

TECMAR

PC-MATE™
CAPTAIN™

**INSTALLATION
MANUAL
USERS
GUIDE**

TECMAR INCORPORATED
Personal Computer Products Division
6225 Cochran Road • Solon (Cleveland), Ohio 44139-3377
Telephone - (216) 349-0600 • Telex - 466692

Limited Warranty

Tecmar, Inc. ("Tecmar") warrants to the **original purchaser** that boards and cables manufactured by Tecmar will be free from defects in materials and workmanship for one year from the date of delivery to purchaser. Tecmar warrants to the **original purchaser** that disc controllers and drives, terminals, printers, and any other non-board item not bearing a Tecmar label will be free from defects in materials and workmanship for a period of ninety days from the date of delivery to purchaser. Tecmar warrants to the **original purchaser** that the media for software will be free from defects in materials and will be readable by the computer system for a period of ninety days from the date of delivery to purchaser. The software is thoroughly tested and thought to be functional when released; at Tecmar's sole option, Tecmar may endeavor to correct any serious problems discovered in the software.

Under the above warranty, Tecmar will, at its option, either repair or replace a non-conforming or defective product, or return to purchaser the purchase price. Tecmar shall have no obligation hereunder if the product has been misused, carelessly handled, defaced (including unauthorized repairs made or attempted by others), modified or altered. The product must be returned with proof of purchase in its original (or other adequate packaging) to the point of purchase or, by prior arrangements with Tecmar, to Tecmar. Claims must be made in accordance with the provisions of this paragraph within the applicable warranty period or they will be barred.

Tecmar reserves the ultimate authority to determine what constitutes warranty repair. If it is determined that the product is not under warranty, it will be repaired using Tecmar's standard rates for parts and labor. Tecmar will use its best efforts to repair the product within three days after receipt thereof. Tecmar shall not be responsible, however, for delays caused by shipping of non-availability of replacement components or other similar or dissimilar causes, events or conditions beyond its reasonable control.

The above warranty is the only warranty authorized by Tecmar. THE ABOVE WARRANTY STATES THE PURCHASER'S EXCLUSIVE REMEDY FOR ANY BREACH OF TECMAR'S WARRANTY AND FOR ANY CLAIM, WHETHER SOUNDING IN CONTRACT, TORT OR NEGLIGENCE, FOR LOSS OR INJURY CAUSED BY THE SALE OF ANY PRODUCT. WITHOUT LIMITING THE GENERALITY OF THE FOREGOING, TECMAR SHALL IN NO EVENT BE RESPONSIBLE FOR ANY LOSS OF BUSINESS OR PROFITS, DOWNTIME OR DELAY, LABOR, REPAIR, OR MATERIAL COSTS, INJURY TO PERSON OR PROPERTY OR ANY SIMILAR OR DISSIMILAR CONSEQUENTIAL LOSS OR DAMAGE INCURRED BY PURCHASER, EVEN IF TECMAR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH LOSSES OR DAMAGES.

SOME STATES DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES OR LIMITATIONS ON HOW LONG AN IMPLIED WARRANTY LASTS, SO THE ABOVE LIMITATION OR EXCLUSION MAY NOT APPLY TO YOU.

THIS WARRANTY SHALL NOT BE APPLICABLE TO THE EXTENT THAT ANY PROVISION OF THIS WARRANTY IS PROHIBITED BY ANY FEDERAL, STATE OR MUNICIPAL LAW WHICH CANNOT BE PREEMPTED. THIS WARRANTY GIVES YOU SPECIFIC LEGAL RIGHTS, AND YOU MAY ALSO HAVE OTHER RIGHTS WHICH VARY FROM STATE TO STATE.

Keep your original sales receipt for the product with this warranty statement.

PRODUCT _____

SERIAL NUMBER _____ DATE OF PURCHASE _____

WHERE PURCHASED _____

Table of Contents

	Page No.
Introduction	1
Features of Captain	4
Manual Overview	5
System Requirements	6
Configuration Section	7
Setting the System Board for Memory Expansion	9
Configuration for the Captain Board	14
Installing a New Battery	20
Installation of the Captain Board	25
Using the Parallel Port	31
PC DOS Printer Commands	32
BASIC Printer Commands	33
Using the Serial Port	35
PC DOS Serial Port Commands	36
BASIC Serial Port Commands	37

Appendix A: Cover Removal and Replacement 1A

Appendix B: Switch Modules and Jumper Blocks 1B

Appendix C: Block Diagram 1C

Appendix D: Using the I/O Sections 1D

Appendix E: Direct Control of the Parallel Port..... 1E

Appendix F: Direct Control of the Serial Port 1F

Appendix G: Direct Control of the Timer Chip 1G

Appendix H: Adding Memory to Captain 1H

Appendix I: Specifications 1I

Appendix J: The Programmable Array Logic (PAL) Option . 1J

Appendix K: Error Messages 1K

INTRODUCTION

Features of Captain	4
Manual Overview	5
System Requirements	6

INTRODUCTION

CONGRATULATIONS ON YOUR PURCHASE OF THE TECMAR PC-MATE CAPTAIN BOARD! Captain is a multifunction board that fits into any slot of your IBM Personal Computer, XT, or PC2. It can also be installed in the Tecmar Expansion Chassis. Captain offers four popular expansion features and the software to get maximum use out of your board:

Tecmar's Captain board gives you:

Maximum Memory

Captain is fully socketed for easy memory expansion from 0-384K RAM in 64K increments. Your IBM PC2 or XT system board must be fully expanded to 256K RAM before installing the Captain. Thus you have the possibility of installing 640K RAM, which is the total maximum memory recognizable to the operating system. With this amount of RAM, all software for the IBM PC can be run. The Captain board can start at either 64K or 256K.

Note: Owners of the earlier IBM PC's with only 64K RAM on the system board will have the possibility of expansion to 448K RAM with Captain.

Serial Input/Output Port

The serial port is an RS-232 port used for controlling a modem, serial printer, or other serial device. It can function as either COM1 or COM2. The serial port connector is a male DB25 mounted on the Option Retaining Bracket at the rear of the board. This connector is identical to the connector on the IBM asynchronous adaptor.

Parallel Port

The Centronics-compatible parallel port interfaces with the IBM printer (or equivalent). The printer port can function as either LPT1 or LPT2. Access to the parallel port is through a female DB25 connector. You don't need any extra parts since the connector, mounting hardware and cable are supplied with the board.

Treasure Chest of Software

The Treasure Chest is 24 programs that include games, business applications, home applications, a security system and hardware tests. They were designed to be easy and fun to use.

The PAL (Programmable Array Logic) Option

Every Captain board is equipped with the appropriate circuitry to accommodate a Programmable Array Logic Chip. You can use PAL to protect restricted software. This feature is especially attractive in an office setting where many personal computers are available, but certain information must be restricted to management groups on a 'need to know' basis. The PAL chip is an option that must be purchased separately.

Features of Captain

- Fully socketed for easy memory expansion from 0 to 384K Bytes in 64K Byte increments.
- Standard IBM PC parity checking.
- RS-232 serial port is identical to the IBM serial port, can be COM1 or COM2, and is compatible with IBM communications software.
- Parallel port replaces the IBM Printer Adaptor board or the parallel port can be used with the IBM Monochrome Printer Adaptor.
- On-board jumper to configure serial port as DTE or DCE without rewiring or soldering.
- Clock/Calendar includes a battery and application software with automatic time-setting functions.
- Clock/Calendar can generate interrupts from once a month to once every tenth of a second.
- Both serial and parallel ports can be interrupt driven.
- Treasure Chest Users Manual is included.
- Treasure Chest Technical Reference Manual is included.

Manual Overview

The Captain manual will help you configure your Captain board, install it in your computer and use the software provided.

The manual is arranged in consecutive order. The steps can be performed as you read them. We do suggest, however, that you familiarize yourself with the entire manual before attempting to install the board.

Before proceeding, you should assemble the following tools:

- 1 regular flat blade screwdriver
- 1 pair tweezers or needle nose pliers
- 1 ballpoint pen
- 1 Phillips head screwdriver (for Expansion Chassis)

The steps you must follow to use Captain are explained in the following pages or Appendixes of this manual. They are:

- 1) Unpack the PC-Mate Captain board and place it on a flat surface.
- 2) Remove the cover of the computer system or Tecmar expansion chassis. (See Appendix A for step by step instructions.)
- 3) Configure the system board and the Captain board.
- 4) Install the board.
- 5) Install the time-setting software (DOSTIME).
- 6) Familiarize yourself with the PC DOS and BASIC commands to control the serial and parallel port.

System Requirements

Your system must meet the following requirements in order to use Captain:

1. The Captain board may be installed in an IBM Personal Computer, XT model computer, Tecmar Expansion Chassis and some IBM look-alike computers such as the Compaq or Columbia Desktop Personal Computer. In the case of the look-alikes, be sure to set the ports as LPT2 and COM2.¹
2. At least one expansion slot in your Personal Computer or Expansion Chassis is required to install your Captain board.
3. If you have other memory expansion boards in the IBM PC, XT, PC2 or Compaq, they must be removed, set to a high-memory area inaccessible to the operating system, or set to memory addresses that do not conflict with the addresses used by the Captain board. Refer to the manual that accompanied the memory expansion board and this manual to get the memory addresses correctly allocated. The Columbia Desktop computer must have a memory expansion board that gives the computer 128K of memory for a total memory amount of 256K.
4. The software for setting the time and date on the clock/calendar, as well as the MEMDISK, PRINTER and COPRINT programs can be run on either DOS V1.1 or V2.n.
5. The system board of your computer system must have its memory fully expanded before Captain can be installed.
6. The Compaq, and Columbia Desktop Personal Computer must have the Captain serial port designated as COM2 and the Captain Parallel port designated as LPT2. The section 'Setting the Captain Jumpers' of this manual will advise you how to set these ports as COM2 and LPT2.

*1 The Captain manual refers to the serial port as COM1 or COM2 and the parallel port as LPT1 or LPT2. These names, assigned by PC-DOS during boot-up, represent the port I/O locations on the Captain board. Time1/Time2 in the manual represent the starting address of the Clock/Calendar. For information about these addresses, refer to Appendix C of this manual.

CONFIGURATION SECTION

Setting the System Board for Memory Expansion . 9

Configuration for Captain Board 14

CONFIGURATION SECTION

Listed below is the amount of memory certain systems must have before Captain can be installed.

IBM PC (64K system board)	64K
IBM PC2	256K
IBM XT	256K
Columbia Desktop	256K
Compaq	256K*

If you have less than the required memory, you must add memory to your system before continuing. (Tecmar supplies RAM expansion kits which can be installed with minimum effort on the Personal Computer's system board or you can remove memory from the Captain board to fill your system board.)

The Columbia Desktop computer can only hold 128K on its system board. You must add an additional memory board (such as the Tecmar Dynamic Memory board) in order to upgrade your computer to 256K. Then you can install Captain.

*The Compaq can only address 512K of memory. Therefore, Compaq cannot address more than 256K on your Captain.

As previously stated, Captain is fully socketed for easy memory expansion from 0-384K Ram in 64K increments. Be sure, before you install this board as normal system memory, that your computer system board, the board on the floor of your computer system, has fully expanded RAM.

The Captain board offers many possible combinations of memory configurations. Each configuration requires unique switch settings on the system board and the Captain board. As the potential user of the Captain, you must set the proper memory configuration for your unique system. In order to accomplish this task, you must know two things about your system:

- 1) The amount of memory (RAM) on your fully expanded system board. A formula for determining the total memory of your system is included in the switch setting section of this manual.
- 2) The amount of memory you acquired on the Captain board.

If you have never set switches or connected jumpers, or if it has been some time since you have done so, looking through Appendix B at this point would be quite useful. Appendix B gives a complete explanation of switch modules and jumper blocks. You will also find instructions for setting the switch modules and connecting jumper block positions in Appendix B.

Setting the System Board for Memory Expansion

You need to know how much total memory you currently have in your system to set the switches on the system board and Captain board. To find the size of your system memory, at the DOS prompt Type:

```
A>CHKDSK ←
```

```
A>CHKDSK
```

```
322560 bytes in total disk space
```

```
9216 bytes in 2 hidden files
```

```
102400 bytes in 7 user files
```

```
210944 bytes available on disk
```

```
262144 bytes total memory
```

```
249734 bytes free
```

Figure 1
Screen produced by CHKDSK command

The second line from the bottom tells you how much total memory you have in your system. The number displayed is the total bytes of memory. To determine the total RAM in your system, bytes must be converted to kilobytes, or K. Use Table 1 to convert total bytes to total K.

Table 1. Byte Conversion Table

BYTES	KILOBYTES (K)
65535	64K
131072	128K
196608	192K
262144	256K
327680	320K
393216	384K

The number that applies to your system is an important number. You may find it helpful to write the RAM number along with the amount of memory you plan to activate on your Captain Board in the spaces provided below.

Total System Memory _____K

Total Captain Memory _____K

If you do not have 64K (65535) or 256K (262144), you will not be able to install the Captain board. If you have 128K or 192K, add more memory to your system board to bring it up to 256K.

If you have more memory than 256K, you must readdress the memory above 256K to start where your Captain memory ends.

The following three pages should be used by persons who will be installing Captain in the IBM Personal Computer, the PC2, the XT or the Compaq computer. If you will be installing Captain in the Columbia Desktop or another IBM 'look-alike' computer that does not use a system switch block to recognize memory, skip the rest of this section and go to 'Configuration for the Captain Board'.

In order for the computer system to recognize the Captain's RAM as system memory, the computer's system configuration switches must be set. The system board is the bottom board on the computer system chassis. Figure 2 below is a diagram showing a switch module 'exploded' out of a system board.

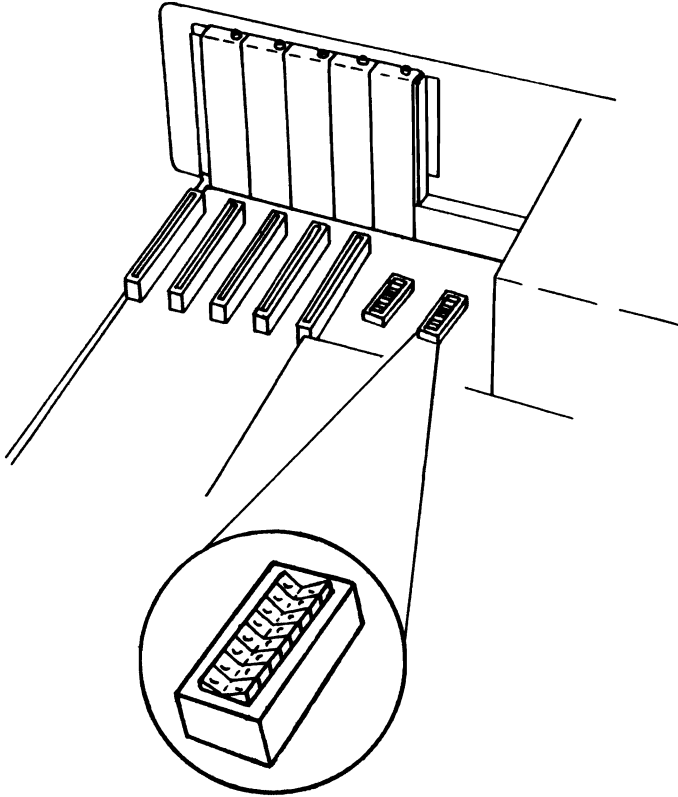


Figure 2
System Board Switch Module

Locate the switch modules on your system board. They are located just beyond the back left corner of Disk Drive A. In the IBM Personal Computer or PC2, the switches consist of switch modules labelled SW1 and SW2.

The switch modules of your computer may have the word 'OPEN' or 'ON' printed on it. In the following sections, OPEN is the same as OFF and CLOSED is the same as ON. When a switch is OPEN or OFF, it is set toward the side opposite the numbers on the switch module.

Setting the System Switches

Before you change switches on your computer, make a note of how they are presently set. Then if you get unusual errors after installing the board, you can check to see if you moved another switch accidentally.

Set SW1 switches 3 and 4 of the system board switch module to the off position. (See Appendix B for information on switch modules.) In the XT, SW1 is the only switch module. Consequently, this is the only setting necessary on the XT system board.




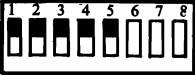
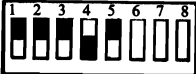
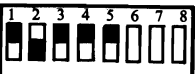
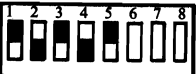
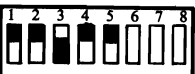
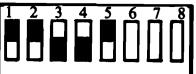

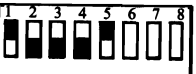
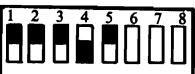
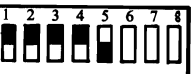
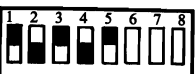
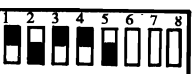
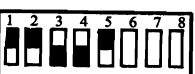
Figure 3
System Board SW1

Do not change any other switches on SW1.

In the other computer models, switch module SW2 must be set. The appropriate settings for SW2 are illustrated for each type of computer on the next pages. Column 1 of Table 2 lists the amount of Captain memory. Column 2 contains the proper switch settings for the IBM PC and PC2 for the amount of Captain memory. Column 3 shows the switch settings for the IBM PC with a 64K system board for the amount of Captain memory. The assumption is that you have fully expanded memory on your system board.

First find the column for your computer system. Next, read down the Captain column until you find the amount of memory on the Captain board. Circle that number. Circle the corresponding diagram in your computer system column. Set the switch SW2 on your system board accordingly.

Table 2. System Board Switch Settings — SW2

Amount of Captain Memory to Activate	IBM PC Models With 256K RAM	IBM PC Models With 64K RAM
0 (all disabled)	ON 	ON 
64K	ON 	ON 
128K	ON 	ON 
192K	ON 	ON 
256K	ON 	ON 
320K	ON 	ON 
384K	ON 	ON 

Double check your settings. **ONLY DESIGNATED SWITCHES SHOULD BE CHANGED.** Follow the diagrams and make no other changes.

Configuration for the Captain Board

Overview

Look at the Captain board and then at Figure 4 below. The labelled symbols in the diagram are Switch 1 (SW1), jumper blocks 1-3 (JPR), and the printer ports for parallel (P2) and serial (P1) ports. The symbols represent parts of the Captain board. Find each of the labelled parts on your board. The parts will be important in configuring and using the Captain board.

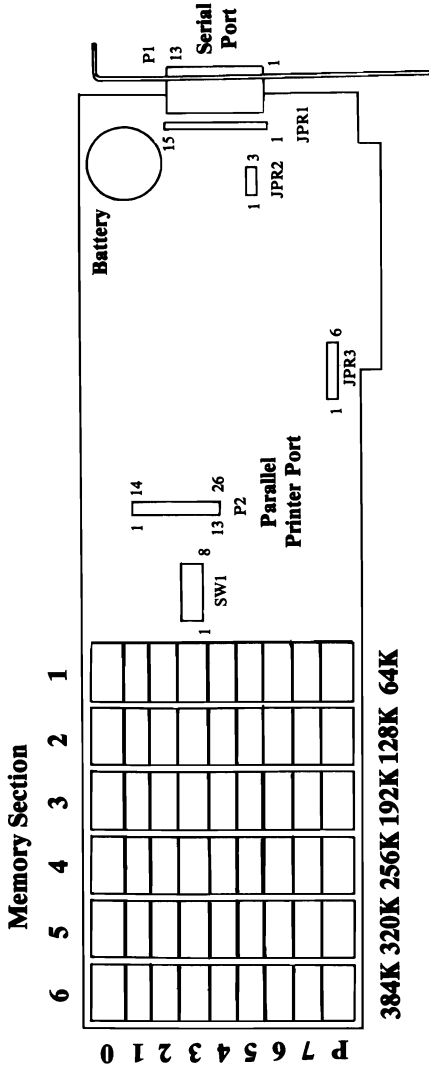


Figure 4
The Captain Board

The symbol labelled SW1 in the Figure 4 diagram is switch module SW1. Eight numbered switches are on the module. The switches serve as indicators to convey information to the Captain board. Figure 5 explains the use of the SW1 switch positions.

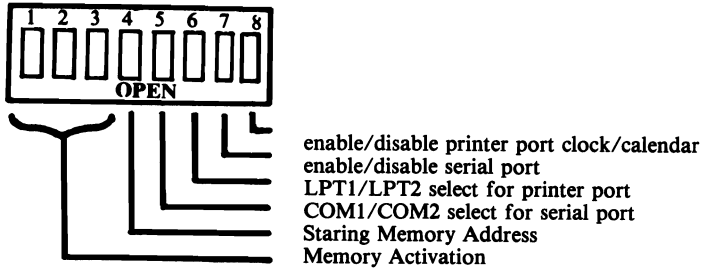


Figure 5

Note: When using the Treasure Chest of software, you must give the -2 option for DOSTIME and SETTIME if your Captain board is jumpered for TIME2. Refer to the Treasure Chest Technical Reference for more detailed information.

Memory Activation

Switches 1, 2, and 3 of switch module SW1 control the amount of RAM on the Captain board. The switch module illustrations below show how switches 1, 2, and 3 of SW1 would be set to activate 0-384K RAM in increments of 64K. Locate and circle the module that represents the amount of memory to be configured for your Captain Board.

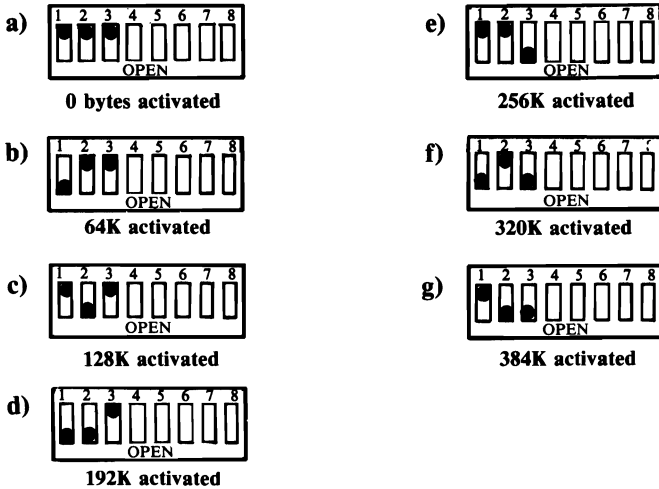


Figure 6
Captain Board Memory Settings

Set switches 1, 2, and 3 of SW1 of your Captain board according to the diagram of the switch module above. Double check the settings.

See Appendix K for error messages that could occur when you turn on your computer.

Starting Memory Address

Switch 4 on the Captain board as SW1 lets you choose between two starting addresses for Captain memory. If you have an XT or PC2 computer, you will have 256K on your system board. If you have the original mode of the IBM Personal Computer, you will have 64K on your system board. Select the diagram that reflects the proper setting for your configuration. Circle the diagram. Set switch 4 of SW1 of your Captain board according to one of the diagrams below. Double check the setting.



**Start at 64K
(Original IBM PC)**



**Start at 256K
(XT, PC2, Columbia
Desktop and Compaq models)**

**Figure 7
SW1 Switch 4 Settings**

Parallel and Serial Port

Switches 5, 6, 7, and 8 on the Captain board SW1 switch module allow you to select COM1/COM2, LPT1/LPT2, and Time1/Time 2 I/O locations. Read each of the following before proceeding with setting the four switches.

- If you have an IBM Parallel Printer Adaptor already in your system, you must set the parallel port on the Captain board to LPT2. If Captain is the third parallel port in your system, the Captain port will be assigned the default name 'LPT3'.
- If you have the IBM Monochrome Display Adaptor in your system, you do not need to set the parallel port as LPT2. The Captain address settings do not conflict with the IBM Monochrome Display Adaptor address. IBM's boot-up procedures will assign the name 'LPT2' to the Captain port if it is the only other parallel port in the system.

- If you have an additional serial port in your system, (as in an XT), you must set the serial port on the Captain board to COM2. This is done to prevent I/O address conflicts between the ports. You will also need to change JPR3, the interrupt jumper.
- If you will be installing Captain in the Columbia Desktop or Compaq personal computers, your Captain serial port must be set as COM2 and your parallel port must be set as LPT2/Time 2.

You must have a thorough understanding of the configuration your system requires. The diagrams in Figure 8 show the settings that are possible for SW1 switches 5, 6, 7, and 8. Find the switch settings that correspond to the requirements you have. Circle each switch setting for your configuration. Set the switches on the Captain board. Double check the settings.

You will select one setting in the first column and one setting in the second column.



COM1 selected



LPT1 and Time 1 selected



COM2 selected



LPT2 and Time 2 selected



Serial Port disabled



**Printer Port and
Clock/Calendar disabled**

Figure 8
Choices for Switch Settings

The I/O addresses corresponding to the selections you made for switches 5, 6, 7, and 8 are listed in Table 3.

Table 3. I/O Addresses for Selected Switches

Selection	Starting Address	
COM1	1016 DEC	3F8 HEX
COM2	760 DEC	2F8 HEX
LPT1	888 DEC	378 HEX
LPT2	632 DEC	278 HEX
TIME1	893 DEC	37D HEX
TIME 2	637 DEC	27d HEX

Jumper Configuration

Look at Figure 9 of the right end of the Captain board below. The symbols labelled JPR1, JPR2, and JPR3 represent jumper blocks on the board. The numbers at each end of the jumper blocks represent the first and last jumper positions of each jumper block.

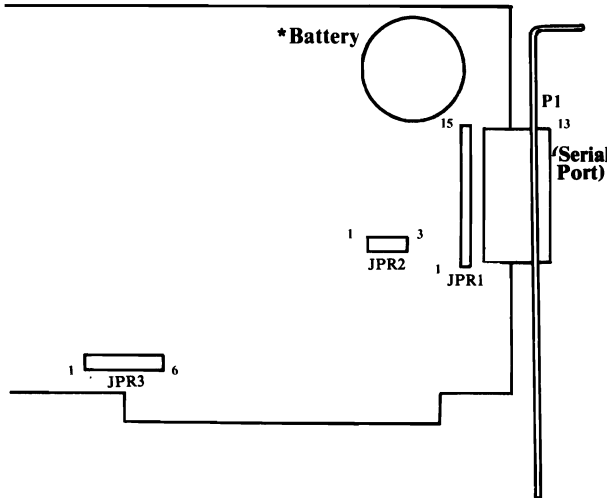


Figure 9
Right End of Captain Board

*The Captain clock/calendar backup battery can be replaced by a Matsushita BR2325 or equivalent. The location of the battery is illustrated in Figure 9. To replace it, simply slip it out from beneath the clip which holds it in place. Slip the new one into place with the same orientation as the original.

A safety fuse is also provided to prevent high voltages from entering the battery. If this fuse blows, contact Tecmar. Do not simply replace the fuse; it can only burn out as a result of some other hardware malfunction.

JPR1 - Serial Port

The jumper block JPR1 is used to select the serial port as either Data Terminal Equipment (DTE) or Data Communication Equipment (DCE). Some devices such as modems are configured as DCE. Others, such as printers are configured as DTE.

Note: DTE ports only communicate with DCE devices.

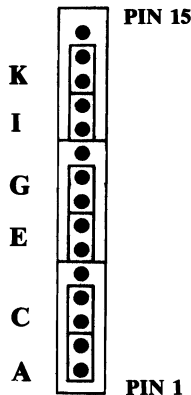
The Captain board is configured for DTE when shipped by Tecmar. Therefore you do not have to change the jumper block if you will be connecting your serial port to a device configured as DCE, for example, a modem. To connect it to a serial printer or another IBM-compatible serial port, jumper the board for DCE.

The configurations differ only in the location of the pin signals on the serial port. The purpose of block JPR1 is to change the location of the pin signals. Thus, two different ports can be connected without any special wiring in the cable.

Remember, a port configured as DTE only communicates with a DCE device. If the ports are configured the same, the Captain board or other device must be jumpered differently. Check the manual included with the device you will be connecting to the serial port for its configuration (DCE or DTE).

Both DTE and DCE connections are given. Circle the appropriate configuration for your application. To change the configuration on your Captain board, connect the appropriate points on JPR1.

DTE: Connect positions A,C,E,G,I and K



DCE: Connect positions B,D,F,H,J and L

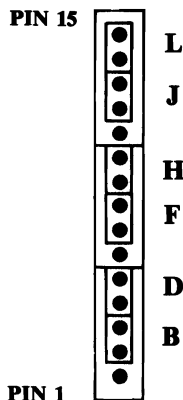


Figure 11
Block JPR1 Connections

Note: The modem control signals RI and RLSD and the two 20mA current loop data signals cannot have their signals changed by changing a jumper. It is necessary to make the appropriate connections in the cable if these signals are required.

Note: If you are connecting a serial printer to P1, you may get a Device Timeout error. This is because IBM's software is looking for handshaking signals on Pin 6. If you do not want to construct a special cable, you may remove jumper wires E and G and connect Pin 6 to Pin 8. Refer to Figure 2F in Appendix F for the pinout of P1.

JPR2 - Serial Port

Jumper block JPR2 allows selection of a particular kind of data input. Connect position B to select the standard RS-232 data input, which is good for communications within 100 feet. Connect position A to select the current loop data input, which is good for communications up to 1000 feet. The board is shipped jumpered in position B.

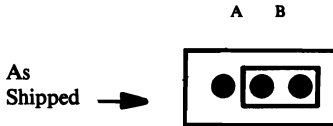


Figure 12
Jumper Block JPR2

JPR3 - Interrupts

Jumper block JPR3 can be used to connect the serial ports and the clock/calendar to various interrupt request lines (IRQ lines) in the IBM Personal Computer. Your Captain board was shipped jumpered at position B. This setting, illustrated below in Figure 13, is sufficient for normal applications.

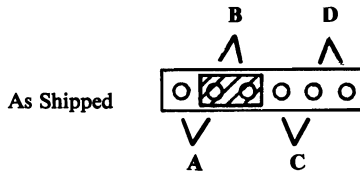


Figure 13
Jumper Block JPR3

Table 4. Connections at Jumper Block JPR3

Position	Interrupt Source	Interrupt Request Line	Interrupt No.
A	Serial Port	IRQ 3 (for COM2)	0B Hex
B	Serial Port	IRQ4 (for COM1)	0C Hex
C	Clock/Calendar	IRQ5	0D Hex
D	Clock/Calendar	IRQ7	0F Hex

To connect the two pins corresponding to one of the above functions, place one of the jumper connectors provided with the board over the two pins of the jumper position. For more information on the serial port interrupt, review Appendix F.



INSTALLATION OF THE CAPTAIN BOARD



INSTALLATION OF THE CAPTAIN BOARD

You can install the Captain board in your Tecmar Expansion Chassis or IBM Personal Computer. The diagrams in this section feature the IBM Personal Computer.

The installation of your Captain board is very simple and only requires the use of a medium-sized flat-blade screwdriver. (Use a Phillips-head screwdriver if you will be installing Captain in the Expansion Chassis.)

1. **TURN OFF THE POWER TO YOUR PERSONAL COMPUTER OR EXPANSION CHASSIS.** It is important that there be no power applied to the unit while the board is being installed or serious damage to the computer could result.
2. Tecmar recommends the use of two adjacent slots in your computer or Expansion Chassis. Captain will be installed in one slot and the Captain parallel port connector, which is attached to the board by a ribbon cable, will be installed in the other slot. Choose the two slots in the computer where Captain will be installed. You may need to move another board to have the two Captain slots adjacent to each other.

If you only have one expansion slot left, the Captain should be installed in the slot and the Captain parallel port ribbon cable can be draped over the top back edge of the unit. Replacing the unit cover will not damage the ribbon cable.
3. Face the front of the computer or expansion chassis. Remove the slot cover in the leftmost chosen slot by loosening and removing the screw at the top which holds it in place (See Figure 14). Set the slot cover and screw aside.

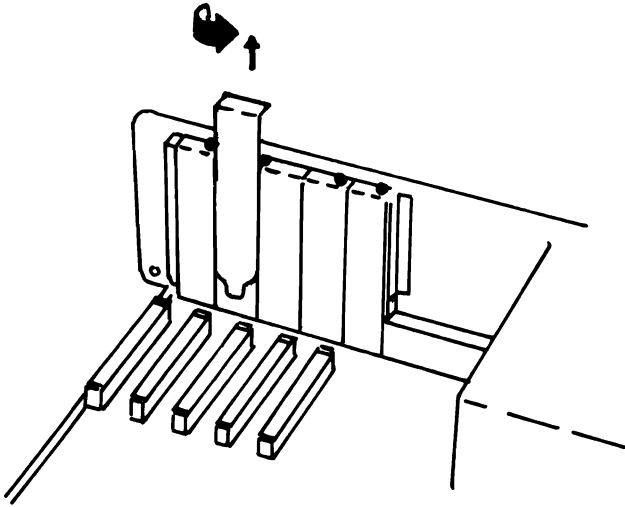


Figure 14

4. If a plastic card edge guide (Figure 15) is not already present on the front wall of your computer, install the one included with Captain. The card edge guide will hold the front edge of the board in place.

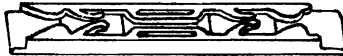


Figure 15

5. Slide the Captain board down into the slot, inserting the black metal Option Retaining Bracket (ORB) attached to the end of the board into the opening created by the removal of the slot cover (See Figure 16). Make sure that the gold 'fingers' of the edge connectors on the bottom of the board are completely in the computer slot. The serial port mounted in the ORB should now be accessible from the back of the computer.

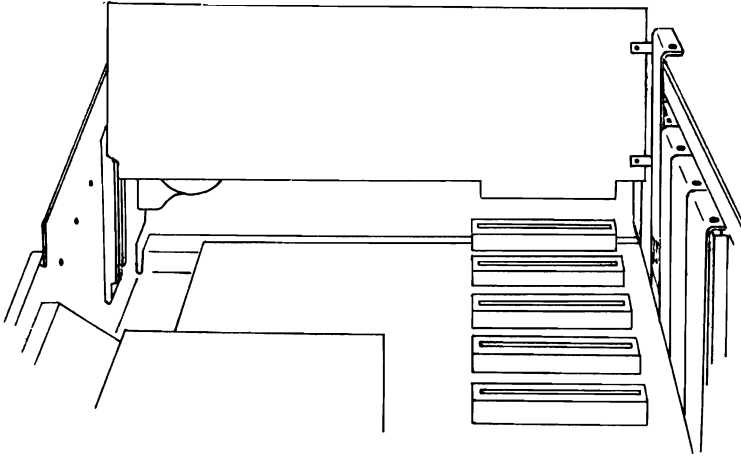


Figure 16

6. Install the screw layed aside in Step 3 into the top of the ORB to hold the board firmly in place.
7. Remove the slot cover from the remaining empty slot adjacent to the Captain board, using the same procedure as in Step 3.
8. The parallel port cable is attached at one end to the P2 connector (See Figure 4) and at the other end to a black metal Option Retaining Bracket (ORB). Place the ORB attached to the end of the parallel port cable into the open slot, as shown in Figure 17.

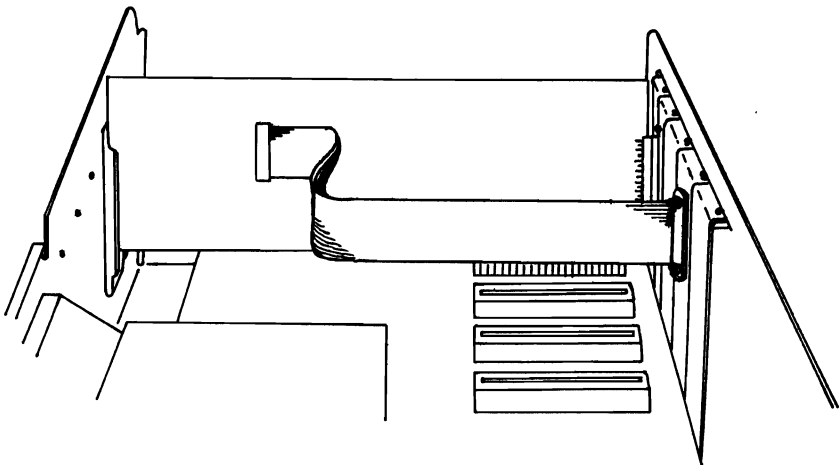


Figure 17

9. Line up the hole in the top of the parallel port ORB with the hole left by the slot cover. Make sure that the cable connector on the board is firmly attached to the parallel port pins (P2).
10. Remove all tools from the inside of the computer.
11. Replace the cover of the IBM Personal Computer or the Tecmar Expansion Chassis using the instructions in Appendix A.

This completes the installation of the Captain board. It is now safe to apply power to the computer.



USING THE PARALLEL PORT

PC DOS Printer Commands 32

BASIC Printer Commands 33

USING THE PARALLEL PORT

Each computer language and operating system uses its own commands (protocols) to send output to the parallel port. The reference below tells you how to use your Captain parallel port with PC DOS or BASIC instructions.

PC DOS Printer Commands

There are three easy ways to use the parallel port in PC-DOS:

1. Copy Command

The COPY command can be used to transfer files to and from Personal Computer peripherals. To print the contents of a file named EXAMPL.FIL, you can type:

COPY EXAMPL.FIL LPT1:

The file must be composed of ASCII characters, which **excludes** all files with the extensions .COM, .EXE, .OBJ and .BAS.

2. CTL/P

Hold the 'CTRL' key down and then press the 'P' key. As you type characters on the keyboard, they will appear on the printer. Hold the 'CTRL' key down and press the 'P' key once more to stop printing. This command will work only if your parallel port is designated as LPT1.

3. SHIFT/PRT-SC

Hold the 'SHIFT' key down and press the 'PrtSC' key to print all the information appearing on the screen. When the end of the present screen is reached, printing stops. Only the information appearing on the screen when SHIFT/PRT-SC is pressed will be printed. This command will only work on the parallel port designated as LPT1.

Refer to your PC DOS manual for more information about these commands.

Basic Printer Commands

The following commands can be used in BASIC to control the printer.

1. **LPRINT [list of expressions][;]
LPRINT USING x\$; list of expressions[;]**

These statements are identical to PRINT and PRINT USING, except that output goes to the printer instead of printing on the screen.

2. **WIDTH LPT1:size**

This statement sets the width of the output line in characters. 'SIZE' can be a number from 0 to 255.

3. **LLIST [line-[line]]**

LLIST is equivalent to LIST, but it sends the lines to the printer instead of displaying them on the screen.

Refer to the IBM BASIC manual for more information about these commands.

If you have configured your parallel port as LPT2, certain print commands may not work. They were designed to work with LPT1 only.

You can be sure your parallel port (LPT2) works by typing the following BASIC program:

```
10 OPEN "LPT2:" FOR OUTPUT AS #1
20 PRINT #1, "HELLO"
30 CLOSE 1
```



USING THE SERIAL PORT

PC DOS Serial Port Commands.....36

BASIC Serial Port Commands37

USING THE SERIAL PORT

Each computer language and operating system has its own commands (protocols) to send output to the serial port. The reference below gives these commands for PC DOS and BASIC.

PC DOS Serial Port Command

To send information to COM1 or COM2:

1. Set the port parameters with the MODE command.

MODE COMn:[baud[,parity[,databits[,stopbits],P]]]

where n = 1 or 2
baud = 110, 150, 300, 600, 1200, 2400, 4800 or 9600
parity = N (none), O (odd), E (even). Default is E
databits = 7 or 8. Default is 7
stopbits = 1 or 2. Default is 2 when baud = 110,
 otherwise default is 1.
P = include if COMn is a printer

2. Then reroute LPT1: output to COMn so the printer commands will send information to the serial port:

MODE LPT1: = COMn:

3. Use the PC DOS or BASIC printer commands (CTL/P, SHIFT/PRT-SC or COPY) to generate printer output.

4. To reestablish normal LPT1: output to the parallel port, type:

MODE LPT1:

BASIC Serial Port Commands

1. You can use the serial port as your printer port in BASIC by using the MODE command explained above. Then enter BASIC and use the BASIC printer commands (explained under the 'Using the Parallel Port') to send output to COM1 or COM2.

To resume normal parallel printer output, re-enter PC DOS and use the MODE command in Step 4 above.

2. Use the BASIC 'OPEN 'COM . . . ' statement to access COM1: or COM2: as a BASIC data file:

**OPEN 'COMn:[baud][,parity][,databits][,stopbits]' AS
[#]filename**

where n = 1 or 2
baud = 75, 11, 150, 300, 600, 1200, 1800, 2400, 4800,
9600. Default is 300.
parity = S (space), O (odd), M (mark), E (even),
N (none). Default is E.
databits = 4, 5, 6, 7 or 8. Default is 7.
stopbits = 1 or 2. Default is 2 for 75 and 110,
otherwise default is 1.
filename = a valid file number.

Then you can use the BASIC file I/O statements to communicate with the serial device:

PRINT #
PRINT # USING
WRITE #
INPUT #
LINE INPUT #

Read your IBM DOS manual carefully for more details of these functions.

LIST OF APPENDIXES

Appendix A:	Cover Removal and Replacement.....	1A
Appendix B:	Jumper Blocks.....	1B
Appendix C:	Block Diagram.....	C1
Appendix D:	Using the I/O Sections.....	1D
Appendix E:	Direct Control of the Parallel Port.....	1E
Appendix F:	Direct Control of the Serial Port.....	1F
Appendix G:	Direct Control of the Timer Chip.....	1G
Appendix H:	Adding Memory to Captain.....	1H
Appendix I:	Specifications.....	1I
Appendix J:	The Programmable Array Logic (PAL) Option.....	1J
Appendix K:	Error Messages.....	1K

Appendix A: COVER REMOVAL AND REPLACEMENT

Removing the IBM Personal Computer Cover

Follow the instructions below to remove the cover of the IBM Personal Computer, PC2 or XT.

1. Position your computer to allow access to the rear. Use a flat-blade screwdriver to remove the six cover mounting screws by turning them counterclockwise as shown in Figure 1A. Save the screws for replacing the cover.

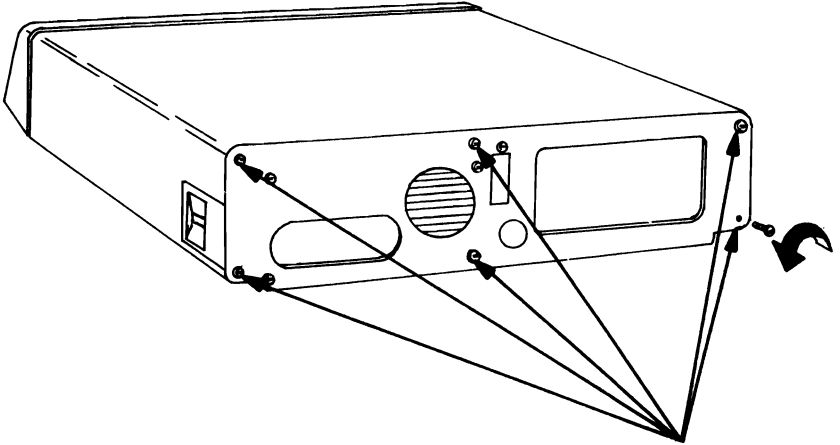


Figure 1A

- Slide the computer's cover towards the front of the computer. When the cover will go no further, tilt it up and remove it, as shown in Figure 2A.

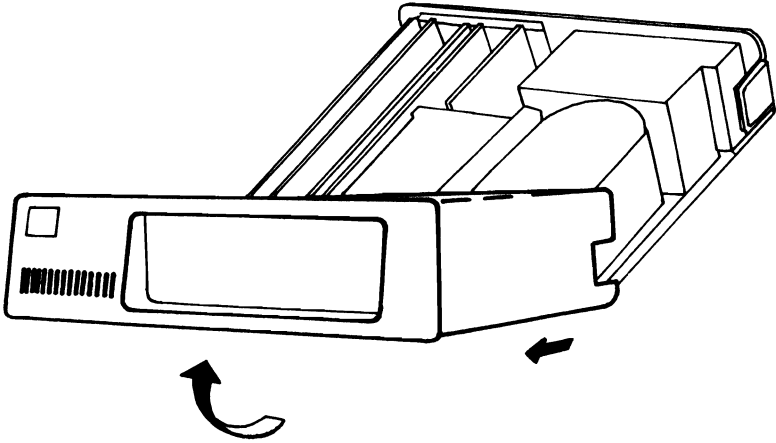


Figure 2A

Replacing the IBM Personal Computer Cover

Use the following instructions to replace the cover of the IBM Personal Computer, PC2 or XT computer.

1. Position the computer's cover, as shown in Figure 3A. Then push down on the cover while gently sliding it towards the rear of the computer.

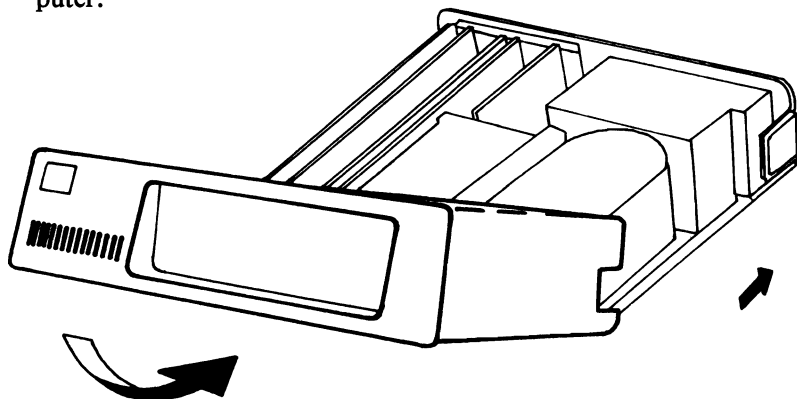


Figure 3A

2. When the cover is in place, insert the six screws into the holes in the rear of the computer and tighten using a flat-blade screwdriver. (See Figure 4A.)

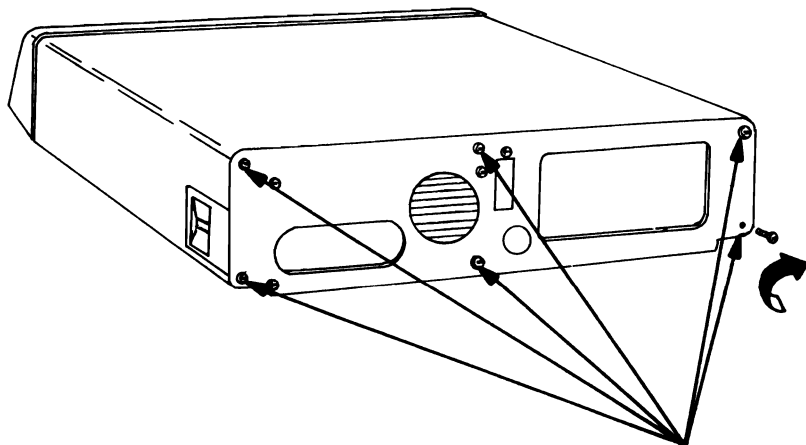


Figure 4A

Removing the Tecmar Expansion Chassis Cover

Follow the instructions given below to remove the cover of the Tecmar expansion chassis.

1. Remove the three screws on each side of the expansion chassis using a phillips-head screwdriver, as shown in Figure 5A. Save the screws.

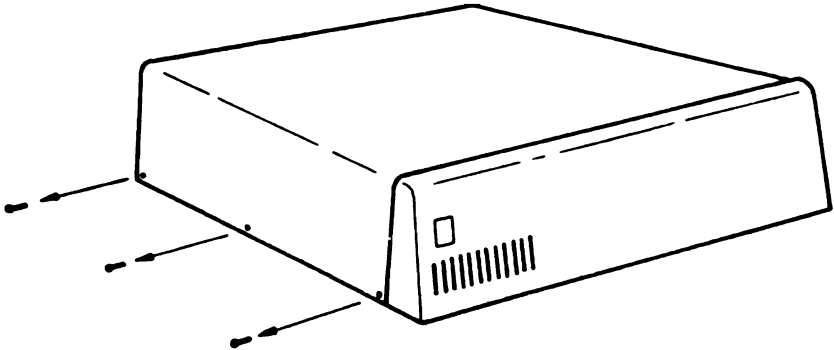


Figure 5A

2. Remove the expansion chassis cover by lifting the rear of the cover and pulling towards the back of the expansion chassis, as shown in Figure 6A.

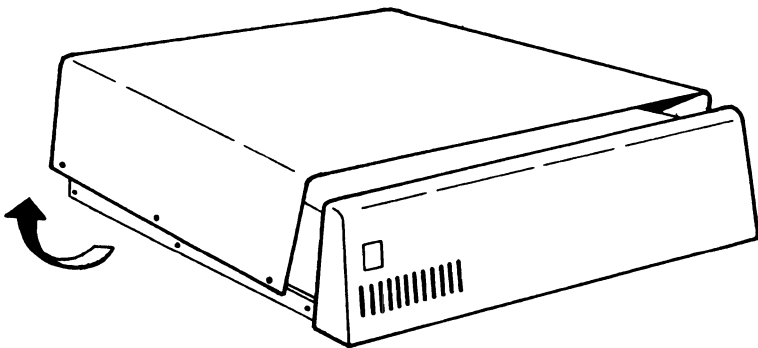


Figure 6A

Replacing the Tecmar Expansion Chassis Cover

Follow the instructions given below to replace the cover of the Tecmar expansion chassis.

1. Slide the front of the cover onto the back of the expansion chassis, while holding the back of the cover up, as shown in Figure 7A. Push the cover forward and down to complete the installation.

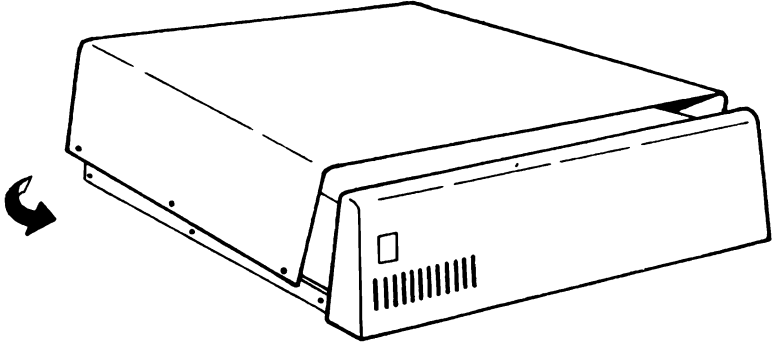


Figure 7A

2. Replace the three screws on each side of the cover as shown in Figure 8A.

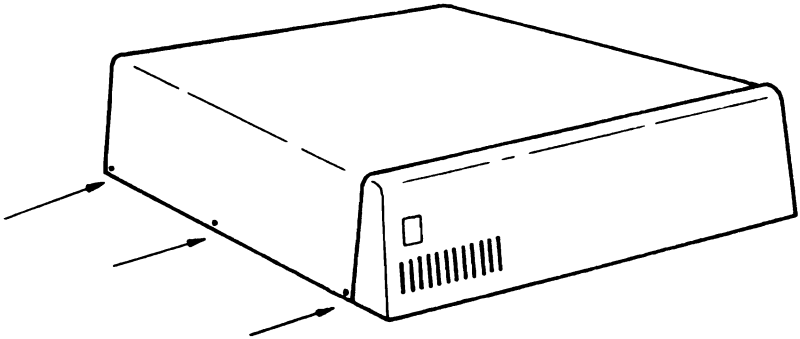


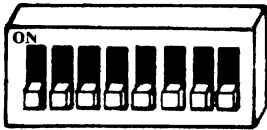
Figure 8A



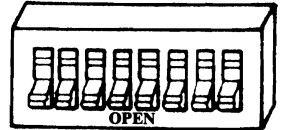
Appendix B: SWITCH AND JUMPER MODULES

Switch Overview

Figure 1B shows the two types of switch modules that may be found on a system board or on an adaptor board.



Slide Switch Module



Rocker Switch Module

Figure 1B
Switch Module

You set the switch modules by depressing one side of the designated rocker switch or by sliding a slide switch towards one side of the switch module. A switch set on OPEN or OFF is depressed or moved towards the bottom side of the switch module (side opposite the numbers) and a switch set on CLOSED or ON is depressed or moved towards the top side of the switch module (slide toward the numbers).

Jumper Blocks

There are two types of jumper blocks. Either type may be found on an adaptor board.

The first type of jumper block is a plastic rectangle with small sockets (holes) in the top. Positions in this type of jumper block are connected by placing a preformed U-shaped piece of wire in the sockets of two adjacent sockets. The two sockets must represent a position of the jumper block. To disconnect a position, remove the U-shaped piece of wire with a pair of tweezers or needle-nose pliers.

The second type of jumper block is a plastic block with perpendicular wire pins. Positions in this type of jumper block are connected by placing a plastic cap over two adjacent pins. The two pins must represent a position of the jumper block. To disconnect a position, remove the plastic cap by hand or with a pair of tweezers or needle-nose pliers.

The two types of jumper blocks and connectors are shown in Figure 2B.

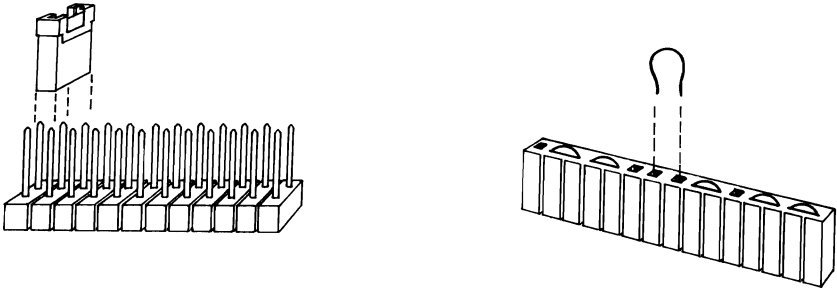


Figure 2B
Jumper Blocks and Connectors

When the connection has been made, the position is 'jumpered'. In Figure 3B Jumper Block JPR1 is jumpered in Positions A and C. Jumper Block JPR5 is jumpered in Position B.

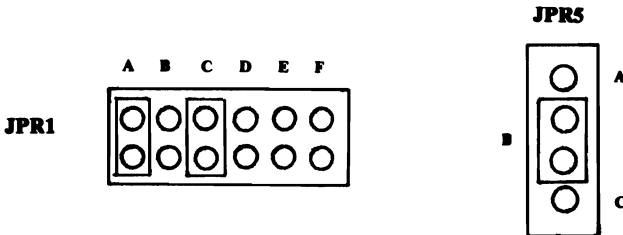
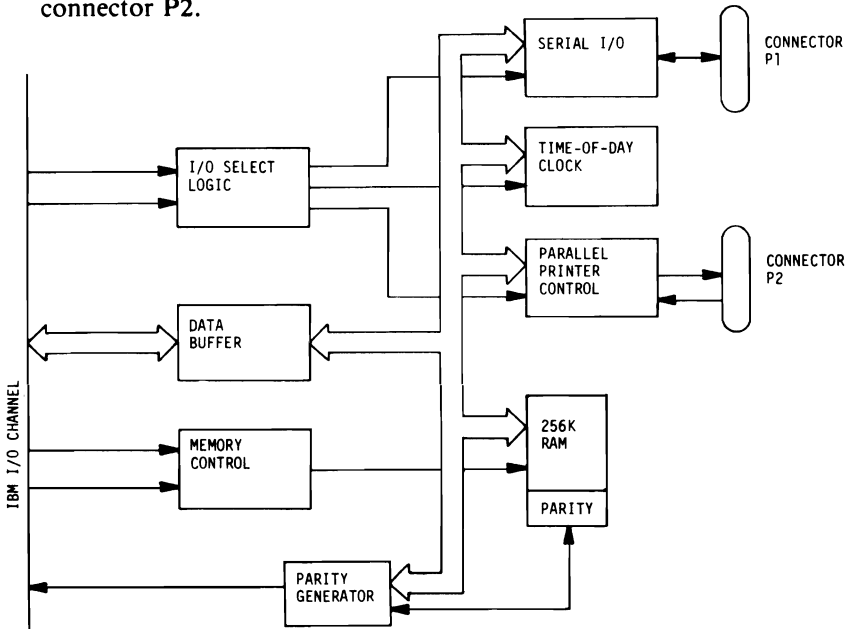


Figure 3B
Jumpered Positions

Appendix C:

BLOCK DIAGRAM DESCRIPTION

1. **Data buffer:** A bidirectional buffer transfers data between the IBM data lines and the RAM and I/O sections on the board.
2. **Memory control:** This sections controls the selection, refresh and address timing of the 256K bytes of RAM. Using the settings of switches SW1 and SW2, this section determines which 64K byte sections of memory will be addressed. The refresh provides the signals to the memory necessary to maintain the integrity of the memory. It is user transparent.
3. **384K Bytes RAM:** An array of RAM modules providing 256K bytes by 8 bits + 1 parity bit of memory space.
4. **Parity generator:** When any word is written into the memory this section produces the correct parity bit, and when any word is read from the 1STMATE Board memory this section checks for the correct parity. (Checks to verify the data is correct).
5. **I/O select logic:** When the address is correct, this section selects one of the I/O sections for I/O read or write operations.
6. **Serial I/O:** An RS-232 standard serial I/O port, jumper configurable as either 'DTE' or 'DCE'. The serial port also can be used as a 20ma current loop port and has modem control signals. The serial port is available through connector P1.
7. **Time-of-day clock:** The real-time clock is made up of an address latch, the clock itself (MM58167) and the battery for standby power.
8. **Parallel printer port:** The printer port is completely compatible with all IBM software and is meant to connect to an IBM, EPSON, or equivalent printer. The parallel printer port is available through connector P2.



Captain Board Block Diagram

Figure 1C



Appendix D: USE OF THE I/O SECTIONS

I/O addresses used:

Address	Function		Read and/or Write
888 Dec (378 Hex)	Printer Data] as LPT1	R/W
889 Dec (379 Hex)	Printer Status		R
890 Dec (37A Hex)	Printer Control		R/W
632 Dec (278 Hex)	Printer Data] as LPT2	R/W
633 Dec (279 Hex)	Printer Status		R
634 Dec (27A Hex)	Printer Control		R/W
893 Dec (37D Hex)	Clock Address latch] as TIME 1	W
895 Dec (37F Hex)	Clock data		R/W
637 Dec (27D Hex)	Clock Address latch] as TIME 2	W
639 Dec (27F Hex)	Clock data		R/W
1016 Dec (3F8 Hex)	Serial Data or Divisor Latch LSB] as COM1	R/W
1017 Dec (3F9 Hex)	Interrupt enable register or Divisor Latch MSB		W
1018 Dec (3FA Hex)	Interrupt identification register		R/W
1019 Dec (3FB Hex)	Line Control Register		W
1020 Dec (3FC Hex)	Modem Control Register		R
1021 Dec (3FD Hex)	Line Status Register		W/R
1022 Dec (3FE Hex)	Modem Status Register		W/R
			R
760 Dec (2F8 Hex)	Serial Data or Divisor Latch LSB] as COM2	R/W
761 Dec (2F9 Hex)	Interrupt enable register or Divisor Latch MSB		W
762 Dec (2FA Hex)	Interrupt identification register		R/W
763 Dec (2FB Hex)	Line control register		W
764 Dec (2FC Hex)	Modem control register		R
765 Dec (2FD Hex)	Line status register		W/R
766 Dec (2FE Hex)	Modem status register		R
			R

In the rest of this manual all programming references assume the Captain board is configured as Time 1, COM1 and LPT1. If any of these three sections is addressed at its alternate location then use the corresponding address for that location.



Appendix E:

DIRECT CONTROL OF THE PARALLEL PRINTER PORT

The printer port is hardware and software compatible with the IBM parallel printer adaptor interface. It can be easily used with an IBM printer or Epson MX80 using the IBM software. It could also be used as a general parallel port for input and output. The board has 12 buffered output points which are latched and can be written to or read from under program control and 5 steady-state input points that may be read at any time.

All signals associated with this port are on connector P2. The signal names apply only when connecting the port to an IBM printer (or equivalent).

The pin-out of P2 is:

Pin No.	Signal Name	Pin No.	Signal Name
1	- Strobe	14	- AUTO FEED
2	D0	15	ERROR
3	D1	16	- INIT
4	D2	17	SEL INP
5	D3	18	GROUND
6	D4	19	GROUND
7	D5	20	GROUND
8	D6	21	GROUND
9	D7	22	GROUND
10	- ACK	23	GROUND
11	+ BUSY	24	GROUND
12	+ PE	25	GROUND
13	+ SEL		

The Parallel Printer Port responds to two output and three input instructions. The output instructions transfer data into two latches whose outputs are available on P2. These outputs are also available to the CPU with two of the input instructions. The third input instruction allows the CPU to read the steady-state input points.

The five instructions, their addresses and functions are:

Output to 378 Hex - Printer data

Output Bit:	7	6	5	4	3	2	1	0
P2 Pin:	9	8	7	6	5	4	3	2

The 8 bit value written to 378 Hex appears on the respective pins of P2. These pins are capable of sourcing 2.6mA and sinking 24mA.

Input from 378 Hex - Printer data

The 8 bit value read from 378 hex is the 8 bit value latched by the most recent output to 378 hex.

Output to 37A Hex - Printer control

Output Bit:	7	6	5	4		3	2	1	0
Function:	No function		Interrupt			17	16	14	1
				Enable		P2 pins			

Bit 0 thru 3 are used as control signals for the printer. Of these, bits 3, 1, and 0 are sent out inverted.

If bit 4 is a 1, the card will interrupt the CPU if pin 10 of P2 goes from high to low. A 0 in bit 4 dissables the interrupt. The power on reset clears all these bits to zero.

The four output pins are driven by open collector drivers pulled up to +5v through 4.7Kohm resistors. They can each sink 7mA.

Input from 37A Hex - Printer control

The value read from 37AH is the data latched by the most recent output to 37AH.

Input from 379 Hex - Printer status

The value read from 379H presents the CPU with the real time status of the following pins:

Data Bit:	7	6	5	4	3	2	1	0
P2 Pin No.:	11	10	12	13	15			
Signal Name:	Busy	ACK	PE	SEL	ERROR	No function		

The Parallel Printer Port may be used with any IBM software that uses a parallel printer.

For a description of each of the printer signals, refer to the printer manual.

Appendix F:

DIRECT CONTROL OF THE SERIAL I/O PORT

The serial port is fully programmable and supports asynchronous communications only. It will add and remove start bits, stop bits, and parity bits. A programmable baud rate generator allows operations from 50 baud to 9600 baud. Five, six, seven or eight bit characters with 1, 1-1/2, or 2 stop bits are supported. A fully prioritized interrupt system controls transmit, receive, error, line status and data set interrupts. Diagnostic capabilities provide loopback functions of transmit/receive and input/output signals. It is implemented using an INS8250 serial communication I.C. The serial port signals are available on connector P1 and the pin-out is listed below.

P1: Serial Port Pinout - configured as 'DTE'

Pin	Signal
7	Ground
20	DTR (Data Terminal Ready)
4	RTS (Request to Send)
2	Serial Data Out
3	Serial Data In
11	Data Out (Current Loop)
9	Current Loop Return for Data Out
18	Current Loop Return for Data In
25	Optically Coupled Data In (Current Loop)
8	RLSD (Received Line Signal Detect)
6	DSR (Data Set Ready)
5	CTS (Clear to Send)
22	RI (Ring Indicator)

Note: There is a small but important difference between the IBM Asynchronous Communications Adapter and Tecmar's IBM-compatible serial ports. IBM's serial port may function even when the Data Set Ready (DSR) line is left 'floating' (either unconnected or not driven by the serial device used); this is, however, an improper and dangerous way to run an RS232 interface. Tecmar serial ports, by contrast, will operate in 'DTE' mode only when the DSR line (pin #6) is driven by the remote device. If this line is not driven, the operating system will respond with a

'Device Timeout' error. Analogously, if the Captain board is used in 'DCE' mode, pin #20 must be driven by the remote device or a 'Device Timeout' error will occur. To use Tecmar boards with a serial device which does not drive DSR, pin #6 must be jumpered to pin #20 on the Tecmar serial port (in either 'DTE' mode or 'DCE' mode).

P2: Serial Port Pinout - configured as 'DCE'

Pin	Signal
7	Ground
20	DSR (Data Set Ready)
4	CTS (Clear To Send)
2	Serial Data In
3	Serial Data Out
11	Data Out (Current Loop)
9	Current Loop Return for Data Out
18	Current Loop Return for Data In
25	Data In (Current Loop)
8	RLSD (Received Line Signal Detect)
6	DTR (Data Terminal Ready)
5	RTS (Request To Send)
22	RI (Ring Indicator)

The functions of these signals are:

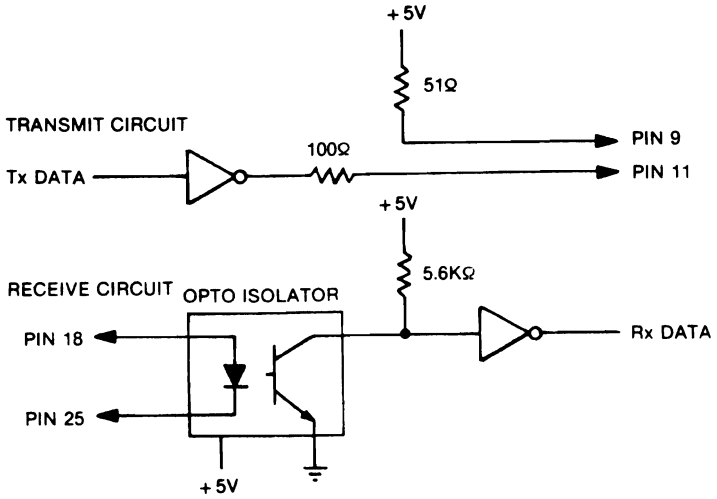
Data Terminal Ready (DTR): When low, informs the MODEM or data set that the INS8250 is ready to communicate. The DTR output signal can be set to an active low by programming Bit 0 (DTR) of the MODEM Control Register to a high level. The DTR signal is reset high upon a Master Reset operation. (See Section on Programming the Serial Port).

Request to Send (RTS): When low, informs the MODEM or data set that the INS8250 is ready to transmit data. The RTS output signal can be set to an active low by programming Bit 1 (RTS) of the MODEM Control Register to be high. The RTS signal is reset high upon a Master Reset operation.

Serial Output (SOUT): Serial data output to the communications link (peripheral, MODEM or data set). The SOUT is set to the Marking (Logic 1) state upon a Master Reset operation.

Serial Input (SIN): Serial data input from the communications link (peripheral device, MODEM, or data set).

Current Loop Data In and Out: The Current Loop Data In and Out is the same as the Serial Data In and Out except that the interface is implemented as illustrated in the diagram below.



Jumper Use

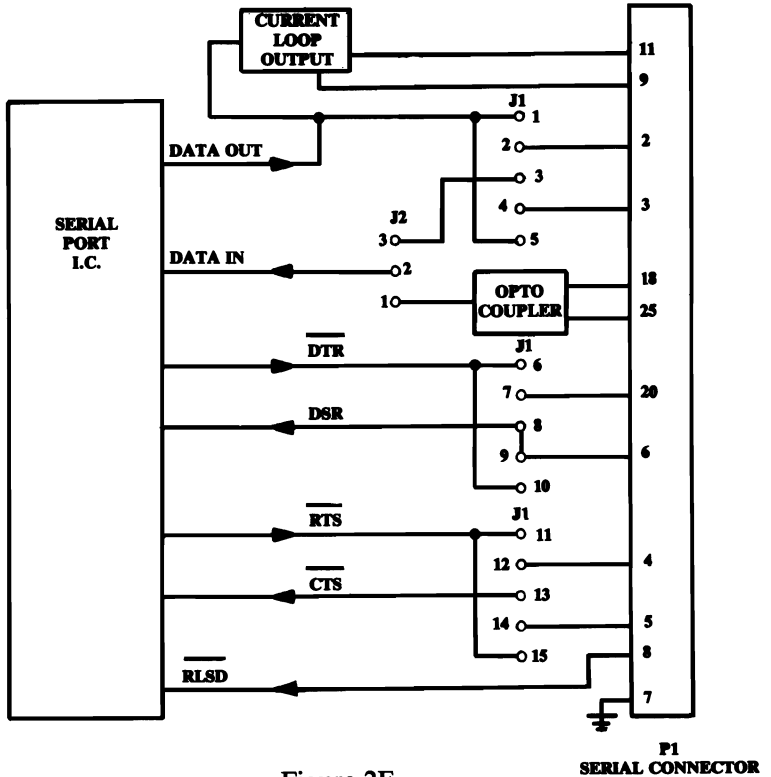


Figure 2F

Received Line Signal Detect (RLSD): When low, indicates that the data carrier has been detected by the MODEM or data set. The RLSD signal is a MODEM-Control function input whose condition can be tested by the CPU by reading Bit 7 (RLSD) of the MODEM Status Register. Bit 3 (DRLSD) of the MODEM Status Register indicates whether the RLSD input has changed state since the previous reading of the MODEM Status Register.

Note: Whenever the RLSD bit of the MODEM Status Register changes state, an interrupt is generated if the MODEM Status Interrupt is enabled.

Data Set Ready (DSR): When low, indicates that the MODEM or data set is ready to establish the communications link and transfer data with the INS8250. The DSR signal is a MODEM control function input whose condition can be tested by the CPU by reading Bit 5 (DSR) of the MODEM Status Register. Bit 1 (DDSR) of the MODEM Status Register indicates whether the DSR input has changed state since the previous reading of the MODEM Status Register.

Note: Whenever the DSR bit of the MODEM Status Register changes state, an interrupt is generated if the MODEM Status Interrupt is enabled.

Clear to Send (CTS): The CTS signal is a MODEM control function input whose condition can be tested by the CPU by reading Bit 4 (CTS) of the MODEM Status Register. Bit 0 (DCS) of the MODEM Status Register indicates whether the CTS input has changed state since the previous reading of the MODEM Status Register.

Note: Whenever the CTS bit of the MODEM Status Register changes state, an interrupt is generated if the MODEM Status Interrupt is enabled.

Ring Indicator (RI): When low, indicates that a telephone ringing signal has been received by the MODEM or data set. The RI signal is a MODEM-control function input whose condition can be tested by the CPU by reading Bit 6 (RI) of the MODEM Status Register. Bit 2 (TERI) of the MODEM Status Register indicates whether the RI input has changed from a low to a high state since the previous reading of the MODEM Status Register.

Note: Whenever the RI bit of the MODEM Status Register changes from a high to a low state, an interrupt is generated if the MODEM Status Interrupt is enabled.

There is one other signal associated with the serial port and that is the serial port interrupt that goes to IRQ4. The interrupt is enabled by setting the 8250 OUT2 output to zero (low).

Interrupt: Goes high whenever any one of the following interrupt types has an active high condition and is enabled via the Interrupt Enable Register: Receiver Error Flag; Received Data Available; Transmitter Holding Register Empty; and MODEM Status. The INTRPT Signal is reset low upon the appropriate interrupt service or a Master Reset operation. This interrupt is routed thru IRQ4 for COM1 or IRQ3 for COM2.

Serial Port Registers

Since the INS8250 is completely programmable, the system programmer may set up the serial port for any of the different modes available as needed for communicating with a particular device. The 8250 must first be initialized to the desired mode. Table 1 shows the reset conditions of all registers in the 8250. The control 'Master Reset' is the power-on reset generated anytime the IBM Personal Computer power is turned on.

Table 1F - Reset Control of Registers and Pinout Signals

Register/Signal	Reset Control	Reset State
Interrupt Enable Register	Master Reset	All Bits Low (0-3 Forced and 4-7 Permanent)
Interrupt Identification Register	Master Reset	Bit 0 is High, Bits 1 and 2 Low Bits 3-7 are Permanently Low
Line Control Register	Master Reset	All Bits Low
MODEM Control Register	Master Reset	All Bits Low
Line Status Register	Master Reset	All Bits Low Except Bits 5 & 6 are High
MODEM Status Register	Master Reset	Bits 0-3 Low Bits 4-7 - Input Signal
SOUT	Master Reset	High
INTRPT (RCVR Errs)	Read LSR/MR	Low
INTRPT (RCVR Data Ready)	Read RBR/MR	Low
INTRPT (RCVR Data Ready)	Read IIR/Write THR/MR	Low
INTRPT (MODEM Status Changes)	Read MSR/MR	Low
OUT 2	Master Reset	High
RTS	Master Reset	High
DTR	Master Reset	High
OUT 1	Master Reset	High

MR = Master Reset

All programming is accomplished by selecting the address of the 8250 register to be used and either reading from it or writing the data to it with the IBM Personal Computer's input or output instructions. The 8250's registers, their addresses and the function of each bit of data are shown in Table 2. Following Table 2 is a description of each register and its use.

Table 2F

Bit No	Register Address									
	0DLAB 0	0DLAB 0	1DLAB 0	2	3	4	5	6	0DLAB 1	1DLAB 1
	Receiver Buffer Register (Read Only)	Holding Register (Write Only)	Interrupt Enable Register	Interrupt Identification Register	Line Control Register	MODEM Control Register	Line Status Register	MODEM Status Register	Divisor Latch (LS)	Divisor Latch (MS)
0	Data Bit 0	Data Bit 0	Enable Received Data Available Interrupt (ERBF1)	0 if Interrupt Pending	Word Length Select Bit 0 (WLS0)	Data Terminal Ready (DTR)	Data Ready (DR)	Delta Clear to Send (DCTS)	Bit 0	Bit 8
1	Data Bit 1	Data Bit 1	Enable Transmitter Holding Register Empty Interrupt (ETBE1)	Interrupt ID Bit (0)	Word Length Select Bit 1 (WLS1)	Request to Send (RTS)	Overrun Error (OR)	Delta Data Set Ready (DDSR)	Bit 1	Bit 9
2	Data Bit 2	Data Bit 2	Enable Receiver Line Status Interrupt (ELSI)	Interrupt ID Bit (1)	Number of Stop Bits (STB)	Out 1	Parity Error (PE)	Trailing Edge Ring Indicator (TERI)	Bit 2	Bit 10
3	Data Bit 3	Data Bit 3	Enable MODEM Status Interrupt (EDSSI)	0	Parity Enable (PEN)	Out 2	Framing Error (FE)	Delta Receive Line Signal Detect (DRLSD)	Bit 3	Bit 11
4	Data Bit 4	Data Bit 4	0	0	Even Parity Select (EPS)	Loop	Break Interrupt (BI)	Clear to Send (CTS)	Bit 4	Bit 12
5	Data Bit 5	Data Bit 5	0	0	Stick Parity	0	Transmitter Holding Register Empty (THRE)	Data Set Ready (DSR)	Bit 5	Bit 13
6	Data Bit 6	Data Bit 6	0	0	Set Break	0	Transmitter Shift Register Empty (TSRE)	Ring Indicator (RI)	Bit 6	Bit 14
7	Data Bit 7	Data Bit 7	0	0	Divisor Latch Access Bit (DLAB)	0	0	Received Line Signal Detect (RLSD)	Bit 7	Bit 15

*Bit 0 is the least significant bit. It is the first bit serially transmitted or received.

**Programmable Baud Rate Generator Divisor Latches -
3F8 (DLAB = 1), 3F9 (DLAB = 1):**

The INS8250 contains a programmable Baud Rate Generator that takes the clock input (1.8432 MHz) and divides it by the 16-bit value in the two divisor latches. The result of this division is 16 times the Baud rate that will be produced. Address 3F8 (DLAB = 1) is the least significant byte of the 16-bit divisor. Address 3F9 (DLAB = 1) is the most significant byte of the divisor. These Divisor latches must be loaded during initialization in order to ensure the desired operation of the baud rate generator. Table 2F indicates the divisors to use for various Baud Rates.

**Table 3F
Interrupt Control Functions**

Desired Baud Rate	Divisor Used to Generate 16x Clock	Percent Error Difference Between Desired and Actual
50	2304 Dec	—
75	1536 Dec	—
110	1047 Dec	0.026
134.5	857 Dec	0.058
150	768 Dec	—
300	374 Dec	—
600	192 Dec	—
1200	96 Dec	—
1800	64 Dec	—
2000	58 Dec	0.69
2400	48 Dec	—
3600	32 Dec	—
4800	24 Dec	—
7200	16 Dec	—
9600	12 Dec	—

Line Status Register — 3FD Hex

This 8-bit register provides status information concerning the data transfer. The contents of the register are described below.

Bit 0: This bit is the receiver Data Ready (DR) indicator. Bit 0 is set to a logic 1 whenever a complete incoming character has been received and transferred into the Receiver Buffer Register. Bit 0 may be reset to a logic 0 either by the CPU reading the data in the Receiver Buffer Register or by writing a logic 0 into it from the CPU.

Bit 1: This bit is the Overrun Error (OE) indicator. Bit 1 indicates that data in the Receiver Buffer Register was not read by the CPU before the next character was transferred into the Receiver Buffer Register, thereby destroying the previous character. The OE indicator is reset after the CPU reads the contents of the Line Status Register.

Bit 2: This bit is the Parity Error (PE) indicator. Bit 2 indicates that the received data character does not have the correct even or odd parity, as selected by the even parity-select bit. The PE bit is set to a logic 1 upon detection of a parity error and is reset to a logic 0 after the CPU reads the contents of the Line Status Register.

Bit 3: This bit is the Framing Error (FE) indicator. Bit 3 indicates that the received character did not have a valid Stop bit. Bit 3 is set to a logic one whenever the Stop bit following the last data bit or parity bit is detected as a zero bit (Spacing level). It is reset after a read from the Line Status Register.

Bit 4: This bit is the Break Interrupt (BI) indicator. Bit 4 is set to a logic 1 whenever the received data input is held in the Spacing (logic 0) state for longer than a full word transmission time (that is, the total time of Start bit + data bits + Parity + Stop bits).

Note: Bits 1 through 4 are the error conditions that produce a Receiver Line Status interrupt whenever any of the corresponding conditions are detected.

Bit 5: This bit is the Transmitter Holding Register Empty (THRE) indicator. Bit 5 indicates that the INS8250 is ready to accept a new character for transmission. In addition, this bit causes the INS8250 to issue an interrupt to the CPU when the Transmit Holding Register Empty Interrupt enable is set high. The THRE bit is set to a logic 1 when a character is transferred from the Transmitter Holding Register into the Transmitter shift Register. The bit is reset to logic 0 concurrently with the loading of the Transmitter Holding Register by the CPU.

Bit 6: This bit is the Transmitter Shift Register Empty (TSRE) indicator. Bit 6 is set to a logic 1 whenever the Transmitter Shift Register is idle. It is reset to logic 0 upon a data transfer from the Transmitter Holding Register to the Transmitter Shift Register. Bit 6 is a read-only bit.

Bit 7: This bit is permanently set to logic 0.

Interrupt Identification Register - 3FAH

The INS8250 has an on-chip interrupt capability that allows for complete flexibility in interfacing to the microprocessor. In order to provide minimum software overhead during data character transfers, the INS8250 prioritizes interrupts into four levels. The four levels of interrupt conditions are as follows: Receiver Line Status (priority 1); Received Data Ready (priority 2); Transmitter Holding Register Empty (priority 3); and MODEM Status (priority 4).

Information indicating that a prioritized interrupt is pending and the type of that interrupt are stored in the Interrupt Identification Register (refer to Table 4). The Interrupt Identification Register (IIR), when addressed, freezes the highest priority interrupt pending and no other interrupts are acknowledged until that particular interrupt is serviced by the CPU. The contents of the IIR are indicated and described below.

One interrupt line is provided to the system. This interrupt is IRQ4 and will be active high. To allow the serial port to send an interrupt to the system, Bit 3 of the Modem Control Register must be set = 0 (low). At this point, any interrupts allowed by the Interrupt Enable Register will cause an interrupt.

Bit 0: This bit can be used in either a hardwired prioritized or polled environment to indicate whether an interrupt is pending. When bit 0 is a logic 0, an interrupt is pending and the IIR contents may be used as a pointer to the appropriate interrupt service routine. When bit 0 is a logic 1, no interrupt is pending and polling (if used) may continue.

Bit 1 and 2: These two bits of the IIR are used to identify the highest priority interrupt pending as indicated in Table 3F.

Bits 3 through 7: These five bits of the IIR are always logic 0.

Table 4F
Interrupt Control Functions

Interrupt Identification Register			Interrupt Set and Reset Functions			
Bit 2 0	Bit 1 0	Bit 0 1	Priority Level —	Interrupt Flag None	Interrupt Source None	Interrupt Reset Control —
1	1	0	Highest	Receiver Line Status	Overrun Error or Parity Error or Framing Error or Break Interrupt	Reading the Line Status Register
1	0	0	Second	Received Data Available	Receiver Data Available	Reading the Receiver Buffer Register
0	1	0	Third	Transmitter Holding Register Empty	Transmitter Holding Register Empty	Reading the IIR Register (if source of interrupt) or Writing into the Transmitter Holding Register
0	0	0	Fourth	MODEM Status	Clear to Send or Data Set Ready or Ring Indicator or Received Line Signal Detect	Reading the MODEM Status Register

Interrupt Enable Register — 3F9 (DLAB = 0):

This register enables the four types of interrupts of the INS8250 to separately activate the chip Interrupt (INTRPT) output signal. It is possible to totally disable the interrupt system by resetting bits 0 through 3 of the Interrupt Enable Register. Similarly, by setting the appropriate bits of this register to a logic 1, selected interrupts can be enabled. Disabling the interrupt system inhibits the Interrupt Identification Register and the active (high) INTRPT output from the chip. All other system functions operate in their normal manner, including the setting of the Line Status and MODEM Status Registers. The contents of the Interrupt Enable Register are indicated and described below.

Bit 0: This bit enables the Received Data Available Interrupt when set to logic 1.

Bit 1: This bit enables the Transmitter Holding Register Empty Interrupt when set to logic 1.

Bit 2: This bit enables the Receiver Line Status Interrupt when set to logic 1.

Bit 3: This bit enables the MODEM Status Interrupt when set to logic 1.

Bits 4 through 7: These four bits are always logic 0.

MODEM Control Register (MCR) — 3FC Hex:

This register controls the interface with a MODEM or device emulating a MODEM. The use of the Modem Control Register is described below.

Bit 0: This bit controls the Data Terminal Ready (DTR) output. When bit 0 is set to a logic 1, the DTR output is forced to a logic 0. When bit 0 is reset to a logic 0, the DTR output is forced to a logic 1.

Note: The outputs of the INS8250 may be applied to an EIA inverting line driver (such as the DS1488) to obtain the proper polarity input at the MODEM or data set.

Bit 1: This bit controls the Request to Send (RTS) output. Bit 1 affects the RTS output in a manner identical to that described above for bit 0.

Bit 2: No function.

Bit 3: This bit controls the Output 2 (OUT 2) signal, which is an auxiliary user-designated output. Bit 3 affects the OUT 2 output in a manner identical to that described above for bit 0. This OUT 2 enables the serial port interrupt when set to zero.

Bit 4: This bit provides a loopback feature for diagnostic testing of the INS8250. When bit 4 is set to logic 1, the following occur: the Transmitter Serial Output (SOUT) is set to the Marking (logic 1) state; the Receiver Serial Input (SIN) is disconnected; the output of the Transmitter Shift Register is 'looped back' into the Receiver Shift Register input; the four MODEM Control inputs (CTS, DSR, RLSD, and RI) are disconnected; and the three MODEM Control outputs (DTR, RTS, OUT 2) are internally connected to the four MODEM control inputs. In the diagnostic mode, data that is transmitted is immediately received. This feature allows the processor to verify the transmit and receive data paths of the INS8250.

In the diagnostic mode, the receiver and transmitter interrupts are fully operational. The MODEM Control Interrupts are also operational but the interrupts' sources are now the lower four bits of the MODEM Control Register instead of the four MODEM Control inputs. The interrupts are still controlled by the Interrupt Enable Register.

The INS8250 interrupt system can be tested by writing into the lower four bits of the MODEM Status Register. Setting any of these bits to a logic 1 generates the appropriate interrupt (if enabled). The resetting of these interrupts is the same as in normal INS8250 operation. To return to normal operation, the registers must be reprogrammed for normal operation and then bit 4 of the MODEM Control Register must be reset to logic 0.

Bits 5 through 7: These bits are permanently set to logic 0.

MODEM Status Register - 3FE Hex:

This 8 bit register provides the current state of the control lines from the MODEM (or peripheral device) to the CPU. In addition to this current-state information, four bits (D0 to D3) of the MODEM Status Register provide change information. These bits are set to a logic 1 whenever a control input from the MODEM changes state. They are reset to logic 0 whenever the CPU reads the MODEM Status Register.

The contents of the MODEM Status Register are indicated and described below.

Bit 0: This bit is the Delta Clear to Send (DCTS) indicator. Bit 0 indicates that the CTS input to the chip has changed state since the last time it was read by the CPU.

Bit 1: This bit is the Delta Data Set Ready (DDSR) indicator. Bit 1 indicates that the DSR input to the chip has changed state since the last time it was read by the CPU.

Bit 2: This bit is the Trailing Edge of Ring Indicator (TERI) detector. Bit 2 indicates that the RI input to the chip has changed from an On (Logic 1) to an Off (logic 0) condition.

Bit 3: This bit is the Delta Received Line Signal Detector (DRLSD) indicator. Bit 3 indicates that the RLSD input to the chip has changed state.

Note: Whenever bit 0, 1, 2, or 3 is set to a logic 1, a MODEM Status interrupt is generated (if enabled).

Bit 4: This bit is the complement of the Clear to Send (CTS) input. If bit 4 (loop) of the MCR (Modem Control Register) is set to a 1, this bit is equivalent to RTS in the MCR.

Bit 5: This bit is the complement of the Data Set Ready (DSR) input. If bit 4 of the MCR is set to a 1, this bit is equivalent to DTR in the MCR.

Bit 6: This bit is the complement of the Ring Indicator (RI) input. If bit 4 of the MCR is set to a 1, this bit is equivalent to OUT1 in the MCR.

Bit 7: This bit is the complement of the Received Line Signal Detect (RLSD) input. If bit 4 of the MCR is set to a 1, this bit is equivalent to OUT 2 of the MCR.

Receiver Buffer Register - 3F8 Hex (DLAB = 0)

The Receiver Buffer Register contains the received character. Bit D0 is the least significant bit and is the first bit received serially.

Transmitter Holding Register - 3F8 Hex (DLAB = 0)

The Transmitter Holding Register is written to with the character to be serially transmitted. Bit D0 is the least significant bit and is the first bit serially transmitted.

Sample Program

A simple example of setting up the INS 8250 Serial Port follows:

```
10 OUT 1019, 128
20 OUT 1016, 12
30 OUT 1017, 0
40 OUT 1019, 3
50 OUT 1016, DATA
```

Line 10 The address 1019 will access the Line Control Register as shown on the previous page as well as Table 2. Data 128 will set bit 7 high, the Division Latch Access Bit, (DLAB=1). With this set high, access can be gained to the Division Latches of the Baud Rate Generator.

Line 20 The address 1016 with DLAB=1 will access the Division Latch as shown in Table 2, which is the least significant of two 8 bit latches. The data output to this latch will be determined by the desired baud rate. Table 3 shows the proper divisor to use with the 1.8432 MHz crystal. In this case we output 12, since 9600 is the desired baud rate.

Line 30 The address 1017 with DLAB=1 will access the Most Significant of two 8 bit latches. Since our divisor of 12 was not above 255, it is represented by the Least Significant word. Therefore our data output to this register will equal zero.

Line 40 Once again the Line Control Register is accessed. Data output is 3, which sets both word length bits to 1, number of stop bits to zero, all parity bits to zero, Sets Break to zero, and resets the DLAB. Setting both word length bits to 1 will result in a word length of 8 bits. Because bit 3 equals zero, parity is disabled, so bits 4 and 5 will have no effect. Bit 6, Set Break, equals zero, so the transmitter output will not be disabled.

Line 50 With DLAB=0, address 1016 will now access the Transmitter Holding Register. Any data sent to this register will now be output in serial format.

To send or receive, the Line Status Register should be read first. By masking the appropriate bits, Data Ready (receive) or Transmitter Holding Register Empty (transmit) can be ascertained.

Since the Serial Port emulates the IBM serial port, all supporting software supplied by IBM can be used.

PC-DOS initializes the COM1/AUX port to 2400 baud, 8 data bits, 1 stop bit, and no parity. Also, note that BASIC reinitializes this port during its initialization process, and at this point the state of this port is undefined. The user must explicitly define the port parameters using the OPEN statement. For example, the statement:

OPEN 'COM1:300,E,7,1' as #1

will set the COM1 port to 300 baud, even parity, seven data bits, and one stop bit. No spaces occur within the 'COM1:' parameter string in the OPEN statement. No error will be reported in the case of spaces occurring, but initialization will not be performed correctly.

Under 1.0 the PC-DOS and BASIC COM1/AUX software is designed for use with a modem, not necessarily for use with a serial printer. When using a serial printer, one must tie CTS (clear-to-send, pin 7 on P1) to DSR (data-set-ready, pin 6). This will prevent Device Timeout or AUX I/O errors from occurring because of modem expectations. Below is a diagram of sample wiring connections to an Integral Data Systems Model 460 serial printer as an example:

Board P1 Port Pin #	Function	IDS-460 Printer Pin #
2	TX	3
3	RCV	2
5	CTS	20
6	DSR	20
7	Signal Ground	7

Also, you may experience difficulty when using a serial printer at high speeds. If the printer has a large character buffer, a Device Timeout or Device Fault error may occur during the time required for the printer to empty its buffer.



Appendix G:

DIRECT CONTROL OF THE TIMER CHIP

This section explains how to program the 58167 Timer chip on the Tecmar Captain board. The I/O ports and their functions are presented. A detailed explanation of the use of the 58167 IC as a time of day counter is given. Example programs are included in this section. The two examples illustrate the setting and reading of the 58167 as a time of day counter, the setting of the latches and the use of the alarm comparator.

I/O Locations and Functions

The Captain Board clock address latch is located at 893 decimal (37D Hex) for TIME1 or at 637 decimal (27D Hex) for TIME2 and the Time-of-Day I.C. data is located at 895 decimal (37F Hex) for TIME2. The use of these two locations is covered below and it is assumed the board is configured as TIME 1.

R - Read Function

W - Write Function

The value written to this location (from 0 through 21 decimal) is the pointer to the internal location in the 58167 that is to be accessed next. The functions of the 22 locations are as follows:

Location		Functional Location in 58167
0	R/W counter	thousandths of seconds
1	R/W counter	hundredths and tenths of seconds
2	R/W counter	seconds
3	R/W counter	minutes
4	R/W counter	hours
5	R/W counter	day of the week
6	R/W counter	day of the month
7	R/W counter	months
8	R/W latch	thousandths of seconds
9	R/W latch	hundredths and tenths of seconds
10	R/W latch	seconds
11	R/W latch	minutes
12	R/W latch	hours
13	R/W latch	day of the week
14	R/W latch	day of the month
15	R/W latch	months
16	R	interrupt status register
17	W	interrupt control register
18	W	counter reset
19	W	latch reset
20	R	rollover bit
21	W	'go' command

The thousandths of seconds, the hundredths and tenths of seconds counters and latches run from 0 to 9. The seconds and minutes run from 0 to 59. The hours use the 24 hour clock and begin with 0 (12 midnight) and run to 23 (11 p.m.). The day of week, day of month and month counters all begin with 1, not 0. The day of week runs from 1 = Sunday to 7 = Saturday. The day of month ranges from 1 to 31 and the months begin with 1 = January through 12 = December.

In the hundredths and tenths of seconds counter and latch, the high order four bits are the tenths of seconds and the low order four bits are the hundredths of seconds.

R/W - 58167 Clock Data

After the location is selected by writing its number (0-21) to the clock address latch, data can be written to or read from the 58167 real time clock from this location. The results of writing and reading each of the 22 locations of the clock will be detailed in the next section "Programming the 58167 Timer".

Overview

The 58167 chip is a time of day counter with alarm capabilities. Eight 8-bit counters are used to count the time of day. In addition, a series of eight 8-bit latches are provided for setting the alarm comparator. An alarm is triggered when the contents of the latches matches the contents of the timer registers. Counters and latches can be read and latches can be written or read without disturbing the real time clock functions. Interrupts can be controlled by any one of 8 timer sources. The use of all these facilities will be explained below.

All of the timer counters and latches count in BCD and therefore must be written to and read from using BCD notation. BCD (Binary Coded Decimal) is an encoding system whereby each four bits of an 8-bit byte has value of 0 to 9 decimal. The usual binary can code from 0 to 15 decimal in four bits. However, for timer purposes, the use of BCD is simpler and more common. In BCD, a two digit number is always coded in two 4-bit groups. For example, 27 would be 0010 0111 in BCD. The high four bits represent the 2 of 27 and the low four bits represent the 7. BASIC does not distinguish between the use of BCD and normal binary. It is therefore necessary to convert from decimal to BCD to write to the timer and to convert from BCD to decimal when reading from the timer. So to write 27 to the timer, the BCD representation must be determined. This BCD representation must then be written to the timer. We determined above that 27 in BCD is 0010 0111 which equals 39 dec. To set the timer to 27 then, a 39 would have to be sent out the appropriate port.

Fortunately, there are equations for converting from BCD to decimal and from decimal to BCD. This saves the user the trouble of writing down a BCD representation for every value to be sent to the timer and then converting that representation to binary. The conversion equations are as follows:

BCD to Decimal: $\text{Decimal} = \text{INT}(\text{BCD}/16)*10 + \text{BCD MOD } 16$

Note: MOD is the remainder of a division operation.

In this case, BCD MOD 16 = the remainder of BCD/16.

INT is the function signalling an integer divide.

Decimal to BCD: $\text{BCD} = \text{INT}(\text{Decimal}/10)*16 + \text{Decimal}$

In the first formula the 'BCD number' divided by 16 (integer division) is multiplied by 10 and then the integer remainder of the 'BCD number' which has been divided by 16 (BCD MOD 16) is added. For example:

The BCD number received was 27 hex, which is 39 decimal ($2 \times 16 + 7 \times 1 = 39$).

$39/16 = 2$ (integer result with a remainder of 7)

$2 \times 10 = 20$

$(39) \bmod 16 = 7$

$20 + 7 = 27$ (the decimal number)

As an example of the second formula, the decimal number to be output is 38.

$38/10 = 3$ (integer result)

$BCD = (38/10 \times 6) + 38 = (3 \times 6) + 38$

$BCD = 56$ decimal which gets output as 38 Hex.

If you are programming in assembly language, any BCD number is equivalent to a hexadecimal number with no digit greater than nine.

Programming the Counters

The eight counters contain the real time clock information. They may be written into to set the time or change it and may be read at any time without affecting the counting.

A. Setting the counters

The procedure for setting or changing the counters is as follows:

1. Write to the clock address latch the number of the counter to which you wish to write. Example: To set the hours, write OUT 893,4.
2. Set the timer by writing to the clock the number to which you wish to set the counter. Example: To set the counter selected to 15, write OUT 895,21. (15 decimal = 21 BCD).
3. Continue with these two steps until the counter registers desired are set.

B. Reading the counters

The procedure for reading the counters is very similar to the process of writing the counters. Read the counters as follows:

1. Write to the clock address latch the number of the counter which you wish to read. Example: To read the minutes, write OUT 893,3.
2. Read the counter from the clock. Example: To read the selected counter, give the command X=INP(895). X will equal the value of the counter selected (in BCD).

If you are programming in assembly code, it is possible to use the rollover bit. This bit will go high if any counter changes its value during the read operation. If the rollover bit is set, then the counter must be re-read within 900 microseconds or the rollover bit will be set again. To use this feature, write 20 to the clock address latch to select the rollover register. Then read the data port location to clear the rollover register. Select and read the counter as described above. Select and read the rollover bit again; if it is a '1' re-read the counter immediately. Otherwise, the read was valid and you can continue to the next operation.

C. Resetting the counters

The counters can be reset through the reset registers on the timer chip. To set all the counters to their lowest values, write 255 to the counter reset register as follows:

1. Write to the clock address latch an '18' to select the counter reset location. (OUT 893,18).
2. Write to the clock a '255' to select reset of all the counters. Example: OUT 895,255 will reset all of the counters to their minimum values.

D. Synchronizing the counters with real time

A special register is provided on the 58167 chip to make synchronizing the counter with real time easier. This register is called the 'Go' register. Writing a '1' to the Go register resets the thousandths, hundredths and tenths of seconds counters and the seconds counter. The purpose of simultaneously resetting these registers is to allow the user to set the slower counters and then issue the Go command at the time when real time indicates that their values should all be zero. Setting each of the faster counters (seconds and faster) individually results in inaccuracy because the setting process takes some time. The time it takes the user to set the tenths of seconds register is more than it takes for the tenths of seconds to advance. Using the Go register allows simultaneous reset of all the fast counting timers, thus reducing this problem.

The procedure for using the Go register is as follows:

1. Use the 'Setting the Counters' procedure to set the months, days of month, days of week, hours, and minutes counters. Example: Set the counter to Thursday, July 29, 10:15 a.m. when your real time clock says 00:14:30.
2. Send a '21' out the clock address latch to select the Go register. (OUT 893,21).
3. Type the command to send a '1' to the real time clock to issue the Go command. (OUT 895,1).
4. Repeat step 3, except this time, do not type the carriage return to send the command.
5. When your real time clock reads exactly 10:15:00:00:00, hit the carriage return to send the Go register command. The command will clear all the counters for seconds and faster, thus synchronizing the timer with your real time clock.

The purpose of setting the timer a few seconds ahead of the real time in step 2 is to assure that the minutes timer is not a minute behind the real time when the Go register is set. Setting the minutes to the actual time minutes would cause the timer to be reset to the beginning of that minute when the Go command is issued the first time. Setting the timer to the next minute allows the counter to be reset to the beginning of that next minute, so that issuing another Go command when the beginning of the next minute actually occurs will synchronize the timer and your real time clock.

Programming the Latches

Overview: The 8 latches are for setting the alarm comparator. When the counters equal the contents of the latches, then the alarm interrupt (interrupts are discussed in the next section) is set. The latches are written, read and reset in a manner very similar to the writing, reading and resetting of the counters.

A. Setting the latches

Each latch is set in the same two step process as setting the counters: the latch is selected and the setting is written to it in BCD. If you wish to set a latch to a 'don't care' state, set it to 204 decimal (CC Hex). This has the effect of having the latch always read the same as the counter. So, if you wished to set the latches so that the alarm interrupt would be triggered every day at 3 p.m., you would set the month, day of week, and day of month to 204, the hours to 15 (decimal), and the minutes, seconds and all other latches to 0.

The following procedure details how to program the latches:

1. Select the latch to be set and send its address value (8-15) to the clock address latch. Example: To set the hours, OUT 893,11.
2. Set the latch to a particular value by sending that value (in BCD) to the clock. Example: To set the latch to 25 (decimal), OUT 895,37. To set any latch to 'don't care', send out '204' decimal (OUT 891,204).
3. Repeat from step 1 to continue setting latches.

B. Reading the latches

The procedure to read the latches is nearly the same as the procedure to read the counters:

1. Select the latch to be read and send its address value (8-15) out the clock address latch. Example: To read the month, OUT 893,15.
2. Read the counter from the real time clock address. Example: To set X equal to the value of the latch, X = INP(895). If the value read is 204, then the latch is set to the 'don't care' state.

C. Resetting the latches

The latches are reset in exactly the same manner as the counters except that the address value to select from the clock address latch is 19, the latch reset, instead of 18, the counter reset. See the section 'Resetting the counters' for details and examples of the reset procedure.

Programming the Interrupts

The 58167 timer includes interrupt control and status registers for setting and reading timer-generated interrupts. The interrupts can be generated from any of eight different sources. By setting the interrupt control register, you determine which source will cause an interrupt. Reading the interrupt status register will tell you which source generated an interrupt that has just occurred.

The interrupt control and status registers are organized as follows:

Bit	Interrupt Source
7	month
6	week
5	day
4	hour
3	minute
2	second
1	tenth of second
0	latch alarm output

A. Setting the interrupt control register

The procedure to enable the timer interrupts involves setting the bits of the control register to select the interrupt source. Writing a '1' into the bit will enable the interrupt to occur when the counter indicated by that timer rolls over to zero. A '0' in a bit will disable that interrupt source. If more than one of the interrupt control bits is set to '1', the interrupt will occur at the fastest rate selected.

An interrupt source is chosen and enabled in the following manner:

1. Write a '17' to the clock address latch to select the interrupt control register. (OUT 893,17).
2. Select the interrupt source by writing the decimal number to the clock that will turn on the desired bit(s). Example: To enable an interrupt to occur at the beginning of every hour, OUT 895,16. (Writing a '16' will set the fourth bit to '1' and enable the hour interrupt.) Example: To enable an interrupt to occur when the time set in the latches is the same as the time of the counters, OUT 895,1.

B. Reading the interrupt status register

The interrupt status register can be read to determine the source of an interrupt. A '1' at any bit indicates that the source for which that bit stands was the cause of the interrupt. The bit will be a '0' if that source did not cause the interrupt. The status bits that are '1' and the interrupt output are reset to '0' after a read of the status register.

Note that the status register will reflect valid interrupt source information whether or not the board is interrupt jumpered. This gives the option of not generating real hardware interrupts, but instead reading the status register and acting upon the occurrence of a '1' at any bit as a 'polled' operation.

The interrupt status register is referenced and read in the following way:

1. Write a '16' to the clock address latch to choose the interrupt status register. (OUT 893,16).
2. Read the register from the clock address. The decimal number returned as a result will indicate which of the status bits is turned on. Remember that performing a read on this register will clear it and reset the interrupt output. Example: To read the contents of the status register into the variable X, X = INP (895). If X equals '1', then the alarm comparator was the cause of the interrupt. If X = '4' then the seconds counter was the cause of the interrupt.

Software Examples

The following two software examples illustrate the programming procedures outlined in the previous section. Each example treats a different function of the board.

Example 1 demonstrates setting and reading the time of day and date from the 58167 timer chip.

Example 2 illustrates the programming of the interrupts and how to read them in a 'polled' manner.

Example 1 is written in BASICA and must be run in BASICA because it uses key trapping functions not available in BASIC. Example 2 may be run in either BASIC or BASICA.

These example programs are provided to you only in these listings. If you wish to actually run these programs, you must first type the complete program in BASIC or BASICA. Once the program has been completely entered, you can then run it. Complete listings and line by line explanations of each example begin on the next page.

User entries required by the programs are clearly presented in menu form in the programs themselves.

**Example 1 - Setting and Reading the Time
from the Captain Board
General Description**

Example 1 illustrates the programming needed to set and read the time of day and date on the Captain board. It allows the user to enter the time of day and then reads the time of day back from the board and prints it once per second.

Line by Line Explanation

- | | |
|---------|--|
| 10-100 | Set the location of the clock address latch and the real time clock. Define functions to convert decimal numbers to BCD and to convert BCD numbers to decimal. |
| 120-170 | Menu to allow choice between setting the time, reading the time and exiting the program. |
| 200-390 | Prompts entry of the current time of day and date. The entry format is described and the meaningful limits on the entries are given. |
| 420-480 | Converts each item entered from decimal to BCD by use of the TOBCD function defined in line 70. All date and time entries must be converted to BCD before being sent to the clock registers. |
| 510-570 | Set the time of day and date entered on the 58167 clock. This is accomplished by referencing each time and date registers and writing to it the entered data in proper BCD form. This completes the setting of the time. |
| 600-640 | Enables the key trapping function which will allow you to terminate program execution when you are satisfied that the program is running correctly and the time is set correctly. |
| 660-700 | Reads the seconds register of the clock and converts the value to decimal using the TODEC function defined in line 100. A value one greater than the current second is determined (PLUSEC). The seconds are read again and again until the seconds roll over making the value of the seconds equal to the value of PLUSEC. |
| 710-750 | Read the rest of the time registers at the end of a one second interval. |
| 770-810 | Print the time of day and date on the screen. The printing will occur every second because of the control loop in lines 690-700. |
| 820 | Go back to line 680 and wait for another second to pass. |
| 850-920 | Sets a string equal to the name of the day of the week so that when the day and date are printed, the name of the day of the week is printed instead of the number associated with that day. |

- 940-1060 Sets a string equal to the name of the month so that when the day and date are printed, the name of the month is printed instead of the number associated with that month.
- 1090 Upon entry of the 'F1' key, this line is referenced. It returns program control to the main menu.
- 1120 Prints end of program message and ends the program.

```

10 REM Example 1 - Setting and Reading the Time of Day and Date
   on the 58167 Time of Day IC.
20 REM
30 REM Set the Clock Address Latch and Real Time Clock Address
40 BASE=893      Clock Address Latch is at 893
50 REM
60 REM Define a Function to Convert from Decimal to BCD
70 DEF FNTOB(X)=(X\10)*6 + X
80 REM
90 REM Define a Function to Convert from BCD to Decimal
100 DEF FNTODEC(X)=(X\16)*10 + X MOD 16
110 REM
120 CLS: PRINT: PRINT "CHOOSE THE FUNCTION YOU WISH TO PERFORM: "
130 PRINT "      1 - SET THE TIME OF DAY AND DATE"
140 PRINT "      2 - READ THE TIME OF DAY AND DATE"
150 PRINT "      3 - EXIT PROGRAM"
150 INPUT "      ";A
170 ON A GOTO 200,600,1120
180 REM
190 REM
200 REM Enter Time of Day and Date from Keyboard
210 CLS
220 PRINT "      WHEN PROMPTED BELOW WITH MTH,DW,DT ENTER:"
230 PRINT "          THE MONTH (1-12)"
240 PRINT "          THE DAY OF THE WEEK (1-7)"
250 PRINT "          THE DAY OF THE MONTH (1-31)"
260 PRINT
270 PRINT "      PLEASE SEPARATE YOUR ENTRIES WITH COMMAS"
280 PRINT
290 INPUT "ENTER MONTH, DAY OF WEEK, AND DATE - MTH,DW,DT  ",MONTH,DW,DATE
300 REM
310 PRINT: PRINT
320 PRINT "      WHEN PROMPTED BELOW WITH HR,MIN,SEC ENTER:"
330 PRINT "          THE HOUR (0-23)"
340 PRINT "          THE MINUTES (0-59)"
350 PRINT "          THE SECONDS (0-59)"
360 PRINT
370 PRINT "      PLEASE SEPARATE YOUR ENTRIES WITH COMMAS"
380 PRINT
390 INPUT "ENTER TIME OF DAY - HR,MIN,SEC  ",HR,MIN,SEC
400 REM
410 REM
420 REM Convert Time of Day and Date from Decimal to BCD
430 MONTH=FNTOB(MONTH)      convert month
440 DATE=FNTOB(DATE)        convert date
450 DW=FNTOB(DW)            convert day of week
460 HR=FNTOB(HR)            convert hour
470 MIN=FNTOB(MIN)         convert minutes
480 SEC=FNTOB(SEC)         convert seconds
490 REM
500 REM
510 REM Set Time of Day and Date as Entered
520 OUT BASE,7: OUT BASE+2,MONTH      set month
530 OUT BASE,6: OUT BASE+2,DATE       set day of month
540 OUT BASE,5: OUT BASE+2,DW         set day of week
550 OUT BASE,4: OUT BASE+2,HR         set hour
560 OUT BASE,3: OUT BASE+2,MIN        set minutes
570 OUT BASE,2: OUT BASE+2,SEC        set seconds
580 REM
590 REM

```

```

600 REM Read Time of Day and Date from Timer
610 CLS: PRINT "PRESS F1 KEY TO STOP PRINTING AND RETURN TO MENU"
620 PRINT
630 KEY(1) ON Turn on F1 key for key trapping
640 ON KEY(1) GOSUB 1090 When F1 key is struck, exit program
650 REM
660 REM Read Time and Print once per Second
670 OUT BASE,2: X=INP(BASE+2): SEC=FNTODEC(X) read seconds
680 IF SEC=59 THEN PLUSEC=0 ELSE PLUSEC=SEC+1 set up rollover check
690 OUT BASE,2: X=INP(BASE+2): SEC=FNTODEC(X) read seconds
700 IF SEC<>PLUSEC GOTO 690 if a second has passed, read the time
710 OUT BASE,3: X=INP(BASE+2): MIN=FNTODEC(X) read minutes
720 OUT BASE,4: X=INP(BASE+2): HR=FNTODEC(X) read hours
730 OUT BASE,5: X=INP(BASE+2): DW=FNTODEC(X) read day of week
740 OUT BASE,6: X=INP(BASE+2): DATE=FNTODEC(X) read day of month
750 OUT BASE,7: X=INP(BASE+2): MONTH=FNTODEC(X) read month
760 REM
770 REM Print Time of Day and Date
780 PRINT: PRINT "Time of Day and Date: ";
790 ON DW GOSUB 860,870,880,890,900,910,920
800 ON MONTH GOSUB 950,960,970,980,990,1000,1010,1020,1030,1040,1050,1060
810 PRINT DW$;" ", " ";MONTH$;" ";DATE;" ";HR;" ":";MIN;" ":";SEC
820 GOTO 680
830 REM
840 REM
850 REM Convert Day of Week Numbers to Day of Week Names
860 DW$="Sunday": RETURN
870 DW$="Monday": RETURN
880 DW$="Tuesday": RETURN
890 DW$="Wednesday": RETURN
900 DW$="Thursday": RETURN
910 DW$="Friday": RETURN
920 DW$="Saturday": RETURN
930 REM
940 REM Convert Month Numbers to Month Names
950 MONTH$="January": RETURN
960 MONTH$="February": RETURN
970 MONTH$="March": RETURN
980 MONTH$="April": RETURN
990 MONTH$="May": RETURN
1000 MONTH$="June": RETURN
1010 MONTH$="July": RETURN
1020 MONTH$="August": RETURN
1030 MONTH$="September": RETURN
1040 MONTH$="October": RETURN
1050 MONTH$="November": RETURN
1060 MONTH$="December": RETURN
1070 REM
1080 REM Stop Printing the Time and Return to the Menu
1090 RETURN 120
1100 REM
1110 REM End of Program
1120 PRINT: PRINT: PRINT "END OF PROGRAM"

```

Example 2 - Programming the Latches and Using the Alarm Comparator

General Description

Example 2 demonstrates how to set the latches on the 58167 timer. The alarm comparator interrupt is also used and the procedure of 'polling' the interrupts is demonstrated. When the interrupt status register indicates that an interrupt has occurred, a message to that effect is printed. The example is a purely artificial one, but it demonstrates how this type of process would be programmed in a context where polling the interrupts would be useful.

Line by Line Explanation

- 10-100 Define functions to convert BCD to decimal and decimal to BCD. Set the clock latch address and real time clock address.
- 120-180 Read the time of day and date from the board starting with the seconds and reading every counter through the month.
- 200-240 Print the time of day and date.
- 261-360 Enter the time which you wish to set in the latches. The time you enter here will be the time at which the alarm will occur.
- 370-380 Call the subroutines to convert the day of the week and month numbers into names.
- 390-460 Print the time at which the interrupt will occur (the time set in the latches) indicating all latches that are set to 'don't care' states. 'Don't care' states are noted by printing that any unit in that counter will satisfy the alarm requirements.
- 480-540 Convert latch settings to BCD. All 'don't care' entries are set to 204 decimal.
- 560-620 Write the 'converted times' and the 'don't care' codes established in lines 480-540 to the latches.
- 640-680 Enable the interrupt latch alarm by writing a '1' to the interrupt control register. Read the interrupt status register to clear it.
- 700-750 Read the interrupt status register until it indicates that the alarm comparator is true, meaning that the time in the latches is the same as the time in the counters. When this occurs, print a message indicating that an interrupt has occurred, print an end of program message, and end the program.
- 770-850 Subroutines to assign day of the week names to the day of the week numbers. This allows the day of the week to be printed for example as 'Sunday' instead of '1'.
- 870-990 Subroutines to assign month names to the month numbers. This allows the month to be printed for example as 'May' instead of '5'.

```

10 REM Example 2 - Program to set latches on the 58167 and to cause interrupt
   on latch comparator alarm
20 REM
30 REM Define a function to convert from BCD to Decimal
40 DEF FNTODEC(X)=(X\16)*10 + X MOD 16
50 REM
60 REM Define a function to convert from Decimal to BCD
70 DEF FNTOBCD(X)=(X\10)*6 + X
80 REM
90 REM Set the Clock Address Latch and Real Time Clock Addresses
100 BASE=893           Clock Address Latch is at 893
110 REM
120 REM Read time of day and date from the board
130 OUT BASE,2: X=INP(BASE+2): SEC=FNTODEC(X)
140 OUT BASE,3: X=INP(BASE+2): MIN=FNTODEC(X)
150 OUT BASE,4: X=INP(BASE+2): HR=FNTODEC(X)
160 OUT BASE,5: X=INP(BASE+2): DW=FNTODEC(X)
170 OUT BASE,6: X=INP(BASE+2): DATE=FNTODEC(X)
180 OUT BASE,7: X=INP(BASE+2): MONTH=FNTODEC(X)
190 REM
200 REM Print Time of Day and Date
210 CLS:PRINT: PRINT "Time of Day and Date:  "
220 ON DW GOSUB 790,800,810,820,830,840,850
230 ON MONTH GOSUB 880,890,900,910,920,930,940,950,960,970,980,990
240 PRINT DW$;" ",LMNTH$;" ";DATE;" ",HR$;" ";MIN$;" ";SEC
250 REM
260 REM Enter the Time to Set in the Latches
270 PRINT: PRINT
280 PRINT "IN RESPONSE TO THE PROMPTS BELOW, ENTER THE ALARM TIME "
290 PRINT "ENTER A CARRIAGE RETURN TO SET A <DON'T CARE> STATE IN A PARTICULAR
   ATCH"
300 PRINT: PRINT
310 PRINT TAB(10): INPUT "DAY OF WEEK: ",LDW$: LDW=VAL(LDW$)
320 PRINT TAB(10): INPUT "MONTH: ",LMNTH$: LMNTH=VAL(LMNTH$)
330 PRINT TAB(10): INPUT "DATE: ",LDATE$: LDATE=VAL(LDATE$)
340 PRINT TAB(10): INPUT "HOUR: ",LHOUR$: LHOUR=VAL(LHOUR$)
350 PRINT TAB(10): INPUT "MINUTE: ",LMIN$: LMIN=VAL(LMIN$)
360 PRINT TAB(10): INPUT "SECOND: ",LSEC$: LSEC=VAL(LSEC$)
370 ON LDW GOSUB 790,800,810,820,830,840,850
380 ON LMNTH GOSUB 880,890,900,910,920,930,940,950,960,970,980,990
390 PRINT
400 PRINT "INTERRUPT WILL OCCUR: "
410 PRINT TAB(10): IF LEN(LDW$)=0 THEN PRINT "ANY DAY": ELSE PRINT DW$:
420 PRINT " ": IF LEN(LMNTH$)=0 THEN PRINT "ANY MONTH": ELSE PRINT MNTH$:
430 PRINT " ": IF LEN(LDATE$)=0 THEN PRINT "ANY DATE": ELSE PRINT LDATE$:
440 PRINT " ": IF LEN(LHOUR$)=0 THEN PRINT "ANY HOUR": ELSE PRINT LHOUR$:
450 PRINT " ": IF LEN(LMIN$)=0 THEN PRINT "ANY MINUTE": ELSE PRINT LMIN$:
460 PRINT " ": IF LEN(LSEC$)=0 THEN PRINT "ANY SECOND" ELSE PRINT LSEC$
470 REM
480 REM Convert data to BCD equivalents. Determine <Don't Care> locations.
490 IF LEN(LMNTH$)=0 THEN LMNTH=204 ELSE LMNTH=FNTOBCD(LMNTH)
500 IF LEN(LDW$)=0 THEN LDW=204 ELSE LDW=FNTOBCD(LDW)
510 IF LEN(LDATE$)=0 THEN LDATE=204 ELSE LDATE=FNTOBCD(LDATE)
520 IF LEN(LHOUR$)=0 THEN LHOUR=204 ELSE LHOUR=FNTOBCD(LHOUR)
530 IF LEN(LMIN$)=0 THEN LMIN=204 ELSE LMIN=FNTOBCD(LMIN)
540 IF LEN(LSEC$)=0 THEN LSEC=204 ELSE LSEC=FNTOBCD(LSEC)
550 REM
560 REM Set times and <Don't Cares> in the latches
570 OUT BASE,15: OUT BASE+2,LMNTH
580 OUT BASE,19: OUT BASE+2,LDW

```

```

590 OUT BASE,14: OUT BASE+2,LDATE
600 OUT BASE,12: OUT BASE+2,LHOUR
610 OUT BASE,11: OUT BASE+2,LMIN
620 OUT BASE,10: OUT BASE+2,LSEC
630 REM
640 REM Enable Interrupt on Latch Alarm
650 OUT BASE,17
660 OUT BASE+2,1      reference interrupt control register
670 OUT BASE,16      select to interrupt on latch alarm
680 X=INP(BASE+2)    reference interrupt status register
690 REM              'read interrupt status register to clear
700 REM Wait for alarm to occur
710 OUT BASE,16      reference interrupt status register
720 X=INP(BASE+2)
730 IF X=0 GOTO 720
740 PRINT: PRINT "INTERRUPT HAS OCCURRED"
750 PRINT: PRINT "END OF PROGRAM": END
760 REM
770 REM Convert day of week numbers to day of week names
780 REM GET DAY OF WEEK AND MONTH NAMES
790 DW$="SUNDAY": RETURN
800 DW$="MONDAY": RETURN
810 DW$="TUESDAY": RETURN
820 DW$="WEDNESDAY": RETURN
830 DW$="THURSDAY": RETURN
840 DW$="FRIDAY": RETURN
850 DW$="SATURDAY": RETURN
860 REM
870 REM Convert month numbers to month names
880 MNTH$="JANUARY": RETURN
890 MNTH$="FEBRUARY": RETURN
900 MNTH$="MARCH": RETURN
910 MNTH$="APRIL": RETURN
920 MNTH$="MAY": RETURN
930 MNTH$="JUNE": RETURN
940 MNTH$="JULY": RETURN
950 MNTH$="AUGUST": RETURN
960 MNTH$="SEPTEMBER": RETURN
970 MNTH$="OCTOBER": RETURN
980 MNTH$="NOVEMBER": RETURN
990 MNTH$="DECEMBER": RETURN

```


Appendix H: ADDING MEMORY TO CAPTAIN

There are four rows or sections on your board where RAM chips may be installed for memory expansion. When a section is filled with nine chips (eight RAM chips and one chip for parity) then 64K of memory has been installed.

The RAM chips must be installed in specified sections on the Captain board. These rows and the order in which they should be filled are shown below.

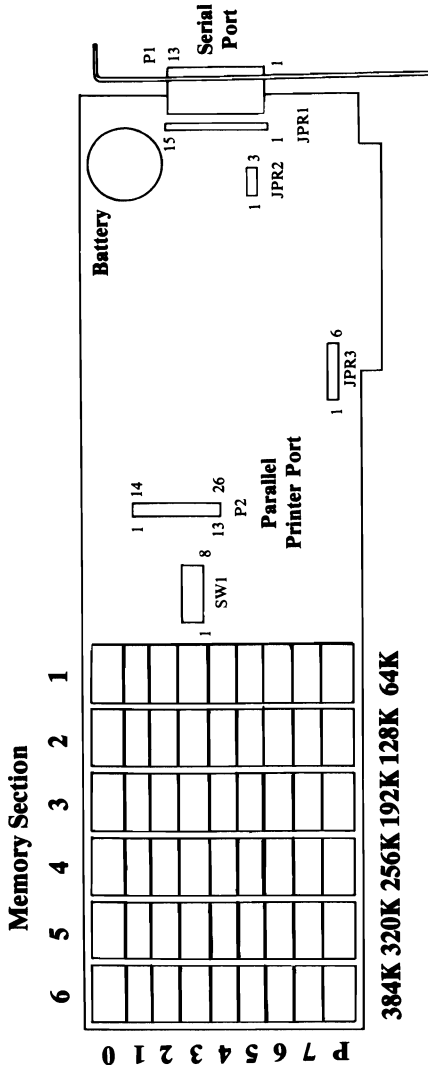


Figure 1H
Memory Section of Captain

When adding memory, insert the modules so that Pin 1 of each module is in the lower left hand corner of the socket. There is always a mark on every chip to indicate which end Pin 1 is on. This mark is usually a notch or a small depression on the top of the plastic case. Do not assume that the printing on the top of the module will orient it correctly.



Appendix I: SPECIFICATIONS

Slots:	One slot in IBM PC, XT or Tecmar expansion chassis
I/O Address:	Fixed at IBM serial and IBM parallel printer I/O locations. (COM1 or COM2, LPT1 or LPT2)
Clock/Calendar Address:	Jumper-selectable to 37D Hex or 27D Hex
Load:	1 TTL load/bus line max
PAL IC:	PAL16R8
RAM IC:	Intel 4164 or equivalent
Battery Life:	Approximately one year
Baud:	(Serial Port) 50-9,600
Power:	(with 256K) 1.6A @ +5V max 25mA @ -12V max 25mA @ +12V max



Appendix J:

THE PROGRAMMABLE ARRAY LOGIC (PAL) OPTION

Programmable Array Logic chips (PALs) perform pre-programmed logical operations on data inputs. The logical operations PALs can perform may be very simple, but are often highly complex. In the microcomputer realm, PALs may perform functions ranging from data-manipulation aids for fast-Fourier transforms and butterfly sorts to software 'locks' to prevent unauthorized software access.

Fast-Fourier transforms and butterfly sorts, for example, require that the bits in a byte of data be reversed. Software algorithms to perform this operation are slow and cumbersome. A PAL could, on the other hand, be programmed to do bit reversals of this kind with much greater speed, and they eliminate the cumbersome software one normally needs for such operations.

One use of the PAL enables you to 'scramble' data according to a set of predefined rules. If a PAL is programmed in such a way, whole disks full of data may be translated via PAL logic into an unintelligible form; if the data is fed back through the PAL again, then with proper programming it can be translated back into its original form.

Another important use of the PAL provides your system with a software 'lock-out'. In such applications, machine-code instructions are imbedded in the software to write an eight-bit byte to the PAL and to read the returned byte. If the appropriate byte is not returned, the software 'knows' that the properly programmed PAL is not installed in the computer being used. It can then abort operation to keep important software from being accessed by non-privileged systems.

For the user who lacks the resources necessary for PAL programming, a limited service is offered by Tecmar to provide custom-programmed software lock-out PALs. You specify a 'key number' and a 'response number', and Tecmar will program your PAL so that when the key number you selected is written to the PAL, the PAL will always return your selected response number.

The user who wishes to custom-program his PAL must have: (1) a thorough understanding of PAL logic, and (2) a properly equipped PROM programmer. Refer to the **PAL Programmable Array Logic Handbook** (Monolithic Memories, 1981) for complete information on PAL programming.



Appendix K:

Error Messages

Parity Check

If you see a Parity Check 2 error message on the screen, you have probably set the switches wrong. Just before this message appears, two numbers will flash on the screen. The second set of numbers will be 201, if you really have a memory error message.

The first set of numbers tells you which address in memory is involved. The number will mean either: 1) the memory on the Captain board is conflicting with existing memory or 2) you have set the switches on your computer's system board to indicate more memory than you have. Check your switch settings.

Look at the charts below to find out what memory bank your error message has occurred.

Table 1K

1st two characters of error	Corresponding section on Captain Board when the system board has:		Range of Memory (Decimal)	Range of Memory (HEX)
	64K	256K		
00	System board		0-64K	0000-FFFF
10	1	system board	64-128K	10000
20	2	system board	128-192K	20000
30	3	system board	192-256K	30000
40	4	1	236-320K	40000
50	5	2	320-384K	50000
60	6	3	384-448K	60000
70		4	448-512K	70000
80		5	512-576K	80000
90		6	576-640K	90000

Table 2K

Last two characters of error number	Row on Captain Board
0 1	0
0 2	1
0 4	2
0 8	3
1 0	4
2 0	5
4 0	6
8 0	7

If the first two characters of the error number are '00', then the switches on your system board have been set wrong. Review Pages 8-13 and then check your system board switches.

If the last two characters of the error number are anything other than the numbers given in Table 2K, then either your parity chip is defective (the parity chip is at the bottom in each column) or there is a problem with another chip on your Captain board. Contact your dealer.

If the last two characters do appear on the table in Figure 2K, then the problem could be one of the following: 1) a chip is in backwards (refer to Appendix H), 2) a chip has a bent leg or 3) a switch on the Captain board is set incorrectly. Check the switches on your Captain board and also the chips in the indicated row.

INDEX

B

BASIC 32, 33, 15F

C

clock/calendar 3, 4
battery-changing 20
direct control 1G-15G
interrupts 1G-15G
I/O function 18, 19, 23
jumpers 20-23, 2B
programming latches 7G
synchronizing counters with real time 6G
software examples 11G-16G
COM1 4, 6, 15, 18, 19, 23, 15F
COM2 4, 6, 15, 18, 19, 23, 15F
connector 2B
configuration 8
COPY command 32,
CTRL/P command 32
current loop 22, 1F-3F

D

DCE 21-22
default drive 34
DTE 21-22

E

Expansion Chassis 6, 26

F

features 4

I

IBM parallel printer adaptor 17
installation 25-29
I/O sections 1D

J

JPR1 21-22
JPR2 23
JPR3 23
jumpers 20-23

L

LLIST 33
LPRINT 33
LPT1 3, 6, 15, 18, 32, 1E-2E
LPT2 3, 6, 15, 18, 32, 1E-2E

M

MODE command 36, 37
modem 12F-13F

O

OPEN 'COM' 37
Option Retaining Bracket (ORB) 27

P

PAL 3, 1J
parallel port 3, 1E-2E, 14
 direct control 1E-2E
 installation 26-28
 location 14
 pin signals 1E
PC DOS 5, 6, 32, 35

S

serial port 4, 1F-14F
direct control 1F-14F
installation 26-28
interrupt 9F-11F
interrupt enable register 11F
interrupt identification register 9F
line status register 8F
location 14
modem control register 12F
modem status register 13F
pin signals 1F
programmable baud rate 7F
sample programs 14F-15F
SHIFT/PRT-SC 32
slot cover 26
system requirements 6

T

time-of-day interrupt 23, 1G

NOTES



NOTES

—

—

—

NOTES



Product Comment Form

Captain

20044

Your comments are a vital tool in assisting us in our efforts to continue the improvement of PC-Mate products and the accompanying manuals.

Suggestions may be used or distributed by Tecmar in any form it believes appropriate without any obligation whatever. This does not limit your use of the information that you originate and supply to Tecmar.

Comments:

TECMAR

Name _____

Address _____

City _____ State _____

Zip Code _____

TECMAR INCORPORATED

PERSONAL COMPUTER PRODUCTS DIVISION

6225 COCHRAN ROAD

SOLON (CLEVELAND), OHIO 44139-3377

TELEPHONE - (216) 349-0600, TELEX - 466692