I/O-8

EIGHT PORT SERIAL COMMUNICATIONS BOARD

OPERATIONS MANUAL

SD #7140172
REVISION D
NOVEMBER 21, 1983

SDSYSTEMS
A SYNTECH COMPANY

EMI NOTICE

This equipment has been designed and constructed to professional standards. However, the equipment must be utilized correctly by the user to obtain proper performance and to comply with applicable industry and governmental regulations.

Since the equipment is supplied as an unconfigured component and cannot be tested for electro-magnetic interference (EMI) in all possible configurations, the equipment is not subject to standards imposed by Subpart J of Part 15 of FCC Rules and Regulations.

Final system configuration will require compliance with applicable FCC regulations. SDSystems recommends the installation of shielded data cables for all external cabling. Electro-magnetic interference (EMI) levels are dependent upon final system hardware configuration and application.

Contact SDSystems Customer Service if additional configuration information is required.

TERMINATION NOTE

SDSystems recommends the use of active termination circuitry on the S-100 bus to avoid system problems.

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

ILLUSTRATIONS--Continued

TABLES

SECTION I
INTRODUCTION

1.0 GENERAL

The I/O-8 is an 8530/8531 based interface board which fully complies with the IEEE-696 specification for the S-100 bus. The I/O-8 operates as a Bus Slave under control of the IEEE-696 permanent or temporary Bus Master. The I/O-8 board is designed to provide serial communication capability from an S-100 computer to various Data Communications Equipment (DCE) or Data Terminal Equipment (DTE).

The following is a list of technical references applicable to the I/O-8:

IEEE-696 Bus Specification
8530 SCC Product Specification (Zilog)
8531 ASCC Product Specification (Zilog)
58167A RTC Product Specification (National)
EIA- RS-232C Specification

1.1 FEATURES

The key features of the I/O-8 board are as follows:

1. Serial Ports

   4 ports asynchronous DTE/DCE
   4 ports synchronous/asynchronous DTE/DCE
   8 I/O ports are addressable to any 8 port boundary in 64K

2. Baud Rate

   Each channel software selectable, full duplex
   50 to 19,200 in asynchronous mode using x16 internal clock
   Up to 1Mbit/sec in synchronous mode using external clock
   Up to 307,200 in synchronous mode using x1 on board internal clock (2.4576 mHz)

3. Data Bits

   5, 6, 7, or 8 data bits/character
   Stop bits - 1, 1.5, 2
   Clock rate - x1, x16, x32, x64
   Parity - odd, even, or none
   Error detection - parity, overrun, CRC or framing
   Data encoding - NRZ, NRZI, FM1, FM0

4. Interrupts

   Receiver ready
   All receive characters
   Special receive condition (parity error, framing error, overrun error, and end of frame in SDLC)

5. Real Time Clock With Battery Backup

6. Software Programmable Interrupt Rates

7. Standby Interrupt While Powered Down

8. Millisecond Through Month Counters

9. User Readable Switch

SECTION II
FUNCTIONAL DESCRIPTION

2.0 GENERAL

A functional block diagram of the I/O-8 board is shown in Figure 2-1. This section gives a brief description of each of the major blocks.

2.1 SERIAL INPUT/OUTPUT

Eight full duplex serial communications channels are contained on the I/O-8.

2.1.1 Serial Communications Controller (SCC/ASCC)

The Z8530A or Z8531A Serial Communications Controller (SCC) is used to interface and control the serial ports. The Z8530A SCC is used for synchronous or asynchronous communications while the Z8531A ASCC will perform asynchronous communications only. Both of these chips have an on-board programmable baud rate generator, complete interrupt control, and four different ways of encoding data. The SCC can also support byte-oriented and bit-oriented synchronous protocols including SDLC, HDLC, and Bisync.

The Z8030/Z8530 SCC Serial Communications Controller Technical Manual describes the options that can be selected under software control. A partial list follows:

* 5, 6, 7 or 8 bits/character
* even, odd, or no parity
* 1, 1-1/2 or 2 stop bits
* CRC generation/checking (sync modes only)
* Polled I/O or Interrupt operation
* Parity, overrun, and framing error detection

2.1.2 Serial Port Interface

The SCC to RS-232 interface is performed by the common MC1488 line driver and MC1489 line receiver. Four ports have a complete interface (TxD, RxD, CTS, RTS, DTR, DSR, DCD, RxCLK, TxCLK, and CLK) allowing synchronous/asynchronous communication with a terminal or modem (DCE or DTE). A header is included on the full ports to make the switch from terminal to modem easier in most cases. The remaining four ports support asynchronous only using four signals (TxD, RxD, CTS, and RTS). The RxD and TxD lines can be swapped to allow a limited conversion for modem use.

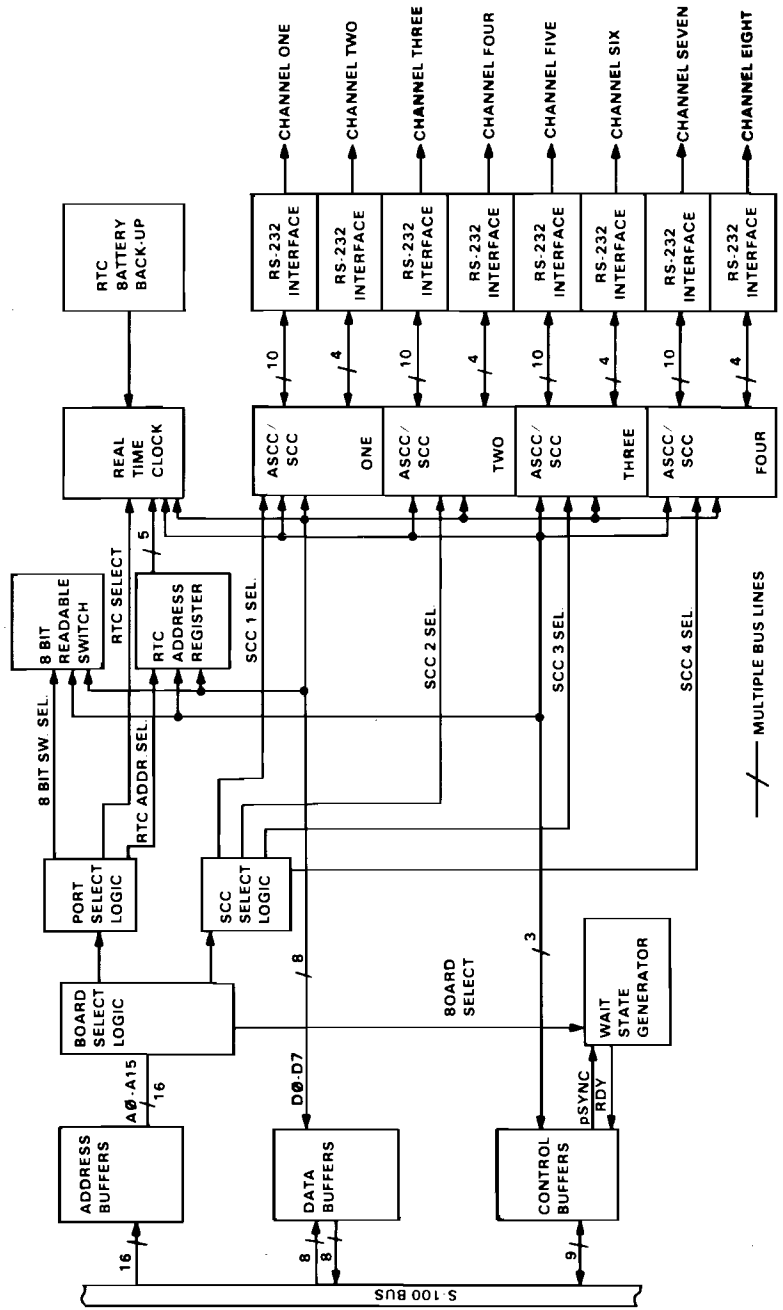


Figure 2-1. I/O-8 BLOCK DIAGRAM

2.1.3 SCC Interrupts

Each of the eight I/O channels can generate three types of interrupts:

1) Transmit
2) Receive
3) External/Status

Transmit interrupt indicates the transmit buffer is empty. Receive interrupts are of three types:

1) First receive character or special receive condition
2) All receive characters or special receive condition
3) Special receive condition only (parity error, framing error)

External/Status interrupts will be generated by various status or error conditions including:

1) Clear To Send (CTS) transition
2) Data Carrier Detect (DCD) transition
3) Transmit underrun
4) Zero baud rate count

All three types of channel interrupts are sent on one interrupt line shared with all eight channels. A jumper strip is provided to allow the serial I/O interrupt line to be connected to any of the bus interrupt lines (Vectored Interrupt 0 to 7 or the Non-Maskable Interrupt).

2.2 REAL TIME CLOCK (RTC)

The I/O-8 uses the 58167A Real Time Clock chip featuring software programmable interrupt rates, battery backup, 56 bits of RAM, and clock counters.

2.2.1 RTC Counters and RAM

The Real Time Clock provides time and day counters and 56 bits of RAM. Nine counters are available: milliseconds, hundredths of seconds, tenths of seconds, seconds, minutes, hours, day of week, day of month, and months. The RAM can be used for storage while the system is powered down or for generating interrupts at preset times.

2.2.2 RTC Battery Backup

An on-board battery provides power to the real time clock and standby interrupt function during system power down. Accurate time can be kept for up to six months in power down mode (dependent upon temperature and battery cycle life).

2.2.3 RTC Interrupts

Two types of interrupts can be generated by the Real Time Clock: the standby interrupt and a general interrupt output. Both interrupts can be programmed for fixed interrupt rates or for a preset time. Also, both interrupts can be independently connected through jumper strips to any bus interrupt line (Vectored Interrupt 0 to 7 or the Non-Maskable Interrupt).

The standby interrupt is used when the host system is powered down and the RTC is battery powered. This interrupt signals another device on the bus (with its own power source) that a set time has passed. Two possible uses would be to have circuitry that would power up the computer when this interrupt occurred or to signal the computer on power up that a preset time has passed.

The general interrupt output is used during normal operation when the host system is powered up. A possible use would be in a multi-user time sharing system. Each time an interrupt occurred it would signal the host system to service a different user.

2.3 BOARD SELECT AND PORT DECODE LOGIC

Eight of the host processor's Input/Output ports are used by the I/O-8. The SCCs require four ports (two for data and two for control), the SCC select register uses one, the RTC address select register uses one, the RTC uses one, and the eight bit readable switch uses one. The address for this block of ports can be placed at any eight byte boundary in 64K. This means the address could be placed at 0 or 8 but not at 1, 2, 3, 4, 5, 6, or 7 because these locations are not evenly divisible by eight.

The board select signal is generated by comparing the S-100 address lines with five or 13 user set switches. The number of switches depends on whether the host processor has an eight bit or a 16 bit I/O address. For example, a Z80 based processor would use eight bits and a 68000 processor would use sixteen bits. A three pin jumper strip is

used to switch between the two modes of addressing. After a board select address is found, the remaining three address bits are tested to determine which port is being accessed. A chip select signal is then generated to enable the proper port.

2.4 SCC SELECT REGISTER

To conserve on the number of Input/Output ports needed by the I/O-8, a register was added to select only one SCC at a time. Without this register, 16 I/O ports would have been needed to control the SCCs. With this register five I/O ports are used, a saving of 11 ports. Outputting the SCC's number (0 - 3) to this register will cause the two data and two control registers of the selected SCC to be placed at the first four ports.

2.5 RTC ADDRESS SELECT/STATUS REGISTER

As an additional saving on the number of Input/Output ports used, a register was added to select a Real Time Clock register. Any RTC register can be written to or read from by first outputting its number (0 - 32) to the RTC Address Select Register and then accessing the RTC port. Using this technique, the host processor I/O ports needed were reduced from 32 to only two.

The function of the RTC Address Select Register changes slightly during a read operation. As expected, the selected RTC register number is contained in the lower five bits. But in a read operation the upper two bits contain the number of the selected SCC. This allows the complete status on which SCC is selected and which RTC register is selected to be determined by reading the one port.

2.6 EIGHT BIT SWITCH PORT

An additional feature of the I/O-8 is a readable switch port. The user assigns a function to each of the eight switches and through the software can determine how the board is configured. An example of its use is in a system that has mixed synchronous and asynchronous channels. The user decides that each bit will correspond to an I/O-8 channel. If the switch is off, the channel will be synchronous and, if on, the channel is asynchronous. The software drivers are then written to read the switch port and initialize each channel to the desired configuration. When it is necessary to add another synchronous port, the switch can be changed and the software will automatically initialize the channel properly.

2.7 WAIT STATE GENERATOR

To synchronize the SCCs and Real Time Clock with the host processor a wait state generator has been added. Any access to the SCCs will automatically have two wait states inserted and any access to the Real Time Clock will have four wait states inserted. The SCC Select Register, RTC Address Select Register, and Eight Bit Switch Port can operate at higher speeds and do not need any wait states.

SECTION III
THEORY OF OPERATIONS

3.0 GENERAL

To make the I/O-8 board more understandable to the end user, the following subsections describe the general operation theory behind the I/O-8. Topics discussed are serial communications in synchronous and asynchronous formats and the need for DCE and DTE interfaces.

3.1 SERIAL COMMUNICATIONS

In serial communications the data bytes to be transmitted are converted from the parallel form in the computer to a serial form for transmission on a single wire. The RS-232-C standard was written to define the serial interface.

3.1.1 DCE And DTE

Two types of RS-232 serial devices exist: Data Communications Equipment (DCE) and Data Terminal Equipment (DTE). These types are necessary since the RS-232-C standard defines signals on the connecting cable to be unidirectional. Obviously the Transmit Data (TxD) pin of one device must be connected to the Receive Data (RxD) pin of the other device. To do this the first device (DTE) must have its TxD pin connected to the RS-232 TxD line, and the second device (DCE) must have its RxD pin connected to the RS-232 TxD line. As a result the two devices that are being interfaced must be of opposite types--one DTE and one DCE. The DTE device always has its interface pins connected to the RS-232 line with the same name. The interface to make a device DCE is not as straightforward. Each line in the RS-232 specification is paired with another line with the same general function (i.e. TxD-RxD, RTS-CTS, DTR-DSR). A DCE interface is made by matching the cable lines with the SCC line that is in the function pair.

3.1.2 Synchronous And Asynchronous Communications Formats

The main difference between synchronous and asynchronous communications formats is in the timing used for receiving information bits. Synchronous transmissions use a separate clock signal for indicating when another bit is to be received, while

asynchronous transmissions use a start bit to signal the beginning of a data character. Figure 3-1 shows the format of an asynchronous data character, and Figure 3-2 shows the format of a synchronous data character. Both of these formats approximately correspond to the voltage waveform expected on the RS-232 Transmit Data (TxD) and Receive Data (RxD) lines.

Before an asynchronous data character is transmitted, the line is always in a high (1) state. The arrival of a start bit signals the beginning of a data character, and a fixed number of data bits along with a parity bit will be received. The end of a data character is marked by 1, 1.5, or 2 stop bits. After the stop bits, another start bit can be transmitted or the line can return to its idle state. The rate that data bits are received or transmitted in asynchronous mode is called the baud rate. The SCC generates its own baud rate clock to sample incoming data at the proper rate. Figure 3-2 shows an example of a monosynchronous data transmisson. Transitions on the CLOCK line indicate that another bit is to be sampled. After each bit is received, a comparison is made for a match with the sync character programmed in the SCC. The example in Figure 3-2 uses a 01101000B for the sync character. After the sync character is recognized, all subsequent bits are assumed to be data and are placed in the receive buffer. Because of the lack of framing bits, synchronous transmissions are more efficient than asynchronous transmissions.

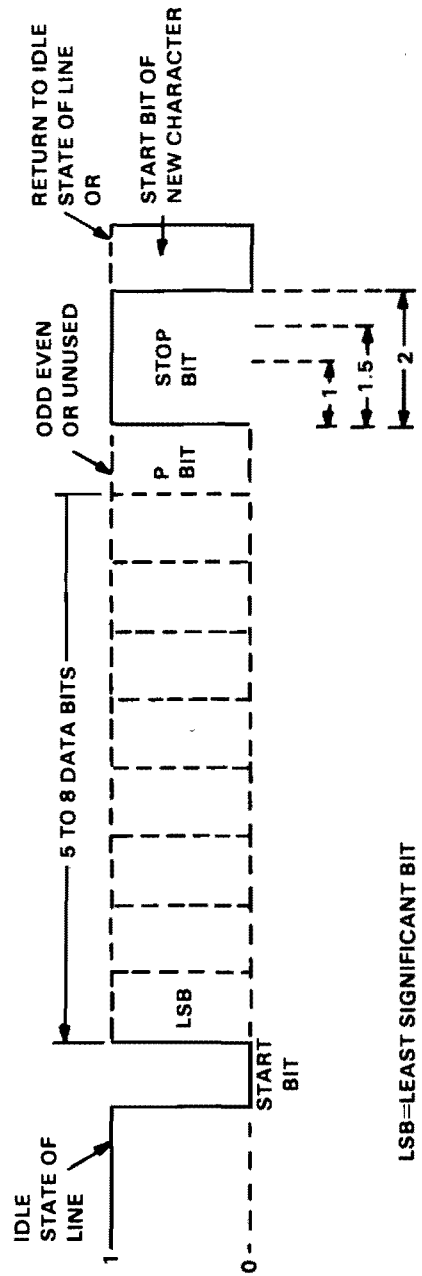
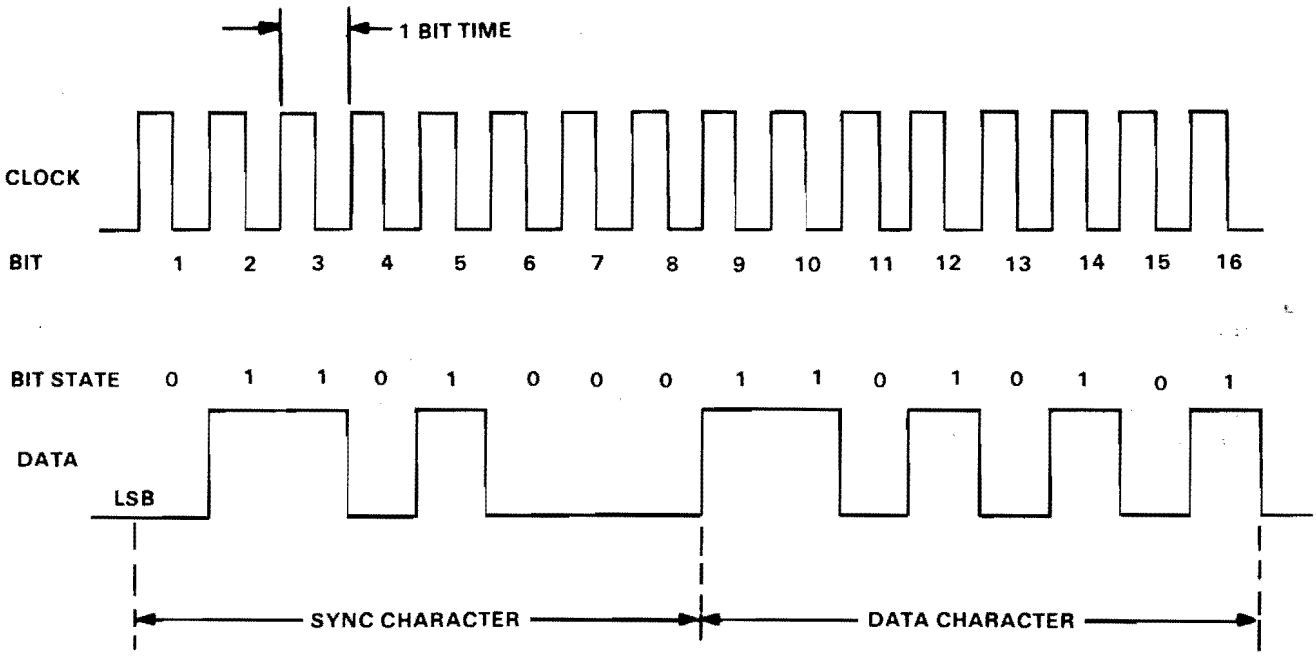


Figure 3-1. ASYNCHRONOUS DATA CHARACTER

Figure 3-2. SYNCHRONOUS DATA CHARACTER



LSB = LEAST SIGNIFICANT BIT

SECTION IV
OPERATIONS

4.0 GENERAL

The I/O-8 can be configured to operate in any of several modes including: synchronous, asynchronous, Data Terminal Equipment (DTE), or Data Communications Equipment (DCE). This section describes the set-up procedure for these modes and the procedures for programming the Serial Communications Controllers (SCC) and the Real Time Clock (RTC).

4.1 BOARD ADDRESSING OPTIONS

The I/O-8 requires eight consecutive ports of the host processor's input/output ports for communicating with the SCCs and RTC. The address of these ports can be 8 or 16 bits long depending on the host processsor. Jumper W14 is used to choose between the two address lengths. Switches S2 and S3 then determine the address of the first port used on the I/O-8.

For most microprocessors, the input/output port addresses are eight bits long (SBC-200). Jumper W14 should be on pins one and two for this case. Switch S2 should have the five most significant bits of the address. With the card edge connector pointing down, address bit seven will be on the right. Switch S3 can be ignored for an eight bit port address.

Example: Initialize the I/O-8 board address to A0H for use with a Z80 processor.

A Z80 has an eight bit I/O address length; so W14 should be on pins one and two. The hexadecimal address A0H is equivalent to 10100000B in binary (consult the Appendix called Base Conversions). Ignore the rightmost three bits and place the remaining five bits on switch S2. For a zero in the address, turn the corresponding switch to the on position and, for a one, turn the switch to the off position. The switch setting for address A0H is shown in Figure 4-1. From the figure, note the order that the bits were placed onto S2.

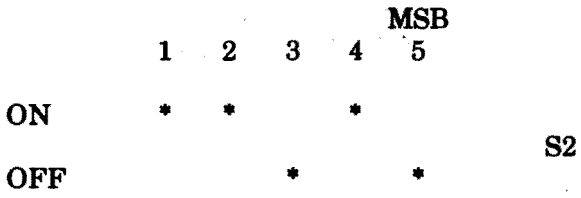


Figure 4-1. S2 SWITCH SETTING FOR ADDRESS A0H

If the host processor being used has 16 bit I/O port addresses (SBC-68000), jumper W14 should be on pins two and three. Switches S2 and S3 will have the 13 most significant bits of the address.

Example: Initialize the I/O-8 board address to 5C38H for use with a 68000 processor.

Since the 68000 processor has I/O port addresses that are 16 bits long, place jumper W14 on pins two and three. Converting the hexadecimal address 5C38H to binary gives 0101110000111000B (consult the Appendix called Base Conversions). Place the left eight bits on S3 and the next five bits on S2. Ignore the rightmost three bits. For a zero in the address, turn the corresponding switch to the on position and, for a one, turn the switch to the off position. Figure 4-2 gives the switch setting for address 5C38H. From the figure, note the order that the bits were placed onto S2 and S3.

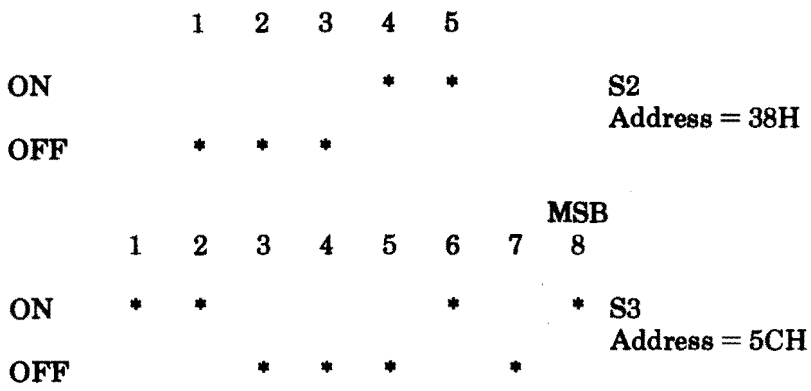


Figure 4-2. SWITCH SETTING FOR THE 16 BIT ADDRESS 5C38H

4.2 SERIAL INTERFACE OPTIONS

The I/O-8 has a flexible serial interface to allow it to communicate with a variety of devices. The following subsections describe the configurations for a terminal (DCE) or modem (DTE) and synchronous or asynchronous communications.

4.2.1 Terminal Or Modem Configuration (DCE or DTE)

RS-232 devices fall into two categories: Data Communications Equipment (DCE) and Data Terminal Equipment (DTE). The device being interfaced determines whether the I/O-8 should be configured for DCE or DTE. A terminal or printer needs a DCE interface and a modem needs a DTE interface. Consult the owner's manual for the device to determine which interface is needed.

All channels on the I/O-8 can support a DCE or DTE interface, but four partial channels (2, 4, 6, and 8) have only four signals available. When choosing which channel to use, the number of signals required must be allowed for. A terminal usually needs only two signals (RxD and TxD); so any of the eight channels can be used. A printer outputs a signal to indicate when it is ready to receive data, and may need certain other signals to be at a known level. One of the four full interface channels (1, 3, 5, or 7) has to be used in this case. Table 4-1 lists the signals available on each channel.

Table 4-1. RS-232 SIGNALS SUPPORTED BY THE I/O-8

Signal Name		Channels (1,3,5,7)	Channels (2,4,6,8)
TxD	(Transmit Data)	Yes	Yes
RxD	(Receive Data)	Yes	Yes
CTS	(Clear To Send)	Yes	Yes*
RTS	(Request To Send)	Yes	Yes*
DCD	(Data Carrier Detect)	Yes	No
DSR	(Data Set Ready)	Yes	No
DTR	(Data Terminal Ready)	Yes	No
RxCLK	(Receive Clock)	Yes	No
TxCLK	(Transmit Clock)	Yes	No
CLK	(Clock)	Yes	No
PGND	(Protective Ground)	Yes	Yes
SGND	(Signal Ground)	Yes	Yes

* A special cable is needed to use this signal in DCE mode.

The procedure for converting a channel from DCE to DTE or DTE to DCE varies depending on whether the channel interface is full or partial. Changing a full interface can usually be done by reversing a header. For the cases that use more signals in the interface (such as a printer), two jumper strips also have to be changed. A partial interface needs only one jumper strip changed to make the conversion. Table 4-2 lists the header, jumper strips, and cable connector associated with each channel.

Table 4-2. CONFIGURATION JUMPERS

Channel Number (Cable Connector)	Interface Jumpers	Sync Jumper	Header
1 (J2)	W1	W10	H1
2 (J3)	W2	--	--
3 (J4)	W3	W11	H2
4 (J5)	W4	--	--
5 (J6)	W5	W9	H3
6 (J7)	W6	--	--
7 (J8)	W7	W12	H4
8 (J9)	W8	--	--

NOTE: Use the Parts Placement Diagram Appendix to locate the headers and jumper strips on the I/O-8.

4.2.2 Asynchronous Communications

After choosing a DCE or DTE interface the communication format (synchronous or asynchronous) must also be chosen. The following subsections describe asynchronous communications on the I/O-8. Synchronous communications are described in Section IV, Subsection 4.2.3.

4.2.2.1 Full Interface Channels (1, 3, 5, 7)

At this point the type of interface (DCE or DTE) and the communications format (synchronous or asynchronous) have been chosen. This subsection describes how to set up a serial channel for DCE/asynchronous or DTE/asynchronous modes.

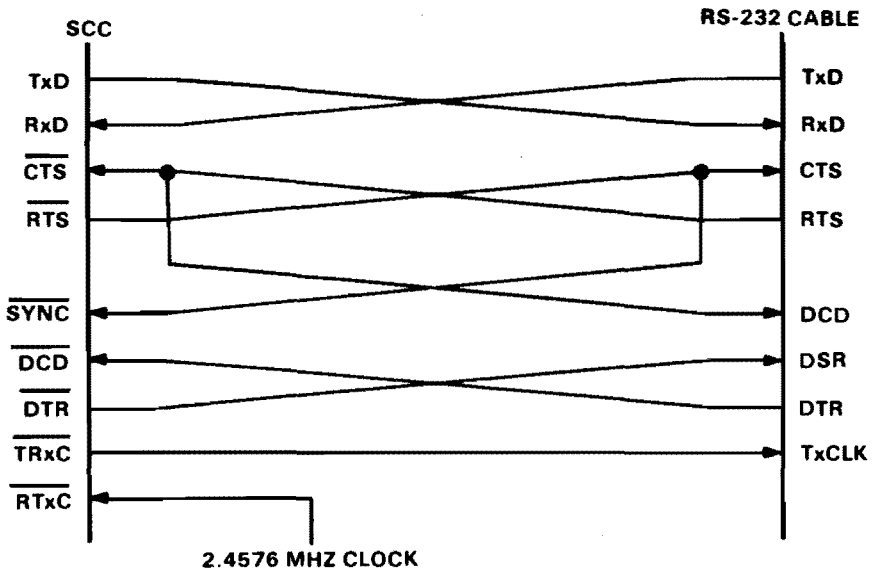
In most cases, such as terminals, the interface type can be chosen by how the header is inserted. From Table 4-2 find the header number for the channel being used (i.e. the header for Channel 7 is H4). Insert the header with the four straight wires pointing away from the heatsinks for a DCE interface (Figure 4-3). Insert the header with the four straight wires pointing toward the heatsinks for a DTE interface (Figure 4-4). The headers shown in Figures 4-3 and 4-4 may not be the same as on an early revision I/O-8 board. Always use the four parallel wires to determine the orientation of a header.

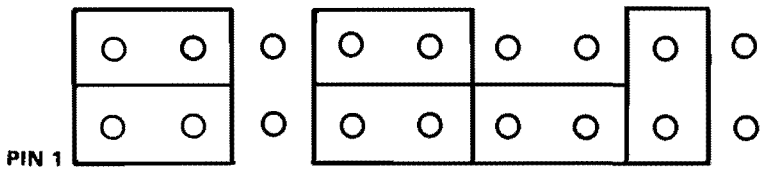
If the headers alone are being used to configure the serial channels, one jumper must be installed on the interface jumper strip. Find the jumper strip for the channel being configured from Table 4-2 (i.e. for Channel 3 the interface jumper strip is W3). Place a jumper on pins 6 and 7 of the interface jumper strip. Figure 4-5 shows the pin numbers of the interface jumper for Channels 1, 3, 5, and 7.

When a complete serial channel is needed the interface jumper strip and sync jumper have to be completely configured in addition to the header. Find the interface jumper strip from Table 4-2 for the channel being configured. If a DTE interface is being used, copy the jumper configuration shown in Figure 4-4 onto the interface jumper strip. If a DCE interface is being used, copy the jumper configuration shown in Figure 4-3 onto the interface jumper strip. While copying the jumper configurations, always hold the I/O-8 board with the card edge connector pointing down. This puts pin 1 of the interface jumpers in the lower lefthand corner. From Table 4-2 find which sync jumper is with the serial channel being configured. For the asynchronous mode, place a jumper on the sync jumper strip.

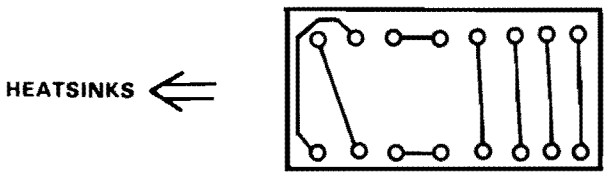
Example: Convert Channel 5 to work with a terminal.

A terminal would need a DCE/asynchronous interface. Inserting header H3 with the correct orientation should be sufficient but, for this example, a complete configuration will be done. Copy the jumper configuration from Figure 4-3 onto W5. Jumper strip W5 has been chosen because it is the interface jumper for Channel 5 (Table 4-2). Insert header H3 with the four straight wires pointing away from the heatsinks (Figure 4-3). Since the terminal is asynchronous, place a jumper on W9.



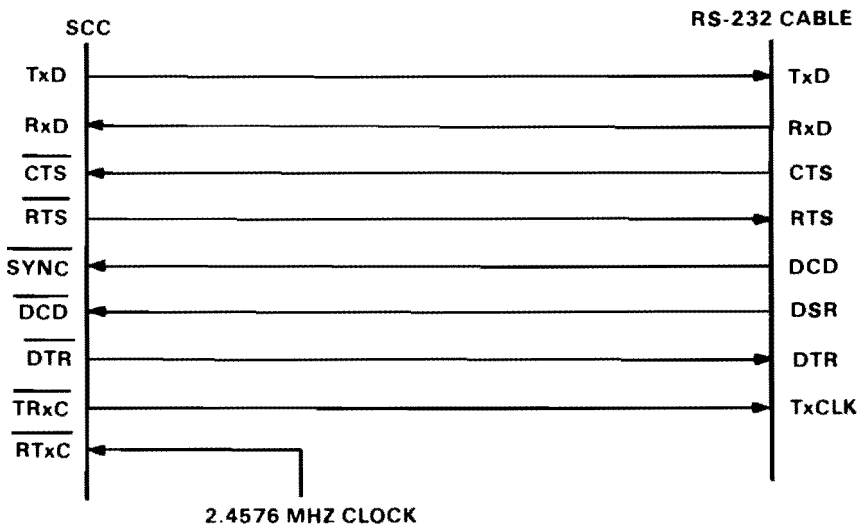


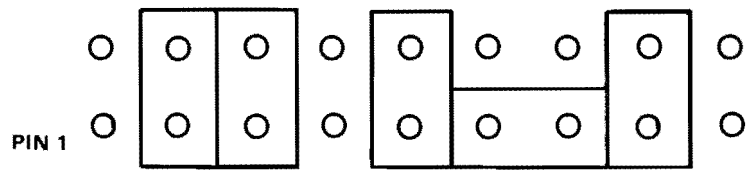
INTERFACE JUMPER CONFIGURATION



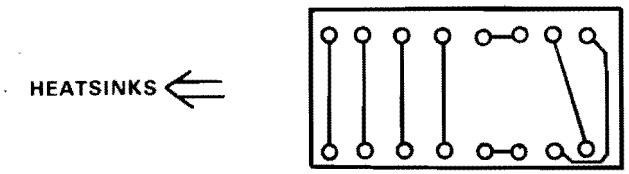
HEADER CONFIGURATION

Figure 4-3. DCE/ASYNCHRONOUS (TERMINAL) CONFIGURATION





INTERFACE JUMPER CONFIGURATION



HEADER CONFIGURATION

Figure 4-4. DTE/ASYNCHRONOUS (MODEM) CONFIGURATION

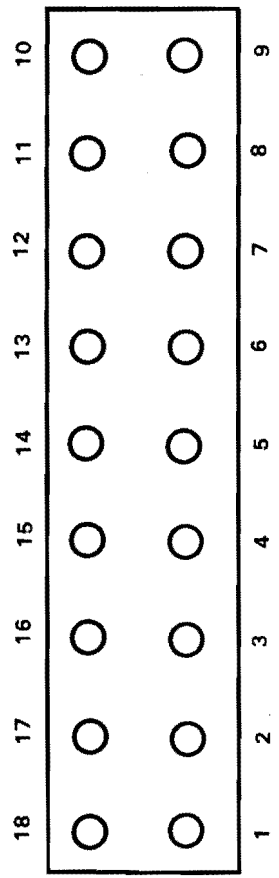


Figure 4-5. INTERFACE JUMPER STRIP PIN NUMBERS

4.2.2.2 Partial Interface Channels (2, 4, 6, 8)

Four channels on the I/O-8 have a partial serial interface that is intended to be used mainly with terminals. These channels are also limited to asynchronous communications only. Configuring a partial channel is done with one 2x3 jumper strip. The interface jumper strip associated with each channel is shown in Table 4-2 (i.e. for Channel 4 the interface jumper strip is W4). Hold the I/O-8 board with the card edge connector pointing down. This puts pin 1 of the interface jumper strip in the lower lefthand corner. Choose which interface type (DCE or DTE) is needed and copy the corresponding jumper configuration from Figure 4-6 onto the interface jumper strip.

Example: Convert channel number 2 to work with a modem.

A modem needs a DTE interface. Also, the modem must be asynchronous. (Use Channel 1, 3, 5, or 7 if the modem is synchronous.) Copy the DTE jumper configuration in Figure 4-6 onto W2.

4.2.2.3 Printers

The four full interface channels of the I/O-8 can be configured to work with a serial asynchronous printer. Most printers use one RS-232 line to indicate when the printer is ready to receive data. Use the printer owner's manual to determine which line is used for printer busy. Configuring the I/O-8 involves setting the header to DCE mode and connecting the printer busy line to the DCD input. The I/O-8 CP/M software drivers have been written to use the DCD input for printer busy. If special software drivers have been written, any of the input lines could be used.

4.2.3 Synchronous Communications

When using Z8530 SCCs, the I/O-8 can support several synchronous communication protocols on the full interface channels (1, 3, 5, and 7). As in asynchronous communications, a DCE or DTE type interface must be chosen. The main difference between an asynchronous configuration and a synchronous configuration is the clock sources. Either the I/O-8 or the device being interfaced must generate all of the clocks. Figures 4-7 and 4-8 show two sample synchronous configurations. In the DTE/synchronous configuration (Figure 4-7) the clock is generated by the interfaced device (i.e. synchronous modem) and received by the SCC at its RTxC pin. In the DCE/synchronous configuration (Figure 4-8) the baud rate clock is generated from the on-board 2.4576 mHz oscillator and transmitted to the interfaced device using the RS-232 CLK line. These

samples were chosen to satisfy the requirements of most synchronous devices. Consult the owner's manual to determine if the device needs a different arrangement of clocks. In addition to configuring the channel, a software driver has to be written that will operate the SCC in synchronous mode.

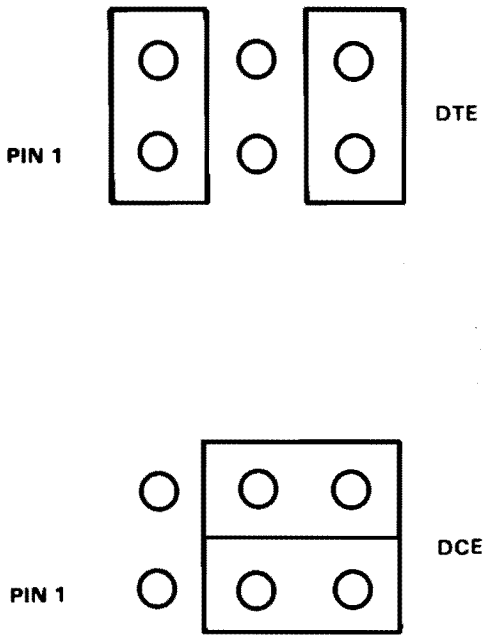


Figure 4-6. JUMPER CONFIGURATIONS FOR PORTS 2, 4, 6, AND 8

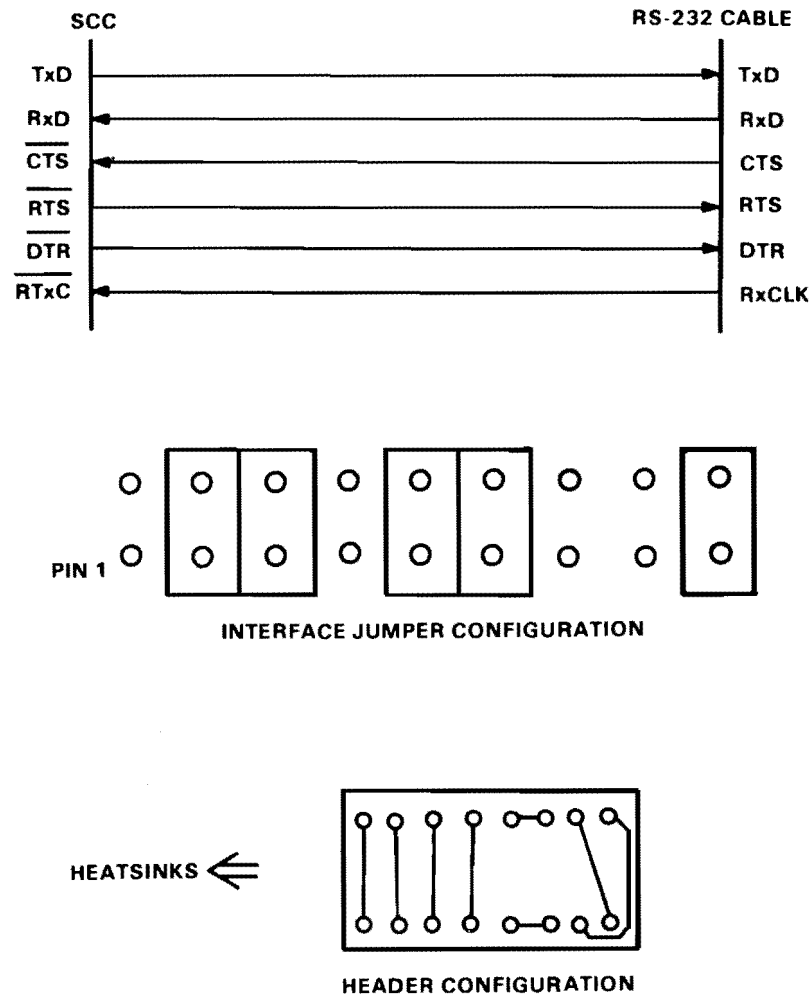
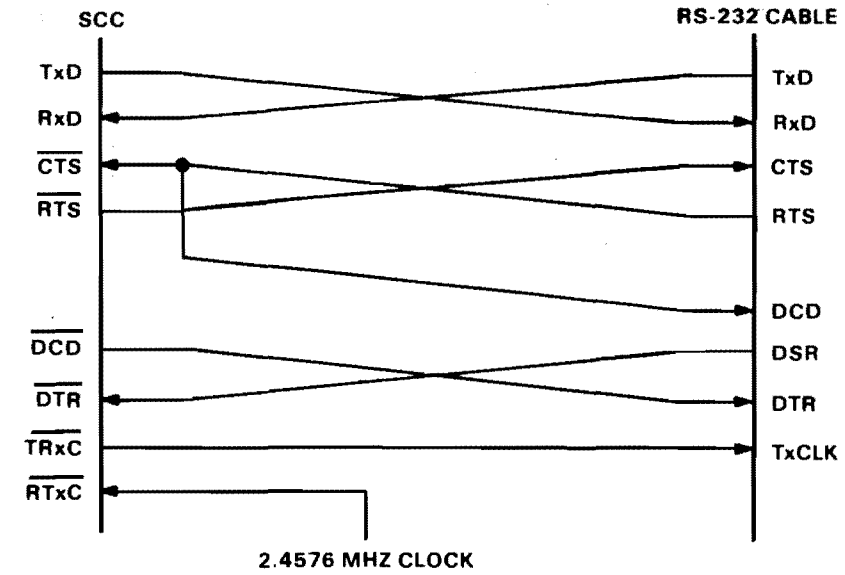
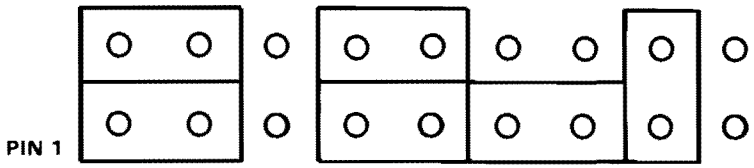
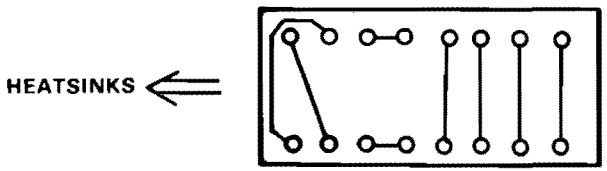


Figure 4-7. DTE/SYNCHRONOUS CONFIGURATION





INTERFACE JUMPER CONFIGURATION



HEADER CONFIGURATION

Figure 4-8. DCE/SYNCHRONOUS CONFIGURATION

Configuring a channel is done in the same manner for synchronous modes as in asynchronous modes. After choosing which channel to use (1, 3, 5, or 7), find the two jumper strips and the header for the channel from Table 4-2. Use Figure 4-7 or 4-8 depending on which interface type (DCE or DTE) has been chosen and insert the header with the orientation shown. Hold the I/O-8 board with the card edge connector pointing down and copy the jumper configuration onto the interface jumper strip (Table 4-2). Since the board is running in synchronous mode, a jumper should not be on the sync jumper strip. Find the sync jumper strip number from Table 4-2 and check to see that a jumper is not on it.

4.3 MISCELLANEOUS OPTIONS

Additional features of the I/O-8 include options for interrupts, reset compatibility, and the SCC wait function.

4.3.1 Interrupt Options

The I/O-8 board contains three interrupt sources: SCC, Real Time Clock, and the standby RTC. Each of the sources can be placed on any of nine S-100 interrupt lines (Vectored Interrupt 0 to 7 and Non-Maskable Interrupt). Table 4-3 shows which jumpers to insert to connect a given interrupt source to a given interrupt line.

Table 4-3. INTERRUPT JUMPER OPTIONS

	SCC INTERRUPTS	RTC INTERRUPTS	RTC STANDBY INTERRUPTS
VI0*	W15-1	W16-1	W17-1
VI1*	W15-2	W16-2	W17-2
VI2*	W15-3	W16-3	W17-3
VI3*	W15-4	W16-4	W17-4
VI4*	W15-5	W16-5	W17-5
VI5*	W15-6	W16-6	W17-6
VI6*	W15-7	W16-7	W17-7
VI7*	W15-8	W16-8	W17-8
NMI*	W15-9	W16-9	W17-9

4.3.2 RESET/SLAVE CLR Option

The IEEE-696 S-100 specification requires that all bus slaves be reset using the SLAVE CLR* signal. Older boards usually do not generate SLAVE CLR* but will reset bus slaves using the RESET* signal. A

jumper has been supplied to allow the I/O-8 to be reset using either the RESET* or SLAVE CLR* signal. If the host processor meets the IEEE-696 specification, place a jumper on pins two and three of W18; otherwise, place a jumper on pins one and two.

4.3.3 SCC Wait Line Option

A feature of the SCCs is a wait line that can signal the host processor when it is ready for more data. This makes it possible to use the Z80 OTIR instruction to transmit a long string of data. To enable this function, place a jumper on W13.

4.4 PORT ADDRESS MAP

The input/output port map for the I/O-8 is shown in Table 4-4.

Table 4-4. PORT ADDRESS MAP

Address	Port Address	Channels
A0	000*	Channel B - control (R/W)
A1	001	Channel A - control (R/W)
A2	010	Channel B - data (R/W)
A3	011	Channel A - data (R/W)
A4	100	SCC select (W)
A5	101	RTC register select - SCC status (R)
A6	110	RTC Data (R/W)
A7	111	Readable switch

* User supplies upper 13 bits with switches S2 and S3.
NOTE: The addresses given are the factory settings.

Port 0 - This port is used for programming Channel B of the selected SCC.

Port 1 - This port is used for programming Channel A of the selected SCC.

Port 2 - Data transmitted or received by Channel B of the selected SCC is sent through this port.

Port 3 - Data transmitted or received by Channel A of the selected SCC is sent through this port.

Table 4-4. PORT ADDRESS MAP--Continued

Port 4 - An SCC is selected by outputting its number (0-3) to this port. The six most significant bits are ignored.

Port 5 - This port will contain the five bit address of a Real Time Clock register. An input will give the RTC address in the lower five bits and the number of the selected SCC in the upper two bits (the format is shown in Table 4-5).

Port 6 - This port is for I/O to the RTC register that was selected by writing to port 5.

Port 7 - An eight bit readable switch is at this port. The value on this switch can be used to determine the system configuration.

Table 4-5. STATUS PORT FORMAT

D7	D6	D5	D4	D3	D2	D1	D0
CS2	CS1	X	A4	A3	A2	A1	A0
			RTC Register Address				

00 = SCC1 (Ports 1/2)
01 = SCC2 (Ports 3/4)
10 = SCC3 (Ports 5/6)
11 = SCC4 (Ports 7/8)

4.5 SCC PROGRAMMING

The Zilog 8530 Serial Communications Controller (SCC) contains 13 write registers for channel configuration and eight read registers for channel status. Four input/output ports are used on the I/O-8 to communicate with the SCCs. Two ports are for control (A0H and A1H) and directly communicate with the SCC register set for each serial channel. The remaining two ports (A2H and A3H) are used for transmitting and receiving data from the serial channels. See Section IV, Subsection 4.4 for a description of each port used by the SCCs.

Programming an SCC register is done in two operations. A write requires two write operations, and reading a register requires a write and a read. The first write is always to write register 0 (WR0) and contains three bits which select the next register to be accessed. The second operation (read or write) uses the register selected by the first write. The pointer bits are automatically cleared after the second operation so that WR0 and RR0 can be accessed again.

4.5.1 Asynchronous SCC Initialization

Seven of the SCC internal registers are used for asynchronous mode initialization (WR3, WR4, WR5, WR11, WR12, WR13, and WR14). Additional registers would be needed if interrupts were used. An example of the initialization routine described here is given in Section IV, Subsection 4.5.2. Also, the SCC Registers Appendix contains the format of all SCC registers.

The first register to be programmed is WR4. The bits in this register control parity, number of stop bits, and clock multiplier. In the example, parity is disabled, a x16 clock is used, and one stop bit is used. Storing a 44H in this register sets the chosen parameters (compare the 44H with register format in SCC Registers Appendix to see how the modes were selected). The second register to be programmed is WR3. Write register 3 contains bits for receiver enable, Auto Enables, and receive bits/character. The example uses C1H to enable the receiver, disable Auto Enables, and receive eight bits/character. WR5 sets transmit bits/character, transmitter enable, RTS, and DTR. The RTS and DTR bits directly control the corresponding RS-232 signals. Having a 1 in the DTR bit sets the RS-232 DTR high, and having a 0 does the converse. The example uses EAH to set the DTR and RTS signals high, enable the transmitter, and transmit 8 bits/character.

The remaining four registers are used to set the baud rate generator parameters. WR12 and WR13 contain the baud rate time constants. The values stored here are used to initialize a counter that determines the rate that data is transmitted or received. Table 4-6 lists the time constants needed to generate standard baud rates. WR11 determines the sources for the receive clock and transmit clock. Storing a 56H here causes the baud rate generator to be used for the receive and transmit clocks. Storing a 01H in WR14 enables the baud rate generator and sets the clock source for the BRG to the RTxC pin (2.4576 mHz oscillator). The BRG clock source could also be the host processor's system clock by storing a 03H in WR14.

Table 4-6. BAUD RATE GENERATOR TIME CONSTANTS

NOTE: The following values are given for the x16 clock using the on-board 2.4576 mHz clock

BAUD RATE	SCC REG 0CH (LOW BYTE)	SCC REG 0DH (HIGH BYTE)	ERROR (%)
50	FEH	05H	0
75	FEH	03H	0
110	68H	02H	+0.03
134.5	39H	02H	0
150	FEH	01H	0
300	FEH	00H	0
600	7EH	00H	0
1200	3EH	00H	0
1800	29H	00H	-0.80
2400	1EH	00H	0
3600*	13H	00H	+1.80
4800	0EH	00H	0
7200*	09H	00H	-3.70
9600	06H	00H	0
19200	02H	00H	0

* These baud rates are approximations as shown by the percentage of error.

4.5.2 Asynchronous Programming Example

```
;
;     ASYNCHRONOUS SCC PROGRAMMING EXAMPLE
;
;          FOR INSTRUCTIONAL PURPOSES ONLY
;
; Consult the following manuals for further information on SCC
; programming. Both manuals are from Zilog.
;
; Z8030/Z8530 Serial Communications Controller Technical Manual
; Z8530 and Z8030 SCC Initialization:  A Worksheet and an Example
;
;
; The port addresses used are for the standard factory settings.
;
        ld      a,00h           ; Select SCC number one
        out     (a4h),a
```

```
;
; Initialize Channel A of SCC one for asynchronous comm.
;
        ld      a,a1h       ; Program Channel A
        ld      c,a
        ld      b,0eh
        ld      h1,sccinit
        otir                ; output initialization table
                            ; to control port of Ch. A
;
; Transmit/ Receive routine
;
 next:  in      a,(a1h)     ; Check for received char.
        and     01h         ; Bit one contains SCC
                            ; received character flag
        jr      z,cont
        in      a,(a3h)     ; Input received char.
        ld      (temp),a    ; Save character
;
; Check for receive errors
;
        ld      a,01h
        out     (a1h),a     ; Point to error status reg.
        in      a,(a1h)
        and     70h         ; Zero all non-error bits
        call    nz,err      ; Call error handling routine
                            ; for receive error
        ld      a,(temp)    ; Restore data
        call    console     ; Output through console
;
; Note:  The console, getchar, and err subroutines are dependent on
;        the system being used.
;
; Transmit character
;
 cont:  in      a,(a1h)     ; Check for transmitter ready
        and     04h         ; Xmit status is in bit 3
        jr      z,next
; Transmitter is ready
        call    getchar     ; Get next char. to be transmitted.
        out     (a3h),a     ; Transmit data
        jr      next        ; Check for received char.
;
```

```
; SCC initialization table
;
sccinit: defb  04h      ; Point to WR4
         defb  44h      ; X16 clock
                        ; No parity
                        ; One stop bit
         defb  03h      ; Point to WR3
         defb  0c1h     ; Receive 8 bits/character
                        ; Enable Receiver
         defb  05h      ; Point to WR5
         defb  0eah     ; Transmit 8 bits/character
                        ; Set RTS and DTR outputs
                        ; Enable receiver
         defb  0bh      ; Point to WR11
         defb  56h      ; Receive clock = BRG
                        ; Transmit clock = BRG
                        ; TRxC outputs BRG
         defb  0ch      ; Point to WR12
         defb  06h      ; Init. low byte of time
                        ; constant for 9600 baud
         defb  0dh      ; Point to WR13
         defb  00h      ; Init. high byte of time
                        ; constant for 9600 baud
         defb  0eh      ; Point to WR14
         defb  01h      ; Use 2.4576 MHz clock
                        ; for baud rate generator
                        ; source
                        ; Enable BRG
```

4.6 REAL TIME CLOCK PROGRAMMING

The Real Time Clock uses 23 registers (counters or RAM) for programming. Reading or writing to any of these registers is done in two steps. First, the register address is written to port 5. Then an input or output operation to port 6 accesses the desired register. Table 4-7 lists the register functions and addresses.

Table 4-7. RTC REGISTER FUNCTIONS AND ADDRESS CODES

NOTE: All other addresses are unused

Function	RTC Register Address
Counter - .0001 of a second	00H
Counter - .01 & .1 of a second	01H
Counter - second	02H
Counter - minute	03H
Counter - hour	04H
Counter - day of the week	05H
Counter - day of the month	06H
Counter - month	07H
RAM - .0001 of a second	08H
RAM - .01 & .1 of a second	09H
RAM - second	0AH
RAM - minute	0BH
RAM - hour	0CH
RAM - day of the week	0DH
RAM - day of the month	0EH
RAM - month	0FH
Interrupt Status Register	10H
Interrupt Control Register	11H
Counter Reset	12H
RAM Reset	13H
Status Bit	14H
GO Command	15H
STANDBY INTERRUPT	16H
Test Mode	1FH

4.6.1 Real Time Counter

The real time counter consists of four-bit digits with two digits being used during any read or write cycle. The digits define a BCD number as shown in Table 4-8. During a read cycle, the unused bits remain at a logical zero and, during a write cycle, the unused bits are disregarded. An unused bit is defined as any bit that does not yield a full BCD number. As an example, there is a limit on the possible numbers for tens of hours. Tens of hours has an upper limit of two. Therefore, two bits are all that is necessary to define tens of hours. This leaves two bits unused. The unused bits appear in Table 4-8 as dashes.

Table 4-8. REAL TIME COUNTER FORMAT

(-- = Unused bits)

Counter Addressed	Units D0	D1	D2	D3	Max BCD Code	Tens D4	D5	D6	D7	Max BCD Code
1/10000 sec	--	--	--	--	0	D4	D5	D6	D7	9
1/100 & 1/10 sec	D0	D1	D2	D3	9	D4	D5	D6	D7	9
second	D0	D1	D2	D3	9	D4	D5	D6	--	5
minute	D0	D1	D2	D3	9	D4	D5	D6	--	5
hour	D0	D1	D2	D3	9	D4	D5	--	--	2
day of the week	D0	D1	D2	--	7	--	--	--	--	0
day of the month	D0	D1	D2	D3	9	D4	D5	--	--	3
month	D0	D1	D2	D3	9	D4	--	--	--	1

4.6.2 RAM

There are 56 bits of RAM located on the chip. The bits provide a means for storage whenever there is a power down. They also serve as an alarm latch for comparison with the real time counter. This comparison is performed on all digits except the ten thousandths of a second and the tens of days of the week. These digits are not used by the real time counter. When the two most significant bits of any RAM digit are ones, the RAM location will always match the real time counter.

The RAM has the same format as the real time counter. There are four bits per digit and a total of 14 digits. Every bit is used. The unused bits in the real time counter are compared to zeros in the RAM.

4.6.3 Interrupts

The MM58167A Microprocessor has two interrupt outputs. The INTERRUPT OUTPUT yields a true high signal. There are eight possible programmable interrupt rates: 10 per second, one per second, one per minute, one per hour, one per day, one per week, one per month, and during a RAM/real time counter comparison. The output is enabled when a one is written in the interrupt control register. The bit location is related to the desired output frequency. When one or more bits are set in the interrupt control register,

the counter's rollover will clock the interrupt status register and the interrupt output will go high. Read the interrupt status register to identify which frequency created the interrupt and to reset the interrupt. A one will appear in the bit position as an identifier for the interrupting frequency. The format of the interrupt status register and interrupt control register is given in Table 4-9. The interrupting frequency is identified by a one in the respective bit position.

Table 4-9. INTERRUPT CONTROL AND INTERRUPT STATUS REGISTER FORMAT

Bit	Interrupt Rates
D0	Counter/RAM comparison
D1	10 per second
D2	one per second
D3	one per minute
D4	one per hour
D5	one per day
D6	one per week
D7	one per month

The STANDBY INTERRUPT occurs when enabled and during a RAM/real time counter comparison. This interrupt mode is intended to be used when the host system is powered down. Additional circuitry would be needed for the host to recognize this signal and turn the power on to service the interrupt. The STANDBY INTERRUPT is enabled by writing a one into RTC register 16H and is disabled by writing a zero. The interrupt is triggered by the level of the compare signal. If the compare is enabled when the STANDBY INTERRUPT is first enabled the interrupt will turn on immediately.

4.6.4 Counter and RAM Resets

The set of digits (counter or RAM) to be reset is determined by the data that is written into the RTC reset registers. Table 4-10 shows the data format needed to reset desired counters or RAM. Writing to RTC register 12H resets counters and writing to RTC register 13H resets RAM. A logical one on the data bus will cause two digits to be reset. Resetting the most significant used bit of any counter will increment the following counter.

Table 4-10. RTC COUNTER AND RAM RESET FORMAT

NOTE: For counters, reset address = 12H
For RAM, reset address = 13H

Counter or RAM Reset	D0	D1	D2	D3	D4	D5	D6	D7
1/10000 sec	1	0	0	0	0	0	0	0
1/100 & 1/10 sec	0	1	0	0	0	0	0	0
second	0	0	1	0	0	0	0	0
minute	0	0	0	1	0	0	0	0
hour	0	0	0	0	1	0	0	0
day of the week	0	0	0	0	0	1	0	0
day of the month	0	0	0	0	0	0	1	0
month	0	0	0	0	0	0	0	1

4.6.5 RTC Programming Example

```
;       REAL TIME CLOCK PROGRAMMING EXAMPLE
;
;           FOR INSTRUCTIONAL PURPOSES ONLY
;
;
; All port addresses are for the standard factory settings.
;
;  a5h = real time clock address select register
;  a6h = real time clock access (using the register address in port
;        a5h)
;
; Reset all real time clock counters
;
        ld    . a,12h       ; Point to counter reset register
        out   (a5h),a
        ld    a,ffh         ; Reset all counters
        out   (a6h),a
;
; Reset all real time clock RAM locations
;
        ld    a,13h         ; Point to RAM reset register
        out   (a5h),a
        ld    a,ffh         ; Reset all RAM locations
        out   (a6h),a
```

```
;
; Set RTC minutes counter to six
;
        ld      a,03h
        out     (a5h),a         ; Point to RTC minutes counter
        ld      a,06h
        out     (a6h),a         ; Set minutes to six
;
; Set RTC hours counter to eight
;
        ld      a,04h
        out     (a5h),a         ; Point to RTC hours counter
        ld      a,08h
        out     (a6h),a         ; Set hours to eight
;
; Give RTC the 'GO' command
;
;  This command resets the seconds, tenths and milliseconds
;  counters. The name 'GO' is misleading since the clock is never
;  stopped.

        ld      a,15h           ; Point to 'GO' register
        out     (a5h),a
        ld      a,0h            ; Give 'GO' command
        out     (a6h),a
;
; Set interrupt rates for one per minute
;
        ld      a,11h           ; Point to interrupt control
        out     (a5h),a         ; register
        ld      a,08h
        out     (a6h),a         ; Set interrupts
;
; Initialize processor's interrupts
;
        di
        im      1               ; Set Z80 interrupt mode one
        ld      a,0c3h
        ld      (38h),a         ; Init interrupt vectors
        ld      hl,rdclk
        ld      a,1
        ld      (39h),a
        ld      a,h
        ld      (3ah),a
        ei
cont:   halt
        jr      cont            ; Wait for interrupts
```

```
;
; Real time clock read routine
;
rdclk:  ld      a,03h
        out     (a5h),a         ; Point to minutes counter
        in      (a6h),a         ; Read minutes counter and
        ld      (min),a         ; store at min.
;
; Read RTC hours counter
;
        ld      a,04h           ; Point to hours counter
        out     (a5h),a
        in      (a6h),a         ; Read hours counter and
        ld      (hour),a        ; store at hour
;
; Reset real time clock interrupts
;
        ld      a,10h           ; Point to interrupt status
        out     (a5h),a         ; register
        in      (a6h),a         ; Input from register resets
                                ; interrupt
        reti                    ; Return for next interrupt
```

SECTION V
ENVIRONMENTAL CONSIDERATIONS

5.0 GENERAL

The I/O-8 is contained on a printed circuit board of the size and characteristics defined by the IEEE-696 specification.

The I/O-8 regulates the voltages supplied by the IEEE-696 bus down to those DC voltages required by the on-board circuitry.

5.1 ELECTRICAL REQUIREMENTS AND SPECIFICATIONS

5.1.1 +5 Volt Regulation

The I/O-8 regulates the +8 volts (unregulated) input down to +5 volts with a maximum or ±.25 volts of ripple.

5.1.2 +12 Volt Regulation

The I/O-8 regulates the ±16 volts (unregulated) input down to ±12 volts with a maximum of ±.5 volts of ripple.

5.1.3 Current Consumption (Estimated)

±12 VDC	150 milliamperes (maximum) 114 milliamperes (typical)
-12 VDC	138 milliamperes (maximum) 108 milliamperes (typical)
+5 VDC	1.78 amps (maximum) 1.19 amps (typical)

5.1.4 Power Consumption (Estimated)

19.94 watts (maximum)
13.68 watts (typical)

5.2 PHYSICAL SPECIFICATIONS

5.2.1 Dimensions

The I/O-8 printed circuit board dimensions are 5.125 x 10.0 inches.

APPENDIX A
SELECTED IEEE-696 SPECIFICATION SHEETS

NOTE: For additional information, see the complete document "IEEE Standard 696 Interface Devices."

IEEE-696 bus pin list

Pin No.	Signal & Type	Active Level	Description
1	+8 V(B)		Instantaneous minimum greater than 7 V, instantaneous maximum less than 25 V, average maximum less than 11 V.
2	+16 V(B)		Instantáneous minimum greater than 14.5 V, instantaneous maximum less than 35 V, average maximum less than 21.5 V.
3	XRDY (S)	H	One of two ready inputs to the current bus master. The bus is ready when both these ready inputs are true. See pin 72.
4	VI0*(S)	L OC	Vectored interrupt line 0.
5	VI1*(S)	L OC	Vectored interrupt line 1.
6	VI2*(S)	L OC	Vectored interrupt line 2.
7	VI3*(S)	L OC	Vectored interrupt line 3.
8	VI4*(S)	L OC	Vectored interrupt line 4.
9	VI5*(S)	L OC	Vectored interrupt line 5.
10	VI6*(S)	L OC	Vectored interrupt line 6.
11	VI7*(S)	L OC	Vectored interrupt line 7.
12	NMI*(S)	L OC	Non-maskable interrupt.

Pin No.	Signal & Type	Active Level	Description
13	PWRFAIL*(B)	L	Power fail bus signal.
14	TMA3* (M)	L OC	Temporary master priority bit 3.
15	A18 (M)	H	Extended address bit 18.
16	A16 (M)	H	Extended address bit 16.
17	A17 (M)	H	Extended address bit 17.
18	SDSB* (M)	L OC	The signal to disable the 8 status signals.
19	CDSB* (M)	L OC	The signal to disable the 5 control output signals.
20	O V(B)		Common with pin 100.
21	NDEF		Not to be defined. Manufacturer must specify any use in detail.
22	ADSB* (M)	L OC	The signal to disable the address signals.
23	DODSB* (M)	L OC	The control signal to disable the data output signals. (D07-0 for 8 bit transfers, ED7-0 and 0D7-0 for 16 bit transfers.
24	ϕ (B)	A	The master timing signal for the bus.
25	pSTVAL*(M)	L	Status valid strobe.
26	pHLDA(M)	H	A control signal used in conjunction with HOLD* to coordinate bus master transfer operations.
27	RFU		Reserved for future use.

Pin No.	Signal & Type	Active Level	Description
28	RFU		Reserved for future use.
29	A5 (M)	H	Address bit 5.
30	A4 (M)	H	Address bit 4.
31	A3 (M)	H	Address bit 3.
32	A15 (M)	H	Address bit 15 (most significant for non-extended addressing).
33	A12 (M)	H	Address bit 12.
34	A9 (M)	H	Address bit 9.
35	DO1 (M)/ED1 (M/S)	H	Data out bit 1, bidirectional even data bit 1.
36	DO0 (M)/ED0 (M/S)	H	Data out bit 0, bidirectional even data bit 0.
37	A10 (M)	H	Address bit 10.
38	DO4 (M)/ED4 (M/S)	H	Data out bit 4, bidirectional even data bit 4.
39	DO5 (M)/ED5 (M/S)	H	Data out bit 5, bidirectional even data bit 5.
40	DO6 (M)/ED6 (M/S)	H	Data out bit 6, bidirectional even data bit 6.
41	DI2 (S)/OD2 (M/S)	H	Data in bit 2, bidirectional odd data bit 2.
42	DI3 (S)/OD3 (M/S)	H	Data in bit 3, bidirectional odd data bit 3.
43	DI7 (S)/OD7 (M/S)	H	Data in bit 7, bidirectional odd data bit 7.

Pin No.	Signal & Type	Active Level	Description
44	sM1 (M)	H	The status signal which indicates that the current cycle is an op-code fetch.
45	sOUT (M)	H	The status signal identifying the data transfer bus cycle to an output device.
46	sINP (M)	H	The status signal identifying the data transfer bus cycle from an input device.
47	sMEMR (M)	H	The status signal identifying bus cycles which transfer data from memory to a bus master, which are not interrupt acknowledge instruction fetch cycle(s).
48	sHLTA (M)	H	The status signal which acknowledges that a HLT instruction has been executed.
49	CLOCK(B)	A	2 MHz (+0.5%) 40-60% duty cycle. Not required to be synchronous with any other bus signal.
50	O V (B)		Common with pin 100.
51	+8 V(B)		Common with pin 1.
52	-16 V(B)		Instantaneous maximum less than -14.5 V, instantaneous minimum greater than -35 V, average minimum greater than -21.5 V.
53	O V(B)		Common with pin 100.

Pin No.	Signal & Type	Active Level	Description
54	SLAVE CLR* (B)	L OC	A reset signal to reset bus slaves. Must be active with POC* and may also be generated by external means.
55	TMA0* (M)	L OC	Temporary master priority bit 0.
56	TMA1* (M)	L OC	Temporary master priority bit 1.
57	DMA2* (M)	L OC	Temporary master priority bit 2.
58	sXTRQ* (M)	L	The status signal which requests 16-bit slaves to assert SIXTN*.
59	A19 (M)	H	Extended address bit 19.
60	SIXTN* (S)	L OC	The signal generated by 16-bit slaves in response to the 16 bit request signal sXTRQ*.
61	A20 (M)	H	Extended address bit 20.
62	A21 (M)	H	Extended address bit 21.
63	A22 (M)	H	Extended address bit 22.
64	A23 (M)	H	Extended address bit 23.
65	NDEF		Not to be defined signal.
66	NDEF		Not to be defined signal.
67	PHANTOM* (M/S)	L OC	A bus signal which disables normal slave devices and enables phantom slaves--primarily used for bootstrapping systems without hardware front panels.
68	MWRT (B)	H	pWR*-sOUT (logic equation). This signal must follow pWR* by not more than 30 ns.

Pin No.	Signal & Type	Active Level	Description
69	RFU		Reserved for future use.
70	O V(B)		Common with pin 100.
71	RFU		Reserved for future use.
72	RDY (S)	H OC	See comments for pin 3.
73	INT* (S)	L OC	The primary interrupt request bus signal.
74	HOLD* (S)	L OC	The control signal used in conjunction with pHLDA to coordinate bus master transfer operations.
75	RESET* (B)	L OC	The reset signal to reset bus master devices. This signal must be active with POC* and may also be generated by external means.
76	pSYNC (M)	H	The control signal identifying BS_1.
77	pWR* (M)	L	The control signal signifying the presence of valid data on DO bus or data bus.
78	pDBIN (M)	H	The control signal that requests data on the DI bus or data bus from the currently addressed slave.
79	A0 (M)	H	Address bit 0 (least significant).
80	A1 (M)	H	Address bit 1.
81	A2 (M)	H	Address bit 2.
82	A6 (M)	H	Address bit 6.

Pin No.	Signal & Type	Active Level	Description
83	A7 (M)	H	Address bit 7.
84	A8 (M)	H	Address bit 8.
85	A13(M)	H	Address bit 13.
86	A14(M)	H	Address bit 14.
87	A11(M)	H	Address bit 11.
88	DO2 (M)/ED2 (M/S)	H	Data out bit 2, bidirectional even data bit 2.
89	DO3 (M)/ED3 (M/S)	H	Data out bit 3, bidirectional even data bit 3.
90	DO7 (M)/ED7 (M/S)	H	Data out bit 7, bidirectional even data bit 7.
91	DI4 (S)/OD4 (M/S)	H	Data in bit 4 and bidirectional odd data bit 4.
92	DI5 (S)/OD5 (M/S)	H	Data in bit 5 and bidirectional odd data bit 5.
93	DI6 (S)/OD6 (M/S)	H	Data in bit 6 and bidirectional odd data bit 6.
94	DI1 (S)/OD1 (M/S)	H	Data in bit 1 and bidirectional odd data bit 1.
95	DI0 (S)/OD0 (M/S)	H	Data in bit 0 (least significant for 8 bit data) and bidirectional odd data bit 0.
96	sINTA (M)	H	The status signal identifying the bus input cycle(s) that may follow an accepted interrupt request presented on INT*.
97	sWO* (M)	L	The status signal identifying a bus cycle which transfers data from a bus master to a slave.

Pin No.	Signal & Type	Active Level	Description
98	ERROR* (S)	L OC	The bus status signal signifying an error condition during present bus cycle.
99	POC* (B)	L	The power-on clear signal for all bus devices; when this signal goes low, it must stay low for at least 10 millisecs.
100	O V(B)		System ground.

pin 1	+8 V(B)	
pin 2	+16 V(B)	
pin 3	XRDY (S)	H
pin 4	VI0* (S)	L
pin 5	VI1* (S)	L
pin 6	VI2* (S)	L
pin 7	VI3* (S)	L
pin 8	VI4* (S)	L
pin 9	VI5* (S)	L
pin 10	VI6* (S)	L
pin 11	VI7* (S)	L
pin 12	NMI* (S)	L
pin 13	PWRFAIL*(B)	L
pin 14	TMA3*(M)	L
pin 15	A18(M)	H
pin 16	A16(M)	H
pin 17	A17(M)	H
pin 18	SDSB*(M)	L
pin 19	CDSB*(M)	L
pin 20	O V	
pin 21	NDEF	
pin 22	ADSB*(M)	L
pin 23	DODSB*(M)	L
pin 24	ϕ(B)	A
pin 25	pSTVAL*(M)	L
pin 26	pHLDA (M)	H
pin 27	RFU	
pin 28	RFU	
pin 29	A5(M)	H
pin 30	A4(M)	H
pin 31	A3(M)	H
pin 32	A15(M)	H
pin 33	A12(M)	H
pin 34	A9(M)	H
pin 35	DO1(M)/ED1 (M/S)	H
pin 36	DOO(M)/EDO(M/S)	H
pin 37	A10(M)	H
pin 38	DO4(M)/ED4(M/S)	H
pin 39	DO5(M)/ED5(M/S)	H
pin 40	DO6(M)/ED6(M/S)	H
pin 41	DI2(S)/OD2(M/S)	H
pin 42	DI3(S)/OD3(M/S)	H
pin 43	DI7(S)/OD7(M/S)	H
pin 44	sM1(M)	H
pin 45	sOUT(M)	H
pin 46	sINP	H
pin 47	sMEMR	H
pin 48	sHLTA(M)	H
pin 49	CLOCK(B)	A
pin 50	OV	
pin 51	+8 V(B)	
pin 52	-16 V(B)	
pin 53	O V	
pin 54	SLAVE CLR* (B)	L
pin 55	TMAO*(M)	L
pin 56	TMA1*(M)	L
pin 57	TMA2*(M)	L
pin 58	sXTRQ*(M)	L
pin 59	A19(M)	H
pin 60	SIXTN*(S)	L
pin 61	A20(M)	H
pin 62	A21(M)	H
pin 63	A22(M)	H
pin 64	A23(M)	H
pin 65	NDEF	
pin 66	NDEF	
pin 67	PHANTOM*(M/S)	L
pin 68	MWRT (B)	H
pin 69	RFU	
pin 70	O V	
pin 71	RFU	
pin 72	RDY (S)	H
pin 73	INT*(S)	L
pin 74	HOLD*(M)	L
pin 75	RESET* (B)	L
pin 76	pSYNC(M)	H
pin 77	pWR*(M)	L
pin 78	pDBIN(M)	H
pin 79	AO (M)	H
pin 80	A1(M)	H
pin 81	A2(M)	H
pin 82	A6(M)	H
pin 83	A7(M)	H
pin 84	A8(M)	H
pin 85	A13(M)	H
pin 86	A14(M)	H
pin 87	A11(M)	H
pin 88	DO2(M)/ED2(M/S)	H
pin 89	DO3(M)/ED3(M/S)	H
pin 90	DO7(M)/ED7(M/S)	H
pin 91	DI4(S)/OD4(M/S)	H
pin 92	DI5(S)/OD5(M/S)	H
pin 93	DI6(S)/OD6(M/S)	H
pin 94	DI1(S)/OD1(M/S)	H
pin 95	DI0(S)/ODO(M/S)	H
pin 96	sINTA(M)	H
pin 97	sWO*(M)	L
pin 98	ERROR*(S)	L
pin 99	POC*(B)	
pin 100	O V	

APPENDIX B
BASE CONVERSIONS

HEXADECIMAL TO DECIMAL

HEXADECIMAL COLUMNS											
6		5		4		3		2		1	
HEX	DEC	HEX	DEC	HEX	DEC	HEX	DEC	HEX	DEC	HEX	DEC
0	0	0	0	0	0	0	0	0	0	0	0
1	1,048,576	1	65,536	1	4,096	1	256	1	16	1	1
2	2,097,152	2	131,072	2	8,192	2	512	2	32	2	2
3	3,145,728	3	196,608	3	12,288	3	768	3	48	3	3
4	4,194,304	4	262,144	4	16,384	4	1,024	4	64	4	4
5	5,242,880	5	327,680	5	20,480	5	1,280	5	80	5	5
6	6,291,456	6	393,216	6	24,576	6	1,536	6	96	6	6
7	7,340,032	7	458,752	7	28,672	7	1,792	7	112	7	7
8	8,388,608	8	524,288	8	32,768	8	2,048	8	128	8	8
9	9,437,184	9	589,824	9	36,864	9	2,304	9	144	9	9
A	10,485,760	A	655,360	A	40,960	A	2,560	A	160	A	10
B	11,534,336	B	720,896	B	45,056	B	2,816	B	176	B	11
C	12,582,912	C	786,432	C	49,152	C	3,072	C	192	C	12
D	13,631,488	D	851,968	D	53,248	D	3,328	D	208	D	13
E	14,680,064	E	917,504	E	57,344	E	3,584	E	224	E	14
F	15,728,640	F	983,040	F	61,440	F	3,840	F	240	F	15
0 1 2 3		4 5 6 7		0 1 2 3		4 5 6 7		0 1 2 3		4 5 6 7	
BYTE				BYTE				BYTE			

HEXADECIMAL TO BINARY

HEXADECIMAL	BINARY
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001
A	1010
B	1011
C	1100
D	1101
E	1110
F	1111

APPENDIX D
DISCLAIMER

SDSystems, INC. makes no representations or warranties with respect to the contents hereof and specifically disclaims any implied warranties of merchantability or fitness for any particular purpose. Further, SDSystems, INC. reserves the right to revise this publication and to make changes from time to time in the content hereof without obligation of SDSystems, INC. to notify any person of such revisions or changes.

APPENDIX E
LIMITED WARRANTY

All SDSystems printed circuit board assemblies are warranted for a period of one (1) year from date of invoice to be free from defects of material and workmanship.

Should an SDSystems board fail to perform to specifications, obtain a Return Material Authorization (RMA) number from your distributor or from SDSystems. Include this number in all correspondence and with the returned product. Ship the item prepaid to SDSystems and it will, at our option, be repaired or replaced free of charge provided the unit is received during the warranty period.

In order to validate this warranty, the enclosed warranty card must be returned to SDSystems. If no warranty card is on file at the time of product return, dated proof of purchase will be required.

This warranty is invalid if product has been misused or improperly modified. Modifications documented in the SDSystems unit publications may be performed without invalidating the warranty. All other modifications will invalidate the warranty. Warranty is limited to replacement of defective parts and no responsibility is assumed for damage to other equipment.

SDSYSTEMS MAKES NO WARRANTIES, GUARANTEES, OR REPRESENTATIONS, EXPRESSED OR IMPLIED, WITH RESPECT TO THE PRODUCTS COVERED HEREBY, EXCEPT AS EXPRESSED HEREIN, AND BUYER EXPRESSLY WAIVES ANY OTHER WARRANTIES, GUARANTEES, OR REPRESENTATIONS INCLUDING, BUT NOT LIMITED TO, ANY IMPLIED WARRANTY OF MERCHANTABILITY OR FITNESS FOR USE. SDSYSTEMS NEITHER ASSUMES NOR AUTHORIZES ANY OTHER PERSON TO ASSUME FOR SELLER ANY OTHER LIABILITIES IN CONNECTION WITH THE SALE OF THE PRODUCTS. IN NO EVENT WILL SDSYSTEMS BE LIABLE FOR ANY SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES.

APPENDIX F
SAMPLE DRIVER

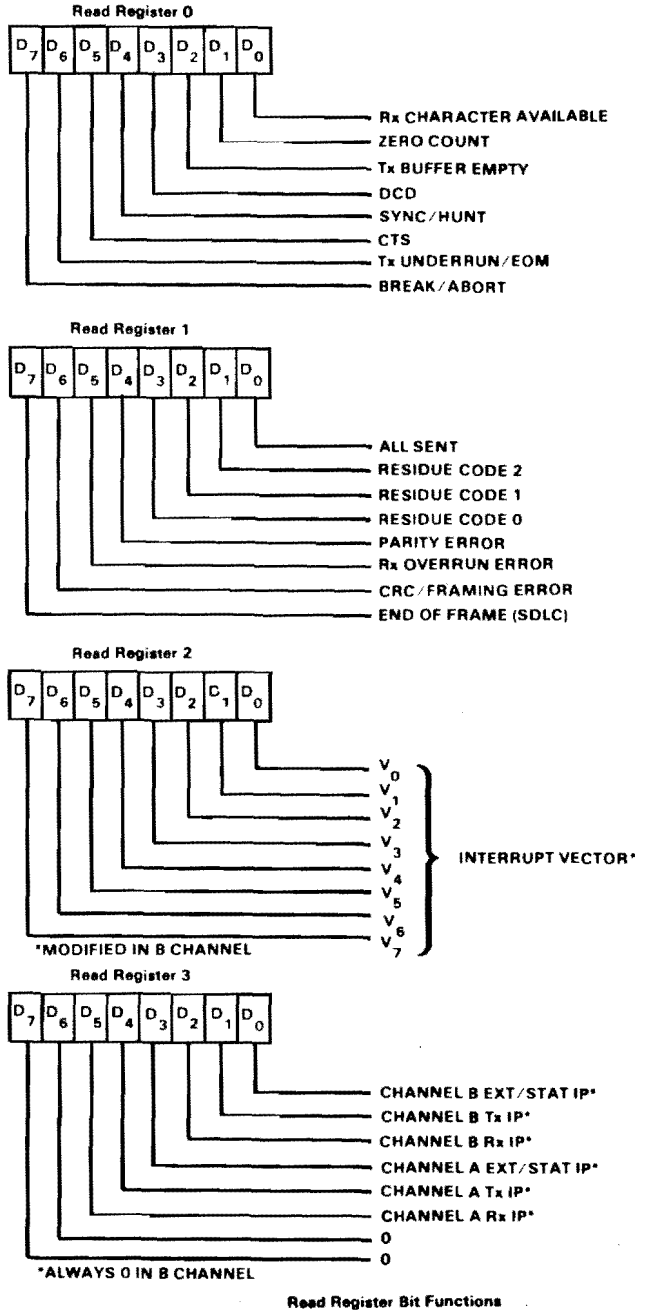
```
;     The MM58167A Clock Chip does not keep track of years or of leap years.
; The following clock driver is intended to be a perpetual calendar based on
; the CP/M Plus method of determining the date. CP/M Plus stores the time
; and date in a data area (the System Control Block) and calls "?time" prior
; to reading this data area. The contents of the "C" register determine if the
; date is to be set or read (C=0, read the time; C=1, set the time). If setting the
; time, the data area must be loaded prior to calling ?time. If reading the
; time, the data area will be updated after calling ?time.

;     This routine will maintain the correct date for 365 days after the last
; access to ?time (either setting or reading) with the power off. However, it is
; strongly recommended that if power has been off for a relatively long
; period of time, the time may need to be corrected slightly, depending on
; how critical precise time may be to the user. The date will be correct.

;     If CP/M Plus is being used, the clock driver below may be used as is and
; inserted in the boot module in place of the return instruction at ?time. If
; another system is being used, the following data area must be set up and
; the appropriate labels declared external in the ?time routine:

;       @date:  dw   00    ; 16-bit binary value denoting the number of
;                          ; days since January 1, 1978. A conversion
;                          ; routine must be written to convert this
;                          ; number to the correct date (MM/DD/YY).
;       @hour   db   0     ; 8-bit (2-BCD values) 10's of hours, hours.
;       @min    db   0     ; 8-bit (2-BCD values) 10's of minutes, minutes
;       @sec    db   0     ; 8-bit (2-BCD values) 10's of seconds, seconds

;-----------------------------------------------------------------------
;                    REAL TIME CLOCK DRIVER FOR IO-8

                maclib  z80
                public  ?time
                extrn   @date,@hour,@min,@sec

00A5 =  rtcrs   equ     0A5h        ; factory set clock register select port addr
00A6 =  rtc     equ     0A6h        ; factory set clock I/O port address

        ?time:
0000 E5D579             push h ! push d ! mov a,c            ; save regs
0003 21BF00E5           lxi h,timout ! push h               ; set return address
0007 0EA60601           mvi c,rtc ! mvi b,1                 ; set c for port, b as filler
000B B7CA4400           ora a ! jz gettm                    ; is it set or get?

        settm:
000F AF320000           xra a ! sta @sec                    ; set seconds to zero
0013 3E15D3A5           mvi a,15h ! out rtcrs ! outp b      ; start clock at 0.000 seconds
0017+ED41
0019 3E03D3A52A         mvi a,03h ! out rtcrs ! lhld @hour ! outp h    ; set the minutes
0020+ED61
0022 3E04D3A5           mvi a,04h ! out rtcrs ! outp l      ; set the hours
0026+ED69
0028 3E06D3A5   updt:   mvi a,06h ! out rtcrs ! outp b      ; set d-o-m to 1
002C+ED41
```

```
002E 3E07D3A5          mvi a,07h ! out rtcrs ! outp b          ; set month to 1
0032+ED41
0034 2A0000            lhld @date                              ; get date from scb
0037 3E0BD3A5          mvi a,0Bh ! out rtcrs ! outp l          ; store msb in clk ram (min
003B+ED69
003D 3E0CD3A5          mvi a,0Ch ! out rtcrs ! outp h          ; store lsb in clk ram (hrs)
0041+ED61
0043 C9                ret                                     ; return to caller

                gettm:
0044 3E06D3A5          mvi a,06h ! out rtcrs ! inp a           ; get day from clock
0048+ED78
004A 21000077AF        lxi h, @date ! mov m,a ! xra a ! rld    ; save it and clear a
004F+ED6F
0051 875F8787          add a ! mov e,a ! add a ! add a         ; high nibble x2, save in e
0055 835FAF            add e ! mov e,a ! xra a ! rld           ; high nib x10, save in e
0058+ED6F
005A 833D5F            add e ! dcr a ! mov e,a                 ; day is now binary (0-30)
005D 3E0BD3A5          mvi a,0Bh ! out rtcrs ! inp l           ; read stored date msb
0061+ED68
0063 3E0CD3A5          mvi a,0Ch ! out rtcrs ! inp h           ; read stored date lsb
0067+ED60
0069 7B856F7C          mov a,e ! add l ! mov l,a ! mov a,h     ; add days to stored date
006D CE0067            aci 0 ! mov h,a
0070 3E07D3A5          mvi a,07h ! out rtcrs ! inp a           ; get month from clock
0074+ED78
0076 F5E6F0            push psw ! ani 0F0h                     ; save and check low nibble
0079 EB21C200          xchg ! lxi h,montbl ! jrz domon
007D+2806
007F 21D400F13C        lxi h,montbl+18 ! pop psw ! inr a ! push psw
                domon:
0085 F1E60F            pop psw ! ani 0Fh
0088 3DCA900023dmon1:  dcr a ! jz stuff ! inx h ! inx h        ; index into table
                       jr dmon1
008E+18F8
                stuff:
0090 7E23666F          mov a,m ! inx h ! mov h,m ! mov l,a
0094 19                dad d
0095 220000            shld @date                              ; put in scb for CP/M
0098 CD2800            call updt                               ; refresh date cycle
009B 3E04D3A5  rerd:   mvi a,04h ! out rtcrs ! inp l           ; read the hours
009F+ED68
00A1 3E03D3A5          mvi a,03h ! out rtcrs ! inp h           ; read the minutes
00A5+ED60
00A7 220000            shld @hour                              ; put in scb for CP/M
00AA 3E02D3A5          mvi a,02h ! out rtcrs ! inp d           ; read the seconds
00AE+ED50
00B0 3E01D3A5DB        mvi a,01h ! out rtcrs ! in rtc          ; read tenth's of sec.
00B6 E6F0              ani 0F0h ! jrz rerd                     ; if tenth's=0, reread
00B8+28E1
00BA 7A320000          mov a,d ! sta @sec                      ; ok- restore seconds
00BE C9                ret                                     ; return
                timout:
00BF D1E1              pop d ! pop h                           ; restore registers
00C1 C9                ret                                     ; return to caller

00C2 00001F003Bmontbl: dw    0,31,59,90,120,151,181,212,243,273,304,334

00DA                   end
```

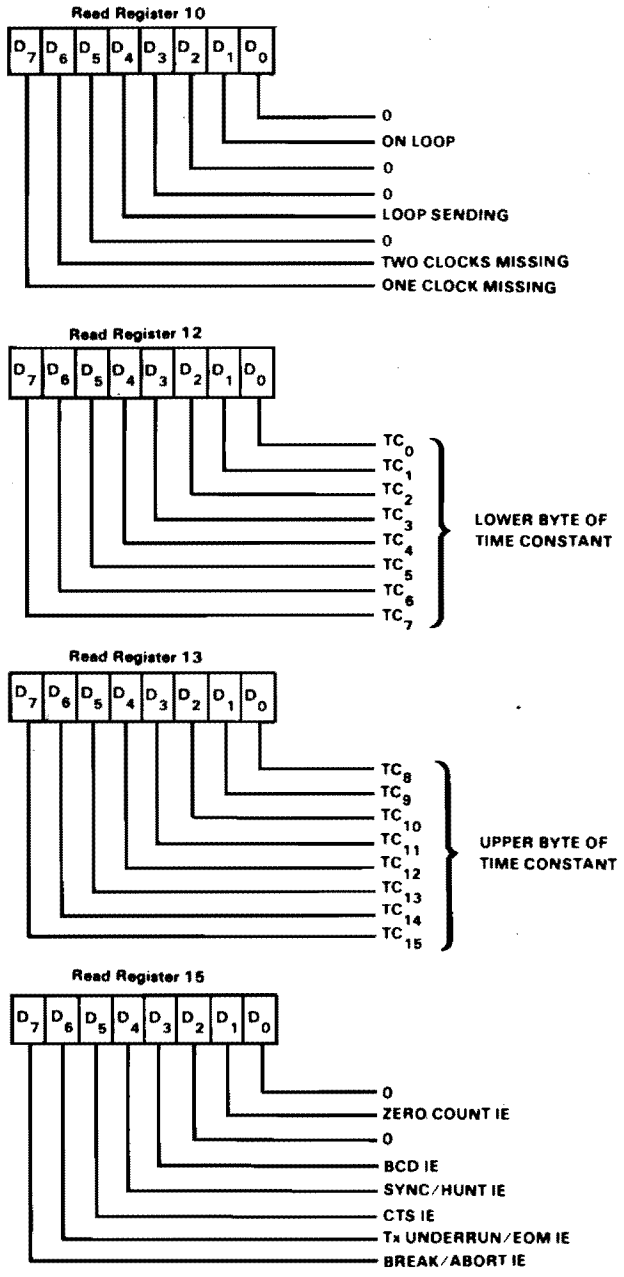
APPENDIX G
SCC REGISTERS

	READ REGISTER FUNCTION		WRITE REGISTER FUNCTION
RR0	Transmit/Receive buffer status, and External status	WRO	Command Register, (Register Pointers, Z8530 only), CRC initialization, resets for various modes
RR1	Special Receive Condition status, residue codes, error conditions	WR1	Interrupt conditions, Wait/DMA request control
RR2	Modified (Channel B only) interrupt vector and Un-modified interrupt vector (Channel A only)	WR2	Interrupt vector (access through either channel)
RR3	Interrupt Pending bits (Channel A only)	WR3	Receive/Control parameters, number of bits per character, Rx CRC enable
		WR4	Transmit/Receive miscellaneous parameters and modes, clock rate, number of sync characters, stop bits, parity
		WR5	Transmit parameters and controls, number of Tx bits per character, Tx CRC enable
		WR6	Sync character or SDLC address field (1st byte)
		WR7	Sync character or SDLC flag (2nd byte)
RR8	Receive buffer	WR8	Transmit buffer
		WR9	Master interrupt control and reset (accessed through either channel), reset bits, control interrupt daisy chain
RR10	Miscellaneous XMTR, RCVR status parameters	WR10	Miscellaneous transmitter/ receiver control bits, NR2I, NR2, FM encoding, CRC reset
		WR11	Clock mode control, source of Rx and Tx clocks

READ REGISTER FUNCTION		WRITE REGISTER FUNCTION	
RR12	Lower byte of baud rate generator time constant	WR12	Lower byte of baud rate generator time constant
RR13	Upper byte of baud rate generator time constant	WR13	Upper byte of baud rate generator time constant
		WR14	Miscellaneous control bits: baud rate generator, Phase-Locked Loop control, auto echo, local loopback
RR15	External/Status interrupt control information	WR15	External/Status interrupt control information-control external conditions causing interrupts

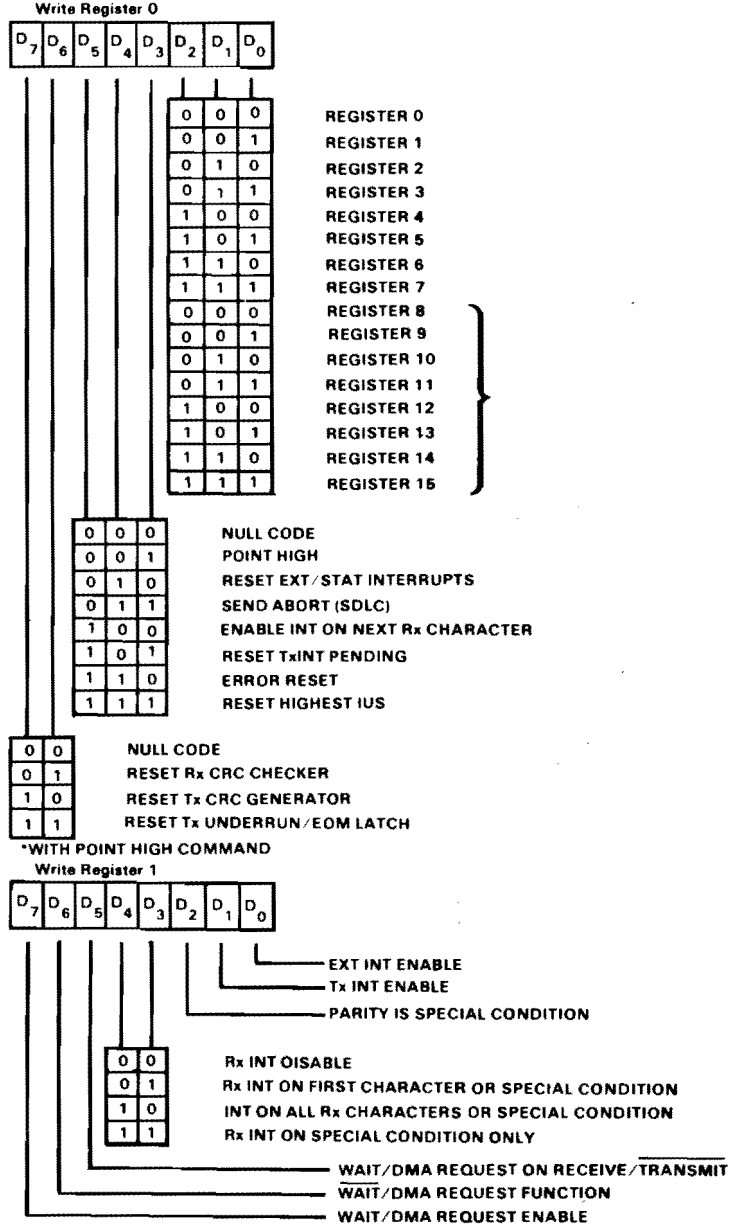


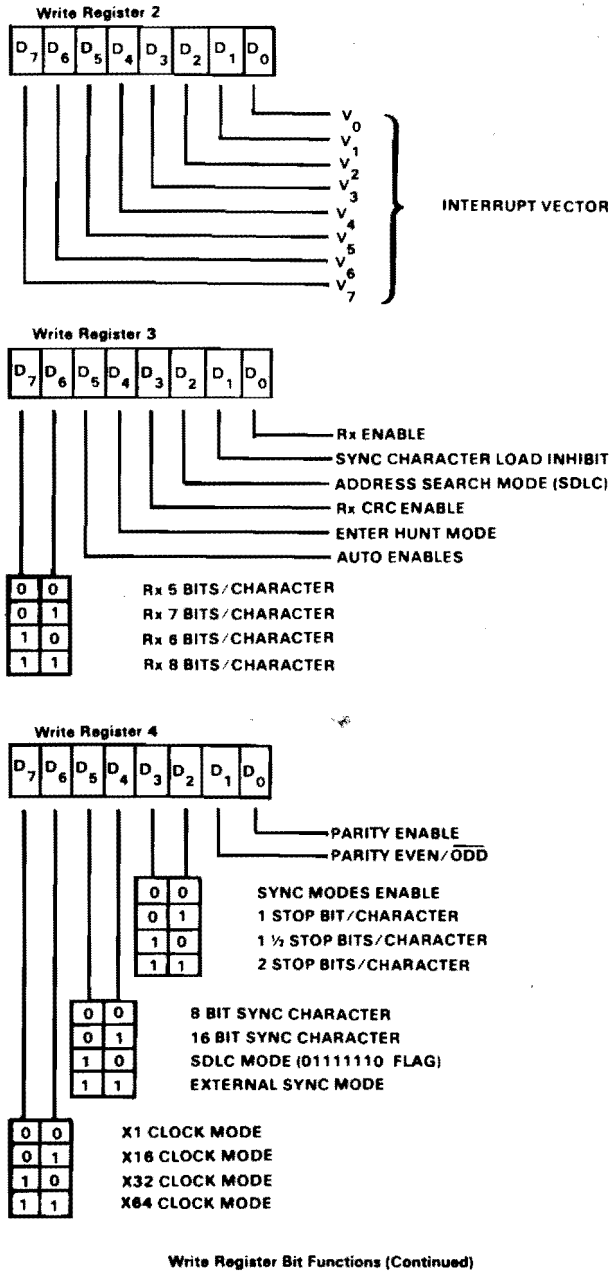
Read Register Bit Functions



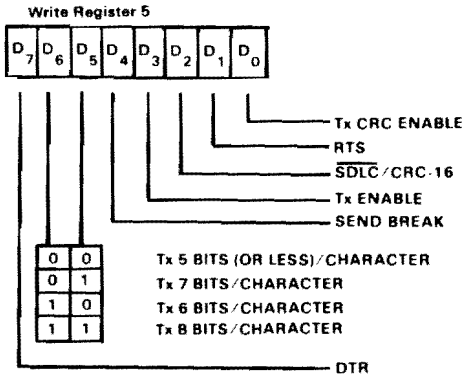
Read Register Bit Functions (Continued)

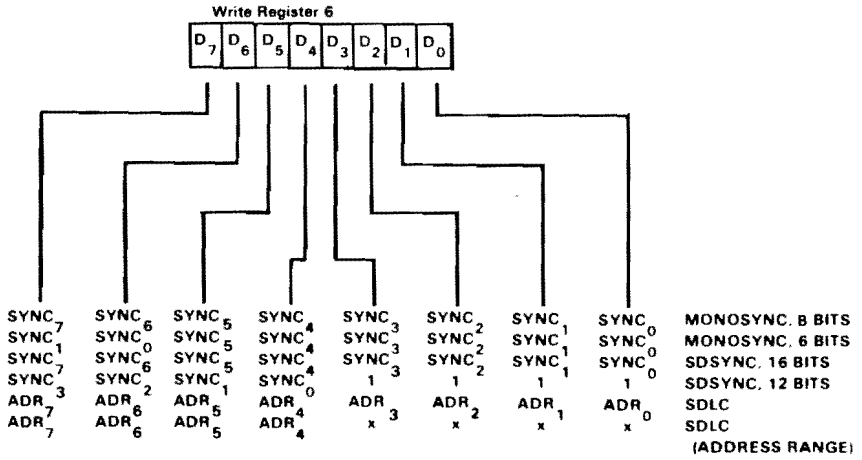
Write Registers. The SCC contains 13 write registers (14 counting WR8, the transmit buffer) in each channel. These write registers are programmed separately to configure the functional "personality" of the channels. In addition, there are two registers (WR2 and WR9) shared by the two channels the may be accessed through either of them. WR2 contains the interrupt vector for both channels, while WR9 contains the interrupt control bits.



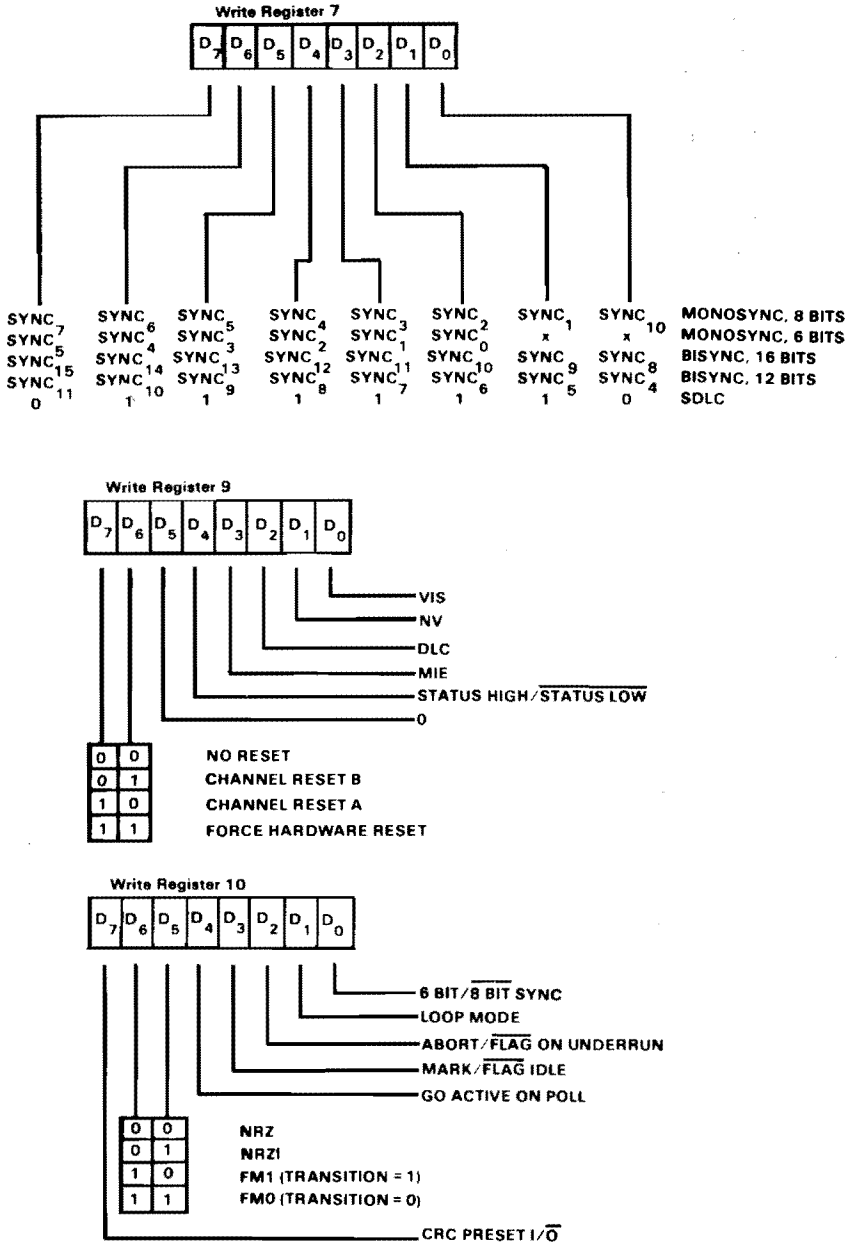


Write Register Bit Functions (Continued)

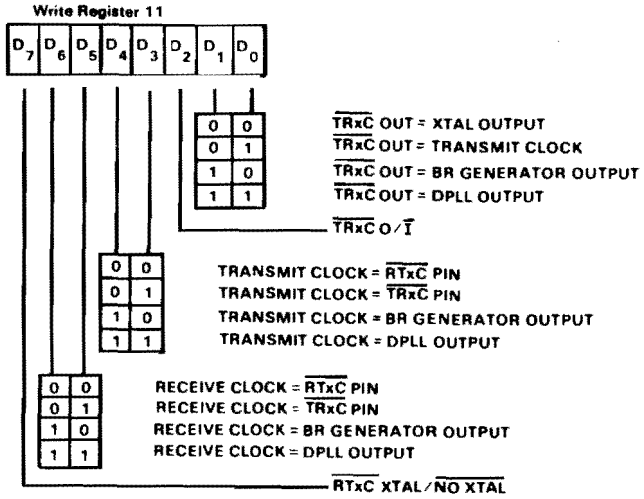


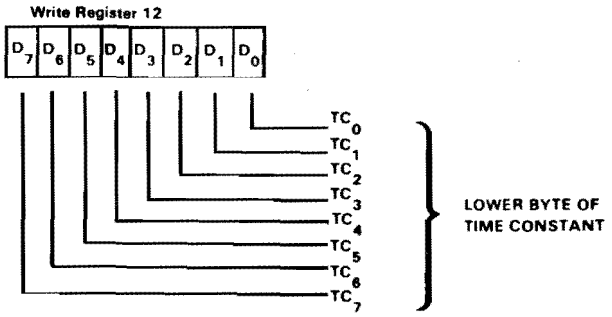


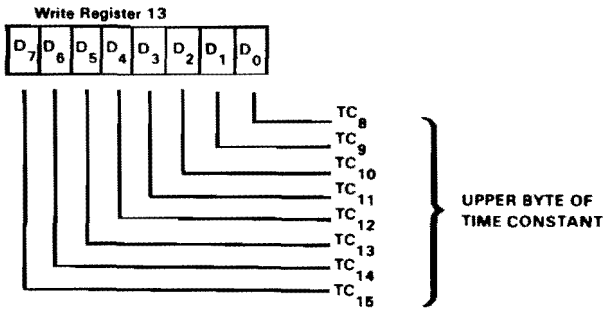
Write Register Bit Functions (Continued)



Write Register Bit Functions (Continued)

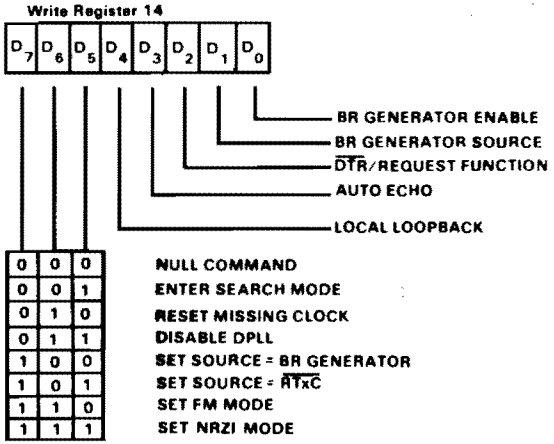






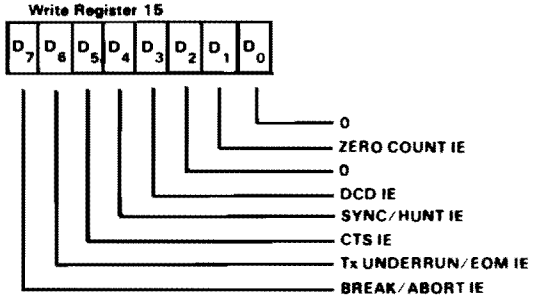
Write Register Bit Functions (Continued)

Write Register 14



0	0	0	NULL COMMAND
0	0	1	ENTER SEARCH MODE
0	1	0	RESET MISSING CLOCK
0	1	1	DISABLE DPLL
1	0	0	SET SOURCE = BR GENERATOR
1	0	1	SET SOURCE = $\overline{RTxC}$
1	1	0	SET FM MODE
1	1	1	SET NRZI MODE

Write Register 15



Write Register Bit Functions (Continued)

APPENDIX H
PARTS LIST FOR I/O-8

QTY REQD	DESCRIPTION	PART/SUB NUMBER	DESIGNATION
	I/O-8 ASCC		
6	IC, 75188 OR MC1488	7010332	U6, U9, U11, U12, U14, U15
8	IC, 75189 OR MC1489	7010333	U2-U5, U7, U8, U10, U13
2	IC, 74LS04	7010164	U28, U31
2	IC, 74LS32	7010181	U22, U25
1	IC, 74LS10	7010168	U29
1	IC, 74LS08	7010166	U32
1	IC, 74LS139	7010220	U23
2	IC, 74LS74	7010195	U21, U24
1	IC, 74LS174	7010241	U26
1	IC, 74LS682	7010518	U38
6	IC, 74LS244	7010264	U1, U33, U35, U37, U39, U40
4	8531A 6 MHZ ASCC	7010520	U16-U19
1	CLOCK CHIP, 58167-A	7010502	U27
1	DM8131	7010513	U34
4	SOCKET 16 PIN GOLD PL	7060023	FOR H1-H4
1	IC, 3.3K 6 PIN RES SIP	7010344	RN2
3	RES. 3.3K OHM 1/4W 5%	7020085	R2, R3, R12
4	RES. 1K OHM 1/4W 5% CC	7020073	R1, R5, R9, R11
1	RES. 1.8K 1/4W 5%	7020079	R10
2	RES. 10K OHM 1/4W 5% CC	7020097	R4, R13
5	CAP 10MF 16V TANT	7030009	C2, C5, C16, C22, C31
2	10 μF CAPACITOR	7030080	C17, C32
23	.1 μF CAPACITOR	7030068	C1, C3, C4, C6-C15, C18-C21, C25, C27-C30, C33
3	22 PF CAPACITOR	7030067	C23, C24, C26
1	7812 LM 340T-12.0	7160003	VR3
1	3.6V ZENER	7040044	CR2
1	REG LM7912 -12V 1A	7160010	VR2

QTY REQD	DESCRIPTION	PART/SUB NUMBER	DESIGNATION
	I/O-8 ASCC (Continued)		
1	DIODE 1N914.1N4148	7040001	CR1
1	TRANS NPN 2N2222	7040005	Q1
1	AST 2907 PNP TRAN	7040009	Q2
1	78H05	7160012	VR1
1	32.768 KHZ .002% CRYSTAL	7080016	Y2
2	8 POSITION DIP SWITCH	7050002	S1, S3
1	5 POSITION DIP SWITCH	7050001	S2
4	HEADER ASSY	0100736	H1-H4
5	BERG 1X2 STR .230 PIN TN	7170018	W9-W13
1	HEATSINK TMH 6103-B	7130004	
2	6-32 X 3/8 PPH SCREW	7130006	
2	6-32 NUT	7130007	
1	ELECTROCHEM BATTERY	7210205	
1	HEATSINK TMH 6106-13	7130003	
7	BERG STIK ST 2X9 TIN PL	7170097	W1, W3, W5, W7, W15-W17
8	BERG STIK ST 2X13 TIN PL	7170098	J2-J9
1	FAB PCB I/O-8	7000067	
0	I/O-8 SCHEMATIC	0300724	
2	BERG 1X3 STR .230 PIN TN	7170021	W14, W18
4	BERG 2X3 STR .318 PIN AG	7170034	W2, W4, W6, W8
1	IC, 3.3K 10 PIN RES SIP	7010345	RN1
2	RES 470 OHM 1/4W 5%	7020065	R7, R8
1	CRYSTAL, 2.4576 MHZ	7080023	Y1
0	I/O-8 TST PROCEDURE	0900724	
1	7407 HEX BUFFERS	7010008	U20
2	LOCKWASHER #6	7130009	
2	NYLON SCREW #4X3/8	7130070	
2	NYLON HEX NUT #4	7130071	
2	MYLAR INSULATORS	7160011	
2	PCB EJECTORS	7130228	
0	GLUE	7130239	
1	150 PF CAPACITOR	7030042	C34
39	PV JUMPERS	7170004	
0	JUMPER CONFIGURATION	0600724	
0	TADIRAN BATTERY	7210204	

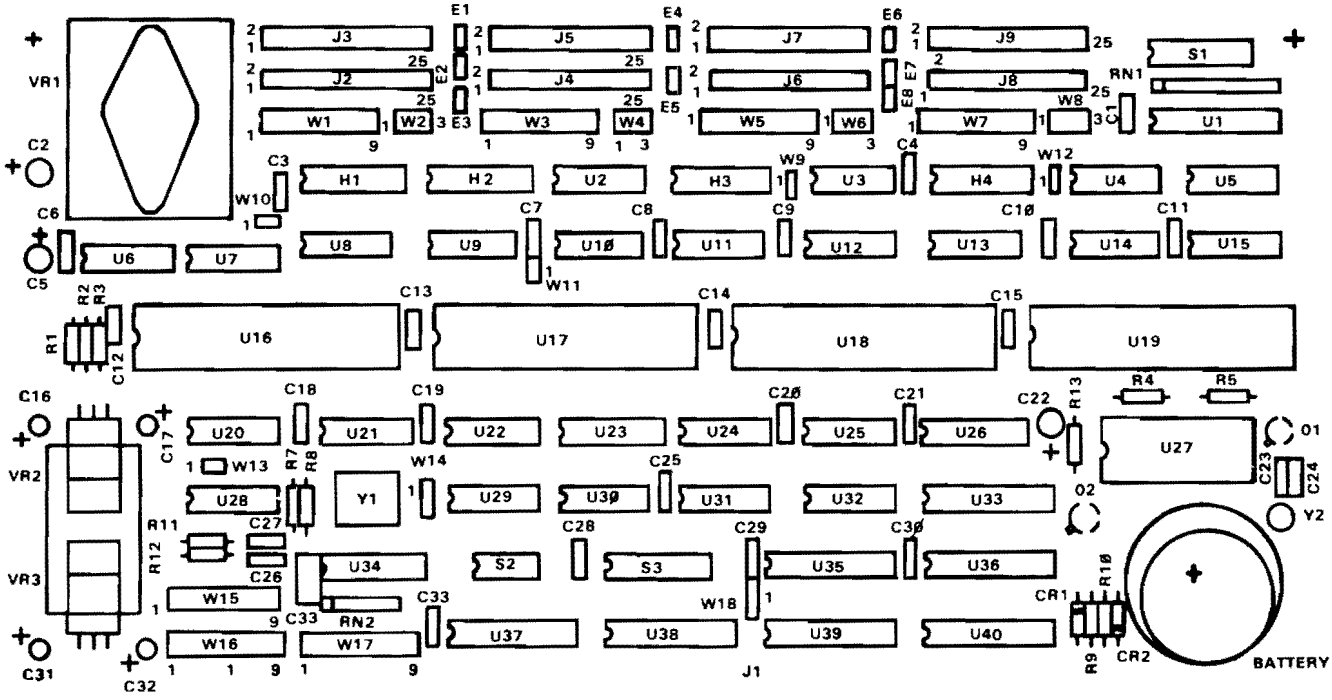
QTY REQD	DESCRIPTION	PART/SUB NUMBER	DESIGNATION
	I/O-8 SCC		
6	IC, 75188 OR MC1488	7010332	U6, U9, U11, U12, U14, U15
8	IC, 75189 OR MC1489	7010333	U2-U5, U7, U8, U10, U13
2	IC, 74LS04	7010164	U28, U31
2	IC, 74LS32	7010181	U22, U25
1	IC, 74LS10	7010168	U29
1	IC, 74LS08	7010166	U32
1	IC, 74LS139	7010220	U23
2	IC, 74LS74	7010195	U21, U24
1	IC, 74LS174	7010241	U26
1	IC, 74LS682	7010518	U38
6	IC, 74LS244	7010264	U1, U33, U35, U37, U39, U40
4	8530 SCC	7010519	U16-U19
1	CLOCK CHIP, 58167-A	7010502	U27
1	DM8131	7010513	U34
4	SOCKET 16 PIN GOLD PL	7060023	FOR H1-H4
1	IC, 3.3K 6 PIN RES SIP	7010344	RN2
3	RES. 3.3K OHM 1/4W 5%	7020085	R2, R3, R12
4	RES. 1K OHM 1/4W 5% CC	7020073	R1, R5, R9, R11
1	RES. 1.8K 1/4W 5%	7020079	R10
2	RES. 10K OHM 1/4W 5% CC	7020097	R4, R13
5	CAP 10MF 16V TANT	7030009	C2, C5, C16, C22, C31
2	10 μF CAPACITOR	7030080	C17, C32
23	.1 μF CAPACITOR	7030068	C1, C3, C4, C6-C15, C18-C21, C25, C27-C30, C33
3	22 PF CAPACITOR	7030067	C23, C24, C26
1	7812 LM 340T-12.0	7160003	VR3
1	3.6V ZENER	7040044	CR2
1	REG LM7912 -12V 1A	7160010	VR2
1	DIODE 1N914.1N4148	7040001	CR1
1	TRANS NPN 2N2222	7040005	Q1
1	AST 2907 PNP TRAN	7040009	Q2

QTY REQD	DESCRIPTION	PART/SUB NUMBER	DESIGNATION
	I/O-8 SCC (Continued)		
1	78H05	7160012	VR1
1	32.768 KHZ .002% CRYSTAL	7080016	Y2
2	8 POSITION DIP SWITCH	7050002	S1, S3
1	5 POSITION DIP SWITCH	7050001	S2
4	HEADER ASSY	0100736	H1-H4
5	BERG 1X2 STR .230 PIN TN	7170018	W9-W13
1	HEATSINK TMH 6103-B	7130004	
2	6-32 X 3/8 PPH SCREW	7130006	
2	6-32 NUT	7130007	
1	ELECTROCHEM BATTERY	7210205	
1	HEATSINK TMH 6106-13	7130003	
7	BERG STIK ST 2X9 TIN PL	7170097	W1, W3, W5, W7, W15-W17
8	BERG STIK ST 2X13 TIN PL	7170098	J2-J9
1	FAB PCB I/O-8	7000067	
0	I/O-8 SCHEMATIC	0300724	
2	BERG 1X3 STR .230 PIN TN	7170021	W14, W18
4	BERG 2X3 STR .318 PIN AG	7170034	W2, W4, W6, W8
1	IC, 3.3K 10 PIN RES SIP	7010345	RN1
2	RES 470 OHM 1/4W 5%	7020065	R7, R8
1	CRYSTAL, 2.4576 MHZ	7080023	Y1
0	I/O-8 TST PROCEDURE	0900724	
1	7407 HEX BUFFERS	7010008	U20
2	LOCKWASHER #6	7130009	
2	NYLON SCREW #4X3/8	7130070	
2	NYLON HEX NUT #4	7130071	
2	MYLAR INSULATORS	7160011	
2	PCB EJECTORS	7130228	
0	GLUE	7130239	
1	150 PF CAPACITOR	7030042	C34
0	PV JUMPERS	7170004	
0	JUMPER CONFIGURATION	0600725	
0	TADIRAN BATTERY	7210204	
	I/O-4 ASCC		
3	IC, 75188 OR MC1488	7010332	U6, U9, U11
4	IC, 75189 OR MC1489	7010333	U2, U7, U8, U10

QTY REQD	DESCRIPTION	PART/SUB NUMBER	DESIGNATION
	I/O-4 ASCC (Continued)		
2	IC, 74LS04	7010164	U28, U31
2	IC, 74LS32	7010181	U22, U25
1	IC, 74LS10	7010168	U29
1	IC, 74LS08	7010166	U32
1	IC, 74LS139	7010220	U23
2	IC, 74LS74	7010195	U21, U24
1	IC, 74LS174	7010241	U26
1	IC, 74LS682	7010518	U38
6	IC, 74LS244	7010264	U1, U33, U35, U37, U39, U40
2	8531A 6 MHZ ASCC	7010520	U16-U17
1	CLOCK CHIP, 58167-A	7010502	U27
1	DM8131	7010513	U34
2	SOCKET 16 PIN GOLD PL	7060023	FOR H1-H2
1	IC, 3.3K 6 PIN RES SIP	7010344	RN2
3	RES. 3.3K OHM 1/4W 5%	7020085	R2, R3, R12
4	RES. 1K OHM 1/4W 5% CC	7020073	R1, R5, R9, R11
1	RES. 1.8K 1/4W 5%	7020079	R10
2	RES. 10K OHM 1/4W 5% CC	7020097	R4, R13
5	CAP 10MF 16V TANT	7030009	C2, C5, C16, C22, C31
2	10 μF CAPACITOR	7030080	C17, C32
18	.1 μF CAPACITOR	7030068	C1, C3, C4, C6-C8, C12, C13, C18-C21, C25, C27-C30, C33
3	22 PF CAPACITOR	7030067	C23, C24, C26
1	7812 LM 340T-12.0	7160003	VR3
1	3.6V ZENER	7040044	CR2
1	REG LM7912 -12V 1A	7160010	VR2
1	DIODE 1N914.1N4148	7040001	CR1
1	TRANS NPN 2N2222	7040005	Q1
1	AST 2907 PNP TRAN	7040009	Q2
1	78H05	7160012	VR1
1	32.768 KHZ .002% CRYSTAL	7080016	Y2
2	8 POSITION DIP SWITCH	7050002	S1, S3
1	5 POSITION DIP SWITCH	7050001	S2

QTY REQD	DESCRIPTION	PART/SUB NUMBER	DESIGNATION
	I/O-4 ASCC (Continued)		
2	HEADER ASSY	0100736	H1, H2
3	BERG 1X2 STR .230 PIN TN	7170018	W10, W11, W13
1	HEATSINK TMH 6103-B	7130004	
2	6-32 X 3/8 PPH SCREW	7130006	
2	6-32 NUT	7130007	
1	ELECTROCHEM BATTERY	7210205	
1	HEATSINK TMH 6106-13	7130003	
5	BERG STIK ST 2X9 TIN PL	7170097	W1, W3, W15-W17
4	BERG STIK ST 2X13 TIN PL	7170098	J2-J5
1	FAB PCB I/O-8	7000067	
0	I/O-8 SCHEMATIC	0300724	
2	BERG 1X3 STR .230 PIN TN	7170021	W14, W18
2	BERG 2X3 STR .318 PIN AG	7170034	W2, W4
1	IC, 3.3K 10 PIN RES SIP	7010345	RN1
2	RES 470 OHM 1/4W 5%	7020065	R7, R8
1	CRYSTAL, 2.4576 MHZ	7080023	Y1
0	I/O-8 TST PROCEDURE	0900724	
1	7407 HEX BUFFERS	7010008	U20
2	LOCKWASHER #6	7130009	
2	NYLON SCREW #4X3/8	7130070	
2	NYLON HEX NUT #4	7130071	
2	MYLAR INSULATORS	7160011	
2	PCB EJECTORS	7130228	
0	GLUE	7130239	
1	150 PF CAPACITOR	7030042	C34
21	PV JUMPERS	7140004	
0	JUMPER TABLE	0600728	
0	TADIRAN BATTERY	7210204	

APPENDIX I
PARTS PLACEMENT DIAGRAM


VR1
VR2
VR3
C2
C3
C4
C5
C6
C7
C8
C9
C10
C11
C12
C13
C14
C15
C16
C17
C18
C19
C20
C21
C22
C23
C24
C25
C26
C27
C28
C29
C30
C31
C32
C33
R1
R2
R3
R4
R5
R7
R8
R9
R10
R11
R12
R13
RN1
RN2
E1
E2
E3
E4
E5
E6
E7
E8
J1
J2
J3
J4
J5
J6
J7
J8
J9
W1
W2
W3
W4
W5
W6
W7
W8
W9
W10
W11
W12
W13
W14
W15
W16
W17
W18
H1
H2
H3
H4
U1
U2
U3
U4
U5
U6
U7
U8
U9
U10
U11
U12
U13
U14
U15
U16
U17
U18
U19
U20
U21
U22
U23
U24
U25
U26
U27
U28
U29
U30
U31
U32
U33
U34
U35
U36
U37
U38
U39
U40
S1
S2
S3
Y1
Y2
Q1
Q2
CR1
CR2
BATTERY

APPENDIX J
SCHEMATIC

APPENDIX K
JUMPER NUMBERING NOTATION

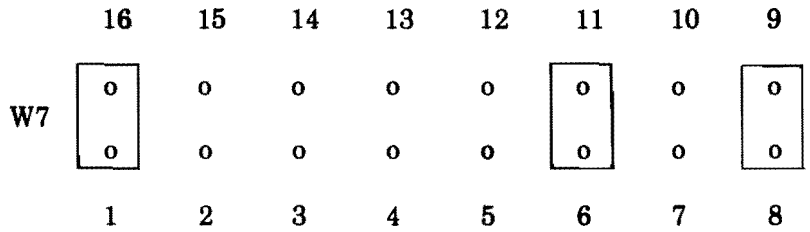
Throughout the manual, jumper locations are referenced by 'W' number and pin number. The following number convention is adhered to for all jumpers on the board.

For horizontal jumpers that are single row (1x3, 1x4, etc.), pins are numbered consecutively from left to right.

For vertical jumpers that are single row (1x3, 1x4, etc.), pins are numbered consecutively from top to bottom.

For horizontal jumpers that are dual row (2x3, 2x4, etc.), pins are numbered consecutively from bottom left to right, continuing on the top right to left.

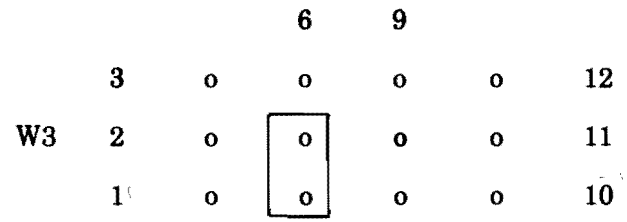
EXAMPLE: W7(1), W7(6), and W7(8) are to be jumpered.



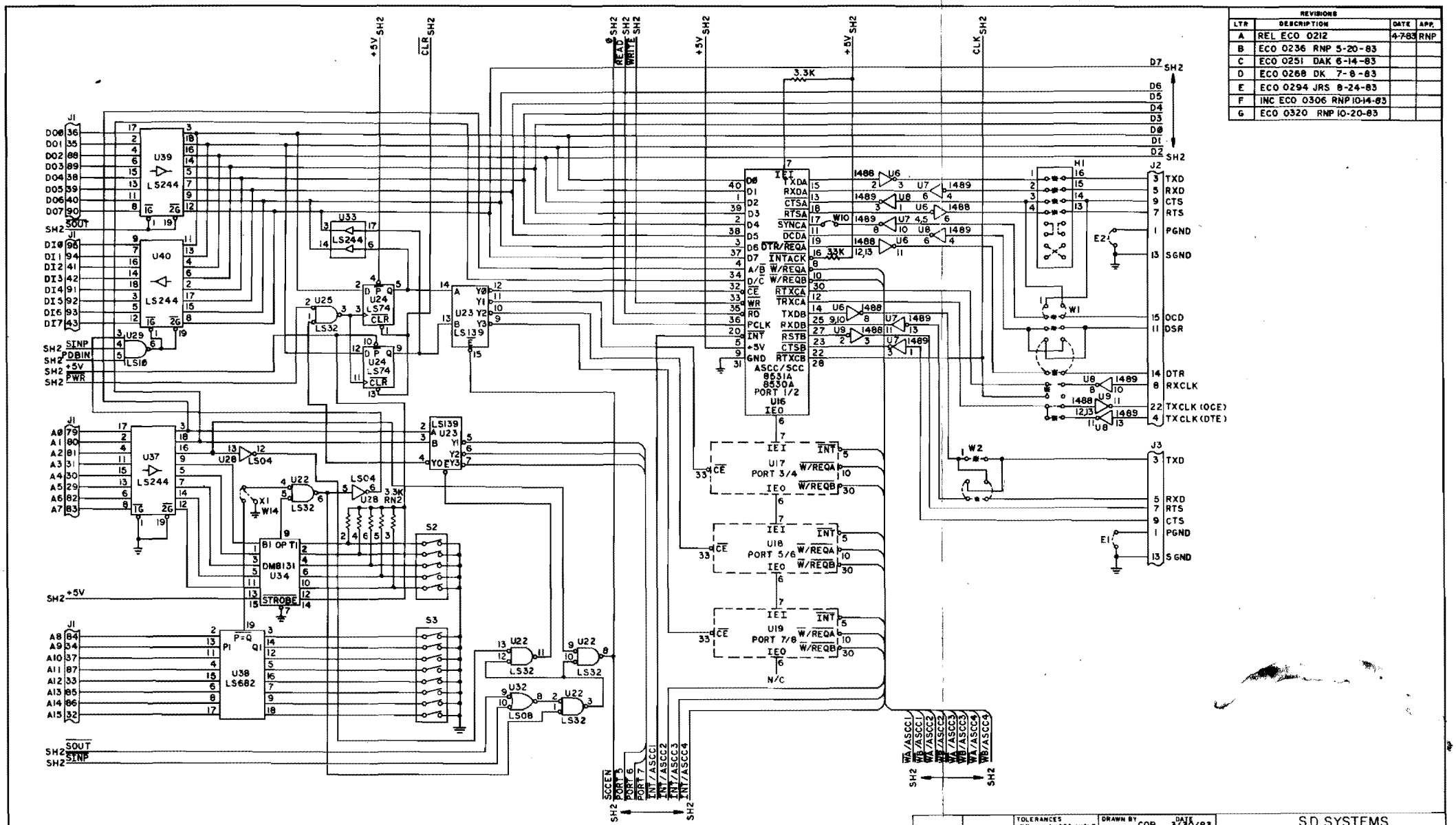
For vertical jumpers that are dual row (2x3, 2x4, etc.), pins are numbered consecutively from left top to bottom, continuing on the right bottom to top.

For horizontal jumpers that are triple row (3x3, 3x4, etc.), pins are numbered consecutively from left bottom to top, continuing from bottom to top for each column from left to right.

EXAMPLE: W3(4-5) is to be jumpered.

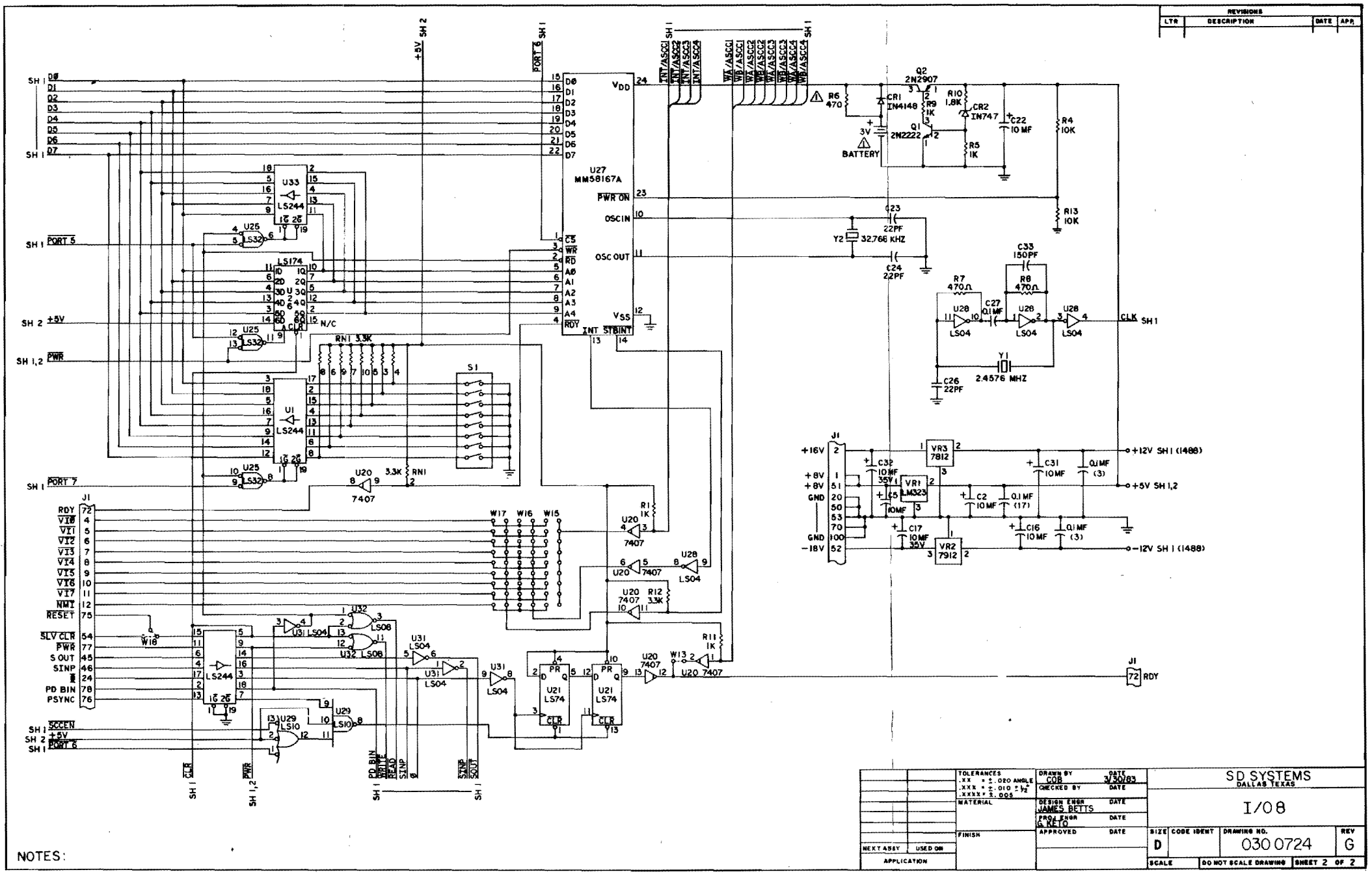


K-1/K-2



⚠ R6 NOT USED WITH CIRCULAR BATTERY TYPES.

NOTES:


REVISIONS
LTR DESCRIPTION DATE APP.
U27 MM58167A
U33 LS244
U1 LS244
U25 LS32
LS174
RNI 3.3K
S1
U20 7407
U28 LS04
U31 LS04
U32 LS08
U29 LS10
U21 LS74
Q2 2N2907
Q1 2N2222
CR1 IN4148
CR2 IN747
BATTERY
Y2 32.766 KHZ
Y1 2.4576 MHZ
VR1 LM323
VR2 7912
VR3 7812
+16V
+8V
GND
-18V
+12V SH 1 (1488)
+5V SH 1,2
-12V SH 1 (1488)
CLK SH1
RDY
SLV CLR
PWR
S OUT
SINP
PD BIN
PSYNC
SCCEN
PORT 5
PORT 6
PORT 7
NOTES:
SD SYSTEMS
DALLAS TEXAS
I/08
DRAWING NO. 030 0724
REV G
SIZE D
SHEET 2 OF 2
DRAWN BY COB
DATE 3/30/83
DESIGN ENGR JAMES BETTS
PROJ ENGR G. KETO
APPROVED
MATERIAL
FINISH
NEXT ASSY
USED ON
APPLICATION
DO NOT SCALE DRAWING