET 40 PIN DIP			2.000
		00034	CPN0010301	CAPACITOR .01 UF			20.000
		00035	CPP0015601	CAPACITOR 15 UF 20V			8.000
		00036	RS20022100	RESISTOR 220 OHM 1/4W 5% CAR			1.000
		00037	RS20033100	RESISTOR 330 OHM 1/4W 5% CAR			2.000
		00038	RSN0000900	RES PACK SIP 4.7 K			3.000
		00039	HDM0000700	HEAT SINK .750WI .375HT .750LG			4.000
		00040	HDS1050606	SCREW,PHMS,SS,PH RECESS, 6-32X3/8		F/S	4.000
		00041	HDN1000006	NUT HEX 6-32 KEP INT/EXT CAD STL		F/S	4.000
		00043	HDM0001200	THERMAL GREASE		F/S	4.000
		00044	ICS0000500	IC OSCILLATOR 5.0688 MHZ			1.000
		00046	CNF0000401	HEADER 8 PIN STRAIGHT			16.000
		00047	DWT0031000	TEST SPEC	AM-310		0.000
		00048	CNA0002100	14 PIN IC SPACER			1.000


J1
PDBIN
PWR
8131 U26
OUTPUT
STRB
74LS138 U17
CMDWRT
DATWRT
STATRD
DATARD
74LS08
74LS32
DMAOP
DREQ
RESET
74LS74
AD7
AD0
SINP
SOUT
DO7
DO0
DATWRT
DATARD
74LS374 U38
74LS374 U40
COMAND
CMDWRT
STATRD
STATUS
74LS374 U39
74LS374 U41
DI7
DI0
GND
+16V
-16V
+8V
7812 U44
7912 U43
7805 U45
7805 U46
+12V
-12V
+5V1
+5V2
U1 4MHZ OSC
R6 330Ω
CLK
5V
Z-80
U13
R3-4 5.1K
R3-3 5.1K
R4-2 5.1K
NMI
WAIT
IRQ
RESET
RD
IOR
WR
IOW
DMASEL
74LS367
BUSAK
BUSRQ
HACK
HREQ
AB8
AB15
U14 DMA 9517
AEN
ADSTB
DREQ0
DREQ3
DACK0
DACK3
READY
CLK
RESET
EOP
CS
MEMR
MEMW
R1 220Ω
R2 330Ω
U31 74LS374
DB0
DB7
U4 74LS139
ROMSEL
RAMSEL
DMASEL
U16 74LS138
STATUS
COMAND
PCI 0
PCI 1
PCI 2
PCI 3
INTCLR
INTSET
BUS


U24
8304B
RD
DB 0
DB 7
BUS
T/R
A 0
A 7
CD
B0
B7
DO
D7
U33
DWB-00310-00
AB 0
AB10
AO (2716)
DO
D7
OE
CS
+5V
ROMSEL
U15
81LS97
A0
A7
U25
74LS367
PRESET
WR
RESET
WRITE
U4
AB11
RAMSEL
YO
Y3
74LS139
RAM 0
RAM 1
A8
A9
A0
A1
A2
BRCLK
U8
5.0688
MHZ. OSC
U27
2651
U18
1489
U7
1488
J4
IRQ
U30
2651
U23
1489
U12
1488
J2
RESET
PCI 0
PCI 2
R/W
CE
TXEMT
TXRDY
RXRDY
RXD
DCD
DSR
CTS
RTS
DTR
TXD
RXC
TXC
R3-5
R3-6
R3-2
R5-4
R5-6
R5-3
R3
R4
R5
= 5.1 K RESIS. PACK TO +5V
U29
2651
U21
1489
U10
1488
J5
PCI 3
U28
2651
U20
1489
U9
1488
J3
PCI 1
R4-5
R5-5
R5-2
R4-4
R4-6
R4-3
U34
U37
U35
U36
WRITE
WE
2114
CS
RAM 0
RAM 1
J1
RESET
+5V
INTSET
INTCLR
U5
74LS74
32
75451
VO
V7

USER COMMENTS

TECHNICAL PUBLICATIONS FILE REFERENCE

FROM: NAME _____

ADDRESS _____

CITY _____

STATE _____ ZIP _____

NOTE: Use this form to communicate any errors, suggested changes, or general comments about this document.

If necessary, call us at:

(714) 957-6076

DOCUMENT: TITLE/NUMBER/REVISION _____

_____

COMMENTS:

FOLD, STAPLE & MAIL