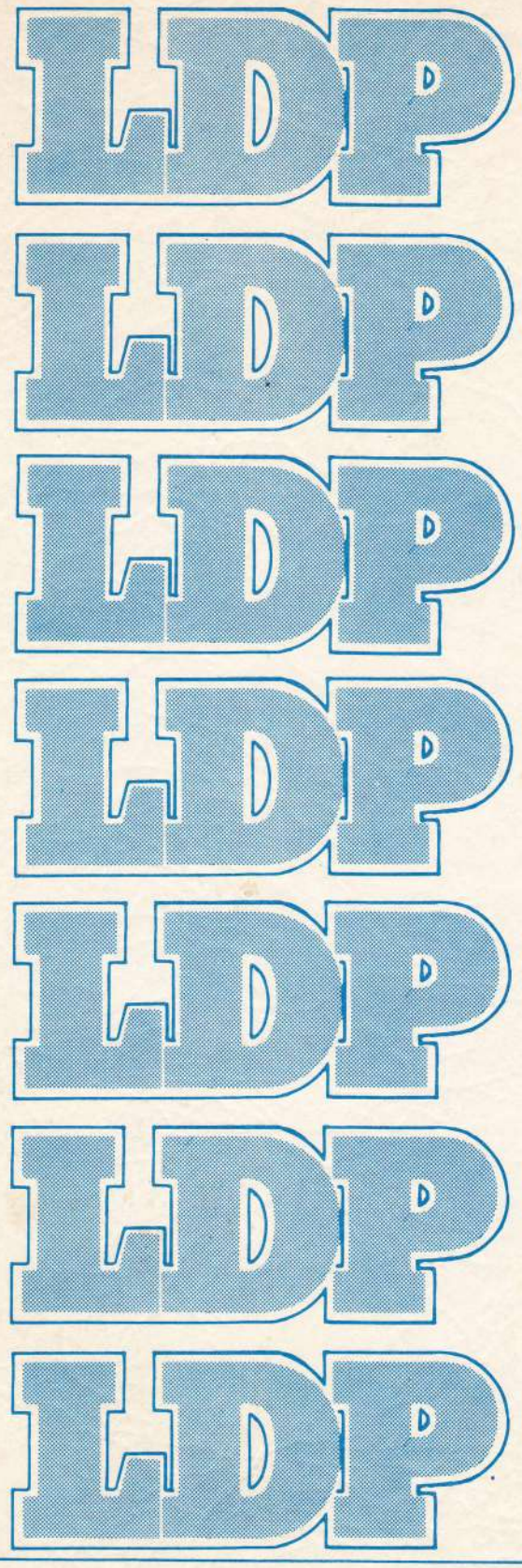


LOMAS DATA PRODUCTS, INC.

HAZITALL OWNER'S MANUAL

REV. 2

PROPRIETARY NOTICE

Lomas Data Products, Inc.
182 Cedar Hill Street
Marlboro, MA 01752
(617) 460-0333

~~4.20V Battery (mercury)~~ (Photo Battery) Duracell PX21 4.5Volts

~~Eveready E133~~

~~Ray-O-Vac T133~~

HAZITALL MANUAL

TABLE OF CONTENTS

HAZITALL MANUAL

1.0 INTRODUCTION

The HAZITALL is a general system support interface. It contains all the I/O functions necessary for a general purpose micro-computer system. It contains the following functions:

Serial Port 1 (asynchrnous)
Serial Port 2 (asynchronous or synchronous)
Strobed Parallel Output Port
Strobed Parallel Input Port
Programmable Real Time Interrupt
Math Processor (option 9511/8231 or 9512/8232)
Battery Backed-up Clock/Calendar
Winchester Disk Controller Support Port

These functions are sufficient in the majority of micro-computer applications. The Winchester Disk Controller support port interfaces to a Western Digital WD1000 Winchester controller. This provides an economical interface to a hard disk without consuming an additional board slot.

2.0 PREPARATION FOR USE

2.1 Introduction

This section provides instructions for preparing the HAZITALL for use in the particular user environment. Before the HAZITALL is installed, the user should be familiar with this entire manual.

2.2 Unpacking and Inspection

Inspect the shipping carton immediately upon receipt for evidence of damage during shipment. If the carton is damaged when received, open the carton in the presence of the carrier's agent. If the carrier's agent is not present, save the packing carton for inspection of the carrier. The HAZITALL is shipped insured to prepaid customers. Any damage incurred during shipment is covered by this insurance. In addition, assembled and tested boards are waranteed for 30 days. If repairs are required, return the HAZITALL in a suitable package to:

LOMAS DATA PRODUCTS, INC.
182 CEDAR HILL STREET
MARLBORO MA 01752

It is suggested that the original packaging carton and material be saved in the event service should be required.

2.3 Installation Considerations

The HAZITALL is designed to be IEEE S100 bus compatible. Be aware that Pins 20,53, and 70 are used for ground. If these grounds conflict with other uses of these pins, they may be cut at the edge connector without adversely affecting its operation.

2.4 User Furnished Components

The HAZITALL uses a 4.5 volt battery to provide power to clock/calendar when 5 volts is not available from the mainframe. The battery is not included with the board. The battery holder is included along with a piece of double sided tape. The battery may be installed anywhere inside your mainframe that is convenient.

In addition, three cables; two serial ports and one parallel port cable, are not provided. These may be purchased seperately from LOMAS DATA PRODUCTS or from a local computer store. All three

cables are the same. A list of the port pinouts is included in the appendix.

2.5 Jumpering

The board address is determined by the 10 position dip switch. The HAZITALL decodes either an 8 or 16 bit address depending on Jumpers, A, B and C on the lower right of the board. With Jumpers A and B connected, 16 bit addressing is selected. With Jumpers B and C connected, 8 bit addressing is selected. Switch 0 corresponds to address bit 7, switch 1 corresponds to address bit 6. Switches 3 through 10 correspond to address bits 8 through 15. When the switch is closed (on), the corresponding address signal must be 0 to match.

The HAZITALL occupies 64 port addresses beginning at the base address selected in the switches. If an 8 bit I/O decode is selected, switch positions 3 through 10 have no effect.

2.5.2 Wait State Generation

The HAZITALL may be jumper selected to provide from 0 to 7 wait states. The number of wait states is jumper selected. The wait state jumpers are labeled with the number of wait states they generate. Table 1 is a list of access times for the peripheral I.C.'s used on the HAZITALL. The number of wait states, if any, necessary for use with your CPU may be determined from the table. The WD1000, clock and math processor generate their own READY without regard to the wait state generator. The usual limiting factor is the width of the READ and WRITE signals (PDBIN and PWR*). The HAZITALL is shipped with one wait state selected. This should be sufficient in most cases. If you are using a CPU with a clock rate faster than 5 MHz, you may have to add more wait states.

2.5.3 Parallel Port Jumpers

A great deal of flexibility has been allowed for use of the parallel ports. The normal mode is for Port A to be a strobed input and Port B to be a strobed output. In this mode, Jumper R-S should be out and jumper T-U should be installed.

Additional jumpering options are allowed in the parallel port jumper group. The pins are numbered from 1 to 20. Pins 1, 2, 3 and 4 determine whether a low true strobe or a high true strobe will latch data into Port A. With Pins 1 to 2 jumpered, a high true strobe will latch data and with Pins 3 to 4 jumpered, a low true strobe will latch data into Port A.

If Port A is not used in MODE 1 (strobed input), the input signal to PC4 is inverted when Pins 1 and 2 are jumpered and not inverted when Pins 3 and 4 are jumpered. If MODE 0 is used for Port A, Pins 9 and 10 should be jumpered. If MODE 1 is used, Pins 11 and 12 should be jumpered.

If MODE 0 is used for Port B, Pins 19 and 20 should be jumpered, otherwise, pins 13 and 15 should be jumpered. In strobed output mode for Port B, either a positive going signal or a negative going signal may be provided for strobing data. When Pins 14 and 16 are jumpered, a negative going signal is provided. When Pins 16 and 18 are jumpered, a positive going signal is provided.

TABLE 1
PERIPHERAL ACCESS TIMES

I.C.	ADDRESS ACCESS TIME	ACCESS TIME FROM PDBIN	MINIMUM CONTROL WIDTH	
			RD	WR
8251A	300	250	250	250
8255A	n/a	250	300	400
8253A	450	400	400	400
Status Port	n/a	50ns	50ns	n/a

2.5.4 2 MHz Clock

Many CPU boards do not generate a 2 MHz clock to drive Pin 49 of the bus. If the 2 MHz clock is not provided elsewhere, it may be provided by jumpering U to W on the HAZITALL.

2.5.5 Serial Port Jumpering

Serial Port may be used in either synchronous or asynchronous mode. When used in synchronous mode, the transmit and receive clocks are provided by the modem. Jumpers, D, E, F, G, H and J allow the selection of the source of the receive and transmit clock. For synchronous mode, Pins D to E and J to H should be jumpered. For use in asynchronous mode, Pins E to F and G to H should be jumpered.

2.5.6 Math Processor Clock Selection

The math processors are available in a high speed version from Intel which run at 4 Mhz. When using standard speed math

processors, a 2 Mhz clock is used. The selection is made by moving the jumper by the math processor to the appropriate end labled either 2 or 4.

3.0 PROGRAMMING

See Table 2 for a list of device addresses. The HAZITALL uses a total of 64 port addresses. The HAZITALL will decode either an 8 bit or 16 bit port address. To select 8 bit decoding, Jumpers B and C should be jumpered. For 16 bit address, decode Jumpers A and B should be jumpered.

3.1 Programmable Timer

The HAZITALL uses an 8253 programmable timer to generate the receive and transmit clocks of the USARTS and to generate a real time interrupt. Counter 0 is used as Serial Port 1's baud rate clock, Counter 1 as Serial Port 2's baud rate clock, and Counter 2 is used as a programmable real time interrupt. A specification sheet for the 8253 is included in the Appendix. All three counters are normally programmed to MODE 3, square wave rate generator. Table 3 shows the necessary divisors for commonally used baud rates. The formula for the calculation of any baud rate is:

divisor = 2.5 * 10E6/baud rate * 16

This formula assumes asynchronous operation of the USARTS in X16 mode.

The counter is programmed by outputting a command word for the particular counter and then outputting the divisor to the counter. The counter command word stays in effect until a new command is sent to the counter command port. All three counter command words may be output before outputting the divisors and the counters may be loaded in any order after the commands are issued. See Figure 7 of the spec sheet for an example output sequence. The Appendix contains a sample program that uses the HAZITALL as a support card in a CP/M-86 system. Counter 2 may be programmed to interrupt at a rate from 2 MHz to 30 HZ. The raising edge of the output clock sets the real time clock flip-flop to request an interrupt. The interrupt service routine should clear the flip-flop by writing any value to the HAZITALL status port.

3.2 8251A USART Programming

The Appendix contains a complete 8251A spec sheet with programming information. A brief description is contained in the following sections.

BASE EQU C0h

TABLE 2

DEVICE ADDRESS

Address	Device
00 + Base	Serial Port 1 Data
01	Serial Port 1 Control
04	Serial Port 2 Data
05	Serial Port 2 Control
08	Parallel Port A
09	Parallel Port B
0A	Parallel Port C
0B	Parallel Port Control
0C	Timer 0
0D	Timer 1
0E	Timer 2
0F	Timer Control
10 D0	Winchester Port 0
11	Winchester Port 1
12	Winchester Port 2
13	Winchester Port 3
14	Winchester Port 4
15	Winchester Port 5
16	Winchester Port 6
17	Winchester Port 7
18	Math Processor Data Stack
19	Math Processor Command Status
1C	HAZITALL Status Port
20 E0	Counter Thousandths of a Second
21	Counter Hundredths and Tenths of a second
22	Counter seconds
23	Counter minutes
24	Counter hours
25	Counter day of week
26	Counter day of month
27	Counter months
28	Latch thousandths of a second
29	Latch hundredths and tenths of a second
2A	Latch seconds
2B	Latch minutes
2C	Latch hours
2D	Latch day of week

2E	Latch day of month
2F	Latch months
30	Interrupt Status Register
31	Interrupt Control Register
32	Counter Reset
33	Latch Reset
34	Status Bit
35	Go Command
36	Stand by Interrupt

TABLE 3

BAUD RATE DIVISORS

Baud Rate	Divisors
110	1420
300	521
600	260
1200	130
2400	65
4800	33
9600	16
19200	8

3.2.1 Mode Initialization

The USART provides for serial to parallel and parallel to serial conversion of data for serial communications. After a reset, hardware or software, both USARTS are ready to accept a mode instruction word. The word defines the following:

1. character length
2. parity enable
3. even/odd parity generation and check
4. baud rate factor (X1, X16, X64)
5. number of stop bits

The HAZITALL has two USARTS, one for Serial Port 1, and one for Serial Port 2. Serial Port 1 should be programmed for asynchronous operation only, while Serial Port 2 may be programmed for either synchronous or asynchronous operation. (See the section on jumpering to determine the proper board jumpering for each mode). The baud rates for each serial port

are generated by the onboard timer. If synchronous mode is used, the baud rate clock is input through the RS232 receivers from the modem device.

The listing in the Appendix initializes each port in asynchronous mode. Port 1 is used as a console interface and Port 2 is used as a serial interface to a printer device.

3.2.2 USART Reset

The USARTS may be reset in either of two ways, through a hardware reset or a software reset. In order to accept a software reset, command (40H), the USART must be in command mode. In order to insure the USART is in command mode, three nulls (00H) may be sent to the USART. This insures that no matter what state the USART was in, it is now in command mode waiting for a command. The RESET may be now sent to the COMMAND/STATUS register followed by a mode word. The USART is now ready to accept commands again. See the spec sheet in the Appendix for the format of the command and mode words.

3.3 Parallel Ports

The HAZITALL uses an 8255A to provide two parallel ports. The normal mode of operation is one strobed output port and one strobed input port. The use of these ports is by no means limited to these modes, but when shipped, the ports are configured for use in these modes. The following sections covers programming of the 8255A as it is used on the HAZITALL. An 8255A spec is included in the Appendix for your reference.

3.3.1 8255A Initialization

After a reset, the 8255A must be initialized before it can be used. Port A is normally configured as a strobed input and Port B as a strobed output with Port C providing the necessary handshake lines to control the data transfers. Ports A and B may be configured as either inputs or outputs in MODE 0. Jumper R-S defines the direction of Port B's buffer. (Jumper IN = Port B input, Jumper Out = Port B output). Jumper T-U defines the direction of Port A's buffer. (Jumper In = Port A input, Jumper Out = Port B output). When Ports A and B are not being used in strobed mode, Port C may be used as inputs and outputs. In this mode, PC0 through PC3 are not connected to the parallel port Connector J2. When the board is used in strobed input and output mode, the mode word is BCH. This defines Port A as a strobed input, Port B as a strobed output and PC6 and PC7 as inputs. PC6 and PC7 in this mode may be used to sense external device status

such as a printer ready signal. Examine the 8255A spec in the Appendix to determine other suitable mode words.

3.4 Calendar Programming

The calendar function on the board uses a National MM58167 CMOS clock/calendar. A connection is provided for a battery to provide battery backed-up operation. A spec and functional description of MM58167 is included in the Appendix. The functional description provides the necessary programming information.

3.5 Math Processor

The math processor is an option. Either an 8231 (9511) or 8232 (9512) may be used in the socket. When the math processor is purchased, a spec sheet is provided for programming information.

3.6 HAZITALL Status Port

There are 11 sources of interrupts on the HAZITALL. When more than one interrupt source is combined on an interrupt request line, the status port may be read to determine the source of the interrupt. Table 4 lists the interrupt sources and their corresponding position in the status register.

The 8231 real time interrupt status is not available and should be connected to its own interrupt request line.

TABLE 4

STATUS REGISTER BIT ASSIGNMENTS

BIT	SOURCE
0	Math processor 8232 1=done 8231 0=done
1	Serial Port 1
2	Serial Port 2
3	Calendar
4	Parallel Port A
5	Parallel Port B
6	Winchester Interrupt Request
7	Winchester Data Request

3.7 Interrupt Selections

The HAZITALL is shipped with the following interrupt assignments:

INTR6	Serial Port 0 Receive Ready
INTR0	Real Time Clock
INTR1	Serial Port 0 Transmit Ready
INTR2	Serial Port 1 Transmit Ready

These assignments may be changed to suit individual needs. Consult the schematics in the Appendix for pin assignments. LDP software assumes the above assignments in our distributed software.

4.0 THEORY OF OPERATION

The design of the HAZITALL is straight forwad and will only be described briefly. Specifications for the LSI devices are included in the Appendix.

4.1 Address Decoding

Either an 8 or 16 bit I/O address may be decoded depending on the position of the shorting plug on Jumpers A, B and C. When B to C are jumpered, only 8 bit address is decoded. When Jumpers A and B are connected, 16 bits of address are decoded by a combination of U1 and U4. The buffering of the data bus and control signals is also shown on Page 1.

4.2 Wait State Generator

The wait state generator consits of U11 and parts of U9, U10 and U14. PSYNC causes the outputs of the serial to parallel to be set to 0. If the board is being selected and a wait state is selected by installing one of the wait state jumpering, the READY line is pulled low causing wait states to be inserted. 0 is used to shift 1's into the serial input register. When the 1 reaches the output to the jumper selected READY is asserted and no more wait states are asserted. When the clock/calendar is selected, 4 wait states are automatically inserted. This allows time for the READY output of the 58167 to become valid.

4.3 Serial Ports and Counter

The serial ports are straight forward interfaces of buffered signals and address decode to 8251A USART's. They both use clocks provided by the programmable counter, 8253, as receive and transmit clocks. Clock 2 of the 8253 is used to generate a real time interrupt. Each raising edge of the clock coming from counter 2 sets 1/2 of a D Blip Flop (U15). The real time INTERRUPT is cleared by writing to the HAZITALL status port.

4.4 Clock/Calendar

The clock/calendar uses a WRITE pulse delayed by one clock cycle. This insures the 100 ns. data setup time to falling edge of the WRITE pulses. The 58167 provides a READY output to insert wait states when used with fast processors.

4.5 Parallel Ports

An 8255A is used to provide two parallel ports. The interface simply uses buffered signals from Page 1 of the schematic. The output drivers are 74LS245 bi-directional transceivers. The direction of the buffers may be changed by removing or installing jumpers. This allows either port to be used as an inport or output port. See the section on jumpering for a full description of jumper options.

HAZITALL PARTS LIST

U1	AM25LS2521 or 74LS688
U2	74LS244
U3	74LS244
U4	74LS266
U5	74LS74
U6	74LS14
U7	10 pos dip sw
U8	74LS08
U9	74LS32
U10	7407
U11	74LS164
U12	74LS244
U13	74LS244
U14	74LS04
U15	74LS74
U16	7406
U17	74LS14
U18	74LS244
U19	74LS138
U20	MM58167
U21	74LS04
U22	74LS32
U23	AMD9511,9512,INTEL8231,8232 (OPTIONAL)
U24	8255A
U25	8251A
U26	8253-5
U27	8251A
U28	74LS245
U29	74LS244
U30	74LS245
U31	74LS245
U32	1488
U33	1489
U34	7406
U35	1488
U36	1489

R1	1K	R6	1K
R2	1K	R7	1K
R3	1K	R8	1K
R4	1K	R9	1K
R5	390	R10	51K

C1	27 pf	C6	6.8 uf 35V
C2	27 pf	C7	.1uf 25V
C3	.1uf 25V	C8	6.8 uf 35V

C4	6.8 uf 35V	C9-C20	.1uf 25V
C5	.1uf 25V	C22-C31	.1uf 25V
C21	.1uf 35V		

VR1	LM323
VR2	7912
VR3	7812

MISCELLANEOUS

	PART NUMBER	QUANTITY
Heat Sink	Thermaloy 6013B	1
Screws	6/32 x 3/8"	4
Nuts	6/32	4
Washer, lock	#6	4
J1 connector	Ansley 609-4007	1
J2,J3,J4 con	Ansley 609-2607	3
Bergstick	2 x 10 65610-420	1
Bergstick	2 x 8 65610-416	1
Bergstick	2 x 3 65610-406	1
Bergstick	1 x 3 65500-403	3
Bergstick	1 x 2 65500-402	3
Berg Mini Jump	65472-002	13
Wire Wrap Pins	Robinson Nugent CP-105-1-T	25

X1	5.00 Mhz
X2	32.768 Khz
CR1-CR2	IN4148
D1	IN4148
14 Pin socket	4
24 Pin socket	3
28 Pin socket	2
40 Pin socket	1

APPENDIX

PIN ASSIGNMENTS FOR J2, J3, AND J4

The following pin assignments are at a DB25 connector assumed to be connected to the corresponding HAZITALL port with a mass terminated ribbon cable. The cables are available from LOMAS DATA PRODUCTS.

SERIAL PORT PIN OUTS(J3 AND J4)

SIGNAL	PIN NUMBER	DIRECTION
GND	1	
TxDATA	2	OUT
RxDATA	3	IN
RTS	4	OUT
CTS	5	IN
DSR	6	IN
GND	7	
DTR	20	OUT
RXCLK	17	IN (J4 ONLY)
TXCLK	15	IN (J4 ONLY)

PARALLEL PORT PINOUT (J2)
PRINTER REQUIRED SIGNALS ONLY

SIGNAL	PIN NUMBER	DIRRECTION
GND	1, 13, 14	
DATA0	9	OUT
DATA1	22	OUT
DATA2	10	OUT
DATA3	23	OUT
DATA4	11	OUT
DATA5	24	OUT
DATA6	12	OUT
DATA7	25	OUT
DATA STROBE	8	OUT
PRINTER BUSH	7	IN

If your application is not a printer reference the achematics and the apecification of the 8255 included in the apendix for the correct pinout.

intel

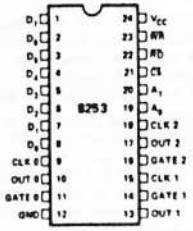
8253/8253-5
PROGRAMMABLE INTERVAL TIMER

- MCS—85™ Compatible 8253-5
- 3 Independent 16-Bit Counters
- DC to 2 MHz
- Programmable Counter Modes
- Count Binary or BCD
- Single +5V Supply
- 24-Pin Dual In-Line Package

The Intel® 8253 is a programmable counter/timer chip designed for use as an Intel microcomputer peripheral. It uses nMOS technology with a single +5V supply and is packaged in a 24-pin plastic DIP.

It is organized as 3 independent 16-bit counters, each with a count rate of up to 2 MHz. All modes of operation are software programmable.

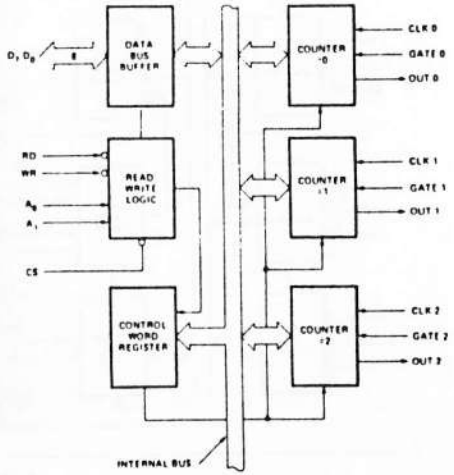
PIN CONFIGURATION



PIN NAMES

D_7-D_0	DATA BUS (8 BIT)
CLK N	COUNTER CLOCK INPUTS
GATE N	COUNTER GATE INPUTS
OUT N	COUNTER OUTPUTS
$\overline{RD}$	READ COUNTER
$\overline{WR}$	WRITE COMMAND OR DATA
$\overline{CS}$	CHIP SELECT
A_0, A_1	COUNTER SELECT
V_{CC}	+5 VOLTS
GND	GROUND

BLOCK DIAGRAM



FUNCTIONAL DESCRIPTION

General

The 8253 is a programmable interval timer/counter specifically designed for use with the Intel™ Microcomputer systems. Its function is that of a general purpose, multi-timing element that can be treated as an array of I/O ports in the system software.

The 8253 solves one of the most common problems in any microcomputer system, the generation of accurate time delays under software control. Instead of setting up timing loops in systems software, the programmer configures the 8253 to match his requirements, initializes one of the counters of the 8253 with the desired quantity, then upon command the 8253 will count out the delay and interrupt the CPU when it has completed its tasks. It is easy to see that the software overhead is minimal and that multiple delays can easily be maintained by assignment of priority levels.

Other counter/timer functions that are non-delay in nature but also common to most microcomputers can be implemented with the 8253.

- Programmable Rate Generator
- Event Counter
- Binary Rate Multiplier
- Real Time Clock
- Digital One-Shot
- Complex Motor Controller

Data Bus Buffer

This 3-state, bi-directional, 8-bit buffer is used to interface the 8253 to the system data bus. Data is transmitted or received by the buffer upon execution of INput or OUTput CPU instructions. The Data Bus Buffer has three basic functions.

1. Programming the MODES of the 8253.
2. Loading the count registers.
3. Reading the count values.

Read/Write Logic

The Read/Write Logic accepts inputs from the system bus and in turn generates control signals for overall device operation. It is enabled or disabled by CS so that no operation can occur to change the function unless the device has been selected by the system logic.

$\overline{RD}$ (Read)

A "low" on this input informs the 8253 that the CPU is inputting data in the form of a counters value.

$\overline{WR}$ (Write)

A "low" on this input informs the 8253 that the CPU is outputting data in the form of mode information or loading counters.

A0, A1

These inputs are normally connected to the address bus. Their function is to select one of the three counters to be operated on and to address the control word register for mode selection.

$\overline{CS}$ (Chip Select)

A "low" on this input enables the 8253. No reading or writing will occur unless the device is selected. The $\overline{CS}$ input has no effect upon the actual operation of the counters.

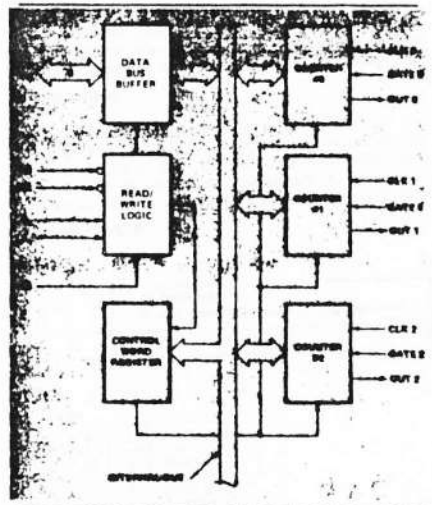


Figure 1. Block Diagram Showing Data Bus Buffer and Read/Write Logic Functions

$\overline{CS}$	$\overline{RD}$	$\overline{WR}$	A_1	A_0	
0	1	0	0	0	Load Counter No. 0
0	1	0	0	1	Load Counter No. 1
0	1	0	1	0	Load Counter No. 2
0	1	0	1	1	Write Mode Word
0	0	1	0	0	Read Counter No. 0
0	0	1	0	1	Read Counter No. 1
0	0	1	1	0	Read Counter No. 2
0	0	1	1	1	No-Operation 3-State
1	X	X	X	X	Disable 3-State
0	1	1	X	X	No-Operation 3-State

8253 READ/WRITE PROCEDURE

Write Operations

The systems software must program each counter of the 8253 with the mode and quantity desired. The programmer must write out to the 8253 a MODE control word and the programmed number of count register bytes (1 or 2) prior to actually using the selected counter.

The actual order of the programming is quite flexible. Writing out of the MODE control word can be in any sequence of counter selection, e.g., counter #0 does not have to be first or counter #2 last. Each counter's MODE control word register has a separate address so that its loading is completely sequence independent (SC0, SC1).

The loading of the Count Register with the actual count value, however, must be done in exactly the sequence programmed in the MODE control word (RL0, RL1). This loading of the counter's count register is still sequence independent like the MODE control word loading, but when a selected count register is to be loaded it must be loaded with the number of bytes programmed in the MODE control word (RL0, RL1). The one or two bytes to be loaded in the count register do not have to follow the associated MODE control word. They can be programmed at any time following the MODE control word loading as long as the correct number of bytes is loaded in order.

All counters are down counters. Thus the value loaded into the count register will actually be decremented. Loading all zeroes into a count register will result in the maximum count (2^{16} for Binary or 10^4 for BCD). In MODE 0 the new count will not restart until the load has been completed. It will accept one of two bytes depending on how the MODE control words (RL0, RL1) are programmed. Then proceed with the restart operation.

	MODE Control Word Counter n
LSB	Count Register byte Counter n
MSB	Count Register byte Counter n

Note: Format shown is a simple example of loading the 8253 and does not imply that it is the only format that can be used.

Figure 6. Programming Format

			A1	A0
No. 1		MODE Control Word Counter 0	1	1
No. 2		MODE Control Word Counter 1	1	1
No. 3		MODE Control Word Counter 2	1	1
No. 4	LSB	Count Register Byte Counter 1	0	1
No. 5	MSB	Count Register Byte Counter 1	0	1
No. 6	LSB	Count Register Byte Counter 2	1	0
No. 7	MSB	Count Register Byte Counter 2	1	0
No. 8	LSB	Count Register Byte Counter 0	0	0
No. 9	MSB	Count Register Byte Counter 0	0	0

Note: The exclusive addresses of each counter's count register make the task of programming the 8253 a very simple matter, and maximum effective use of the device will result if this feature is fully utilized.

Figure 7. Alternate Programming Formats

Read Operations

In most counter applications it becomes necessary to read the value of the count in progress and make a computational decision based on this quantity. Event counters are probably the most common application that uses this function. The 8253 contains logic that will allow the programmer to easily read the contents of any of the three counters without disturbing the actual count in progress.

There are two methods that the programmer can use to read the value of the counters. The first method involves the use of simple I/O read operations of the selected counter. By controlling the A0, A1 inputs to the 8253 the programmer can select the counter to be read (remember that no read operation of the mode register is allowed A0, A1=11). The only requirement with this method is that in order to assure a stable count reading the actual operation of the selected counter must be inhibited either by controlling the Gate input or by external logic that inhibits the clock input. The contents of the counter selected will be available as follows:

first I/O Read contains the least significant byte (LSB).

second I/O Read contains the most significant byte (MSB).

Due to the internal logic of the 8253 it is absolutely necessary to complete the entire reading procedure. If two bytes are programmed to be read then two bytes must be read before any loading WR command can be sent to the same counter.

Read Operation Chart

A1	A0	RD	
0	0	0	Read Counter No. 0
0	1	0	Read Counter No. 1
1	0	0	Read Counter No. 2
1	1	0	Illegal

Reading While Counting

In order for the programmer to read the contents of any counter without effecting or disturbing the counting operation the 8253 has special internal logic that can be accessed using simple WR commands to the MODE register. Basically, when the programmer wishes to read the contents of a selected counter, on the fly, he loads the MODE register with a special code which latches the present count value into a storage register so that its contents contain an accurate, stable quantity. The programmer then issues a normal read command to the selected counter and the contents of the latched register is available.

MODE Register for Latching Count

A0, A1 = 11

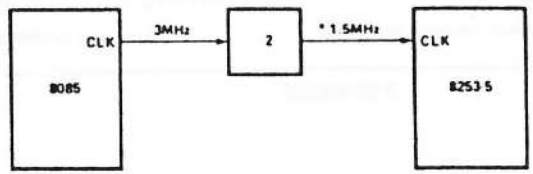
D7	D6	D5	D4	D3	D2	D1	D0
SC1	SC0	0	0	X	X	X	X

SC1, SC0 — specify counter to be latched

D5, D4 — 00 designates counter latching operation

X — don't care

The same limitation applies to this mode of reading the counter as the previous method. That is, it is mandatory to complete the entire read operation as programmed. This command has no effect on the counter's mode.



*If an 8085 clock output is to drive an 8253-5 clock input, it must be reduced to 2 MHz or less.

Figure 8. MCS-85™ Clock Interface*

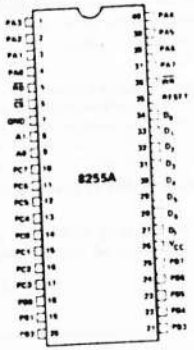
intel

8255A/8255A-5
PROGRAMMABLE PERIPHERAL INTERFACE

- MCS-85™ Compatible 8255A-5
- 24 Programmable I/O Pins
- Completely TTL Compatible
- Fully Compatible with Intel® Microprocessor Families
- Improved Timing Characteristics
- Direct Bit Set/Reset Capability Easing Control Application Interface
- 40-Pin Dual In-Line Package
- Reduces System Package Count
- Improved DC Driving Capability

The Intel® 8255A is a general purpose programmable I/O device designed for use with Intel® microprocessors. It has 24 I/O pins which may be individually programmed in 2 groups of 12 and used in 3 major modes of operation. In the first mode (MODE 0), each group of 12 I/O pins may be programmed in sets of 4 to be input or output. In MODE 1, the second mode, each group may be programmed to have 8 lines of input or output. Of the remaining 4 pins, 3 are used for handshaking and interrupt control signals. The third mode of operation (MODE 2) is a bidirectional bus mode which uses 8 lines for a bidirectional bus, and 5 lines, borrowing one from the other group, for handshaking.

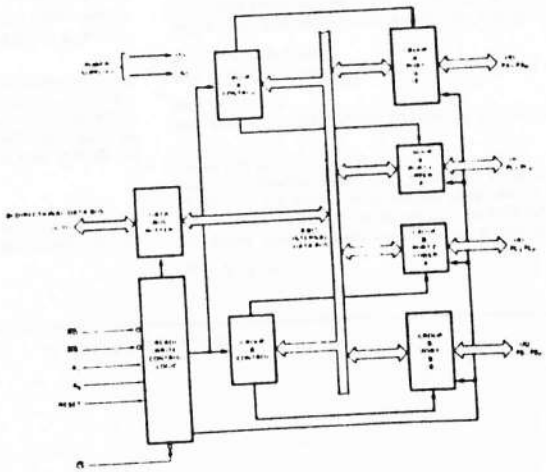
PIN CONFIGURATION



PIN NAMES

D_7-D_0	DATA BUS (BI-DIRECTIONAL)
RESET	RESET INPUT
$\overline{CS}$	CHIP SELECT
$\overline{RD}$	READ INPUT
$\overline{WR}$	WRITE INPUT
A0, A1	PORT ADDRESS
PA7-PA0	PORT A (BIT)
PB7-PB0	PORT B (BIT)
PC7-PC0	PORT C (BIT)
V_{CC}	+5 VOLTS
GND	0 VOLTS

8255A BLOCK DIAGRAM



8255A FUNCTIONAL DESCRIPTION

General

The 8255A is a programmable peripheral interface (PPI) device designed for use in Intel® microcomputer systems. Its function is that of a general purpose I/O component to interface peripheral equipment to the microcomputer system bus. The functional configuration of the 8255A is programmed by the system software so that normally no external logic is necessary to interface peripheral devices or structures.

Data Bus Buffer

This 3-state bidirectional 8-bit buffer is used to interface the 8255A to the system data bus. Data is transmitted or received by the buffer upon execution of input or output instructions by the CPU. Control words and status information are also transferred through the data bus buffer.

Read/Write and Control Logic

The function of this block is to manage all of the internal and external transfers of both Data and Control or Status words. It accepts inputs from the CPU Address and Control busses and in turn, issues commands to both of the Control Groups.

($\overline{CS}$)

Chip Select. A "low" on this input pin enables the communiction between the 8255A and the CPU.

($\overline{RD}$)

Read. A "low" on this input pin enables the 8255A to send the data or status information to the CPU on the data bus. In essence, it allows the CPU to "read from" the 8255A.

($\overline{WR}$)

Write. A "low" on this input pin enables the CPU to write data or control words into the 8255A.

(A_0 and A_1)

Port Select 0 and Port Select 1. These input signals, in conjunction with the RD and WR inputs, control the selection of one of the three ports or the control word registers. They are normally connected to the least significant bits of the address bus (A_0 and A_1).

8255A BASIC OPERATION

A_1	A_0	$\overline{RD}$	$\overline{WR}$	$\overline{CS}$	INPUT OPERATION (READ)
0	0	0	1	0	PORT A → DATA BUS
0	1	0	1	0	PORT B → DATA BUS
1	0	0	1	0	PORT C → DATA BUS
					OUTPUT OPERATION (WRITE)
0	0	1	0	0	DATA BUS → PORT A
0	1	1	0	0	DATA BUS → PORT B
1	0	1	0	0	DATA BUS → PORT C
1	1	1	0	0	DATA BUS → CONTROL
					DISABLE FUNCTION
X	X	X	X	1	DATA BUS → 3-STATE
1	1	0	1	0	ILLEGAL CONDITION
X	X	1	1	0	DATA BUS → 3-STATE

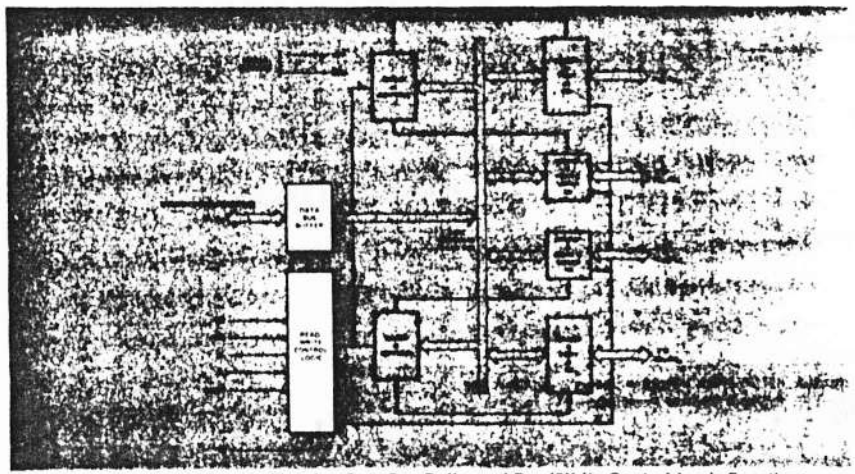


Figure 1. 8255A Block Diagram Showing Data Bus Buffer and Read/Write Control Logic Functions

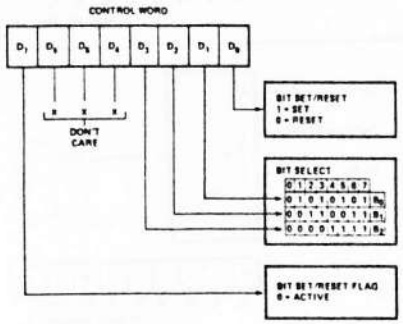


Figure 5. Bit Set/Reset Format

When Port C is being used as status/control for Port A or B, these bits can be set or reset by using the Bit Set/Reset operation just as if they were data output ports.

Interrupt Control Functions

When the 8255A is programmed to operate in mode 1 or mode 2, control signals are provided that can be used as interrupt request inputs to the CPU. The interrupt request signals, generated from port C, can be inhibited or enabled by setting or resetting the associated INTE flip-flop, using the bit set/reset function of port C.

This function allows the Programmer to disallow or allow a specific I/O device to interrupt the CPU without affecting any other device in the interrupt structure.

INTE flip-flop definition:

(BIT-SET) – INTE is SET – Interrupt enable
(BIT-RESET) – INTE is RESET – Interrupt disable

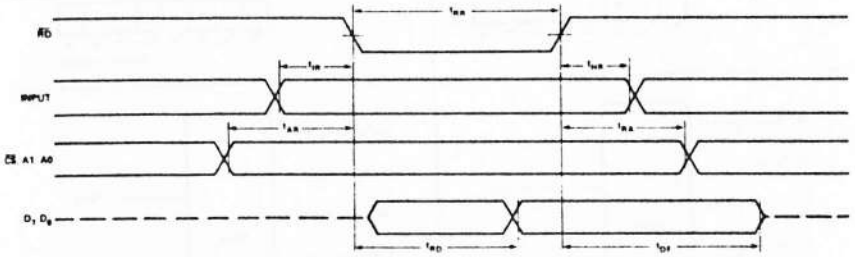
Note: All Mask flip-flops are automatically reset during mode selection and device Reset.

Operating Modes

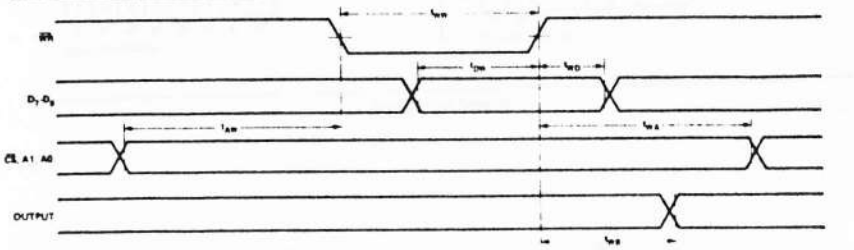
MODE 0 (Basic Input/Output). This functional configuration provides simple input and output operations for each of the three ports. No "handshaking" is required, data is simply written to or read from a specified port.

Mode 0 Basic Functional Definitions:

- Two 8-bit ports and two 4-bit ports.
- Any port can be input or output.
- Outputs are latched.
- Inputs are not latched.
- 16 different Input/Output configurations are possible in this Mode.



MODE 0 (Basic Input)

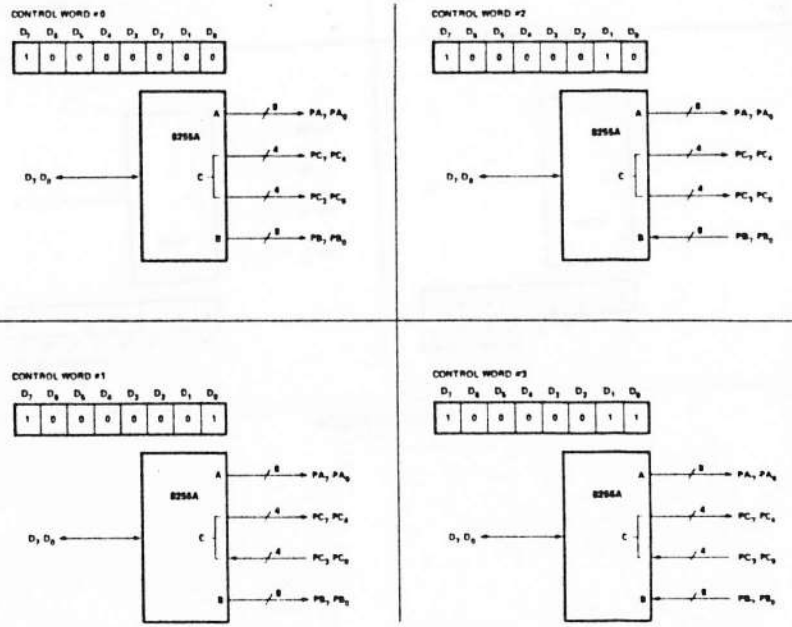


MODE 0 (Basic Output)

MODE 0 Port Definition

A		B		GROUP A		#	GROUP B	
D_4	D_3	D_1	D_0	PORT A	PORT C (UPPER)		PORT B	PORT C (LOWER)
0	0	0	0	OUTPUT	OUTPUT	0	OUTPUT	OUTPUT
0	0	0	1	OUTPUT	OUTPUT	1	OUTPUT	INPUT
0	0	1	0	OUTPUT	OUTPUT	2	INPUT	OUTPUT
0	0	1	1	OUTPUT	OUTPUT	3	INPUT	INPUT
0	1	0	0	OUTPUT	INPUT	4	OUTPUT	OUTPUT
0	1	0	1	OUTPUT	INPUT	5	OUTPUT	INPUT
0	1	1	0	OUTPUT	INPUT	6	INPUT	OUTPUT
0	1	1	1	OUTPUT	INPUT	7	INPUT	INPUT
1	0	0	0	INPUT	OUTPUT	8	OUTPUT	OUTPUT
1	0	0	1	INPUT	OUTPUT	9	OUTPUT	INPUT
1	0	1	0	INPUT	OUTPUT	10	INPUT	OUTPUT
1	0	1	1	INPUT	OUTPUT	11	INPUT	INPUT
1	1	0	0	INPUT	INPUT	12	OUTPUT	OUTPUT
1	1	0	1	INPUT	INPUT	13	OUTPUT	INPUT
1	1	1	0	INPUT	INPUT	14	INPUT	OUTPUT
1	1	1	1	INPUT	INPUT	15	INPUT	INPUT

MODE 0 Configurations



Input Control Signal Definition

STB (Strobe Input). A "low" on this input loads data into the input latch.

IBF (Input Buffer Full F/F)

A "high" on this output indicates that the data has been loaded into the input latch; in essence, an acknowledgement. IBF is set by STB input being low and is reset by the rising edge of the RD input.

INTR (Interrupt Request)

A "high" on this output can be used to interrupt the CPU when an input device is requesting service. INTR is set by the $\overline{STB}$ is a "one", IBF is a "one" and INTE is a "one". It is reset by the falling edge of $\overline{RD}$. This procedure allows an input device to request service from the CPU by simply strobing its data into the port.

INTE A

Controlled by bit set/reset of PC_4.

INTE B

Controlled by bit set/reset of PC_2.

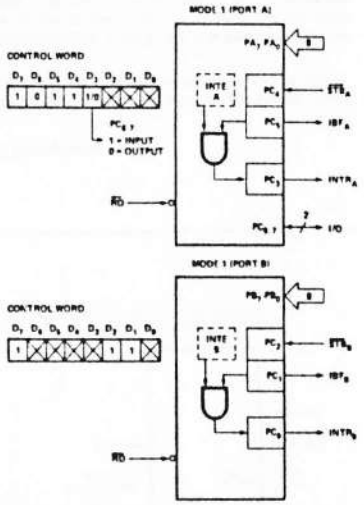


Figure 6. MODE 1 Input

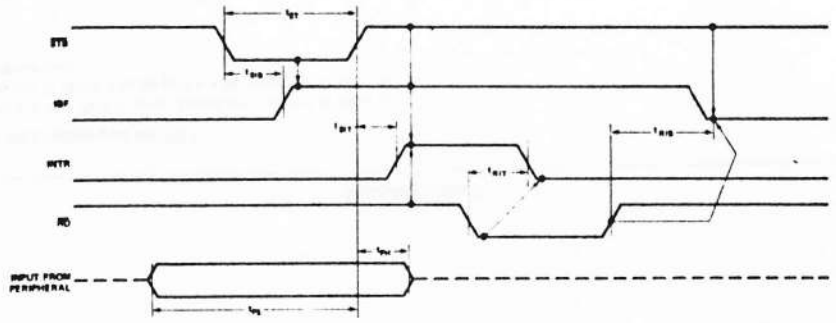


Figure 7. MODE 1 (Strobed Input)

Output Control Signal Definition

$\overline{OBF}$ (Output Buffer Full F/F). The OBF output will go "low" to indicate that the CPU has written data out to the specified port. The OBF F/F will be set by the rising edge of the WR input and reset by ACK input being low.

$\overline{ACK}$ (Acknowledge Input). A "low" on this input informs the 8255A that the data from port A or port B has been accepted. In essence, a response from the peripheral device indicating that it has received the data output by the CPU.

INTR (Interrupt Request). A "high" on this output can be used to interrupt the CPU when an output device has accepted data transmitted by the CPU. INTR is set when ACK is a "one", OBF is a "one" and INTE is a "one". It is reset by the falling edge of WR.

INTE A

Controlled by bit set/reset of PC_6.

INTE B

Controlled by bit set/reset of PC_2.

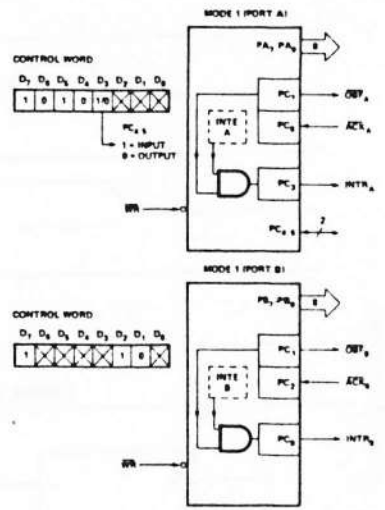


Figure 8. MODE 1 Output

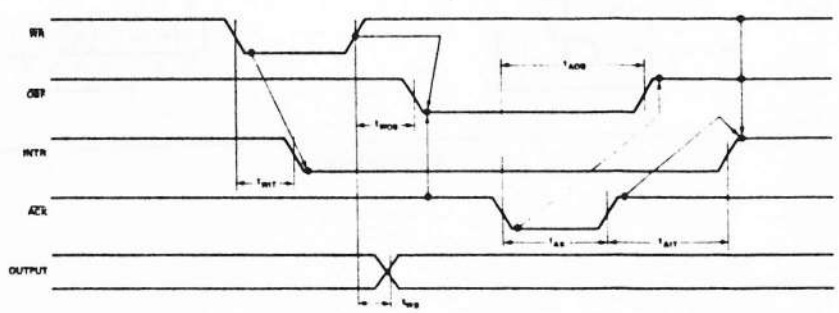


Figure 9. Mode 1 (Strobed Output)

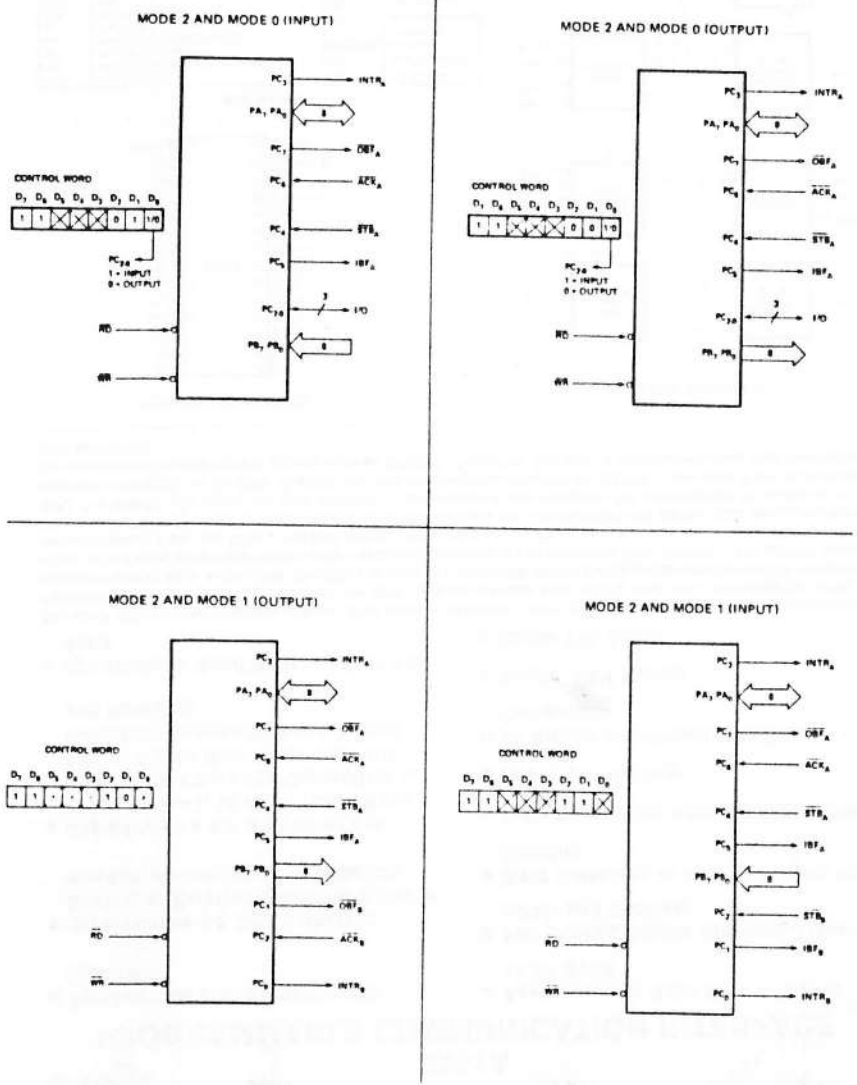


Figure 14. MODE 2 Combinations

Mode Definition Summary

	MODE 0		MODE 1		MODE 2	
	IN	OUT	IN	OUT	GROUP A ONLY	
PA_0	IN	OUT	IN	OUT	⟷	
PA_1	IN	OUT	IN	OUT	⟷	
PA_2	IN	OUT	IN	OUT	⟷	
PA_3	IN	OUT	IN	OUT	⟷	
PA_4	IN	OUT	IN	OUT	⟷	
PA_5	IN	OUT	IN	OUT	⟷	
PA_6	IN	OUT	IN	OUT	⟷	
PA_7	IN	OUT	IN	OUT	⟷	
PB_0	IN	OUT	IN	OUT	—	MODE 0 OR MODE 1 ONLY
PB_1	IN	OUT	IN	OUT	—	
PB_2	IN	OUT	IN	OUT	—	
PB_3	IN	OUT	IN	OUT	—	
PB_4	IN	OUT	IN	OUT	—	
PB_5	IN	OUT	IN	OUT	—	
PB_6	IN	OUT	IN	OUT	—	
PB_7	IN	OUT	IN	OUT	—	
PC_0	IN	OUT	$INTR_B$	$INTR_B$	I/O	
PC_1	IN	OUT	IBF_B	$\overline{OBF}_B$	I/O	
PC_2	IN	OUT	$\overline{STB}_B$	$\overline{ACK}_B$	I/O	
PC_3	IN	OUT	$INTR_A$	$INTR_A$	$INTR_A$	
PC_4	IN	OUT	$\overline{STB}_A$	I/O	$\overline{STB}_A$	
PC_5	IN	OUT	IBF_A	I/O	IBF_A	
PC_6	IN	OUT	I/O	$\overline{ACK}_A$	$\overline{ACK}_A$	
PC_7	IN	OUT	I/O	$\overline{OBF}_A$	$\overline{OBF}_A$	

Special Mode Combination Considerations

There are several combinations of modes when not all of the bits in Port C are used for control or status. The remaining bits can be used as follows:

If Programmed as Inputs –
All input lines can be accessed during a normal Port C read.

If Programmed as Outputs –
Bits in C upper (PC_7-PC_4) must be individually accessed using the bit set/reset function.

Bits in C lower (PC_3-PC_0) can be accessed using the bit set/reset function or accessed as a threesome by writing into Port C.

Source Current Capability on Port B and Port C

Any set of eight output buffers, selected randomly from Ports B and C can source 1mA at 1.5 volts. This feature allows the 8255 to directly drive Darlington type drivers and high-voltage displays that require such source current.

Reading Port C Status

In Mode 0, Port C transfers data to or from the peripheral device. When the 8255 is programmed to function in Modes 1 or 2, Port C generates or accepts "hand-shaking" signals with the peripheral device. Reading the contents of Port C allows the programmer to test or verify the "status" of each peripheral device and change the program flow accordingly.

There is no special instruction to read the status information from Port C. A normal read operation of Port C is executed to perform this function.

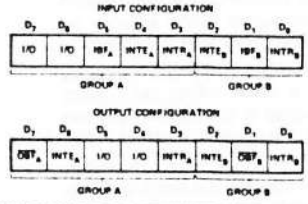


Figure 15. MODE 1 Status Word Format

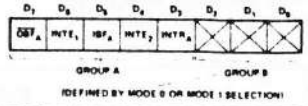


Figure 16. MODE 2 Status Word Format

Receiver Buffer

The Receiver accepts serial data, 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. Serial data is input to RxD pin, and is clocked in on the rising edge of $\overline{RxC}$.

Receiver Control

This functional block manages all receiver-related activities which consist of the following features:

The RxD initialization circuit prevents the 8251A from mistaking an unused input line for an active low data line in the "break condition". Before starting to receive serial characters on the RxD line, a valid "1" must first be detected after a chip master Reset. Once this has been determined, a search for a valid low (Start bit) is enabled. This feature is only active in the asynchronous mode, and is only done once for each master Reset.

The False Start bit detection circuit prevents false starts due to a transient noise spike by first detecting the falling edge and then strobing the nominal center of the Start bit (RxD = low).

The Parity Toggle F/F and Parity Error F/F circuits are used for parity error detection and set the corresponding status bit.

The Framing Error Flag F/F is set if the Stop bit is absent at the end of the data byte (asynchronous mode), and also sets the corresponding status bit.

RxRDY (Receiver Ready)

This output indicates that the 8251A contains a character that is ready to be input to the CPU. Rx RDY can be connected to the interrupt structure of the CPU or, for Polled operation, the CPU can check the condition of RxRDY using a Status Read operation.

Rx Enable off both masks and holds RxRDY in the Reset Condition. For Asynchronous mode, to set RxRDY, the Receiver must be Enabled to sense a Start Bit and a complete character must be assembled and transferred to the Data Output Register. For Synchronous mode, to set RxRDY, the Receiver must be enabled and a character must finish assembly and be transferred to the Data Output Register.

Failure to read the received character from the Rx Data Output Register prior to the assembly of the next Rx Data character will set overrun condition error and the previous character will be written over and lost. If the Rx Data is being read by the CPU when the internal transfer is occurring, overrun error will be set and the old character will be lost.

$\overline{RxC}$ (Receiver Clock)

The Receiver Clock controls the rate at which the character is to be received. In Synchronous Mode, the Baud Rate (1x) is equal to the actual frequency of $\overline{RxC}$. In Asynchronous Mode, the Baud Rate is a fraction of the actual $\overline{RxC}$ frequency. A portion of the mode instruction selects this factor; 1, 1/16 or 1/64 the $\overline{RxC}$.

For Example:

Baud Rate equals 300 Baud, if
$\overline{RxC}$ equals 300 Hz (1x)
$\overline{RxC}$ equals 4800 Hz (16x)
$\overline{RxC}$ equals 19.2 kHz (64x).

Baud Rate equals 2400 Baud, if
$\overline{RxC}$ equals 2400 Hz (1x)
$\overline{RxC}$ equals 38.4 kHz (16x)
$\overline{RxC}$ equals 153.6 kHz (64x).

Data is sampled into the 8251A on the rising edge of $\overline{RxC}$.

NOTE: In most communications systems, the 8251A will be handling both the transmission and reception operations of a single link. Consequently, the Receive and Transmit Baud Rates will be the same. Both $\overline{TxC}$ and $\overline{RxC}$ will require identical frequencies for this operation and can be tied together and connected to a single frequency source (Baud Rate Generator) to simplify the interface.

SYNDET (SYNC Detect)/BRKDET (Break Detect))

This pin is used in SYNChronous Mode for SYNDET and may be used as either input or output, programmable through the Control Word. It is reset to output mode low upon RESET. When used as an output (internal Sync mode), the SYNDET pin will go "high" to indicate that the 8251A has located the SYNC character in the Receive mode. If the 8251A is programmed to use double Sync characters (bisync), then SYNDET will go "high" in the middle of the last bit of the second Sync character. SYNDET is automatically reset upon a Status Read operation.

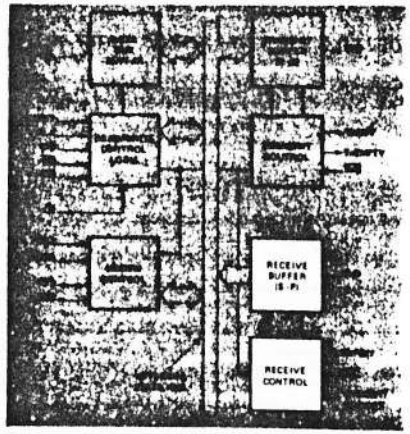


Figure 3. 8251A Block Diagram Showing Receiver Buffer and Control Functions

When used as an input (external SYNC detect mode), a positive going signal will cause the 8251A to start assembling data characters on the rising edge of the next $\overline{RxC}$. Once in SYNC, the "high" input signal can be removed. When External SYNC Detect is programmed, the Internal SYNC Detect is disabled.

BREAK DETECT (Async Mode Only)

This output will go high whenever the receiver remains low through two consecutive stop bit sequences (including the start bits, data bits, and parity bits). Break Detect may also be read as a Status bit. It is reset only upon a master chip Reset or Rx Data returning to a "one" state.

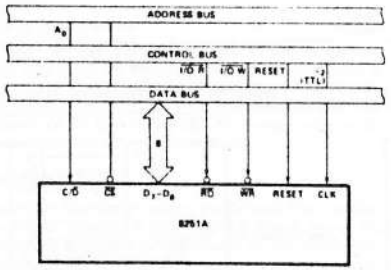


Figure 4. 8251A Interface to 8080 Standard System Bus

DETAILED OPERATION DESCRIPTION

General

The complete functional definition of the 8251A is programmed by the system's software. A set of control words must be sent out by the CPU to initialize the 8251A to support the desired communications format. These control words will program the: BAUD RATE, CHARACTER LENGTH, NUMBER OF STOP BITS, SYNCHRONOUS or ASYNCHRONOUS OPERATION, EVEN/ODD/OFF PARITY, etc. In the Synchronous Mode, options are also provided to select either internal or external character synchronization.

Once programmed, the 8251A is ready to perform its communication functions. The TxRDY output is raised "high" to signal the CPU that the 8251A is ready to receive a data character from the CPU. This output (TxRDY) is reset automatically when the CPU writes a character into the 8251A. On the other hand, the 8251A receives serial data from the MODEM or I/O device. Upon receiving an entire character, the RxRDY output is raised "high" to signal the CPU that the 8251A has a complete character ready for the CPU to fetch. RxRDY is reset automatically upon the CPU data read operation.

The 8251A cannot begin transmission until the Tx Enable (Transmitter Enable) bit is set in the Command Instruction and it has received a Clear To Send (CTS) input. The TxD output will be held in the marking state upon Reset.

Programming the 8251A

Prior to starting data transmission or reception, the 8251A must be loaded with a set of control words generated by the CPU. These control signals define the complete functional definition of the 8251A and must immediately follow a Reset operation (internal or external).

The control words are split into two formats:

1. Mode Instruction
2. Command Instruction

Mode Instruction

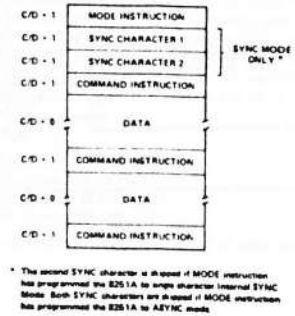
This format defines the general operational characteristics of the 8251A. It must follow a Reset operation (internal or external). Once the Mode Instruction has been written into the 8251A by the CPU, SYNC characters or Command Instructions may be inserted.

Command Instruction

This format defines a status word that is used to control the actual operation of the 8251A.

Both the Mode and Command Instructions must conform to a specified sequence for proper device operation. The Mode Instruction must be inserted immediately following a Reset operation, prior to using the 8251A for data communication.

All control words written into the 8251A after the Mode Instruction will load the Command Instruction. Command Instructions can be written into the 8251A at any time in the data block during the operation of the 8251A. To return to the Mode Instruction format, the master Reset bit in the Command Instruction word can be set to initiate an internal Reset operation which automatically places the 8251A back into the Mode Instruction format. Command Instructions must follow the Mode Instructions or Sync characters.



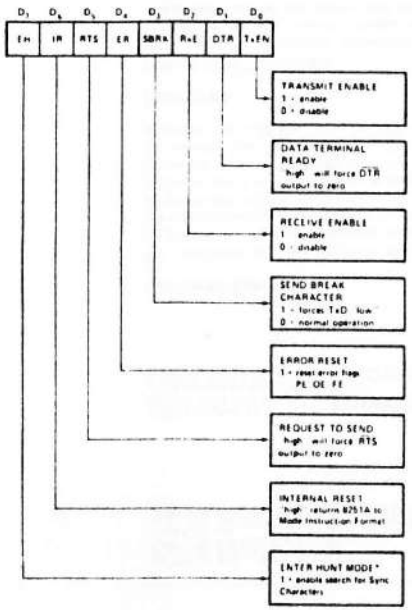
* The second SYNC character is skipped if MODE instruction has programmed the 8251A for single character internal SYNC Mode. Both SYNC characters are skipped if MODE instruction has programmed the 8251A for ASYNC mode.

Figure 5. Typical Data Block

COMMAND INSTRUCTION DEFINITION

Once the functional definition of the 8251A has been programmed by the Mode Instruction and the Sync Characters are loaded (if in Sync Mode) then the device is ready to be used for data communication. The Command Instruction controls the actual operation of the selected format. Functions such as: Enable Transmit/Receive, Error Reset and Modem Controls are provided by the Command Instruction.

Once the Mode Instruction has been written into the 8251A and Sync characters inserted, if necessary, then all further "control writes" ($C/\overline{D}$ = 1) will load a Command Instruction. A Reset Operation (internal or external) will return the 8251A to the Mode Instruction format.



Note: Error Reset must be performed whenever RxEnable and Enter Hunt are programmed

Figure 10. Command Instruction Format

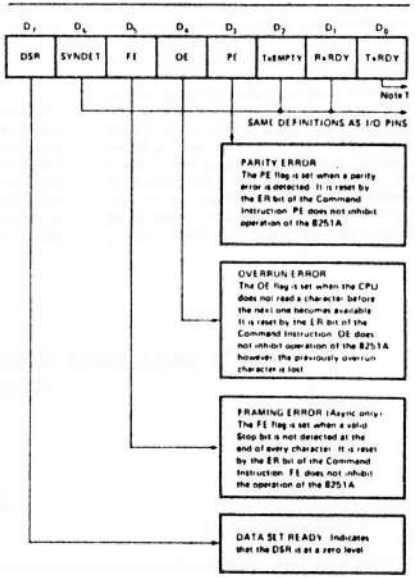
STATUS READ DEFINITION

In data communication systems it is often necessary to examine the "status" of the active device to ascertain if errors have occurred or other conditions that require the processor's attention. The 8251A has facilities that allow the programmer to "read" the status of the device at any time during the functional operation. (The status update is inhibited during status read).

A normal "read" command is issued by the CPU with $C/\overline{D}$ = 1 to accomplish this function.

Some of the bits in the Status Read Format have identical meanings to external output pins so that the 8251A can be used in a completely Polled environment or in an interrupt driven environment. TxRDY is an exception.

Note that status update can have a maximum delay of 28 clock periods from the actual event affecting the status.



Note 1: TxRDY status bit has different meanings from the TxRDY output pin. The former is not conditioned by $\overline{CTS}$ and TxEN, the latter is conditioned by both $\overline{CTS}$ and TxEN

i.e. TxRDY status bit = DB Buffer Empty

TxRDY pin out: DB Buffer Empty · (CTS · 0) · (TxEN · 1)

Figure 11. Status Read Format

APPLICATIONS OF THE 8251A

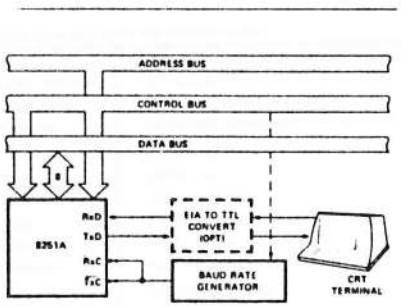


Figure 12. Asynchronous Serial Interface to CRT Terminal, DC—9600 Baud

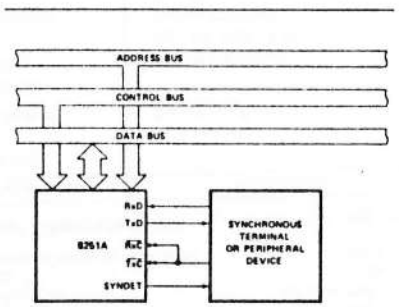


Figure 13. Synchronous Interface to Terminal or Peripheral Device

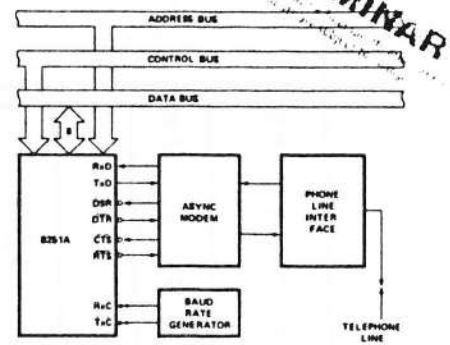


Figure 14. Asynchronous Interface to Telephone Lines

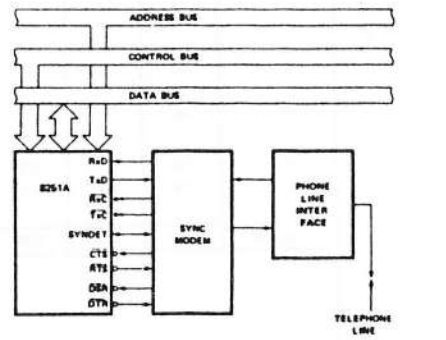


Figure 15. Synchronous Interface to Telephone Lines

Functional Description (Continued)

TABLE II. ADDRESS CODES AND FUNCTIONS

A4	A3	A2	A1	A0	FUNCTION
0	0	0	0	0	Counter – Thousandths of Seconds
0	0	0	0	1	Counter – Hundredths and Tenths of Seconds
0	0	0	1	0	Counter – Seconds
0	0	0	1	1	Counter – Minutes
0	0	1	0	0	Counter – Hours
0	0	1	0	1	Counter – Day of the Week
0	0	1	1	0	Counter – Day of the Month
0	0	1	1	1	Counter – Months
0	1	0	0	0	Latches – Thousandths of Seconds
0	1	0	0	1	Latches – Hundredths and Tenths of Seconds
0	1	0	1	0	Latches – Seconds
0	1	0	1	1	Latches – Minutes
0	1	1	0	0	Latches – Hours
0	1	1	0	1	Latches – Day of the Week
0	1	1	1	0	Latches – Day of the Month
0	1	1	1	1	Latches – Months
1	0	0	0	0	Interrupt Status Register
1	0	0	0	1	Interrupt Control Register
1	0	0	1	0	Counter Reset
1	0	0	1	1	Latch Reset
1	0	1	0	0	Status Bit
1	0	1	0	1	"GO" Command
1	0	1	1	0	Standby Interrupt
1	1	1	1	1	Test Mode

All others unused.

TABLE III. COUNTER AND LATCH RESET FORMAT

D0	D1	D2	D3	D4	D5	D6	D7	COUNTER OR LATCH RESET
1	0	0	0	0	0	0	0	Thousandths of Seconds
0	1	0	0	0	0	0	0	Hundredths and Tenths of Seconds
0	0	1	0	0	0	0	0	Seconds
0	0	0	1	0	0	0	0	Minutes
0	0	0	0	1	0	0	0	Hours
0	0	0	0	0	1	0	0	Days of the Week
0	0	0	0	0	0	1	0	Days of the Month
0	0	0	0	0	0	0	1	Months

FOR COUNTER RESET A4–A0 MUST BE 10010

FOR LATCH RESET A4–A0 MUST BE 10011

Functional Description (Continued)

Following a read of any real time counter a status bit read should be done. If during a counter read cycle the clock rolls over, the data read out could be invalid. Thus, during a read if the clock rolls over the status bit will be set. The status bit will appear on D0 when read, D1 through D7 will be zeros.

To synchronize the clock with real time a "GO" command exists which can be used to reset the thousandths of seconds, hundredths and tenths of seconds, and seconds counters. After setting the lower frequency counters (minutes through months), the appropriate address and a write pulse can be sent to reset all counters mentioned above. This allows the clock to be started at an exactly known time. It can also be used as a stopwatch function. The "GO" command is the start and a counter read is the stop point. The clock does not stop during or following a read, so each read would be a split time.

A second special command will enable the standby interrupt output. The standby interrupt output is the only input or output enabled during the power down or standby mode. Power down occurs when the power down input goes to a logical zero level. In this mode the outputs are TRI-STATED and the inputs ignored regardless of the state of the chip select. The standby interrupt is enabled by writing a 1 on the D0 line with the standby interrupt address selected. On the next counter-latch comparison the open drain output device turns on, sinking current. The output will be turned on immediately upon writing a 1 on D0 if the comparison occurred before the write, yet is still in effect. To disable the output a zero on D0 is written at the standby interrupt address. The write cycles must occur during normal operation, but the output can become active during power down. This feature can be used to turn the power back on during a power down mode (see *Figure 4* for a typical application). Refer to Tables II and III for the address input codes and functions and for the counter and latch reset format.

The interrupt output is controlled by the interrupt status register (8 bits) and the interrupt control register (8 bits). The status register contains the present state of the comparator (compares the counters and latches) and the outputs (1 bit each) of the tenths of seconds, seconds, minutes, hours, week, day of the month, and month counters *(Figure 1)*. The interrupt status register can only be read. The interrupt control register is a mask register that regulates which of the 8 bits in the status register goes out as an interrupt. The control register cannot be read from. A 1 is written into the control register to select the appropriate interrupt output. If more than a single 1 exists in the control register each selected bit will come out as an interrupt. This will appear as an interrupt occurring at the highest frequency selected. The interrupt is acknowledged by addressing and reading the status register. Once acknowledged the interrupt output and status register are reset. The only way to disable the interrupt output is to write all 0's into the control register or to enable the power down input.

The I/O bus is controlled by the read, write, ready and chip select lines. During a read cycle ($\overline{RD}$ = 0, $\overline{WR}$ = 1, $\overline{CS}$ = 0, RDY = 0) the data on the I/O bus is the data contained in the addressed counter or latch. During a write cycle ($\overline{RD}$ = 1, $\overline{WR}$ = 0, $\overline{CS}$ = 0, RDY = 0) the data on the I/O bus is latched into the addressed counter or latch. At the start of each read or write cycle the RDY signal goes low and will remain low until the clock has placed valid data on the bus or until it has completed latching data in on a write. The chip select line is used to enable or disable the device outputs. When the chip is selected the device will drive the I/O bus for a read or use the I/O bus as an input for a write. The I/O bus will not be affected when the chip is deselected. The outputs driving the bus will go to the TRI-STATE or high impedance state. The chip will not respond to any inputs when deselected. Refer to *Figures 2 and 3* for read and write cycle timing.

The clock's time base is a 32,768 crystal controlled oscillator. Externally, the crystal, the input tuning capacitor, and the output load capacitor are required. Included internally are a high gain inverter, an RC delay, and the bias resistor. To tune the oscillator a constant read can be done on one of the higher frequency counters. For example, a constant read of the thousandths of seconds counter will place a 500 Hz signal on the D4 bus line.

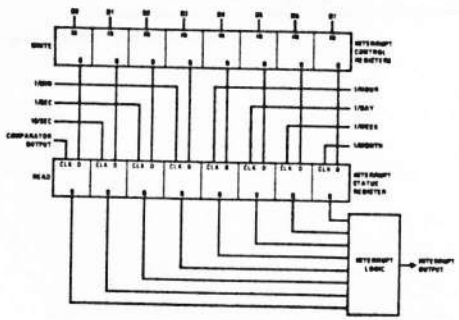
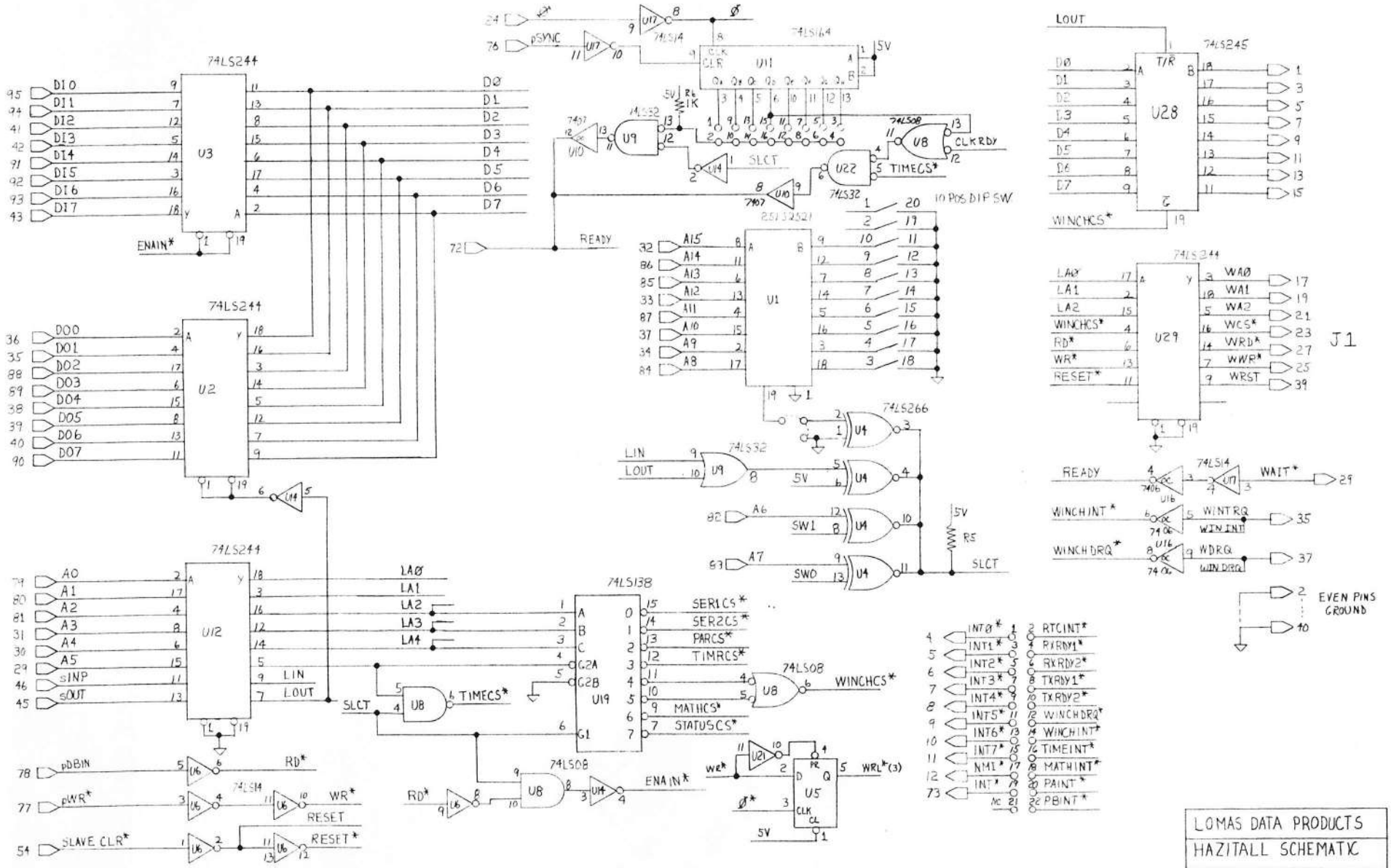


FIGURE 1. Interrupt Register Format

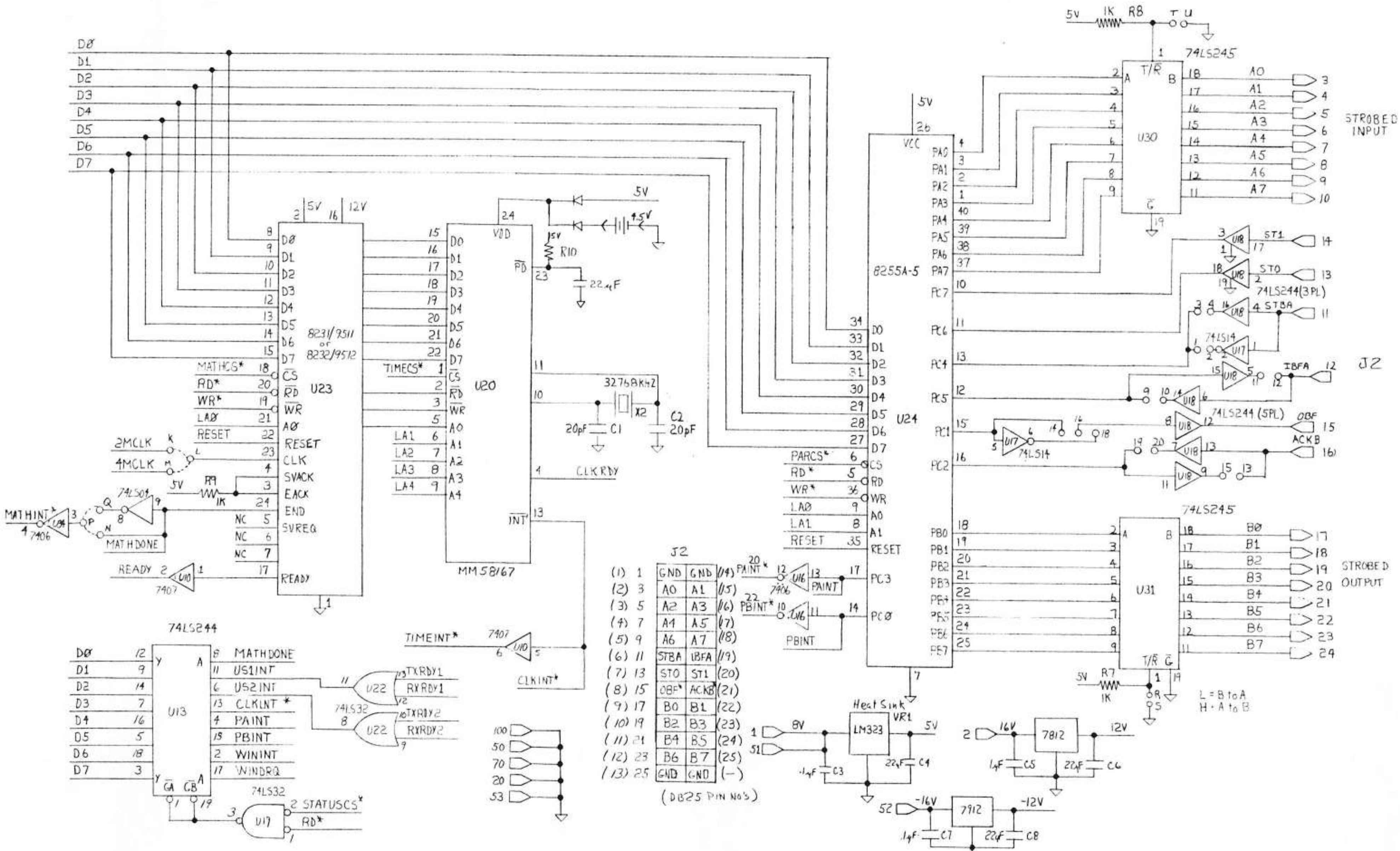
```
00BC                MODEBYTE        EQU     0BCH    ;PORTB STROPED OUT,PORTA STROBED INPUT
00C0                BASE            EQU     0C0H    ;BASE ADDRESS OF THE HAZITALL
00C0                SER1DATA        EQU     BASE+0
00C1                SER1CTL         EQU     BASE+1
00C1                SER1STATUS      EQU     BASE+1

00C4                SER2DATA        EQU     BASE+4
00C5                SER2CTL         EQU     BASE+5
00C5                SER2STATUS      EQU     BASE+5

00CF                TIMERCTL        EQU     BASE+0FH
00CC                COUNTER1        EQU     BASE+0CH
00CD                COUNTER2        EQU     BASE+0DH
00CE                COUNTER3        EQU     BASE+0EH

00CB                PPCTL           EQU     BASE+0BH
00C8                PORTA           EQU     BASE+08H
00C9                PORTB           EQU     BASE+09H
00CA                PORTC           EQU     BASE+0AH
00DC                HAZITALL_STATUS EQU     BASE+1CH
00DC                HAZITALL_CTL    EQU     BASE+1CH
                    ;HAZITALL DRIVERS FOR CPM86
                    ;
                    ;       SERIAL PORT 1= CONSOLE PORT
                    ;       SERIAL PORT 2= PRINTER PORT
                    ;       PARRALLEL OUTPUT=AUXILLIARY OUTPUT
                    ;       PARRALLEL INPUT=AUXILLIARY INPUT
                    ;
                    HAZITALL_INIT:
                    ;INITIALIZE THE 2 USARTS - 2 STOP BITS,NO PARITY,X16 MODE
0000 BE6900                 MOV     SI,OFFSET SERIAL_TABLE
0003 B90600                 MOV     CX,6            ;6BYTES TO INIT EACH USART
                    INITLOOP:
0006 AC                     LODS    AL              ;GET BYTE FROM INIT TABLE
0007 E6C1                   OUT     SER1CTL,AL      ;SEND TO FIRST USART
0009 E6C5                   OUT     SER2CTL,AL      ;SEND SAME BYTE TO USART 2
000B E2F9           0006    LOOP    INITLOOP        ;DO UNTILL 6 BYTES SENT
                    TIMER_INIT:
000D BE6F00                 MOV     SI,OFFSET TIMER_TABLE
0010 B90300                 MOV     CX,3            ;INIT 3 TIMERS SO THREE TIME THROUGH
0013 BACC00                 MOV     DX,COUNTER1     ;POINT TO COUNTER 1
                    TIMER_LOOP:
0016 AC                     LODS    AL              ;GET BYTE FROM TABLE
0017 E6CF                   OUT     TIMERCTL,AL     ;SEND TO MODE REGISTER
0019 AC                     LODS    AL              ;GET LSB OF DIVISOR
001A EE                     OUT     DX,AL           ;SEND IT TO THE COUNTER
001B AC                     LODS    AL              ;GET MSB
001C EE                     OUT     DX,AL           ;SEND IT TO THE COUNTER
001D 42                     INC     DX              ;NEXT COUNTER
001E E2F6           0016    LOOP    TIMER_LOOP      ;DO AGAIN IF CX NOT 0

                    PARRALLEL_INIT:
```

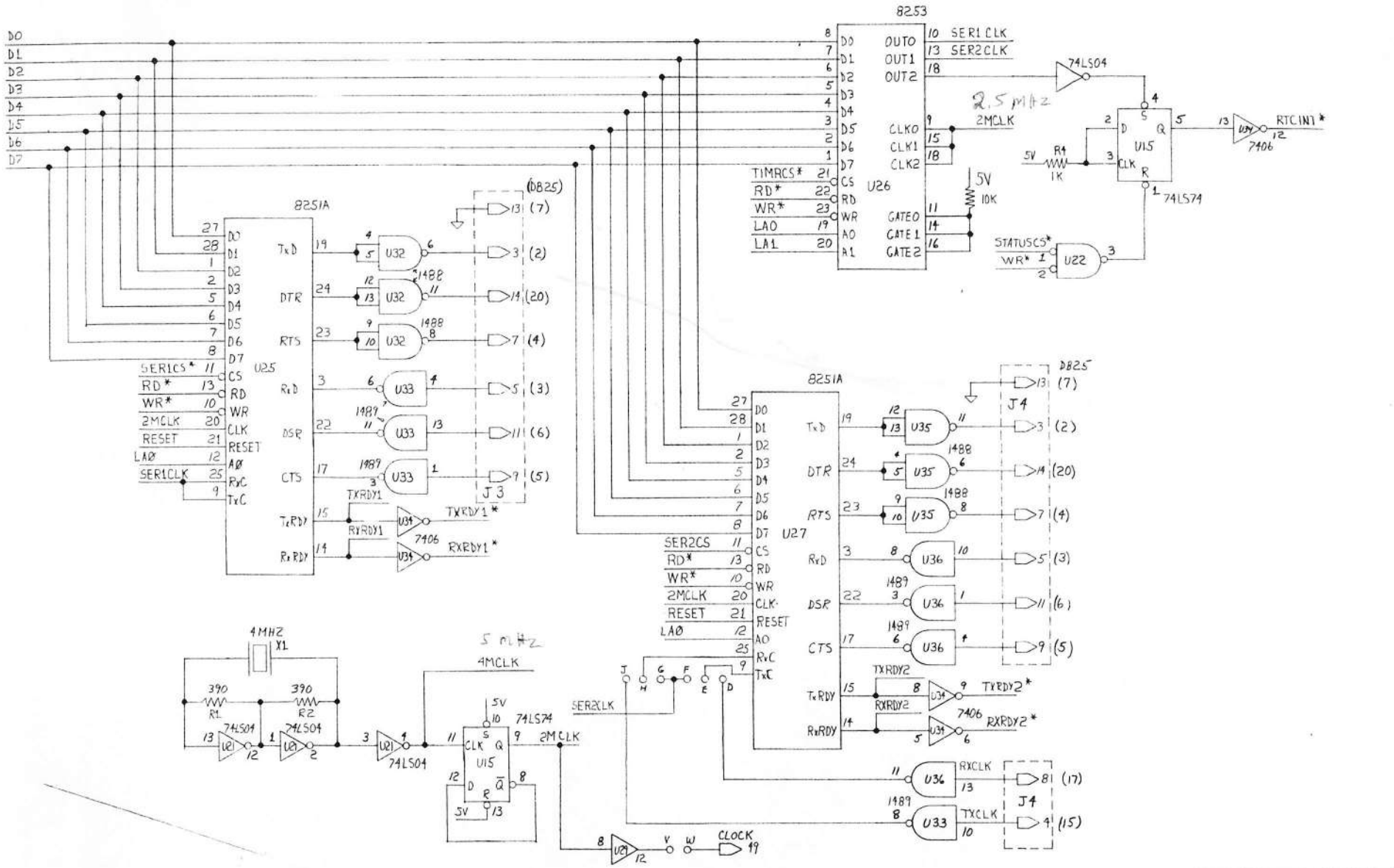
```
0020 B0BC                       MOV     AL,MODEBYTE     ;GET MODE WORD FOR PARRALLEL PORT
0022 E6CB                       OUT     PPCTL,AL        ;SET MODE
0024 C3                         RET                     ;INITILIZATION OF HAZITALL COMPLETE

                        CONOUT:
                        ;CONSOLE OUTPUT ROUTINE
                        ;CHARACTER TO BE SENT IS IN REG CL
0025 E4C1                       IN      AL,SER1STATUS   ;GET USART STATUS
0027 A801                       TEST    AL,01           ;CHECK FOR TX READY
0029 74FA       0025            JZ      CONOUT          ;IF NOT READY WAIT UNTILL READY
002B 8AC1                       MOV     AL,CL           ;GET DATA TO SEND
002D E6C0                       OUT     SER1DATA,AL     ;SEND THE DATA
002F C3                         RET
                        CONSOLE_STATUS:
                        ;CHECKS THE CONSOLE STATUS AND RETURNS WITH AL=0FFH IF A CHARACTER
                        ;IS WAITING ELSE AL IS 0'ED INDICATING NO CHARACTER AVAILIABLE.
0030 E4C1                       IN      AL,SER1STATUS   ;GET USART STATUS
0032 A802                       TEST    AL,02H          ;RX READY?
0034 7403       0039            JZ      FALSE_RETURN    ;IF ZERO THEN NO CHARACTER
0036 0CFF                       OR      AL,0FFH         ;SET TO 1'S TO INDICATE CHARACTER
0038 C3                         RET
                        FALSE_RETURN:
0039 32C0                       XOR     AL,AL           ;ZERO AL
003B C3                         RET

                        CONSOLE_INPUT:
003C E8F1FF     0030            CALL    CONSOLE_STATUS  ;CHECK IF CHARACTER READY
003F 74FB       003C            JZ      CONSOLE_INPUT   ;IF ZERO WAIT TILL AVAILABLE
0041 E4C0                       IN      AL,SER1DATA     ;GET THE CHARACTER
0043 247F                       AND     AL,7FH          ;STRIP ANY HIGH-ORDER BIT
0045 C3                         RET                     ;RETURN WITH CHARACTER

                        PRINTER_OUTPUT:
0046 E4C5                       IN      AL,SER2STATUS   ;GET USART 2 STATUS
0048 A801                       TEST    AL,01H          ;TX READY?
004A 74FA       0046            JZ      PRINTER_OUTPUT  ;IF NOT WAIT
004C 8AC1                       MOV     AL,CL           ;GET CHARACTER TO SEND
004E E6C4                       OUT     SER2DATA,AL     ;SEND THE CHARACTER TO THE PRINTER
0050 C3                         RET                     ;BACK TO CPM

                        ;PARRALLEL INTERFACE ROUTINES
                        ;
                        ;PORT A IS USED AS A STROBED INPUT PORT
                        ;PORT B IS USED AS A STROBED OUTPUT PORT
                        ;THE TWO SPARE LINES ON PORT C ARE USED AS INPUTS
                        ;
                        ;THE MODE WORD FOR THIS CONFIGURATION IS 0BCH
                        ;
                        ;THE BITS OF THE 8255 STATUS PORT ARE INTERPRITED AS FOLLOWS
                        ;WHEN USED IN THIS MODE:
                        ;
                        ;       BIT 0   INTERRUPT B ASSERTED
```

```
                        ;       BIT 1   OUTPUT BUFFER EMPTY
                        ;       BIT 2   PORT B INTERRUPT ENABLE
                        ;       BIT 3   INTERRUPT A ASSERTED
                        ;       BIT 4   PORT A INTERRUPT ENABLE
                        ;       BIT 5   PORT A INPUT BUFFER FULL STATUS
                        ;       BIT 6   I/O BIT 6
                        ;       BIT 7   I/O BIT 7
                        ;
                        ;PORT B IS USED AS THE AUXILLIARY OUTPUT, AND PORT A IS USED AS
                        ;THE AUXILLIARY INPUT. PORT C BIT 6 IS USED AS A STATUS INPUT
                        ;FROM THE RECEIVING DEVICE. A 1 INDICATES WILLINGNESS TO RECEIVE,
                        ;AND A 0 INDICATES THE DEVICE IS BUSY.

                        AUXIN:
0051 E4CA                       IN      AL,PORTC        ;GET DEVICE STATUS
0053 A820                       TEST    AL,20H          ;TEST BUFFER FULL STATUS
0055 74FA       0051            JZ      AUXIN           ;IF BUFFER NOT FULL WAIT
0057 E4C8                       IN      AL,PORTA        ;GET THE INPUT DATA
0059 C3                         RET                     ;RETURN WITH IT

                        AUXOUT:
005A E4CA                       IN      AL,PORTC        ;GET THE STATUS
005C A880                       TEST    AL,80H          ;CHECK TO SEE IF PERIPHERAL READY
005E 74FA       005A            JZ      AUXOUT          ;IF NOT READY WAIT TILL IT IS
0060 A802                       TEST    AL,02H          ;IS THE BUFFER EMPTY
0062 74F6       005A            JZ      AUXOUT          ;BUFFER IS NOT EMPTY WAIT SOME MORE
0064 8AC1                       MOV     AL,CL           ;GET THE DATA TO SEND
0066 E6C9                       OUT     PORTB,AL        ;SEND IT
0068 C3                         RET
                        ;THE ABOVE ROUTINE COULD BE USED TO OUTPUT TO A PRINTER. IF IT IS,
                        ;INSURE THAT THE POLARITY OF THE PRINTER BUSY STATUS MATCHES THE
                        ;ABOVE ASSUMPTIONS.


                        SERIAL_TABLE:
0069 000000                     DB      0,0,0           ;THREE NULLS
006C 40                         DB      40H             ;RESET COMMAND
006D CE                         DB      0CEH            ;MODE - 8 BIT DATA,NOPARITY,X16 MODE
006E 37                         DB      37H             ;COMMAND - RTS,DTR ASSERTED, RX AND
                                                        ;TX ENABLED, CLEAR ANY ERRORS

                        TIMER_TABLE:
006F 36                         DB      00110110B       ;COUNTER 1 MODE WORD
0070 0D00                       DW      13              ;DIVISOR FOR 9600 BAUD FOR CONSOLE
0072 76                         DB      01110110B       ;COUNTER 2 MODE WORD
0073 A101                       DW      417             ;300 BAUD FOR PRINTER
0075 B6                         DB      10110110B       ;COUNTER 3 MODE WORD
0076 D0                         DB      2000            ;1 MS INTERRUPT RATE FOR RTC - NOT USED


END OF ASSEMBLY.  NUMBER OF ERRORS:  0.  USE FACTOR:  1%
```

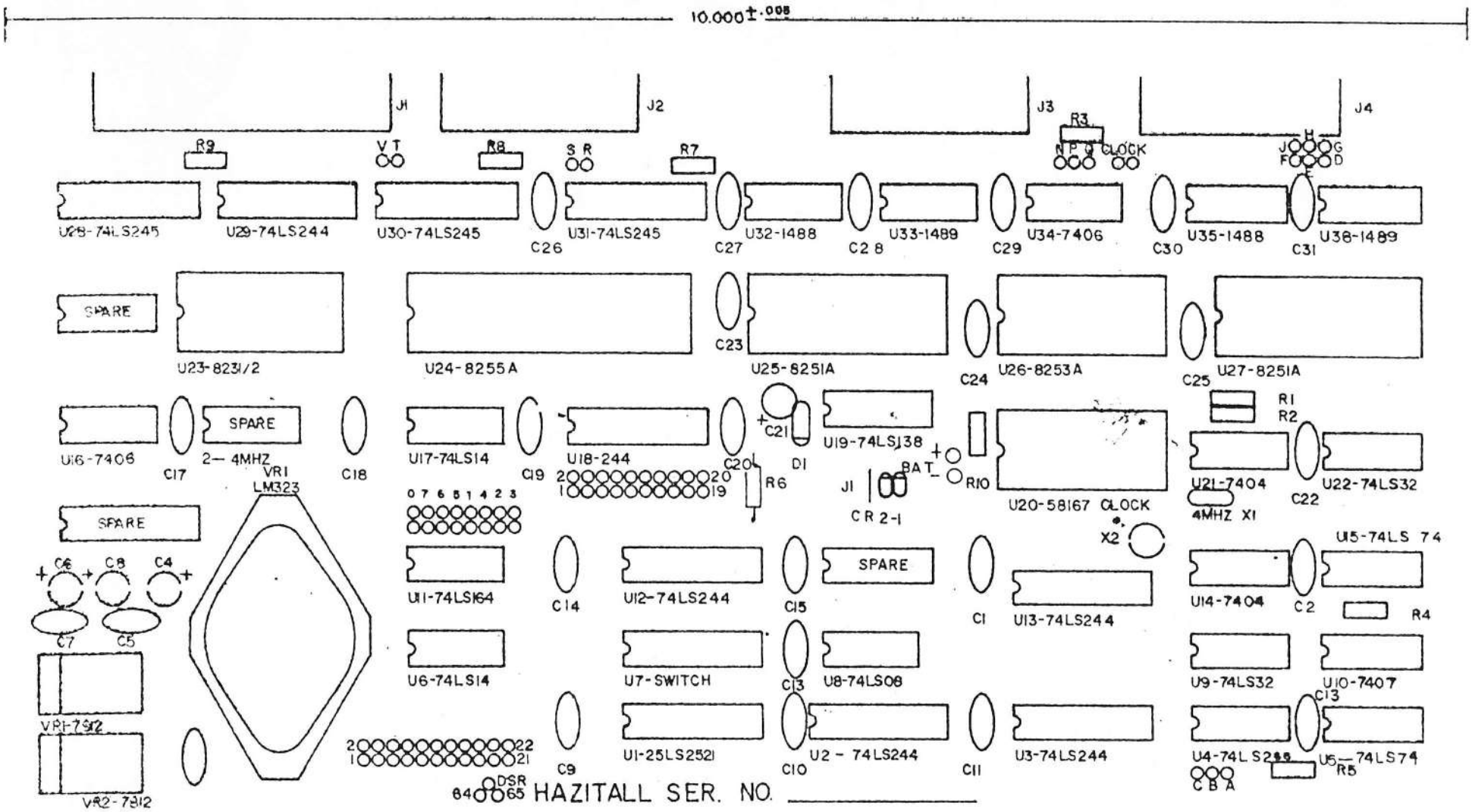


LOMAS DATA PRODUCTS	
HAZITALL SCHEMATIC	
DATE 5-15-81	
REV 0	PAGE 1


D0 D1 D2 D3 D4 D5 D6 D7
8231/9511 or 8232/9512
U23
MATHCS* RD* WR* LA0 RESET
2MCLK 4MCLK
SVACK EACK END SVREQ READY
MATHINT* 7406 74LS04 MATHDONE
READY 7407
MM58167
U20
TIMECS* LA1 LA2 LA3 LA4
32768KHZ X2 C1 C2 20pF
CLKRDY
R10 22µF
5V 4.5V 12V
74LS244
U13
MATHDONE US1INT US2INT CLKINT* PAINT PBINT WININT WINDRQ
74LS32 U22 TXRDY1 RXRDY1 TXRDY2 RXRDY2
U17 STATUSCS* RD*
TIMEINT* U10 CLKINT*
8255A-5
U24
PARCS* RD* WR* LA0 LA1 RESET
PA0 PA1 PA2 PA3 PA4 PA5 PA6 PA7
PC7 PC6 PC4 PC5 PC1 PC2 PC3 PC0
PB0 PB1 PB2 PB3 PB4 PB5 PB6 PB7
74LS245
U30
T/R
A0 A1 A2 A3 A4 A5 A6 A7
STROBED INPUT
ST1 STO STBA IBFA OBF ACKB
74LS244(3PL) 74LS14 74LS244 (5PL)
U18 U17
J2
PAINT* PBINT* 7406 U16 PAINT PBINT
U31
B0 B1 B2 B3 B4 B5 B6 B7
STROBED OUTPUT
L = B to A
H = A to B
R7 1K R8 1K
J2
GND GND A0 A1 A2 A3 A4 A5 A6 A7 STBA IBFA STO ST1 OBF* ACKB B0 B1 B2 B3 B4 B5 B6 B7 GND GND
(DB25 PIN NO's)
Heat Sink
LM323 VR1
8V 5V
7812 16V 12V
7912 -16V -12V
C3 C4 C5 C6 C7 C8



Clock measured 15 Dec 91 J.S.
5 mHz
2.5 mHz



LOMAS DATA PRODUCTS

HAZITALL SILK SCREEN REV I

DEC-28-81