O C T A G O N

S Y S T E M S

Embedded PCs For Extreme Environments

2050 PC/104 CPU
User's Manual

5867(0403)

Copyright

Disclaimer

OCTAGON SYSTEMS®

6510 W. 91st Ave.
Westminster, CO 80031

Technical support: 303–426–4521
Telephone: 303–430–1500
FAX: 303–426–8126
Web site: www.octagonsystems.com

IMPORTANT!

Please read the following section before installing your product:

Octagon's products are designed to be high in performance while consuming very little power. In order to maintain this advantage, CMOS circuitry is used.

CMOS chips have specific needs and some special requirements that the user must be aware of. Read the following to help avoid damage to your card from the use of CMOS chips.

Using CMOS circuitry in industrial control

Industrial computers originally used LSTTL circuits. Because many PC components are used in laptop computers, IC manufacturers are exclusively using CMOS technology. Both TTL and CMOS have failure mechanisms, but they are different. Described below are some of the failures that are common to all manufacturers of CMOS equipment. However, much of the information has been put in the context of the Micro PC.

Octagon has developed a reliable database of customer–induced, field failures. The average MTBF of Micro PC cards exceeds 11 years, yet there are failures. Most failures have been identified as customer–induced, but there is a small percentage that cannot be identified. As expected, virtually all the failures occur when bringing up the first system. On subsequent systems, the failure rate drops dramatically.

- Approximately 20% of the returned cards are problem–free. These cards, typically, have the wrong jumper settings or the customer has problems with the software. This causes frustration for the customer and incurs a testing charge from Octagon.
- Of the remaining 80% of the cards, 90% of these cards fail due to customer misuse and accident. Customers often cannot pinpoint the cause of the misuse.
- Therefore, 72% of the returned cards are damaged through some type of misuse. Of the remaining 8%, Octagon is unable to determine the cause of the failure and repairs these cards at no charge if they are under warranty.

The most common failures on CPU control cards are over voltage of the power supply, static discharge, and damage to the serial and parallel ports. On expansion cards, the most common failures are static discharge, over voltage of inputs, over current of outputs, and misuse of the CMOS circuitry with regards to power supply sequencing. In the case of the video cards, the most common failure is to miswire the card to the flat panel display. Miswiring can damage both the card and an expensive display.

- **Multiple component failures:** The chance of a random component failure is very rare since the average MTBF of an Octagon card is greater than 11 years. In a 7-year study, Octagon has never found a single case where multiple IC failures were not caused by misuse or accident. It is very probable that multiple component failures indicate that they were user–induced.

- **Testing "dead" cards:** For a card that is "completely nonfunctional", there is a simple test to determine accidental over voltage, reverse voltage or other "forced" current situations. Unplug the card from the bus and remove all cables. Using an ordinary digital ohmmeter on the 2,000 ohm scale, measure the resistance between power and ground. Record this number. Reverse the ohmmeter leads and measure the resistance again. If the ratio of the resistances is 2:1 or greater, fault conditions most likely have occurred. A common cause is miswiring the power supply.

- **Improper power causes catastrophic failure:** If a card has had reverse polarity or high voltage applied, replacing a failed component is not an adequate fix. Other components probably have been partially damaged or a failure mechanism has been induced. Therefore, a failure will probably occur in the future. For such cards, Octagon highly recommends that these cards be replaced.

- **Other over–voltage symptoms:** In over–voltage situations, the programmable logic devices, EPROMs and CPU chips, usually fail in this order. The failed device may be hot to the touch. It is usually the case that only one IC will be overheated at a time.

- **Power sequencing:** The major failure of I/O chips is caused by the external application of input voltage while the Micro PC power is off. If you apply 5V to the input of a TTL chip with the power off, nothing will happen. Applying a 5V input to a CMOS card will cause the current to flow through the input and out the 5V power pin. This current attempts to power up the card. Most inputs are rated at 25 mA maximum. When this is exceeded, the chip may be damaged.

- **Failure on power-up:** Even when there is not enough current to destroy an input described above, the chip may be destroyed when the power to the card is applied. This is due to the fact that the input current biases the IC so that it acts as a forward biased diode on power-up. This type of failure is typical on serial interface chips but can apply to any IC on the card.

- **Under rated power supply:** The board may fail to boot due to an under rated power supply. It is important that a quality power supply be used with the 2050 that has sufficient current capacity, line and load regulation, hold up time, current limiting, and minimum ripple. It is extremely import to select a supply that ramps up in 10ms or less. This assures that all the circuitry on the CPU control card sequences properly and avoids system lockup.

- **Excessive signal lead lengths:** Another source of failure that was identified years ago at Octagon was excessive lead lengths on digital inputs. Long leads act as an antenna to pick up noise. They can also act as unterminated transmission lines. When 5V is switch onto a line, it creates a transient waveform. Octagon has seen submicrosecond pulses of 8V or more. The solution is to place a capacitor, for example 0.1 µF, across the switch contact. This will also eliminate radio frequency and other high frequency pickup.

Note Any physical damage to the CPU control card is **not** covered under warranty.

Table of Contents

List of Figures

List of Tables

Overview: Section 1 – Installation

Section 1 provides installation and programming instructions, startup options, and system configuration program examples. The following chapters are included:

Chapter 1:	Overview
Chapter 2:	Quick start
Chapter 3:	Setup programs
Chapter 4:	Save and run programs

Chapter 1: Overview

Description

The 2050 PC/104 CPU control card is intended for higher–performance, low–power embedded control applications. The 2050 integrates serial communications, IDE hard disk port, floppy disk port, a multifunctional parallel port, keyboard and speaker ports, a CompactFlash socket, and a 10/100BaseT Ethernet port.

The 2050 can be used in a stand–alone mode or expanded through the PC/104 interface.

The 2050 comes with Datalight ROM-DOS installed and it is compatible with Windows NT, Windows 98, Windows CE, Linux, QNX, and DOS. Since the 2050 uses the same functional blocks as the Octagon Micro PC™ cards, the circuitry has been fully proven as reliable and the software is compatible with the software in the Micro PC series.

2050 major hardware features

CPU processor

The CPU is designed for low–power applications. It can be configured to run at 33, 50, 64, 66, 100, or 128 MHz (jumper selectable).

32 MB Surface Mount SDRAM

The 2050 comes with 32 MB of surface mount SDRAM. In OEM quantities it can be ordered with 16 MB surface mounted SDRAM.

Solid–state disk SSD1

SSD1 is a 2MB SMT boot flash that contains a 128K BIOS. In OEM quantities it can be ordered with 4 or 8MB SMT flash. The BIOS on SSD1 can be reprogrammed through the Z–tag interface.

CompactFlash socket

The CompactFlash socket accepts a CompactFlash card. The CompactFlash appears as an IDE device to the system.

Hard disk and floppy disk ports

The IDE hard drive port is terminated with a 44–pin, 2-mm connector. The IDE connector supplies +5V to the hard drive. The AT BIOS supports up to two IDE drives. The floppy drive port is terminated with a 26–pin ZIF connector.

Ethernet

The 2050 provides a 10/100 BaseT Ethernet port and supports the IEEE 802.3 Ethernet standard. The Ethernet controller IC chip provides the following:

- Integrated 10/100 BaseT transceiver interface
- Two LEDs for link and traffic status
- Auto negotiating between full and half-duplex modes
- Intel 82559ER chipset

Caution

Use a strain relief loop when connecting to the 2050 Ethernet connector to avoid damaging the connector.

Multipurpose connector

A multipurpose connector provides connection to the serial, parallel, keyboard and mouse, and speaker ports as well as the reset and battery interfaces.

Serial ports protected against ESD

The 2050 has two serial ports with combinations of RS–232C and RS–422/485 interfaces. These serial ports have the following common specifications:

- IEC1000, level 3, ESD protection specification
  - — Contact discharge ±6 kV
  - — Air–gap discharge ±8 kV
- Backdrive protection

- 16550 compatible
- Up to 115.2K baud
- 16–byte FIFO buffers
- Jumper–selectable terminations for RS–422/485
- Enabled and disabled in SETUP

Multifunctional printer port

The 2050 incorporates the latest enhanced parallel port. It includes the following features:

- Unidirectional
- Bidirectional
- IEEE 1284, ECP and EPP modes
- 12 mA of drive current
- Backdrive protection

The following represent applications in the multifunctional parallel port:

- LPT1 for PC compatible printers
- 17 general purpose digital I/O lines
- Up to a 4 x 4 matrix keypad
- 4–line alphanumeric display

Hardware reset

A hardware reset ensures complete reset of the system and all attached peripherals. A hardware reset can be done by any of the following methods:

- An expired watchdog timer cycle
- Depressing the reset switch
- Cycling power
- Power supervisor reset

Real time calendar/clock with battery–backup

The real time clock is fully AT compatible. An optional off–card battery powers the real time clock when the 5 volt supply is removed.

Watchdog timer added for safety

The watchdog timer resets the system or generates an NMI (nonmaskable interrupt) if the program stops unexpectedly. The watchdog is enabled, disabled and strobed under software control. The time–out period is programmable from 0.5 to 2 seconds. Using INT17 functions in DOS, the timeout period can be extended to 64 seconds.

PC/104 16–bit interface

The PC/104 interface accepts an 8– or 16–bit PC/104 expansion board. PC/104 expansion boards are available from several manufacturers. Up to two PC/104 expansion boards may be stacked on the 2050.

5 Volt operation lowers system cost

The 2050 operates from a single 5V ±5% supply.

- 5V ±5%
- +12V and –12V supplied to PC/104 connector; not required for 2050 operation

Rugged environmental operation

■ Operating temperature	–40° to 85°C, 99MHz and below –40° to 70°C, 128MHz
Nonoperating temperature	–55° to 95°C, nonoperating
■ Relative humidity	5% to 95% noncondensing
■ Altitude	–100 to 10,000 m
■ Shock	40g, 3 axis
■ Vibration	5g, 3 axis

Size

PCB size 3.55" x 3.775" x 0.68". See page 21 for size with connectors.

2050 major software features

Diagnostic software verifies system integrity automatically

The 2050 has built–in diagnostic software that can be used to verify on–card I/O and memory functions. On power-up, a series of tests is performed. If a problem occurs, the failed test can be identified by a flashing LED or a beep code. The test is performed automatically every time the system is reset or powered up. Memory verification does not require software, test equipment, monitor, keyboard, disks, or test fixtures. See the *"Troubleshooting"* chapter for a listing of tests and failures and their descriptions.

SETUP information stored in EEPROM for high reliability

Loss of SETUP data is serious in industrial applications. Most PCs store SETUP information in battery–backed CMOS RAM. If the battery fails or is replaced during routine maintenance, this information is lost. Without a keyboard and monitor in embedded applications, time consuming re–initialization is required. The 2050 stores the SETUP information in EEPROM with 512 bytes available to the user. Software routines to use this available memory come with the 2050.

Phoenix software BIOS

The 2050 has a Phoenix Software AT BIOS with Octagon BIOS extensions. The BIOS extensions include Octagon's INT17 functions.

Octagon BIOS extensions

On–board BIOS extensions allow easy access to watchdog timer functions, CMOS memory, etc.

Boot sequence

A 2050 can be configured to boot from a floppy, hard disk, CD ROM or CompactFlash.

Chapter 2: Quick start

This chapter covers the basics of setting up a 2050 system. Refer to the 2050 component diagrams, Figures 2–1 and 2–2, for the location of the various connectors. The following topics are discussed:

- Mounting the 2050
- Setting up a serial communications console I/O link between the 2050 and your desktop PC
- Downloading files to the 2050 and running a program from the SSD1 flash drive.

Hardware installation

WARNING!

The 2050 contains static-sensitive CMOS components. To avoid damaging your card and its components:

- **Ground yourself before handling the card and observe proper ESD precautions**
- **Disconnect power before removing or inserting a PC/104 expansion board**
- **When programming a memory device, place the device in the socket before applying power.**

Figure 2–1 2050 component diagram – top

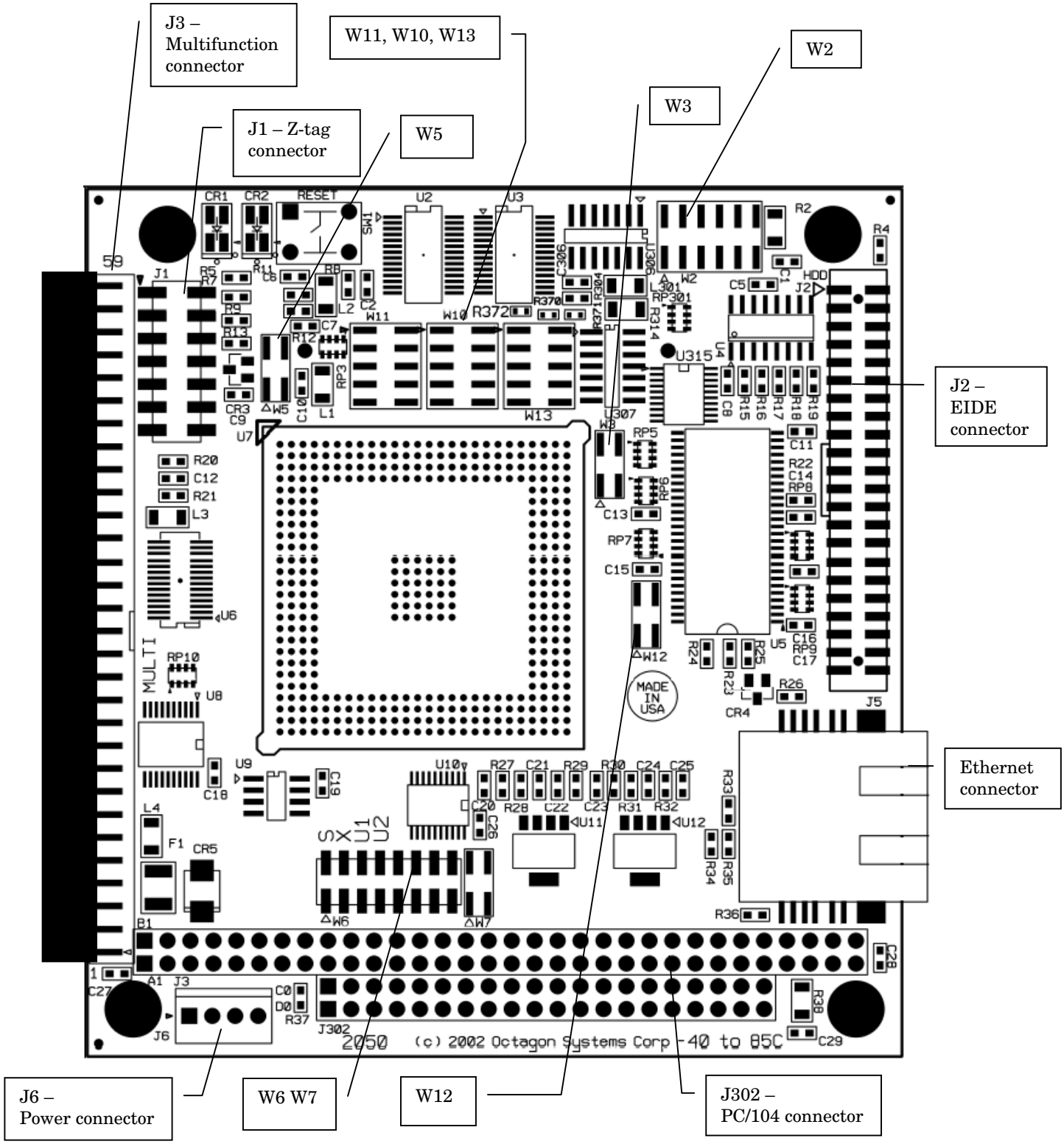


Figure 2–2 2050 component diagram - bottom

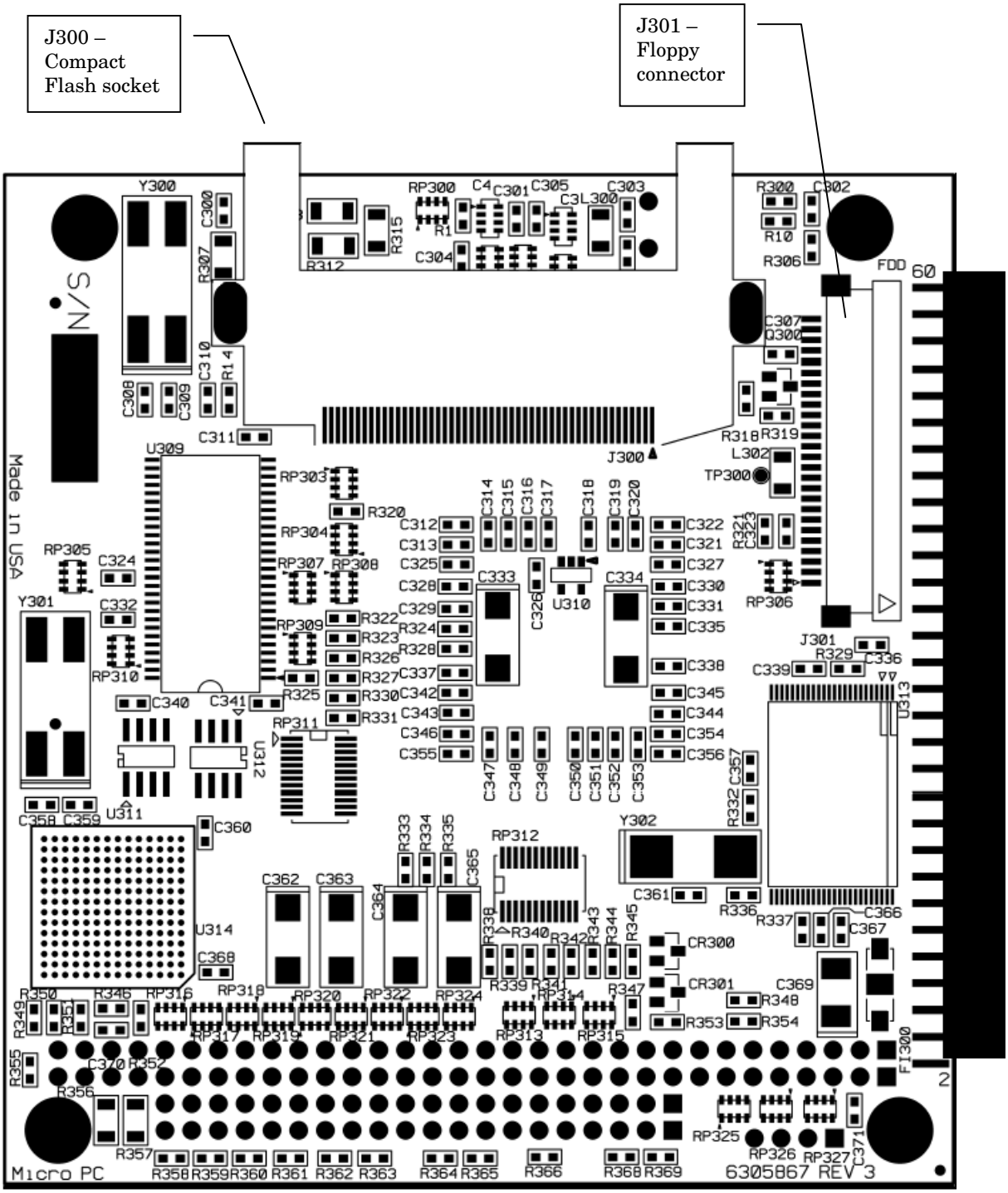
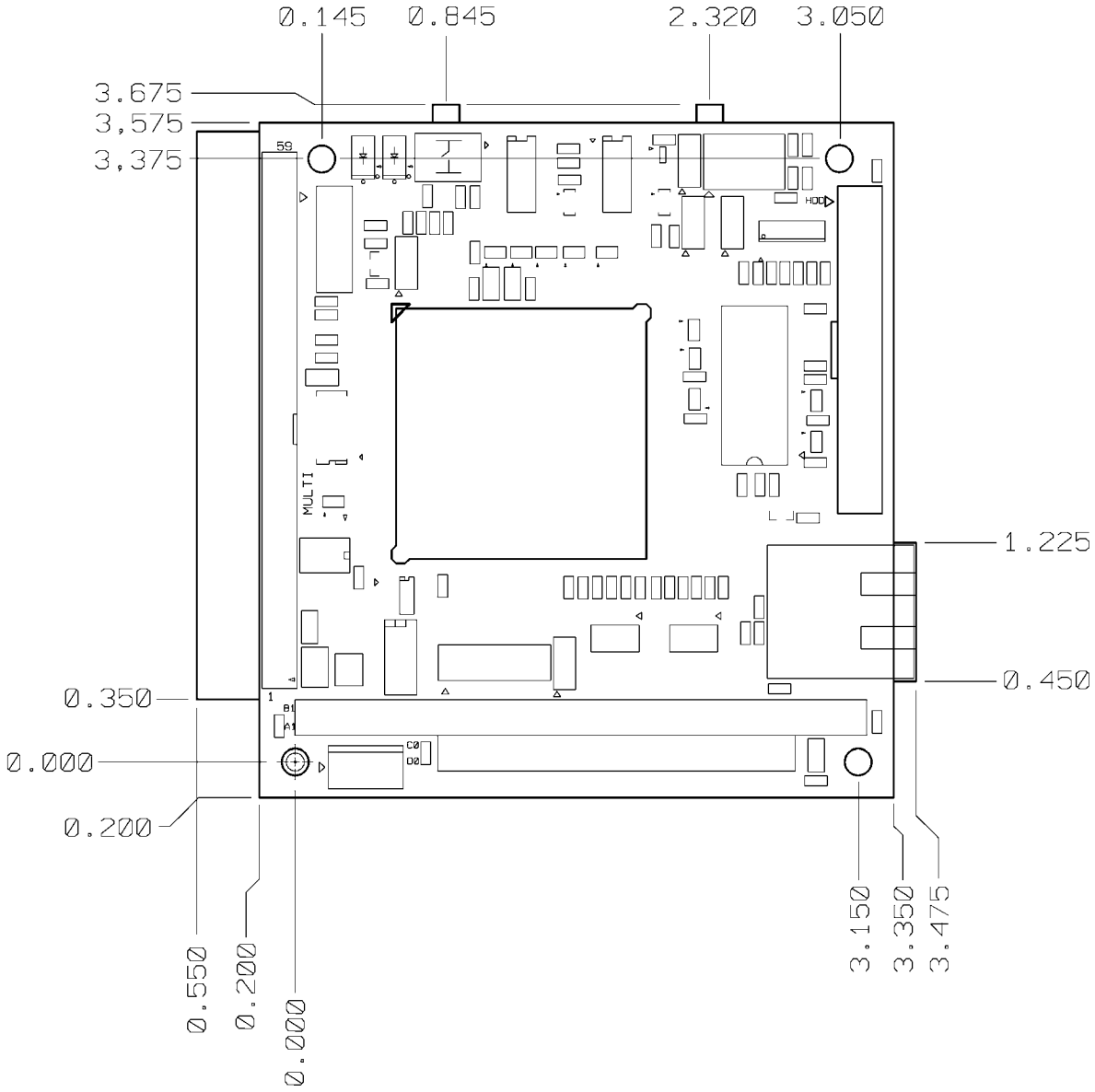


Figure 2–3 2050 center-to-center hole dimensions (inches)



Installing the 2050 CPU control card

Installation

To install the 2050 you will need the following equipment (or equivalent):

- 2050 PC/104 CPU control card
- Optional - a device with an operating system. The device could be floppy, hard disk, CompactFlash, or CD ROM. The operating system can be Windows NT, Windows 98, Windows CE, Linux, QNX, or DOS. This is optional because the 2050 comes with ROM-DOS installed in SSD1.
- +5V power supply - see *Power Supply Requirements* section
- MPC-18, multipurpose cable, #5905
- Null modem adapter, #2740
- Octagon Products, Manuals, and Catalog CD
- Windows HyperTerminal or equivalent terminal emulation software
- Your PC

Hardware components required to mount the 2050 (included):

- 4 threaded hex standoffs (4–40 x 3/4")
- 4 screws (4–40 x 3/16")
- 4 internal star lock washers (#4)
- 4 nuts (4–40)

Refer to the 2050 component diagram, Figures 2–1 and 2–2, for the location of various connectors, and to the mounting hole diagram, Figure 2-3, for installing your 2050 system.

Hardware mounting

1. Use the standoffs, washers, and screws and place them in the four holes on the 2050 board. Refer to Figure 2–3 for the center–to–center mounting hole dimensions and for the location of the designated holes used for mounting the hardware.

   WARNING!

   All four standoffs, screws and washers must be used to secure the 2050. The standoffs ensure full support of the board.

   WARNING!

   Verify that the washers and standoffs do not touch any of the component pads adjacent to the mounting holes. Damage will occur at power-up.

2. Connect a 5V power source to the 2050. Refer to the *Power Supply Requirements* section, page 25. The power supply connector is located at J6. Refer to Figure 2–4 and Table 2-1.

Note The +12V and –12V signals are routed to the PC/104 bus only.

Note See *Appendix A - Connectors* for additional mating information.

WARNING!

Accidentally crossing the wires, i.e., plugging +5V wires into the ground connector or the ground wires into the +5V connector will damage the 2050.

Figure 2–4 Power connector, J6 diagram

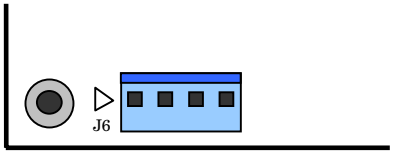


Table 2–1 Power connector: J6

Pin	Function
1	+5v
2	GND
3	+12V
4	-12V

Multipurpose connector

The multipurpose connector, J3, provides connection to the serial port, parallel port, keyboard and mouse, and speaker as well as the reset and battery interfaces. The MPC-18 multipurpose cable, or equivalent, is required. This cable breaks out to two DB-9 female serial connectors (COM1 and COM2), a parallel port connector with both a 25-pin and a 26-pin header, a PS/2 type keyboard connector, a PS/2 type mouse connector, a two-conductor speaker connector, an AT battery connector, and a reset switch. See Figures 2-5 and 2-6.

Note See *Appendix A - Connectors* for additional mating information.

Figure 2–5 Multipurpose connector, J3 diagram

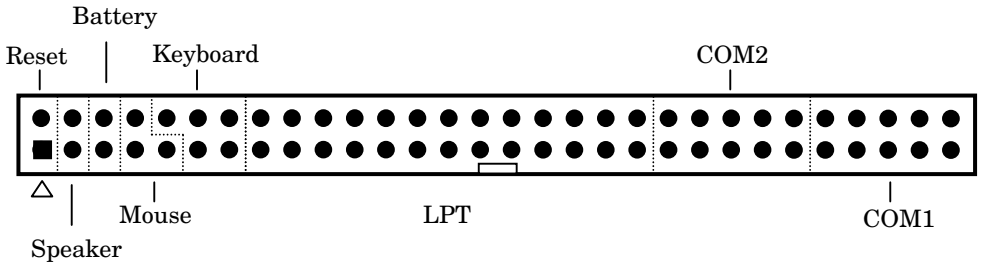
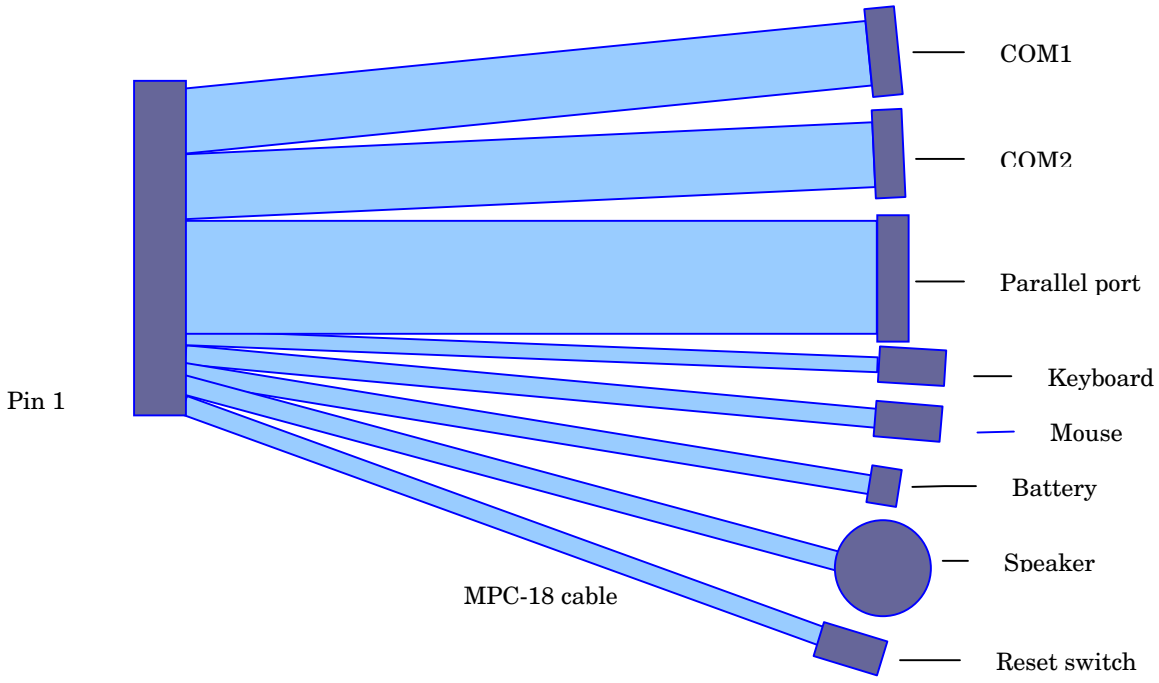
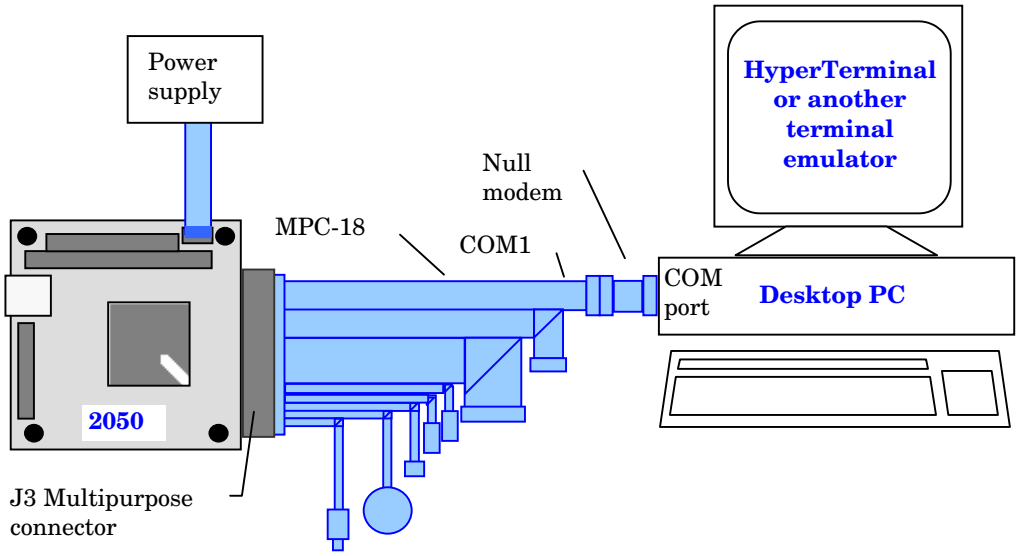


Figure 2–6 MPC-18 cable



Connecting a serial console

Figure 2–7 Serial console connections



1. Connect the equipment as per Figure 2-7.

Note You must use COM1 on the 2050 in order to establish a serial console connection.

2. Connect the null modem adapter to any serial port (COM1 through COM4) on your PC.

Note Refer to the HyperTerminal manual for more information on using a desktop COM port other than COM1.

2050 power supply requirements

The 2050 is designed to operate from a single +5 VDC supply, connected at J6. The typical current requirements for the 2050 is listed in the *Technical data* appendix. If you are using the PC/104 interface, you may also require ±12 VDC.

The user should consider factors such as the power cable conductor gauge, number and length of conductors, mating connectors, and the power requirements of external devices such as hard drives, floppy drives, displays, mouse, and keyboard.

It is important that a quality power supply be used that has sufficient current capacity, line and load regulation, hold up time, current limiting, and minimum ripple.

It is extremely import to select a supply that ramps up in 10ms or less. This assures that all the circuitry on the 2050 sequences properly and avoids system lockup.

Also, select a power supply that discharges quickly. If large power supply output capacitors are used, powering the system down and then up may lock up the 2050. If the power supply does not drain below 0.7V, the CMOS components on the 2050 will act like diodes and forward bias, potentially damaging the 2050 circuitry.

The proper selection of a quality power supply ensures reliability and proper functioning of the 2050.

WARNING!

Make sure the power supply is OFF when connecting the power cable to the 2050 board. Damage to the 2050 may occur if the power is ON when connecting the power cable.

Installing a different operating system

The 2050 comes with ROMDOS 7.1 installed on SSD1. You can install a different operating system. The operating system can be Windows NT, Windows 98, Windows CE, Linux, QNX, or DOS.

To install an operating system:

1. Connect a CD ROM drive to the EIDE connector, and insert the operating system CD into the drive.

2. Establish communication with the 2050, either using a serial console or a video card and keyboard. Refer to *Connecting a Serial Console* on page 25 for information on connecting a serial console. Note that a Linux console is 38400 baud regardless of the BIOS settings.

3. Enter SETUP by pressing the F2 key during BIOS POST sequence (this occurs between the memory test and bootup).

4. Configure the CD-ROM as a slave device, and change the boot

sequence to CD-ROM first.

5. Reboot the system.

6. Follow the on-screen dialog to load the operating system.

7. Reboot, either without the CD-ROM or after setting the BIOS to boot from a fixed disk first.

Establishing communications with the 2050

The following examples are given for a DOS operating system. If you are using a different operating system, refer to the operating system documentation for the appropriate commands. You can also refer to the Octagon web site at www.octagonsystems.com for information on Linux, QNX, and CE.net.

1. For communication using HyperTerminal (or equivalent), the following settings must be used:
   Baud rate: 38400
   Communications parameters: no parity, 8 data bits, 1 stop bit
   Flow control: none
   Terminal support: ANSI
   ANSI terminal option- wrap lines that exceed terminal width: Yes

2. Copy the 2050 files from the supplied CD-ROM to a subdirectory on your PC hard drive.

```
C:
MD C:\MPC
XCOPY D:\2050\*.* C:\MPC /S
```

3. Start HyperTerminal. You are now ready to establish communications between your PC and the 2050.

4. Power on the 2050.

5. A logon message similar to the one below will appear on your PC monitor:

```
PhoenixBIOS 4.0 Release 6.0
Copyright (C) 2001, ZF Micro Devices, Inc. All Rights Reserved.
ZFx86 BIOS Version 1.04 (PN 270-0012-0104ad)
Copyright 1985-2000 Phoenix Technologies Ltd.
All Rights Reserved
Octagon Release: V1.00-4/11/02

CPU = ZFx86  128 MHz
637K System RAM Passed
```

```
Press <F2> to enter SETUP
```

```
                        PhoenixBIOS Setup Utility

CPU Type              : ZFx86 CPU          System ROM     : E6E2 - FFFF
Speed                 : 127 MHz            BIOS Date      : 04/11/02

System Memory         : 640 KB             COM Ports      : 03F8 02F8
Extended Memory       : 31744 KB           LPT Ports      : 0378
Shadow Ram            : 384 KB             Display Type   : Monochrome
                                           PS/2 Mouse     : Not Installed


Hard Disk 0           : 2168MB             Diskette A     : 1.44/1.25 MB 3 1/2"
Hard Disk 1           : None               Diskette B     : Disabled
Hard Disk 2           : None
Hard Disk 3           : None
```

Note Your display message may be slightly different

```
Starting ROM-DOS...

2050 C:\>_
```

If you do not get the proper logon message:

- Check the HyperTerminal serial parameters of your PC to make sure they are set correctly. Parameters should be 38400 baud, 8 data bits, no parity, and 1 stop bit
- Make sure all jumpers are set to factory defaults
- If the system still does not respond, refer to the *Troubleshooting* chapter.

6. Use the directory command to make sure your equipment and software are working properly. Enter:

```
2050 C:\> DIR
```

A similar directory listing of ROM-DOS files stored in SSD1:

```
Volume in drive C is SSD1
Volume Serial Number is 281F-9D7D
Directory of C:\

COMMAND  COM        34,565 01-21-2000   6:22a
CONFIG   SYS            78 04-26-2000   1:51p
AUTOEXEC BAT            43 09-13-1999   2:14p
DOS          <DIR>         04-28-2000  12:09a
UTILS        <DIR>         04-28-2000  12:09a
5 file(s)         34,686 bytes
```

```
469,472 bytes free
```

7. You are now ready to transfer files between your PC and the 2050.

Transferring files between the 2050 and your PC

The following example for transferring files is for a ROM-DOS operating system. If you are using a different operating system, refer to your operating system documentation for the appropriate commands.

Once you have established communications between your PC and the 2050, you can serially download files to any read/write drive used by the 2050. You can then test and debug your application files. You can also upload files from the 2050 to your desktop PC for editing and debugging.

When booting from the 2050 BIOS drive, the default C: drive is SSD1. Drive D: is a virtual drive. All drives assigned can be accessed as read/write drives and files can be serially transferred to and stored on any of these drives.

Note The virtual drive is optional. If you do not need a virtual drive, do not use VDISK.SYS.

You can download files through the serial port to the 2050 using the TRANSFER utility. With this utility you download files, one at a time, to the 2050 using the XMODEM protocol. TRANSFER.EXE is used to send or receive files via the serial port (e.g., COM1). TRANSFER.EXE uses the XMODEM protocol, as does HyperTerminal. (See the note below on XMODEM).

Note XMODEM only transfers files in which the file size is exactly on a 128 byte boundary. If the file size does not fall exactly on the boundary, XMODEM automatically rounds the file size up to the next 128 byte boundary with padding characters. For example, a file with a size of 10,000 bytes, will be rounded up to 10,112 bytes, transferred, and written with the new file size. In most cases, this is not a concern, but in some instances the XMODEM padding causes problems. The padding problems become apparent when an application program is expecting a specific file size or is expecting characters other than the padding characters to be at the end of the file.

The following information on downloading files between the 2050 and your PC uses the example program DEMO.EXE. This file is on the Octagon Products, Manuals, and Catalog CD in the \2050\DEMO

directory.

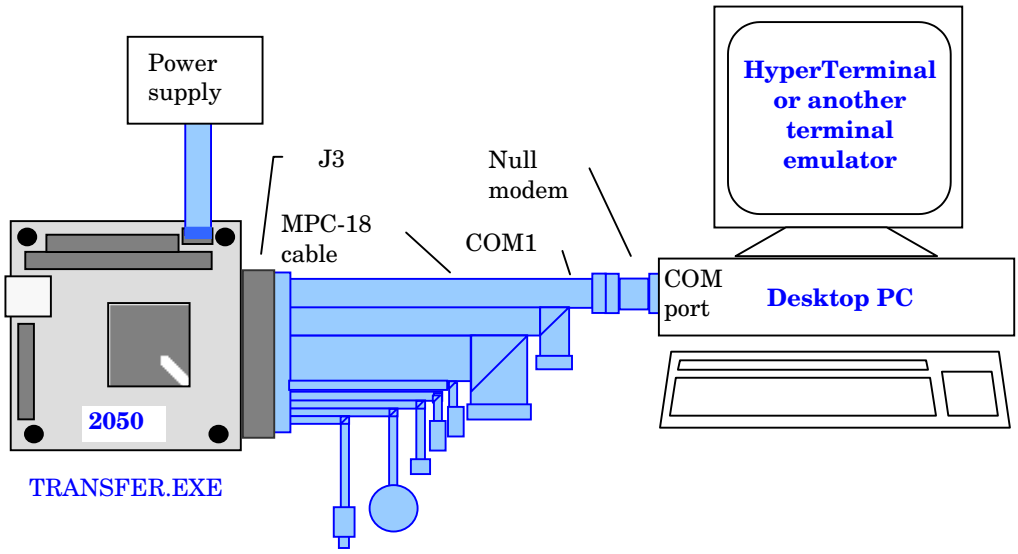
Downloading files to the 2050 using TRANSFER.EXE

The following procedures assume you are using HyperTerminal and that it is included in your directory path. For other communication programs, refer to their instructions on sending a file from your PC to a target system. Refer to the *Software utilities* chapter for specific information on using TRANSFER.EXE.

Hardware and software requirements:

- Desktop PC, running HyperTerminal, connected by a MPC–18 cable and a null modem adapter to J3 of the 2050
- A 2050 running TRANSFER.EXE out of COM1.

Figure 2–8 Downloading files using TRANSFER



1. Connect the equipment as per Figure 2-8.
2. On the desktop PC, log into the directory which contains the file(s) you will download to the 2050, for example:

```
C:\MPC\2050\DEMO
```

3. Start HyperTerminal and power on the 2050.
4. Execute the TRANSFER.EXE program from the 2050 by entering:

```
2050 C:\> TRANSFER D:DEMO.EXE
```

Note In this case, D: is the virtual drive assigned in CONFIG.SYS. Any

2050 read/write drive could be substituted.

Note When sending a file, enter the following:

```
2050 C:\> TRANSFER /S
```

The following message is displayed from the 2050:

```
Receiving E:DEMO.EXE . . .
```

5. Execute the following steps using HyperTerminal:
   a. Click Transfer
   b. Click Send File…
   c. Enter file name to send. E.g. DEMO.EXE
   d. Change protocol to 1K Xmodem (or Xmodem)
   e. Press <ALT><D> to enter the download screen.

Note TRANSFER.EXE will time-out if the program has not been started after approximately 40 seconds. If the time-out occurs, the following message from the 2050 is displayed:

```
Failed to receive D:DEMO.EXE!
Deleting D:DEMO.EXE
```

6. When the file transfer is complete, type the following DOS command to view the D: drive directory and confirm that your file has been transferred to the 2050:

```
2050 C:\> DIR D:
```

The system will display the contents of drive D:

```
Volume in drive D is VDISK vX.XX
Directory of D:\
DEMO          EXE            27264          06-07-96       2:57p
                             1 file(s)      27264 bytes
```

7. To execute the program you have just downloaded, type:

```
2050 C:\> D:DEMO
```

The DEMO program displays a message on your PC.

Chapter 3: Setup programs

This chapter discusses running the SETUP configuration program on the 2050 CPU Card. SETUP configures devices set up by the BIOS such as serial ports, floppy drives, etc.

SETUP

SETUP can be entered by pressing the “F2” key during the BIOS POST sequence (this occurs between the memory test and boot).

Also, by removing the USESETUP jumper from the “S” position at W6[1-2], you may force the setup to temporarily revert to the defaults shown in the following menus, which allows the user to reconfigure the setup.

The system will display the 2050 CPU Card PhoenixBIOS Setup Utility Main menu. Select the submenu by using the up/down arrows, then press <ENTER> (when using a video card). For a serial console, configuration, Ctrl + E is up and Ctrl + X is down.

Note Options having an asterisk are default settings.

Main menu

The Main menu allows you to set the basic system configuration.

```
                        PhoenixBIOS Setup Utility
    Main     Advanced     Power     Boot     Exit
```

System Time: [00:00:36] System Date: [01/01/1988] Legacy Diskette A: [1.44/1.25 MB 3 1/2"] Legacy Diskette B: [Disabled] > Primary Master [3253MB] > Primary Slave [None] > Secondary Master [None] > Secondary Slave [None] Memory Cache: [Enabled] NumLock: [Auto] System Memory: 640 KB Extended Memory: 31744 KB	Item Specific Help <Tab>, <Shift-Tab>, or <Enter> selects field.

```
F1  Help   ^v Select Item   -/+   Change Values      F9  Setup Defaults
Esc Exit   <> Select Menu   Enter Select > Sub-Menu  F10 Save and Exit
```

System Time: Sets the time for the system clock

System Date: Sets the date for the system clock

Legacy Diskette A: Enables or disables a legacy floppy disk drive. Choices are Disabled, 360 KB 5 ¼", 1.2 MB 5 ¼", 720 KB 3 ½", 1.44/1.25 MB 3 ½", 2.88 MB 3 ½"

Legacy Diskette B: Enables or disables a second legacy floppy disk drive. Note, however, that the 2050 only supports one floppy disk drive.

> Primary Master Accesses submenu for a Primary Master disk drive. Options are None, IDE Removable, CD-ROM, ATAPI Removable, Other ATAPI, User, and Auto.

> Primary Slave Same as Primary Master

> Secondary Master Same as Primary Master. Note, however, that the 2050 only supports two IDE devices.

> Secondary Slave Same as Primary Master. Note, however, that the 2050 only supports two IDE devices.

Memory Cache: Enables or Disables the memory cache.

NumLock: Auto, On, or Off

System Memory: Displays the amount of system memory which is on the card

Extended Memory: Displays the amount of extended memory on the card

Hard drive submenus

The Hard drive submenus allow you to set the primary/secondary/master/slave parameters. Except for older disk drives, the Auto selection will detect and display the correct parameters.

PhoenixBIOS Setup Utility

Main

Primary Master [3253MB]		Item Specific Help
Type:	[Auto]	User = you enter parameters of hard-disk drive installed at this connection. Auto = autotypes hard-disk drive installed here. 1-39 = you select pre-determined type of hard-disk drive installed here. CD-ROM = a CD- ROM drive is installed here. ATAPI Removable = removable disk drive is installed here.
Cylinders:	[6304]	
Heads:	[16]	
Sectors:	[63]	
Maximum Capacity:	3253MB	
Multi-Sector Transfers:	[16 Sectors]	
LBA Mode Control:	[Enabled]	
32 Bit I/O:	[Disabled]	
Transfer Mode:	[Fast PIO 4]	
Ultra DMA Mode:	[Disabled]	

F1 Help ^v Select Item -/+ Change Values F9 Setup Defaults
Esc Exit <> Select Menu Enter Select > Sub-Menu F10 Save and Exit

Advanced menu

The Advanced menu allows you to set advanced system configuration. Note that if items are incorrectly set in this menu, the system might malfunction.

```
                        PhoenixBIOS Setup Utility
   Main     Advanced     Power     Boot     Exit
```

	Item Specific Help
Setup Warning Setting items on this menu to incorrect values may cause your system to malfunction. >I/O Device Configuration >PCI Configuration Serial Video: [Enabled] Baud Rate: [38.4K] Secured Setup Configurations [No] Installed O/S: [Other] Reset Configuration Data: [No] Large Disk Access Mode: [DOS]	Peripheral Configuration

```
F1  Help   ^v Select Item   -/+   Change Values      F9  Setup Defaults
Esc Exit   <> Select Menu   Enter Select > Sub-Menu  F10 Save and Exit
```

Serial Video: Enabled, Disabled. Enables redirection of video and keyboard to COM1.

Baud Rate: 9600, 19.2K, 38.4K, 57.6K, 115K. Selects baud rate for serial console.

Secured Setup Configurations Yes or No. Yes prevents the operating system from overriding selections you have made in Setup.

Installed O/S: Other, Win95. Selects the operating system you use most often.

Reset Configuration Data: Yes or No. Yes erases all configuration data in a section of memory for ESCD (Extended System Configuration Data) which stores the configuration settings for non-PnP plug in devices. Select Yes when required to restore the manufacturer's defaults.

Large Disk Access Mode: DOS, Other. Select DOS if you have DOS. Select Other for another operating system such as Unix.

I/O Device Configuration submenu

The I/O Device Configuration submenu allows you to set the I/O configurations.

PhoenixBIOS Setup Utility

Advanced

I/O Device Configuration		Item Specific Help
PS/2 Mouse	[Auto Detect]	
Serial port A:	[Enabled]	
Base I/O address:	[3F8]	
Interrupt:	[IRQ 4]	
Serial port B:	[Enabled]	
Base I/O address:	[2F8]	
Interrupt:	[IRQ 3]	
Parallel port:	[Enabled]	
Mode:	[Bi-directional]	
Base I/O address:	[378]	
Interrupt:	[IRQ 7]	
Floppy disk controller:	[Enabled]	
Local Bus IDE Adapter:	[Enabled]	

F1 Help ^v Select Item -/+ Change Values F9 Setup Defaults
Esc Exit <> Select Menu Enter Select > Sub-Menu F10 Save and Exit

PS/2 Mouse	Disabled, Enabled, Auto Detect. Frees up IRQ12 if disabled.
Serial port A:	Disabled, Enabled, Auto. Enabled allows user to set configuration, while Auto uses the BIOS or OS configuration.
Base I/O address:	3F8, 2F8, 3E8, 2E8
Interrupt:	IRQ3, IRQ4
Serial port B:	Same as Serial Port A.
Base I/O address:	3F8, 2F8, 3E8, 2E8
Interrupt:	IRQ3, IRQ4
Parallel port:	Disabled, Enabled, Auto. Enabled allows user to set configuration, while Auto uses the BIOS or OS configuration.
Mode:	Output only, Bi-directional, EPP, ECP
Base I/O address:	378, 278, 3BC
Interrupt:	IRQ5, IRQ7
Floppy disk controller:	Disabled, Enabled, Auto. Enabled allows user to set configuration, while Auto uses the BIOS or OS configuration.
Local Bus IDE Adapter:	Disabled, Enabled. Enables the integrated local bus IDE adapter.

PCI Configuration submenu

The I/O Device Configuration submenu allows you to set the PCI configurations.

```
                     PhoenixBIOS Setup Utility
        Advanced
+-----------------------------------------------------+------------------------+
|               PCI Configuration                     | Item Specific Help     |
+-----------------------------------------------------+------------------------+
|                                                     |                        |
| >PCI/PNP ISA UMB Region Exclusion                   | Reserve specific       |
| >PCI/PNP ISA IRQ Resource Exclusion                 | upper memory blocks    |
| >PCI/PNP ISA DMA Resource Exclusion                 | for use by legacy ISA  |
| Ethernet IRQ                         [10]           | devices                |
|                                                     |                        |
+-----------------------------------------------------+------------------------+

 F1  Help   ^v Select Item   -/+   Change Values      F9  Setup Defaults
 Esc Exit   <> Select Menu   Enter Select > Sub-Menu  F10 Save and Exit
```

PCI/PNP ISA UMB Region Exclusion	See submenu
PCI/PNP ISA IRQ Resource Exclusion	See submenu
PCI/PNP ISA DMA Resource Exclusion	See submenu
Ethernet IRQ:	Disabled, Auto Select, 3, 4, 5, 7, 9, 10, 11, 12, 14, 15. Specifies IRQ for use by Ethernet. PCI cannot use an interrupt that is being used by an ISA or EISA device. Select Auto only if no ISA or EISA devices are on the system.

PCI/PNP ISA UMB Region Exclusion submenu

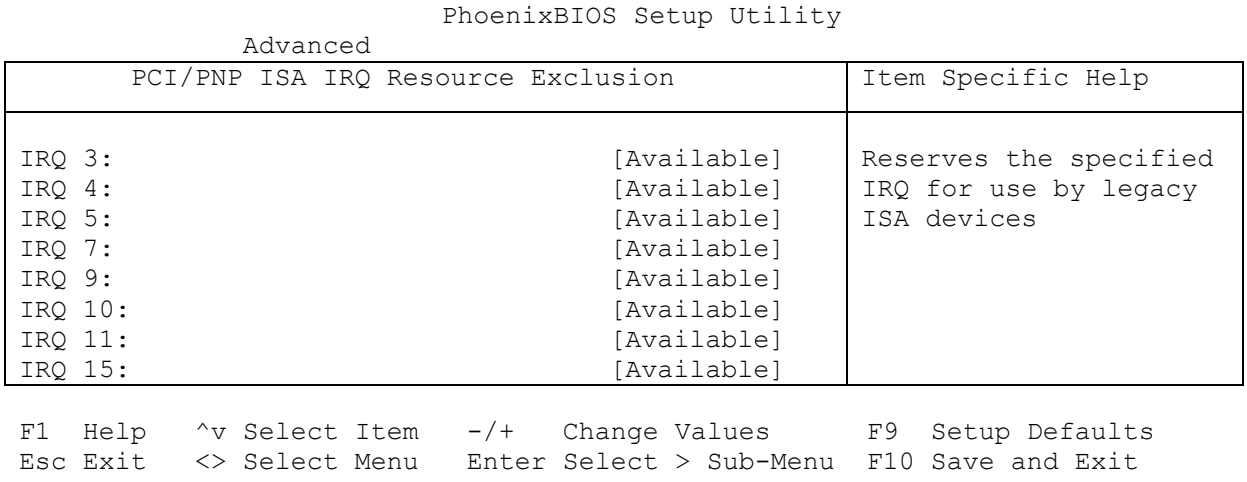
The PCI/PNP ISA UMB Region Exclusion submenu reserves the specified block of upper memory for use by legacy ISA devices. Options are Available or Reserved.

```
                     PhoenixBIOS Setup Utility
        Advanced
+-----------------------------------------------------+---------------------------+
|          PCI/PNP ISA UMB Region Exclusion           | Item Specific Help        |
+-----------------------------------------------------+---------------------------+
|                                                     |                           |
| C800 - CBFF:                         [Available]    | Reserves the specified    |
| CC00 - CFFF:                         [Reserved]     | block of upper memory     |
| D000 - D3FF:                         [Available]    | for use by legacy ISA     |
| D400 - D7FF:                         [Available]    | devices                   |
| D800 - DBFF:                         [Available]    |                           |
| DC00 - DFFF:                         [Available]    |                           |
+-----------------------------------------------------+---------------------------+

 F1  Help   ^v Select Item   -/+   Change Values      F9  Setup Defaults
 Esc Exit   <> Select Menu   Enter Select > Sub-Menu  F10 Save and Exit
```

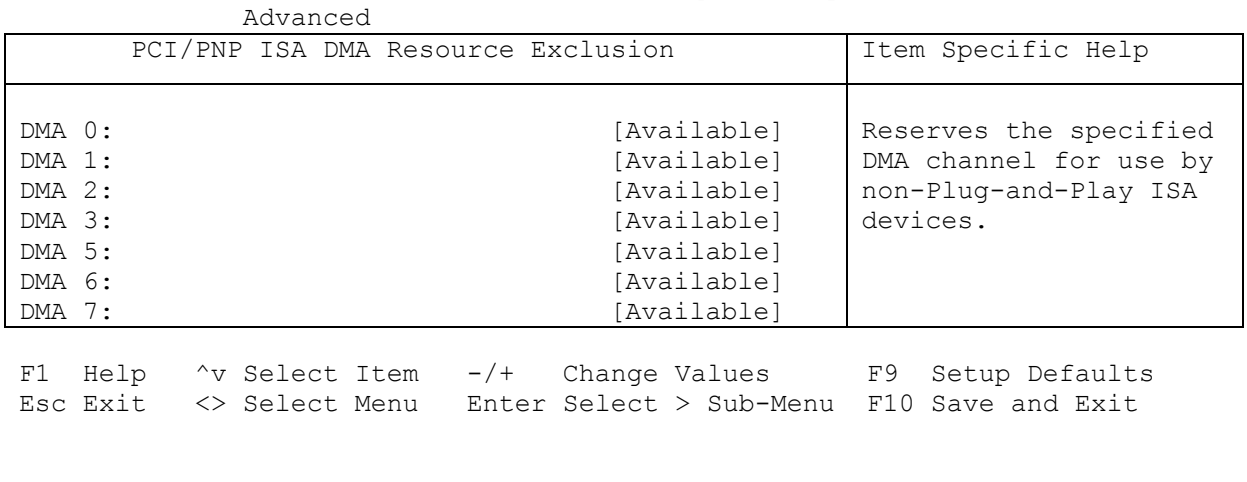
PCI/PNP ISA IRQ Resource Exclusion submenu

The PCI/PNP ISA IRQ Resource Exclusion submenu reserves the specified IRQ for use by legacy ISA devices. Options are Available or Reserved.



PCI/PNP ISA DMA Resource Exclusion submenu

The PCI/PNP ISA DMA Resource Exclusion submenu reserves the specified DMA channels for use by legacy ISA devices. Options are Available or Reserved.



Power menu

The Power menu allows you to set the power management configuration.

```
                        PhoenixBIOS Setup Utility
    Main     Advanced     Power     Boot     Exit
```

	Item Specific Help
Power Savings: [Disabled] Standby Timeout: Off Auto Suspend Timeout: Off Video Timeout: Off IRQ 3: [Enabled] IRQ 4: [Enabled] IRQ 12: [Enabled] Resume on Modem Ring: [On]	Maximum Power Savings conserves the greatest amount of system power. Maximum Performance conserves power but allows greatest system performance. To alter these settings, choose Customized. To turn off power management, choose Disabled.

```
F1  Help   ^v Select Item   -/+   Change Values      F9  Setup Defaults
Esc Exit   <> Select Menu   Enter Select > Sub-Menu  F10 Save and Exit
```

Power Savings: Disabled, Customize, Maximum Power Savings, Maximum Performance. Disabled disables all power management, Customize allows you to set parameters in the three menus below, the two Maximum settings use predefined values.

Standby Timeout: Off, 4, 8, 12, 16, 20, 24, 28 minutes. Inactivity period before system goes into Standby mode.

Auto Suspend Timeout: Off, 10, 20, 30, 40, 50, 60, 70 minutes. Inactivity period before system goes from Standby to Suspend mode.

Video Timeout: Off, 4, 8, 12, 16, 20, 24, 28, 32, 36, 40, 44 minutes. Inactivity period to power down monitor. Disabled turns CRT off in Standby mode.

IRQ 3/4/12: Disabled, Enabled. Enabling interrupt causes it to restore full On during Standby or Suspend.

Resume on Modem Ring: Modem ring serves as wakeup event

Boot menu

The Boot menu allows you to set the Boot configuration.

```
                        PhoenixBIOS Setup Utility
    Main     Advanced     Power     Boot     Exit
```

	Item Specific Help
Summary screen: [Disabled] Skip memory test [Yes] Floppy check: [Disabled] >Boot Order	Display system configuration on boot

```
F1  Help   ^v Select Item   -/+   Change Values      F9  Setup Defaults
Esc Exit   <> Select Menu   Enter Select > Sub-Menu  F10 Save and Exit
```

Summary screen: Enables or disables summary screen during bootup

Skip memory test Yes or No to skip memory test

Floppy check: Enables or Disables search for floppy drives during bootup

Boot Order Brings up Boot Order submenu, to set the order of drives to boot from.

Boot Order submenu

The Boot Order submenu allows you set the order of drives for booting.

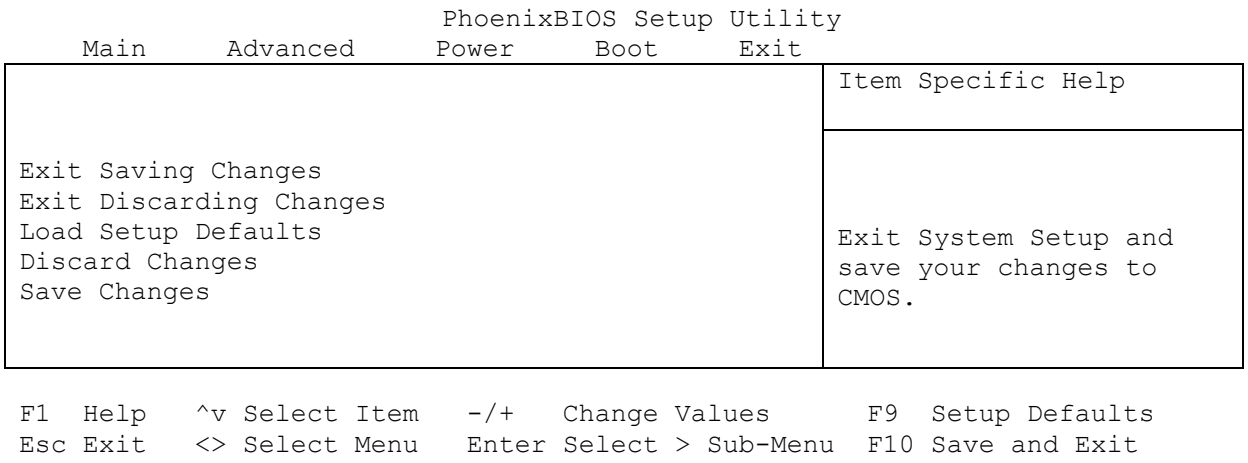
```
                        PhoenixBIOS Setup Utility
              Advanced
```

Boot Order	Item Specific Help
+Removable Devices +Hard Drive CD-ROM Drive	Order of Boot Devices <+> and <-> moves the device up or down.

```
F1  Help   ^v Select Item   -/+   Change Values      F9  Setup Defaults
Esc Exit   <> Select Menu   Enter Select > Sub-Menu  F10 Save and Exit
```

Exit menu

The Exit menu allows you to save or discard changes made during Setup. Esc does not exit this menu, you must select one of the menu items and press Enter. You can also press F9 or F10 at any time to exit Setup. When using the serial console F9 and F10 are not available; you must press down/up arrow to get to the proper option then press enter.



Chapter 4: *Save and run programs*

Save and run your programs on the 2050

Once you have written, tested and debugged your application, you can then save it to SSD1 or to another device. When you reboot the 2050, your program can automatically load and execute. This assumes your device already contains a bootable DOS.

This chapter describes the following:

- Saving an application program
- Autoexecuting the program from the 2050
- Overriding autoexecution of your program.

The information in this chapter assumes you are using ROM–DOS in your application. Some Microsoft programs make undocumented DOS calls. With ROM–DOS, an error returns when an undocumented DOS call is made, causing your program to operate erratically. We recommend using Microsoft's MSDOS when using programs with undocumented DOS calls. Refer to the section *Adding operating system startup files* in the *SSD1, Z-tag interface, CompactFlash, DRAM, and battery backup* chapter for more information on saving and autoexecuting programs.

Adding your application

To add your application, do the following:

1. Depending on your operating system you have installed, you may have the utility TRANSFER.EXE or COPY.EXE. Refer to the documentation included with your operating system to determine what utilities are available and how to use them.
2. Add or remove any device drivers for your application. You may want to do the same for the CONFIG.SYS file. Remember to add these drivers to your drive as well.
3. To autoexecute your application, add your application name to the AUTOEXEC.BAT file.

Overriding the autoexecution of your application

You may stop the autoexecution of your application by doing one of the following options:

Option 1

1. Press F5 or F8 on your local keyboard. For more information, see your ROM–DOS manual.

Option 2

1. Press Ctrl–C when the system is first starting. This halts all batch files.
2. Change AUTOEXEC.BAT and/or CONFIG.SYS to **not** call out your program.

Option 3

1. Install a floppy.
2. Change SETUP option “Boot Order” to “Boot 1st: Drive A:”.
3. Change SETUP to enable the floppy.
4. Boot from floppy.
5. Change AUTOEXEC.BAT on C:.

Overview: Section 2 – Hardware

Section 2 discusses usage, functions, and system configurations of the 2050 major hardware features. The following chapters are included:

Chapter 5:	Serial ports
Chapter 6:	LPT1 parallel port
Chapter 7:	Console devices
Chapter 8:	SSD1, Z-tag interface, CompactFlash, DRAM, and battery backup
Chapter 9:	External drives
Chapter 10:	Ethernet
Chapter 11:	PC/104 expansion

Chapter 5: Serial ports

Description

The 2050 has two serial ports, COM1 and COM2, which are accessed through the multipurpose connector, J3 . These serial ports interface to a printer, terminal, or other serial device. All ports support 5–, 6–, 7–, or 8–bit word lengths, 1, 1.5, or 2 stop bits, and baud rates up to 115.2K.

COM1 and COM2 are 8 wire interfaces and can be configured as RS–232, RS–422, or RS–485 interfaces.

Note RS-422 and RS-485 does not function correctly on revision 2 or lower 2050 cards. Contact Octagon Technical Support for additional information.

Both serial ports have the following specifications:

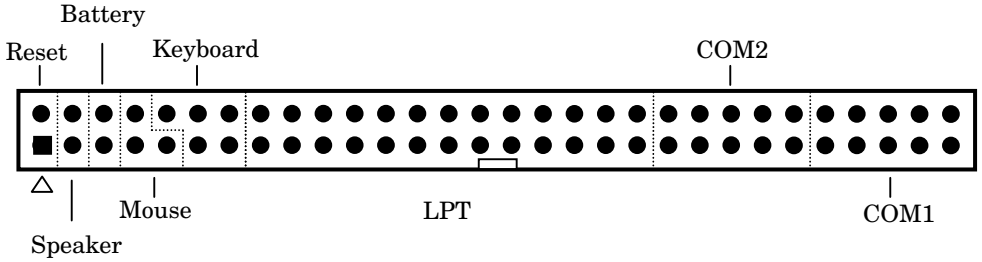
- 16550 compatible
- 16–byte FIFO buffers
- IEC 1000, level 3, ESD protection
  — Contact discharge ±4 kV
  — Air–gap discharge ±8 kV
- Backdrive protection
- Up to 115.2k Baud operation

Mating receptacle

Use an MPC-18 cable to connect the COM ports to external serial equipment. The P1 and P2 connectors of the MPC-18 are DB–9 female connectors that plug directly into a 9–pin PC serial cable.

Note See the Accessories appendix for mating information on the multipurpose connector.

Figure 5–1 Multipurpose connector, J3



Serial port configurations

The COM ports are defined in Table 5-1. Table 5–2 shows the jumper settings, and Table 5–3 shows the COM pinouts for J3.

Table 5-1 Serial port configurations

COM Port	Address	IRQ	Interface	Connector
COM1	3F8h*, 2F8h, 3E8h, 2E8h	IRQ4*, IRQ3	RS-232 – 8 wire RS-422 – 4 wire RS-485 – 4 wire	J3 – COM1
COM2	2F8h*, 3F8h, 3E8h, 2E8h	IRQ3*, IRQ4	RS-232 – 8 wire RS-422 – 4 wire RS-485 – 4 wire	J3 – COM2
* = default				

Table 5–2 COM1 and COM2 connector pinouts

COM1 J3			COM2 J3		
Pin	**RS–232 Signal**	**RS–422/485 Signal**	**Pin**	**RS–232 Signal**	**RS–422/485 Signal**
51	DCD	Tx+	41	DCD	Tx+
52	DSR	Tx–	42	DSR	Tx–
53	RxD		43	RxD	
54	RTS		44	RTS	
55	TxD		45	TxD	
56	CTS		46	CTS	
57	DTR	Rx+	47	DTR	Rx+
58	RI	Rx–	48	RI	Rx–
59	GND	GND	49	GND	GND
60	Nc		50	Nc	

Table 5–3 2050 COM port jumper settings: W2, W10, W11, W13

COM Port	Communication Type	Jumper Settings
COM1	RS–232C	W2[4–6], [10–12]* W11[1–2], [3–4], [5–6]* W13[1–2]*
	RS–422	W2[2–4], [8–10]** W11[1–2], [3–4], [5–6]* W13[1–3]
	RS–485	W2[4–6], [10–12]* W11[1–3], [7–9] W11[8–10]** W13[2–4]
COM2	RS–232C	W2[3–5], [9–11]* W10[1–2], [3–4], [5–6]* W13[7–8]*
	RS–422	W2[1–3], [7–9]** W10[1–2], [3–4], [5–6]* W13[7–9]
	RS–485	W2[3–5], [9–11]* W10[1–3], [7–9] W10[8–10]** W13[8–10]

• = default jumper installed
** These jumpers terminate the network. If the 2050 is not at an end of the network, leave these jumpers off.

Function and use of serial ports

COM1 as serial console device

You can use COM1 as a console device. See the *Console devices* chapter for more information.

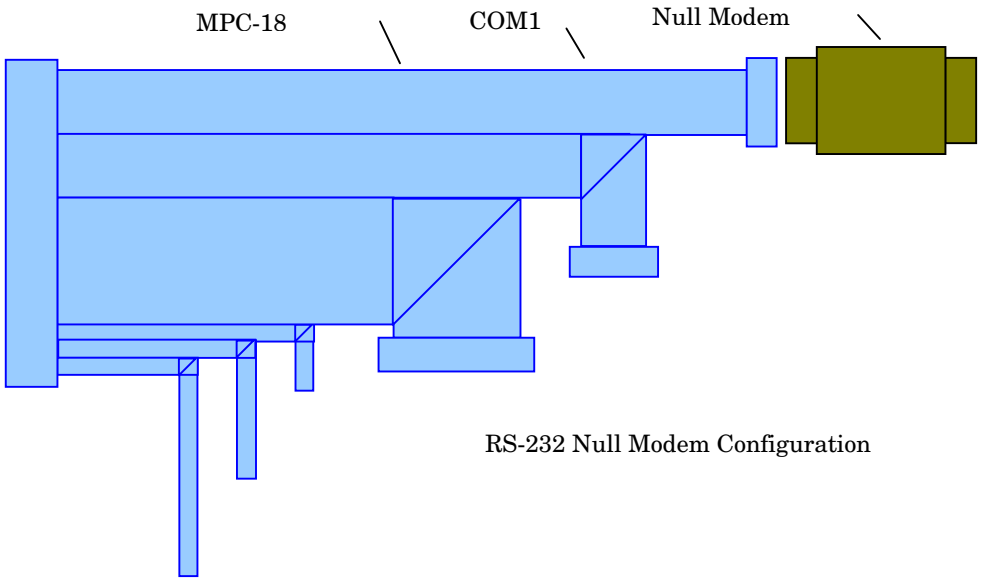
Mating receptacle

Use an MPC-18 cable to connect the COM ports to external serial equipment. The P1 and P2 connectors are DB–9 female connectors that plug directly into a 9–pin PC serial cable.

Note When interfacing the 2050 to your desktop PC, you must use a null modem adapter.

Note See the Accessories appendix for mating information on the multipurpose connector.

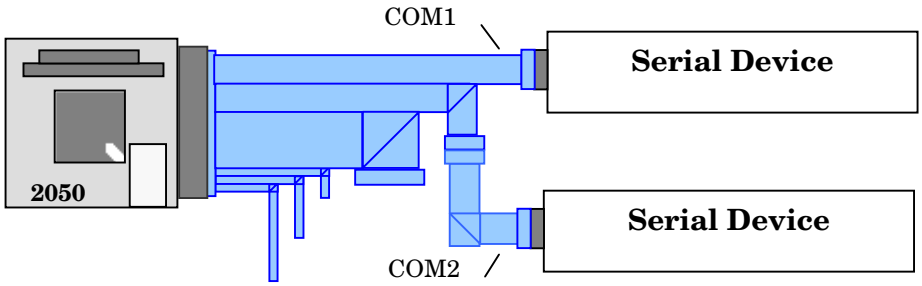
Figure 5–2 Null modem adapter and the MPC-18 cable



COM Ports as RS–232 I/O

COM1 and COM2 are 8–wire RS–232 interfaces.

Figure 5–3 2050 RS-232 serial devices



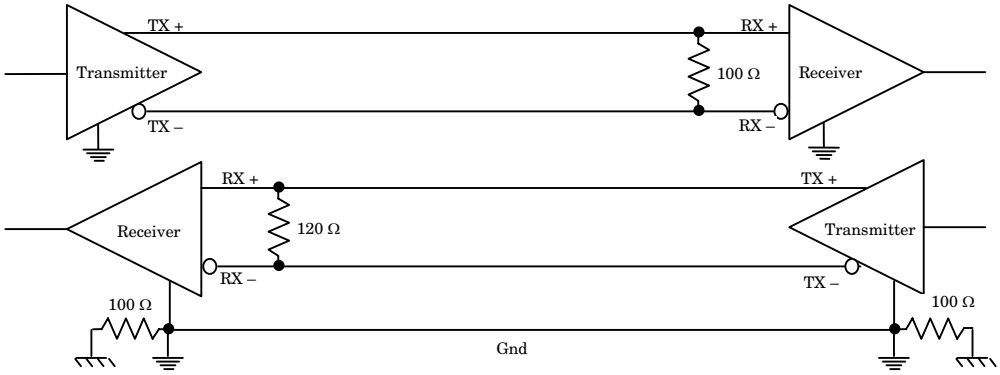
RS-422

Note RS-422 and RS-485 does not function correctly on revision 2 or lower 2050 cards. Contact Octagon Technical Support for additional information.

COM1 and COM2 can be used as RS-422 ports. RS–422 is typically a point-to-point configuration using differential signaling to communicate between the devices on a network. Differential signal reduces the effect of environmental noise, allowing communication over distances up to 1200 meters. The 2050 uses RTS internally to enable the transmit function.

RS-422 is also specified for multi-drop (party-line) applications where only one driver is connected to, and transmits on, a “bus” of up to 10 receivers. The device at the end of an RS–422 network must be terminated. The 2050 optionally terminates with a 120 ohm resistor. Refer to Table 5-4. Figure 5-5 shows a typical RS–422 four wire interface circuit. Figure 5-6 shows a typical connection.

Figure 5–4 Typical RS-422 four-wire interface circuit



RS-485

Note RS-422 and RS-485 does not function correctly on revision 2 or lower 2050 cards. Contact Octagon Technical Support for additional information.

An application may implement a node as either the “host” node or as a “remote” node in an RS–485 network. There can be as many as 32 nodes without any bus repeaters in the network. A host is referred to as the node that initiates communication; while a remote is referred to as a node that is addressed by the host.

In any given communication sequence in an RS–485 network, there can only be one host. The host is responsible for initiating communication, maintaining network registration, and providing housekeeping tasks with other nodes. Remotes, however, cannot initiate a communication. They can only respond to messages that are addressed to them from the host. The 2050 uses RTS internally to enable the transmit function.

The devices at each end of an RS–485 network must be terminated. Any node located between the end points should not be terminated. The 2050 optionally terminates with a 120 ohm resistor. Refer to Table 5-4. Figure 5–6 shows a typical RS–485 network.

Figure 5–5 Typical RS-485 half duplex interface circuit

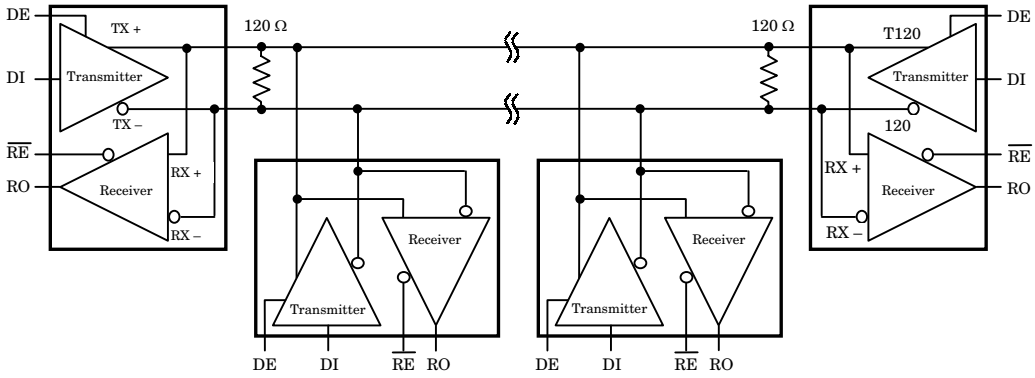
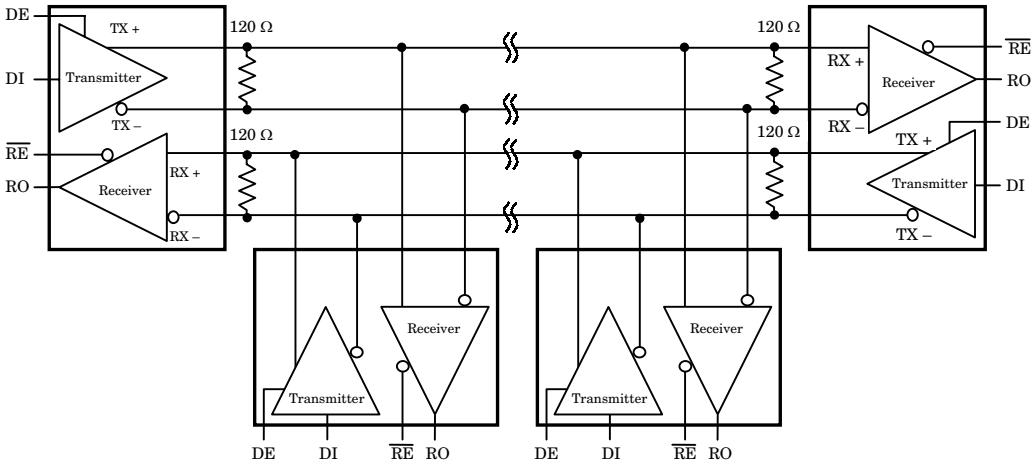


Figure 5–6 Typical RS-485 full duplex interface circuit



Chapter 6: LPT1 parallel port, LCD and Keypad

LPT1 parallel port

LPT1 is a multifunction parallel port, which is accessed through the multipurpose connector, J3 It supports the unidirectional standard mode, bidirectional mode, enhanced parallel port (EPP) mode, and extended capabilities port (ECP) mode. The default I/O address for LPT1 is 378h, with the default interrupt is IRQ7. You can choose the addresses 278h or 3BCh, or interrupt IRQ5, in the 2050 SETUP utility.

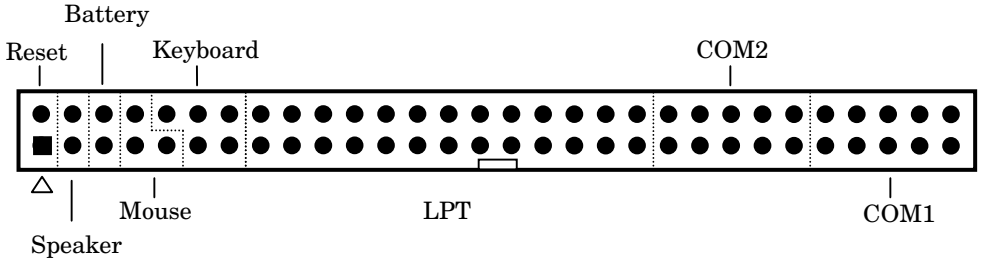
The LPT1 port supports a number of devices including a PC compatible printer, an LCD display, or a keypad.

Mating receptacle

Use an MPC-18 cable to connect the LPT port to external parallel equipment. The P3 connector is a DB–25 female connector which plugs directly into a 25–pin parallel cable.

Note See the *Accessories* appendix for mating information on the multipurpose connector.

Figure 6–1 Multipurpose connector, J3

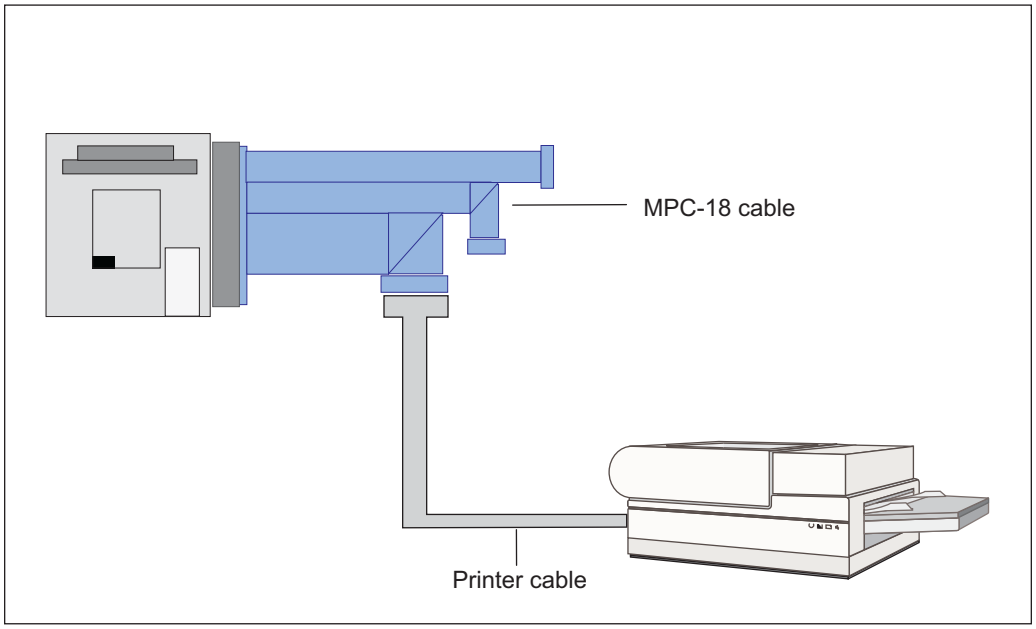


Printer

Installing a printer

1. Make sure that the LPT1 port is in standard or bidirectional mode. This is done in SETUP.
2. Connect the MPC-18 cable to the 2050 card.
3. Connect the DB-25 of the MPC-18 cable to the printer.

Figure 6–2 LPT1 as a printer port



Display

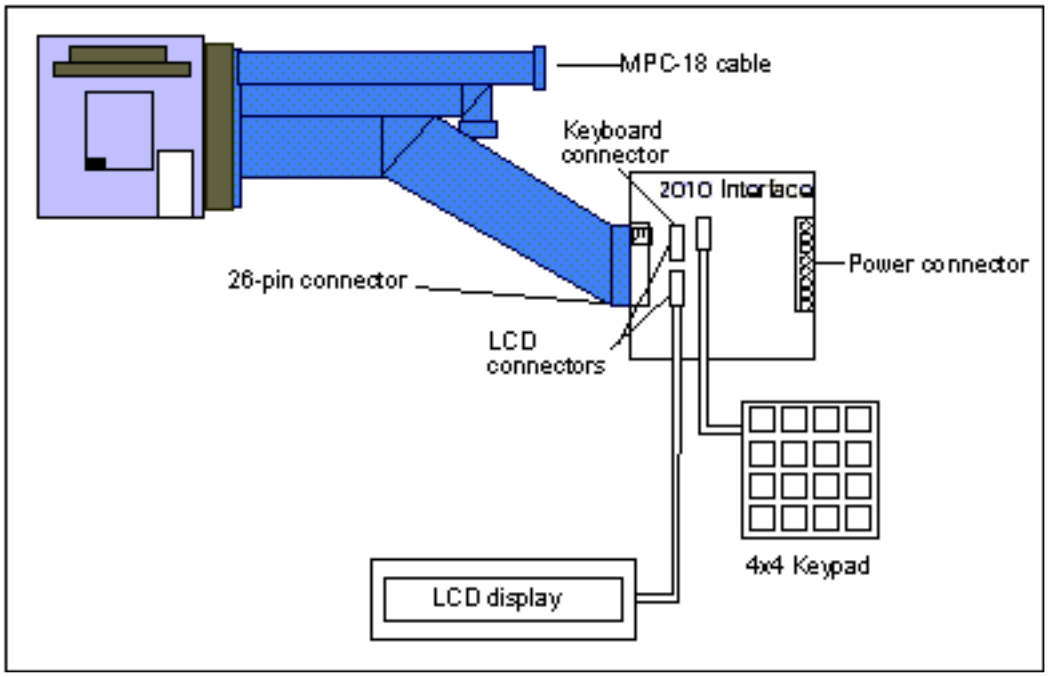
The LPT1 port supports either a 4 x 20 or a 4 x 40 liquid crystal display (LCD). To interface the displays to the 2050, an MCP-18 cable and an Octagon 2010 interface board are required.

The program DISPLAY.EXE (found on the CD ROM) provides an easy method to use the display. Refer to the file DISPLAY.DOC on the CD ROM for information on initializing and using the display. Also, refer to the *2010 product sheet* for more information on the interface board.

Installing a display

1. Connect the 26-pin connector on the MPC-18 to J3 on the 2010. See Figure 6–3.
2. Connect the display cable to either the 14–pin or 16–pin header on the 2010. The size of the display will determine which header to use.
4. Refer to the file DISPLAY.DOC for more information on initializing and using the display.

Figure 6–3 LPT1 as a display or keypad port



Keypad

LPT1 also supports 4 x 4 matrix keypads. To interface the keypad to the 2050, use the Octagon 2010 interface board. The program DISPLAY.EXE (found on the 2050 CD ROM) provides an easy method to use the keypad. Refer to the file DISPLAY.DOC on the 2050 CD ROM for information on initializing and using the keypad. Also, refer to the *2010 product sheet* for information on the interface board.

Installing a keypad

1. Connect the 26-pin connector on the MPC-18 to J3 on the 2010. See Figure 6–3.

2. Connect the keypad cable to the 10–pin header on the 2010.
3. Refer to the DISPLAY.DOC file for more information on reading the keypad.

Chapter 7: Console devices

Description

The 2050 has three options for console devices. You can use a PC/104 video card with a monitor and a keyboard as your console. You can use COM1 as the console, or you can run the system without a console device.

Selecting console devices

The following represent the options on the 2050 for console devices:

- A standard PC/104 video card, such as an Octagon 2430 SVGA card, and a keyboard.
- Serial console from COM1. A serial cable/null modem adapter plugged into a host PC running HyperTerminal provides both input and output. The local keyboard also allows input.
- No console device means no video output, either from a PC/104 card or the serial console. The local keyboard allows input.

If you are using a PC/104 video card, refer to the instructions included with the card for installation procedures.

Serial console

If the serial console is enabled COM1 is used as the console device.

Follow these steps to use the serial console:

1. For communication using HyperTerminal (or equivalent), the following settings must be used:
   Baud rate: 38400
   Communications parameters: no parity, 8 data bits, 1 stop bit
   Flow control: none
   Terminal support: ANSI
   ANSI terminal option- wrap lines that exceed terminal width: Yes
2. Copy the 2050 files from the supplied CD ROM to a subdirectory on your PC hard drive.

   ```
   C:
   ```

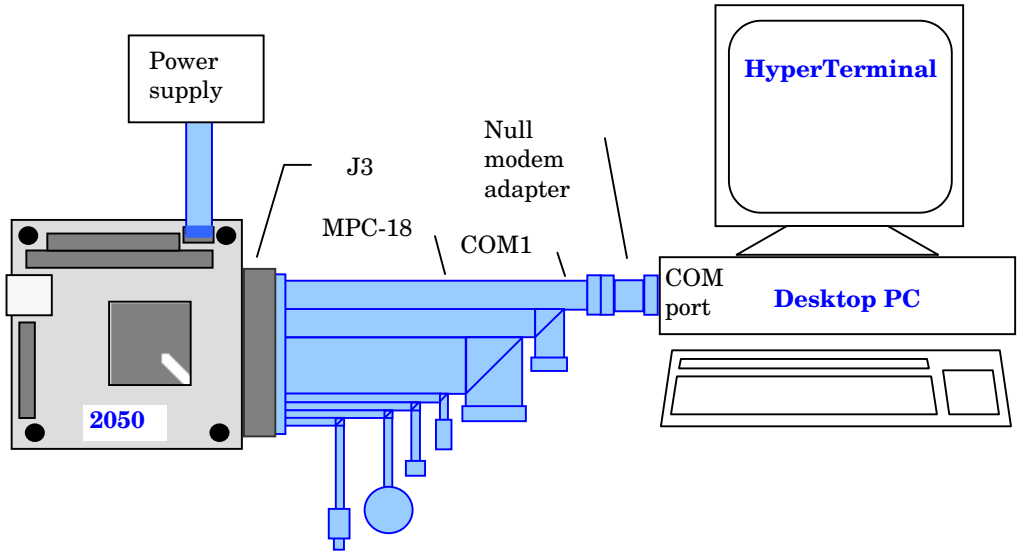
```
MD C:\MPC
XCOPY D:\2050\*.* C:\MPC /S
```

3. Start HyperTerminal. You are now ready to establish communications between your PC and the 2050.

4. Power on the 2050.

   If you do not get the proper logon message check the HyperTerminal serial parameters of your PC to make sure they are set correctly. Parameters should be 38400 baud, 8 data bits, no parity, and 1 stop bit

Figure 7–1 The 2050 and a serial console



Keyboard / Mouse

You can use any of the console options mentioned in the *Selecting console devices* section with a local keyboard. The MPC-18 cable has a keyboard and a mouse connector. The keyboard controller accepts an AT style keyboard and has a PS–2 type connector. The mouse port also has a PS–2 type connector.

You can also connect a serial mouse to one of the COM ports, and load a mouse driver. Neither the keyboard nor the mouse is required for operation.

Note See the *Accessories* appendix for mating information on the keyboard, speaker, and mouse connectors.

Chapter 8: SSD1, Z-tag interface, CompactFlash, SDRAM, and battery backup

Description

The 2050 is shipped with a 2 MB SMT Flash and 32 MB SDRAM.

SSD1

SSD1 is a 4 MB SMT flash soldered directly onto the PCB board. It contains the BIOS drive. SSD1 can be used as a hard drive.

Z-tag interface

The Z-tag interface allows you to reload the BIOS should it become corrupted. This procedure requires a Z-tag Dongle, available from ZF Microdevices. Refer to the ZFx86 Data book for information on how to use the Dongle to reprogram the BIOS.

CompactFlash

The CompactFlash appears to the system as an IDE device. It is automatically detected and configured as a hard drive during bootup. To configure the 2050 to boot from a CompactFlash, refer to the following section “Creating a Bootable CompactFlash.”

The CompactFlash can be configured as a master or slave using jumper W5. It can also be configured for 3V or 5V operation using jumper W12. Table 8–1 shows the jumper settings.

Note At this time, Octagon Systems only recommends SanDisk Industrial Grade CompactFlash. Currently SanDisk Compact flash comes in two technologies, NAND and NOR. NAND technology is best used in situations with sustained reads and writes. NOR technology is best used in situations with random reads and writes. Selecting the wrong CompactFlash may cause your application to run slow. For further information contact SanDisk at www.sandisk.com or your SanDisk representative.

Table 8–1 CompactFlash configuration jumpers

Configuration	Jumper
Master	W5[1–3] *
Slave	W5[1–2]
5V	W12[1–3]
3V	W12[2–4] *
* = default	

Creating a Bootable CompactFlash

A CompactFlash as shipped from the factory may or may not be formatted; even if formatted, it may or may not be bootable. The following sequence shows how to create a bootable CompactFlash, and how to configure the 2050 to boot from the CompactFlash.

CAUTION

The CompactFlash cannot be partitioned, formatted, or sys'ed from the SSD on the 2050. You must use an external drive such as a hard drive, floppy, or CD.

1. Create a bootable external device.

Note Octagon offers OS Embedders that include a CD boot disk for a variety of operating systems. Contact your Octagon representative for additional information.

2. Change the boot sequence in SETUP so the 2050 boots from the external drive first. Reboot from the external device.
3. Use FDISK to create partitions on the CompactFlash. Refer to your operating system manual for the appropriate parameters for using FDISK. You might also have to refresh the MBR (Master Boot Record). For ROMDOS, the command for refresh is `fdisk 80/r`.
4. Reboot, using the external device.
5. Format the CompactFlash.
6. Copy your operating system from the external device to the CompactFlash.
7. Change the boot sequence in SETUP so that the CompactFlash is first. Remove the external device and power off the 2050.
8. Ensure the CompactFlash is configured as a Master with jumper W5[1–3] (default).
9. Reboot.

SDRAM

The 2050 comes with 32 MB of surface mount SDRAM. In OEM quantities it can be ordered with 16 MB surface mounted SDRAM.

Battery backup for real time calendar clock

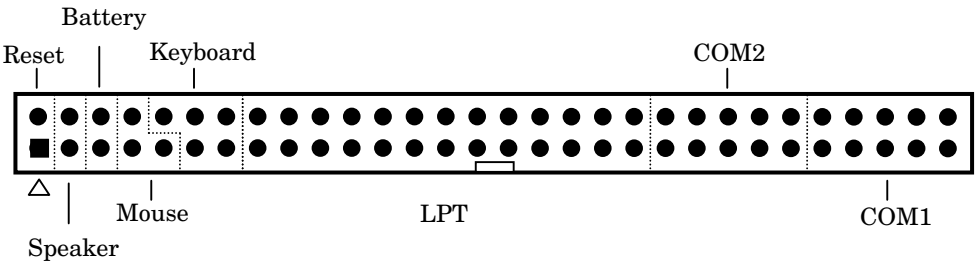
An AT battery can be installed to back up the CMOS real time clock.

Installing an AT battery

1. Power off the 2050.
6. Install the 3.6V AT clock battery at the J3 connector. Figure 8–3 shows the pinout of J3. There is a battery connector on the MPC-18 cable, which breaks out the J3 connector into its various functions.

Note See the *Accessories* appendix for mating information on the battery connector.

Figure 8–1 The J3 pinout



Chapter 9: External drives

Description

The 2050 is compatible with any standard IDE hard drive that has a 16–bit IDE interface. The BIOS extension ROM for the hard drive is supplied on the card so that no additional software is needed. The floppy drives use DMA channel 2.

Floppy disk controller

The 2050 supports a single .500" height FDD through a 26–pin ZIF connector.

Note See the *Accessories* appendix for mating information on the floppy disk connector.

Power requirements

The 2050 requires +5V for operation. Power to a single .500" height FDD is supplied from the 2050.

Installing a floppy disk drive

1. Disconnect power to the 2050.
2. Insert one end of your cable into the rear of the floppy drive. Make sure pin 1 on the cable is connected to pin 1 on the drive.

Note The floppy disk drive and the 2050 CPU control card both contain ZIF sockets for the FDD cable. To install a cable in either device, the tabs on each end of the connector must be pulled out, the cable installed, then the tabs pushed back in.

3. Insert the other end of the cable into J301 on the 2050.
4. Enter SETUP to set up the AT BIOS. You can execute this program either by pressing "F2" during system bootup. The system steps you through the configuration. Also, refer to the *SETUP programs* chapter for more information on the AT BIOS SETUP program.

Hard disk controller

The 2050 will interface to hard drives that have 16–bit IDE interfaces via a 44–pin connector at J2. The CompactFlash is seen by the system as an IDE hard drive, so one additional IDE drive can be installed.

Note See the *Accessories* appendix for mating information on the IDE hard drive connector.

Installing a hard drive:

1. Disconnect power to the 2050.
2. Insert one end of the Octagon hard drive adapter cable into the rear of the hard drive. Make sure pin 1 on the cable is connected to pin 1 on the drive.
3. Insert the other end of the IDE cable into J2 on the 2050. Make sure pin 1 on the cable is connected to pin 1 on the 2050.
4. Execute the BIOS SETUP program to configure your system for a hard drive. You can execute this program either by pressing “F2” during system bootup. The system steps you through the configuration. Also, refer to the *SETUP programs* chapter for more information on the BIOS SETUP program.
5. If you want to boot the system from the hard drive, you need to format the drive accordingly.

Chapter 10: Ethernet

Description

The 2050 provides a 10/100Base–T Ethernet port and supports the IEEE 802.3 Ethernet standard. The Ethernet controller IC chip provides the following:

- 8K x 16 SRAM buffer
- Integrated 10/100 Base–T transceiver interface
- Two LEDs for link and traffic status integrated into connector

The 2050 Ethernet uses twisted–pair wiring cable, which is built in a star configuration. The interface terminates at the standard, 8–position, RJ–45 latching phone jack that can be vertically accessed.

Caution

Use a strain relief loop when connecting to the 2050 Ethernet connector to avoid damaging the connector.

For more information on programming the Ethernet port, see the README.DOC in the Ethernet directory on the CD ROM. By default the Ethernet port connects to IRQ10, but can be reconfigured to IRQ11 via the SETIRQ.DOC program found on the 2050 CD ROM.

Table 11–1 Ethernet LEDs

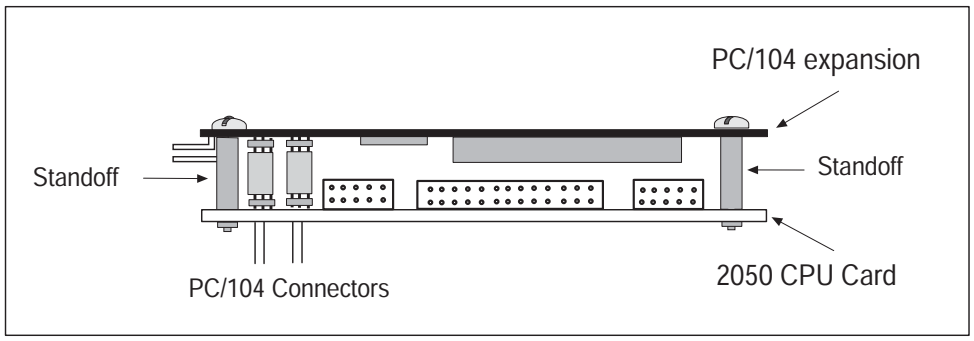
Color	CR7
Amber	Activity LED: Activated by access to I/O space
Green	Link LED: Activated by network link

Chapter 11: PC/104 expansion

Description

This connector allows you to interface to one or two PC/104 modules including A/D converters, digital I/O, serial ports, etc. The 2050 supports 8– and 16–bit cards**.** These modules can be stacked on top of the 2050 to form a highly integrated control system.

Figure 13–1 Typical PC/104 module stack



WARNING!

When installing any PC/104 module, avoid excessively flexing the 2050 card. Mate pins correctly and use the required mounting hardware.

Note See the *Accessories* appendix for mating information on the PC/104 connector.

Overview: *Section 3 – System management*

Section 3 provides information on managing the 2050 in the areas of internal control and troubleshooting. The following chapters are included:

Chapter 12:	Watchdog timer and hardware reset
Chapter 13:	Serial EEPROM
Chapter 14:	Temperature Sensor
Chapter 15:	Troubleshooting

Chapter 12: Watchdog timer and hardware reset

Description

The watchdog timer is a fail–safe against program crashes or processor lockups. It has a programmable timeout period, ranging from 0.5 seconds to 2 seconds. INT17 software calls, a built–in function on the 2050, are used to enable and set the timeout, extend the timeout, strobe, and disable the watchdog timer from your application. If the timer expires, it performs a hardware reset.

Watchdog function definitions using enhanced INT 17h handler

This section provides definitions for the watchdog functions using the INT17 handler (I17HNDLR.EXE). I17HNDLR.EXE is a TSR program. It is called out by the 2050 BIOS. Once executed it is active, but it must be executed each time the system is rebooted. If you use a different BIOS the INT17 functions can still be used by your application. Copy the utility to your hard drive and add it to your AUTOEXEC.BAT.

Note The INT17 functions can only be used with DOS operating systems.

Enable watchdog

```
Function:            fdh
Subfunction:         01h
Purpose:             To enable the watchdog.
Calling registers:   AH     fdh
                     AL     01h
                     BX     0-7 (0=1/2 sec., 1=1 sec., 2=2 sec.
                            3=4 sec. 4=8 sec. 5=16 sec.6=32 sec.
                            7=64 sec.)
                     DX     ffffh
Return registers:    None
Comments:            This function enables the watchdog.  Once
                     the watchdog is enabled, it has to be
                     strobed at a period of not less than the
                     time specified in the BX register or until
                     the watchdog is disabled.  Otherwise, a
                     system reset will occur.
```

Programming example:

```
/* Inline assembly code for Borland C++ 3.1 */
asm {
              mov           ax,0fd01h
              mov           bx,4
              mov           dx,0ffffh
              int           17h
              }
```

Strobe watchdog

Function:	fdh
Subfunction:	02h
Purpose:	To strobe the watchdog.
Calling registers:	AH fdh
	AL 02h
	DX ffffh
Return registers:	None
Comments:	This function strobes the watchdog. Once the watch dog is enabled, it has to be strobed at a period of not less than the time specified when enabled (Function 0fdh, Sub-function 1) or until the watchdog is disabled (Function 0fdh, Sub-function 3). Otherwise, a system reset will occur.

Programming example:

```
/* Inline assembly code for Borland C++ 3.1 */
asm {
mov           ax,0fd02h
mov           dx,0ffffh
int           17h
              }
```

The watchdog timer can also be strobed by reading address 20Ch. This may be faster than strobing the watchdog timer with an interrupts function call, for example:

```
A=INP(20Ch)
```

Disable watchdog

Function:	fdh
Subfunction:	03h
Purpose:	To disable the watchdog.
Calling registers:	AH fdh
	AL 03h
	DX ffffh
Return registers:	None
Comments:	This function disables the watchdog Once the watch dog is enabled, it has to be strobed at a period of not less than the time specified when enabled (see Function 0fdh, Sub-function 2) or until the watchdog is disabled. Otherwise, a system reset will occur.

Programming example:

```
/* Inline assembly code for Borland C++ 3.1 */
asm {
mov           ax,0fd03h
mov           dx,0ffffh
int           17h
    }
```

Hardware reset

The 2050 has a button which allows you to reset the system without turning off the power. This provides a more complete reset than the <CTRL><ALT><DEL> method. The J3 connector also has a reset function. By depressing the button (connecting the two lines), the circuit is pulled to ground and resets the system.

The RESET command accomplishes the same thing as the reset button. Refer to the component diagram in the *Quick start* chapter for the location of the reset button.

WARNING!

When using COM1 as the console, the <CTRL><ALT><DEL> commands on the host system keyboard only reset the host system. Use the RESET command to issue a hardware reset on the 2050.

Chapter 13: Serial EEPROM

Description

Up to 512 bytes of user–definable data can be saved in the serial EEPROM. The serial EEPROM does not require battery backup to maintain the data when the system power is off. The serial EEPROM is easily accessible via software interrupts by most programming languages.

The calendar/clock provides the user with 242 bytes of user–defined CMOS RAM. This RAM requires battery backup to maintain data. If a battery is not desirable, this data can be stored in serial EEPROM, written to CMOS RAM on power-up, changed and written back to serial EEPROM.

Enhanced INT 17h function definitions

This section provides definitions for both serial EEPROM and CMOS RAM functions. The serial EEPROM definitions include the following functions: Read a single word from serial EEPROM, Write a single word to serial EEPROM, Read multiple words from serial EEPROM, Write multiple words to serial EEPROM, and Return serial EEPROM size.

The CMOS RAM definitions include the following functions: Read extended CMOS RAM, Write extended CMOS RAM, Check CMOS battery, Copy contents of serial EEPROM to extended CMOS RAM, and Copy contents of extended CMOS RAM to serial EEPROM.

Serial EEPROM

Read a single word from the serial EEPROM

Function:	fch	
Subfunction:	00h	
Purpose:	To read a single word from the on–board serial EEPROM.	
Calling registers:	AH	fch
	AL	00h
	BX	Word address (zero based)
	DX	ffffh (User area relative address)
		9876h (Absolute address)
Return registers:	Carry flag cleared if successful	
	AX	Word read
	Carry flag set if error	
	AL	Error code

Error code	**Meaning**
ffh	Unknown error
01h	Function not implemented
02h	Defective serial EEPROM
03h	Illegal access

Comments:	This function reads a word from the user area of the serial EEPROM.

Programming example:

```
/* Read word 2 */
unsigned int seeData;
/* Inline assembly code for Borland C++ 3.1 */
asm {
mov           ax,0fc00h
mov           bx,02h         /* Read word 2 */
mov           dx,0ffffh
int           17h
mov           seeData,ax     /* store data in c environment */
  }
```

Write a single word to the serial EEPROM

Function:	fch	
Subfunction:	01h	
Purpose:	To write a single word to the on–board serial EEPROM.	
Calling registers:	AH	fch
	AL	01h
	BX	Word address (zero based)
	CX	Data word to write
	DX	ffffh (User area relative address)
		9876h (Absolute address)
Return registers:	Carry flag cleared if successful	
	Carry flag set if error	
	AL	Error code

	Error code	**Meaning**
	ffh	Unknown error
	01h	Function not implemented
	02h	Defective serial EEPROM
	03h	Illegal access
Comments:	This function writes a word to the user area of the serial EEPROM.	

Programming example:

```
/* Write 0x1234 to word 3*/
unsigned int seeData = 0x1234;
/* Inline assembly code for Borland C++ 3.1 */
asm {
              mov           ax,0fc01h
              mov           bx,03h        /* Write word 3 */
              mov           cx,seeData    /* Get write data
                                             from c environment */
              mov           dx,0ffffh
              int           17h
                }
```

Read multiple words from the serial EEPROM

Function:	fch	
Subfunction:	02h	
Purpose:	To read multiple words from the on–board serial EEPROM.	
Calling registers:	AH	fch
	AL	02h
	BX	Word address (zero based)
	CX	Word count
	DX	ffffh
	ES:DI	Destination pointer
Return registers:	Carry flag cleared if successful	
	AX	Word read
	Carry flag set if error	
	AL	Error code
	Error Code	**Meaning**
	ffh	Unknown error
	01h	Function not implemented
	02h	Defective serial EEPROM
	03h	Illegal access
Comments:	This function reads multiple words from the user area of the serial EEPROM.	

Programming example:

```
/* Read 10 words starting at word 5 */
unsigned int far *seeDataPtr = new unsigned int[10];
/* Allocate storage*/
/* Inline assembly code for Borland C++ 3.1 */
asm {
              mov           ax,0fc02h
              mov           bx,05h        /* Read starts at
                                             word 5 */
              mov           cx,10         /* Read 10 words */
              mov           dx,0ffffh
              les           di,seeDataPtr
```

```
int           17h
}
```

Write multiple words to the serial EEPROM

Function:	fch	
Subfunction:	03h	
Purpose:	To write multiple words to the on–board serial EEPROM.	
Calling registers:	AH	fch
	AL	03h
	BX	Word address (zero based)
	CX	Word count
	DX	ffffh
	DS:SI	Source pointer
Return registers:	Carry flag cleared if successful	
	Carry flag set if error	
	AL	Error code
	Error Code	**Meaning**
	ffh	Unknown error
	01h	Function not implemented
	02h	Defective serial EEPROM
	03h	Illegal access
Comments:	This function writes multiple words to the user area of the serial EEPROM.	

Programming example:

```
/* Write 8 words starting at word 6*/
unsigned int far *seeDataPtr = new unsigned int[8];
/* Allocate storage*/
              unsigned int far* tmpPtr = seeDataPtr;
              for(int I=0;I<8;I++)
              *seeDataPtr = I;             /* initialize data */
/* Inline assembly code for Borland C++ 3.1 */
              asm {
              push          ds
              mov           ax,0fc03h
              mov           bx,06h         /* Write starts at
                                           word 6 */
              mov           cx,8           /* Write 8 words */
              mov           dx,0ffffh
              lds           si,seeDataPtr
              int           17h
              pop           ds
                 }
```

Return serial EEPROM size

Function:	fch	
Subfunction:	04h	
Purpose:	To obtain the size of the on–board serial EEPROM.	
Calling registers:	AH	fch
	AL	04h
	DX	ffffh
Return registers:	Carry flag cleared if successful	
	AX	Size of the serial EEPROM (in words)

	BX	Size available to user (in words)
	Carry flag set if error	
	AL	Error code
Error code		**Meaning**
	ffh	Unknown error
	01h	Function not implemented
	02h	Defective serial EEPROM
	03h	Illegal access

Comments: This function returns the size (in words) of the serial EEPROM. Since the user cannot access all of the serial EEPROM, this function determines how much space is available to the user. This avoids the user from accessing unavailable addresses.

Programming example:

```
unsigned int seeUserSize;
/* Inline assembly code for Borland C++ 3.1 */
asm {
        mov     ax,0fc04h
        mov     dx,0ffffh
        int     17h
        mov     seeUserSize,bx
        }
```

Chapter 14: Temperature sensor and user jumpers

Description

The temperature sensor is a thermometer located on the board close to the CPU. The user can set an upper and lower temperature limit. If the temperature measured exceeds the upper limit the OS bit becomes active, indicating an over-temperature condition. The OS bit remains active until the measured temperature falls below the lower limit. By reading the OS bit to see if it is set or not, an external device such as a fan can be activated or deactivated.

The user jumpers are W6[5–6] and W6[7–8]. The INT17 functions provide an easy method to implement software routines according to whether or not a jumper has been installed.

Temperature sensor INT17h function definitions

The temperature sensor allows you to set over and under temperature limits, and to read the ambient temperature. There is also a configuration register. For complete details on the configuration register, refer to the Dallas Semiconductor DS1775 SOT23 Digital Thermometer and Thermostat Product Review sheet.

Digital data is written to/read from the component with MSb first, in two's complement format, with the MSb denoting positive or negative temperature. All temperatures are in Celsius.

Write TEMP SENSOR register pointer

Function:	0edh	
Subfunction:	00h	
Purpose:	To set the TEMP SENSOR pointer to internal registers	
Calling Registers:	AH	0edh
	AL	00h
	BH	Number of bits in internal register 8 or 16. Only the configuration register is 8 bit, all others are 16 bit.

```
                    BL      TEMP SENSOR register pointed to:
                            0=TEMPERATURE
                            1=CONFIGURATION
                            2=HYST over temp
                            3=OS under temp
                            4=0x0ff, reserved
                    DX      ffffh

Return Registers:           Carry flag cleared if successful

                            Carry flag set if error
                    AL      Error code

Comments:           This function shall be used to set the TEMP
                    SENSOR internal register pointer.

Programming example 1:      /* Inline assembly code for Borland
                    C++ 3.1 */
                    unsigned char aData;
                    asm {
                            mov             ax,0ed00h
                            mov             bx,0801h
                            mov             dx,ffffh
                            int             17h
                            mov             aData,al
                        }
```

Read TEMP SENSOR current register

```
Function:           0edh
Subfunction:        01h
Purpose:            Reads the register currently pointed to by
                    the TEMP SENSOR register pointer.
Calling Registers:  AH      0edh
                    AL      01h
                    DX      ffffh

Return Registers:   Carry flag cleared if successful
                    Carry flag set if error
                    AL      Error code
                    BX      Data read from the TEMPERATURE SENSOR
Comments:           This function shall be used to read the
                    TEMP SENSOR register currently pointed to.

Programming example 1:      /* Inline assembly code for Borland
                    C++ 3.1 */
                    unsigned char aData;
                    unsigned int bData;
                    asm {
                            mov             ax,0ed01h
                            mov             dx,ffffh
                            int             17h
                            mov             aData,al
                            mov             bData,bx
```

```
}
```

Write TEMP SENSOR current register

```
Function:            0edh
Subfunction:         02h
Purpose:             Writes the register currently pointed to by
                     the TEMP SENSOR register pointer.
Calling Registers:   AH     0edh
                     AL     02h
                     BX     Data to write
                     DX     ffffh
Return Registers:    Carry flag cleared if successful
                     Carry flag set if error
                     AL     Error code
Comments:            This function shall be used to write the
                     TEMP SENSOR register currently pointed to.
Programming example 1:      /* Inline assembly code for Borland
                     C++ 3.1 */
                     unsigned char aData;
                     unsigned int bData;
                     asm {
                           mov             ax,0ed01h
                           mov             bx,bData
                           mov             dx,ffffh
                           int             17h
                           mov             aData,al
                         }
```

Read TEMP SENSOR Int Status bit

```
Function:            0edh
Subfunction:         03
Purpose:             Reads the TEMP SENSOR INT STATUS bit set
                     when TEMPERAUTRE is greater than THYS value
                     or TEMPERATURE is less than OS value.

Calling Registers:   AH     0edh
                     AL     03h
                     DX     ffffh

Return Registers:    Carry flag cleared if successful
                     AL     State of int bit
Carry flag set if error
                     AL     Error code

Comments:            This function shall be used to write the
TEMP SENSOR register currently pointed to.

Programming example 1:      /* Inline assembly code for Borland
C++ 3.1 */
                     unsigned char aData;
                     asm {
```

```
                mov             ax,0ed01h
                mov             bx,bData
                mov             dx,ffffh
                int             17h
                mov             aData,al
                 }
```

User jumpers

The user jumpers are W6[5–6] and W6[7–8]. The INT17 functions provide an easy method to implement software routines according to whether or not a jumper has been installed.

Read user jumper

```
Function:           0fbh
Sub-Function:       0bh
Purpose:            To read user jumpers

Calling Registers:  AH     0fbh
                    AL     0bh
                    DX     0ffffh

Return Registers:   Carry flag cleared if successful
                    AL     Jumper data
                           bit 0 user jumper A. 1=on, 0=off
                           bit 1 user jumper B. 1=on, 0=off
                    Carry flag set if error
                    AL     Error code

Comments:           This function shall be used to read the user
jumper


Programming example 1:     /* Inline assembly code for Borland
C++ 3.1 */
                    unsigned char aData;
                    asm {
                           MOV             AX, 0fb0bh
                           MOV             DX, 0ffffh
                           INT             17h
                           MOV             aData, AL
                           }
                    if (aData & 1)
                      printf("U1 jumper is ON\n");
                    else
                      printf("U1 jumper is OFF\n");
```

Chapter 15: Troubleshooting

If your system is not working properly, check the following items.

No screen activity–checking console serial communications

If you do not get the sign–on message after bootup, check the following:

- Make sure all PC/104 expansion cards are removed from the 2050. This ensures that other cards are not interacting with the 2050.
- Remove the jumper from the "S" position at W6[1–2].
- The MPC-18 cable turns the 2050 J3 port into a 9–pin AT serial port. Make sure a null modem adapter is installed on the other end, and that the assembly is inserted into the proper serial port on the PC. Make sure the COM1 connector on the MPC-18 is used.
- Make sure your power module provides +5V (+/–0.25V) and at least 2.5A of current.
- After verifying the above conditions, you can monitor voltage levels by connecting an oscilloscope between the TxD* line on COM1 and ground. After power-up, you should see a burst of activity on the oscilloscope screen. The voltage level should switch between +/–8V.

Garbled console screen activity

If you do get activity on your console screen but the message is garbled, check the following:

- Remove the jumper from the "S" position at W6[1–2] to ensure the default settings for COM1.
- If you are using HyperTerminal, make sure you have configured the software for 38400 baud and have selected the correct serial port for communicating with your PC. Refer to the HyperTerminal manual for information on selecting the baud rate.
- If you are using communications software other than HyperTerminal, Octagon cannot guarantee the operation. Make sure that the software parameters are set to match the default settings of the 2050. These settings are:
  1) Baud rate: 38400

2) Communications parameters: no parity, 8 data bits, 1 stop bit
3) Flow control: none
4) Terminal support: ANSI
5) ANSI terminal option- wrap lines that exceed terminal width: Yes.

System generates a BIOS message but locks up when booting from SSD1

- Remove the jumper from the "S" position at W6[1–2] and reboot.
- Display the directory of SSD1 and verify that all the necessary boot files exist. Copy any missing files to SSD1.
- If no files are missing, remake SSD1 to overwrite any files which may have become corrupted. In addition, you may want to do a **FORMAT** and **SYS** to SSD1.

Note FORMAT requires a floppy disk drive to restore system files.

System will not boot from CompactFlash

Many CompactFlashes as shipped from the factory are not bootable devices. Refer to the procedure on page 60 to make your CompactFlash bootable.

System locks up on power-up; may or may not respond to reset switch

A common cause is using a non–Octagon power supply such as a PC desktop supply. Most of these PC supplies are rated at 5V at 20A or more. Switching supplies usually requires a 20% load to operate properly, that is, 4A or more. Since a typical Micro PC system takes less than 2A, the supply does not regulate properly. Output drift up to 6–7V and/or 7–8 voltage spikes have been reported. If the power supply comes up slowly (that is, longer than 10 ms), the sequencing of ICs on the board may be out of sync, thus, causing the system to lock up.

Octagon supplies are designed to ramp up fast, discharge fast on power-down and to regulate properly under a no load condition.

System locks up after power-down/power-up

If the power supply does not drain below 0.7V, the CMOS components on the card will act like diodes and forward bias. This is typically caused by using power supplies that have large output capacitors.

Either use a different power supply that discharges faster, leave the power off until the supply has adequate time to discharge or place a 100 ohm, large wattage resistor across the output capacitor.

Octagon supplies are designed to ramp up fast, discharge fast on power-down and to regulate properly under a no load condition.

LED signaling of “beep” codes

Description

The 2050 has a bicolor LED that is used by the BIOS to signal system status.

Immediately after the 2050 powers on, both LEDs are lit and display an amber color. Upon completion of the boot sequence, the amber LED turns off and the green LED remains on.

If a failure occurs during the boot sequence, visual beep codes are displayed to the LEDs. The visual beep codes are defined in the following table.

The bicolor LED also indicates memory suspend status. Upon entering memory suspension, the green LED turns off and the amber LED turns on. On a resume condition, the amber LED turns off and the green LED turns on.

When the system enters the cool down clocking state, the green LED and the amber LED turn on. When the system exits the cool down clocking state, the amber LED turns off and the green LED remains on.

Table 14–1 BIOS beep codes

Port 80 Code	Beep Sequence	POST Routine Description
02h		Verify Real Mode
03h		Disable Non-Maskable Interrupt (NMI)
04h		Get CPU type
06h		Initialize system hardware
07h		Disable shadow and execute code from the ROM
08h		Initialize chipset with initial POST values
09h		Set IN POST flag
0Ah		Initialize CPU registers
0Bh		Enable CPU cache
0Ch		Initialize caches to initial POST values
Oeh		Initialize I/O component
0Fh		Initialize the local bus IDE
10h		Initialize Power Management
11h		Load alternate registers with initial POST values
12h		Restore CPU control word during warm boot
13h		Initialize PCI Bus Mastering devices
14h		Initialize keyboard controller
16h	1-2-2-3	BIOS ROM checksum
17h		Initialize cache before memory Auto size
18h		8254 timer initialization
1Ah		8237 DMA controller initialization
1Ch		Reset Programmable Interrupt Controller
20h	1-3-1-1	Test DRAM refresh
22h	1-3-1-3	Test 8742 Keyboard Controller
24h		Set ES segment register to 4 GB
28h		Auto size DRAM
29h		Initialize POST memory manager
2Ah		Clear 512 KB base RAM
2Ch	1-3-4-1	ROM failure on address line xxxx
2Eh	1-3-4-3	RAM failure on data bits xxxx of low byte of memory bus
2Fh		Enable cache before system BIOS shadow
32h		Test CPU bus-clock frequency
33h		Initialize Phoenix Dispatch Manager
36h		Warm start shutdown
38h		Shadow system BIOS ROM
3Ah		Auto size cache
3Ch		Advanced configuration of chipset registers
3Dh		Load alternate registers with CMOS values
41h		Initialize extended memory for ROMPilot
42h		Initialize interrupt vectors
45h		POST device initialization
46h	2-1-2-3	Check ROM copyright notice
47h		Initialize I20 support
48h		Check video configuration against CMOS
49h		Initialize PCI bus and devices
4Ah		Initialize all video adapters in system
4Bh		QuietBoot start (optional)
4Ch		Shadow video BIOS ROM
4Eh		Display BIOS copyright notice
4Fh		Initialize MultiBoot
50h		Display CPU type and speed
51h		Initialize EISA board
52h		Test keyboard
54h		Set key click if enabled
55h		Enable USB devices

58h	2-2-3-1	Test for unexpected interrupts
59h		Initialize POST display service
5Ah		Display prompt “Press F2 to enter Setup”
5Bh		Disable CPU cache
5Ch		Test RAM between 512 and 640 KB
60h		Test extended memory
62h		Test extended memory address lines
64h		Jump to UserPatch1
66h		Configure advanced cache registers
67h		Initialize Multi Processor APIC
68h		Enable external and CPU caches
69h		Setup System Management Made (SMM) area
6Ah		Display external L2 cache size
6Bh		Load custom defaults (optional)
6Ch		Display shadow-area message
6Eh		Display possible high address for UMB recovery
70h		Display error messages
72h		Check for configuration errors
76h		Check for keyboard errors
7Ch		Set up hardware interrupt vectors
7Dh		Initialize Intelligent System Monitoring
7Eh		Initialize coprocessor if present
80h		Disable onboard Super I/O ports and IRQs
81h		Late POST device initialization
82h		Detect and install external RS232 ports
83h		Configure non-MCD IDE controllers
84h		Detect and install external parallel ports
85h		Initialize PC-compatible PnP ISA devices
86h		Re-initialize onboard I/O ports
87h		Configure Motherboard Configurable Devices (optional)
88h		Initialize BIOS Data Area
89h		Enable Non-Maskable Interrupts (NMIs)
8Ah		Initialize Extended BIOS Data Area
8Bh		Test and initialize PS/2 mouse
8Ch		Initialize floppy controller
8Fh		Determine number of ATA drives (optional)
90h		Initialize hard-disk controllers
91h		Initialize local-bus hard-disk controllers
92h		Jump to UserPatch2
93h		Build MPTABLE for multi-processor boards
95h		Install CD ROM for boot
96h		Clear huge ES segment register
97h		Fix up Multi Processor table
98h	1-2	Search for option ROMs. One long, two short beeps on checksum failure.
99h		Check for SMART drive (optional)
9Ah		Shadow options ROMs
9Ch		Set up Power Management
9Dh		Initialize security engines (optional)
9Eh		Enable hardware interrupts
9Fh		Determine number of ATA and SCSI drives
A0h		Set time of day
A2h		Check key lock
A4h		Initialize typematic rate
A8h		Erase F2 prompt
Aah		Scan for F2 keystroke
Ach		Enter SETUP
Aeh		Clear Boot flag
B0h		Check for errors

B1h		Inform RomPilot about the end of POST
B2h		POST done – prepare to boot operating system
B4h	1	One short beep before boot
B5h		Terminate QuietBoot (optional)
B6h		Check password (optional)
B7h		Initialize ACPI BIOS
B9h		Prepare Boot
Bah		Initialize SMBIOS
BBh		Initialize PnP Option ROMs
BCh		Clear parity checkers
BDh		Display MultiBoot menu
Beh		Clear screen (optional)
BFh		Check virus and backup reminders
C0h		Try to boot with Int 19h
C1h		Initialize POST Error Manager (PEM)
C2h		Initialize error logging
C3h		Initialize error display function
C4h		Initialize system error handler
C5h		PnPnd dual CMOS (optional)
C6h		Initialize note dock (optional)
C7h		Initialize note dock late
C8h		Force check (optional)
C9h		Extended checksum (optional)
Cah		Redirect Int15h to enable remote keyboard
CBh		Redirect Int 13h to Memory Technologies Devices such as ROM, RAM, PCMCIA, and serial disk
CCh		Redirect Int 10h to enable remote serial video
CDh		Re-map I/O and memory for PCMCIA
Ceh		Initialize digitizer and display message
D2h		Unknown interrupt
		The following are for boot block in Flash ROM
E0h		Initialize the chipset
E1h		Initialize the bridge
E2h		Initialize the CPU
E3h		Initialize system timer
E4h		Initialize system I/O
E5h		Check force recovery boot
E6h		Checksum BIOS ROM
E7h		Go to BIOS
E8h		Set Huge Segment
E9h		Initialize Multi Processor
Eah		Initialize OEM special code
Ebh		Initialize PIC and DMA
Ech		Initialize Memory type
Edh		Initialize Memory size
Eeh		Shadow Boot Block
Efh		System memory test
F0h		Initialize interrupt vectors
F1h		Initialize Run Time Clock
F2h		Initialize video
F3h		Initialize System Management Manager
F4h		Output one beep
F5h		Clear Huge Segment
F6h		Boot to Mini DOS
F7h		Boot to full DOS

Technical assistance

Carefully recheck your system before calling Technical Support. Run as many tests as possible; the more information you can provide, the easier it will be for Technical Support staff to help you solve the problem. For additional technical assistance, try the following:

Technical Support telephone: 303-426-4521

E-mail Technical Support: Support@octagonsystems.com

Applications Notes (via web): http://www.octagonsystems.com/
Contact Us/Application Notes/application notes.html

FAQ (via web): http://www.octagonsystems.com/
Contact Us/FAQ/faq.html

Overview: *Section 4 – Appendices*

Section 4 contains a series of appendices which provides additional information about the 2050.

Appendix A:	Technical data
Appendix B:	Software utilities
Appendix C:	Accessories

Appendix A: 2050 technical data

Technical specifications

CPU

586 128 MHz

SYSCLK

33, 50, 64, 66, 100, or 128 MHz (jumper selectable)

BIOS

AT compatible with industrial extensions

SDRAM

32MB SMT SDRAM supplied
OEM option – 16MB SDRAM.

Solid–state disk 1

2 MB flash supplied
OEM option – 4 or 8 MB

Floppy drive

Floppy drive support via 26–pin ZIF connector and on-card floppy drive controller.

Hard drive

Hard drive BIOS supported using on-card hard drive controller which allows extended IDE drives.

DOS

ROM-DOS 7.1 included

Other operating system

Compatible with Windows NT, Windows 98, Windows CE, Linux, QNX, and DOS

Serial I/O

COM1 and COM2 are 16C550 compatible

Parallel port

LPT1 is PC compatible with multifunctional capability

Watchdog timer

Time–out is from 0.5 seconds to 2 seconds, software enabled and strobed. Software extensions increase timeout. Disabled on power-up and reset. Controls are through built–in, enhanced INT 17h function calls.

Bus mastering

Bus mastering is not supported

Power requirements

5V ±0.25V @ 1.0 Amp. maximum

Full 128 MHz operation:	800 mA typical
Suspend:	167 mA typical

Environmental specifications

–40° to 85°C, operating, 99MHz and below
–40° to 70°C, operating, 128MHz
–55° to 90° C, nonoperating
RH 5% to 95%, noncondensing

Size

PCB size 3.55” x 3.775” x 0.68”. See page 21 for size with connectors.

Mating connectors

J302 – PC/104 interface, PC/104 8/16–bit receptacle:	
For 8–bit:	Samtec #ESQ-132-14-G-D
For 16–bit:	Samtec #ESQ-120-14-G-D
J3 – 60-pin connector:	Digi-Key CKR60G-ND, or equivalent
J2 – IDE hard drive port:	Molex 87332-4420
J6 – power:	Molex, 22-01-3047, female housing 4 position
	Molex, 08-52-0123, crimp to wire terminal

Maps

Table A–1 2050 DMA map

Channel	Description
Channel 0	Reserved for bus memory refresh
Channel 1	Reserved for ECP Parallel Port
Channel 2	Floppy Drive Interface
Channel 3	IDE Interface
Channel 4	Slave
Channel 5	Available
Channel 6	Not available
Channel 7	Not available

Table A–2 2050 I/O map

Hex range	Function
000h to 0FFh	System I/O Functions
100h to 1Efh	Off Card I/O Space
1F0h to 1F7h	IDE Controller
218h to 21Ch	CPU Functions
278h to 27Bh	Bi-directional Parallel Port (LPT1)
2F8h to 2FFh	COM2
378h to 37Bh	LPT1
3F0h to 3F7h	Floppy Controller
3F8h to 3FFh	COM1
1000h to 1020h	Ethernet
8100h to 8124h	GPIO/LEDs

Table A–3 2050 interrupt map

IRQ	Default Device	Alternate
IRQ0	System Timer	
IRQ1	Keyboard	
IRQ2	Cascade to IRQ9	
IRQ3	COM2	PC/104
IRQ4	COM1	PC/104
IRQ5	Available	PC/104
IRQ6	Floppy	PC/104
IRQ7	LPT1	PC/104
IRQ8	RTC Alarm	
IRQ9	Cascade from IRQ2	PC/104
IRQ10	Ethernet	PC/104
IRQ11	Available	PC/104
IRQ12	Mouse	PC/104
IRQ13	Reserved for FPU	
IRQ14	Primary IDE Drive	PC/104
IRQ15	Available	PC/104

Table A–4 2050 memory map

Address	Size	Description	Shadowing
00000H to 9FFFFH	640KB	System SDRAM Memory	
A0000H to BFFFFH	128KB	Off Card Video Memory	
C0000H to C7FFFH	32KB	Off Card ISA Video BIOS	Enabled
C8000H to CDFFFH	24KB	Off Card Memory	Disabled
CE000H to CFFFFH	8KB	SSD1 window when X jumper installed	Disabled
D0000H to D7FFFH	32KB	Off Card Memory	Setup Option
D8000H to DFFFFH	32KB	Extended BIOS Area (enabled when “X” jumper is present)	Always on when “X” is present
E0000H to FFFFFH	128KB	Primary BIOS Area (Phoenix)	Enabled
10000H to FFFFFFFFH	4GB	System Extended SDRAM Memory	

Jumper settings

Table A–5 2050 COM port jumper settings: W2, W10, W11, W13

COM Port	Communication Type	Jumper Settings
COM1	RS–232C	W2[4–6], [10–12]* W11[1–2], [3–4], [5–6]* W13[1–2]*
	RS–422	W2[2–4], [8–10]** W11[1–2], [3–4], [5–6]* W13[1–3]
	RS–485	W2[4–6], [10–12]* W11[1–3], [7–9] W11[8–10]** W13[2–4]
COM2	RS–232C	W2[3–5], [9–11]* W10[1–2], [3–4], [5–6]* W13[7–8]*
	RS–422	W2[1–3], [7–9]** W10[1–2], [3–4], [5–6]* W13[7–9]
	RS–485	W2[3–5], [9–11]* W10[1–3], [7–9] W10[8–10]** W13[8–10]

• = default jumper installed
** These jumpers terminate the network. If the 2050 is not at an end of the network, leave these jumpers off.

Table A–6 2050 SYSCLK frequency select: W3

SYSCLK Frequency	**Jumper Settings**
33 MHz	No jumpers
50 MHz	W3[1–2], W3[3–4]
64 MHz	W3[1–2] *
66MHz	W3[3–4] **
*= default jumper installed ** NOTE: Ensure that W6[9–10] and W6[11–12] are properly jumpered, to avoid exceeding system capability. Default is 2x SYSCLK frequency; 2x of 66 MHz exceeds CPU capability. Also, note that if the system is configured to exceed 100 MHz, jumper W7 must be changed.	

Table A–7 2050 Flash configuration jumper settings: W5 and W12

Configuration	**Jumper**
CompactFlash is Master	W5[1–3]*
CompactFlash is Slave	W5[1–2]
5V (CompactFlash voltage)	W12[1–3]
3V (CompactFlash voltage)	W12[2–4] *

Table A–8 2050 jumper settings: W6 system configuration

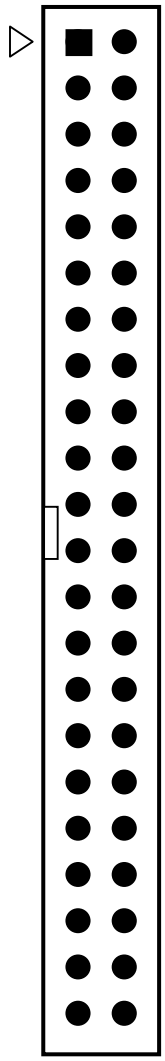
Jumper position	**Pins**	**Description**
"S"	W6[1–2]*	USESETUP
"X"	W6[3–4]*	BIOS extension enable
"U1"	W6[5–6]*	User jumper A
"U2"	W6[7–8]*	User jumper B
CPU Multiplier	W6[9–10]	Installed – SYSCLK speed x1. *Removed – SYSCLK speed x2.
SYSCLK Enable	W6[11–12]*	MUST REMAIN INSTALLED.
External BUR	W6[13–14]	ON – Use external Boot Update ROM OFF – Use onboard BIOS
ZTAG Enable	W6[15–16]	ON – ZTAG enable Off – Normal boot
* = default jumper installed		

Table A–9 2050 voltage select jumper settings: W7

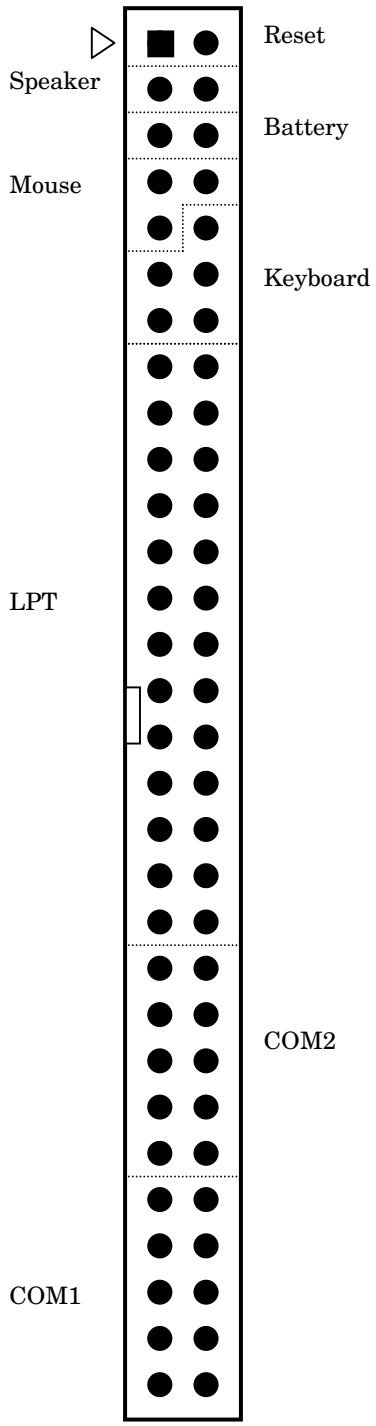
Jumper	**Voltage**
W7[1–2]	2.25V (<100 MHz CPU)
W7[3–4] *	2.7V (>100 MHz CPU)
*= default jumper installed	

Connector pinouts

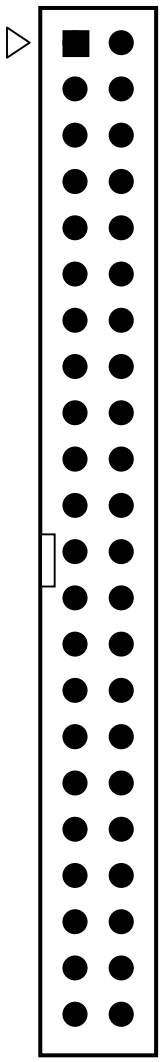
The following tables show the pinouts for the connectors on the 2050 CPU control card.

EIDE connector – J2		
Pin#	**Signal**	**Connector diagram**
1	RST*	
2	GND	
3	DD7	
4	DD8	
5	DD6	
6	DD9	
7	DD5	
8	DD10	
9	DD4	
10	DD11	
11	DD3	
12	DD12	
13	DD2	
14	DD13	
15	DD1	
16	DD14	
17	DD0	
18	DD15	
19	GND	
20	NC – keypin	
21	DMARQ	
22	GND	
23	DIOW*	
24	GND	
25	DIOR*	
26	GND	
27	IORDY	
28	NC	
29	DMACK*	
30	GND	
31	INTRQ	
32	NC	
33	DA1	
34	PDIAG*	
35	DA0	
36	DA2	
37	HDCS0*	
38	HDCS1*	
39	ACTLED	
40	GND	
41	+5V	
42	+5V	
43	GND	
44	+5V	

Multi-purpose connector – J3						
Pin#	**Signal**		**Pin#**	**Signal**		**Function**
1	Reset		2	GND		Reset
3	Speaker		4	+5V		Speaker
5	BATT+		6	BATT –		Battery
7	MCLK		8	GND		Mouse
9	MDATA					
			10	GND		Keyboard
11	Keyboard clock		12	GND		
13	Keyboard data		14	+5V		
15	OSTB*		16	AFD*		LPT
17	PD0		18	ERR*		
19	PD 1		20	INIT*		
21	PD2		22	SLIN*		
23	PD3		24	GND		
25	PD4		26	GND		
27	PD5		28	GND		
29	PD6		30	GND		
31	PD7		32	GND		
33	ACK*		34	GND		
35	BUSY		36	GND		
37	PE		38	GND		
39	SLCT		40	+5V		
41	DCD2*	RS-422 TX+	42	DSR2*	RS-422 TX–	COM2
43	RXD2		44	RTS2*		
45	TXD2		46	CTS2*		
47	DTR2*	RS-422 RX+	48	RI2*	RS-422 RX–	
49	GND	RS-422 GND	50	NC		
51	DCD1*	RS-422 TX+	52	DSR1*	RS-422 TX–	COM1
53	RXD1		54	RTS1*		
55	TXD1		56	CTS1*		
57	DTR1*	RS-422 RX+	58	RI1*	RS-422 RX–	
59	GND	RS-422 GND	60	NC		



Power connector – J6		
Pin#	**Signal**	**Connector diagram**
1	+5V	
2	GND	
3	+12V	
4	– 12V	

CompactFlash connector – J300		
Pin#	**Signal**	**Connector diagram**
1	GND	
2	DD3	
3	DD4	
4	DD5	
5	DD6	
6	DD7	
7	HDCS0*	
8	GND	
9	GND	
10	GND	
11	GND	
12	GND	
13	+5V	
14	GND	
15	GND	
16	GND	
17	GND	
18	A2	
19	A1	
20	A0	
21	DD0	
22	DD1	
23	DD2	
24	NC	
25	NC	
26	NC	
27	DD11	
28	DD12	
29	DD13	
30	DD14	
31	DD15	
32	HDCS1*	
33	NC	
34	IOR*	
35	IOW	
36	+5V	
37	IRQ14	
38	+5V	
39	M/S select	
40	NC	
41	RST*	
42	IORDY	
43	NC	
44	+5V	
45	ACTLED	
46	PDIAG	
47	DD8	
48	DD9	
49	DD10	
50	GND	

Note: Connector pinout is as follows:
Pins 1-25 are on the “pin 1” side of the connector, sequentially numbered (1, 2, 3, etc.).
Pins 26-50 are on the opposite side of the connector.

PC/104 Connector

Pin	Row A	Row B	Row C	Row D
0	—	—	Gnd	Gnd
1	IOCHK*	Gnd	SBHE*	MEMCS16*
2	SD7	RESETDRV	LA23	IOCS16*
3	SD6	+5V	LA22	IRQ10
4	SD5	IRQ9	LA21	IRQ11
5	SD4	–5V	LA20	IRQ12
6	SD3	DRQ2	LA19	IRQ15
7	SD2	–12V	LA18	IRQ14
8	SD1	0 WS**	LA17	DACK0*
9	SD0	+12VDC	MEMR*	DRQ0
10	IOCHRDY	Key	MEMW*	DACK5*
11	AEN	SMEMW*	SD8	DRQ5
12	SA19	SMEMR*	SD9	DACK6*
13	SA18	IOW*	SD10	DRQ6
14	SA17	IOR*	SD11	DACK7*
15	SA16	DACK3*	SD12	DRQ7
16	SA15	DRQ3	SD13	+5V
17	SA14	DACK1*	SD14	Master*
18	SA13	DRQ1	SD15	Gnd
19	SA12	Refresh*	Key	Gnd
20	SA11	SYSCLK	—	—
21	SA10	IRQ7	—	—
22	SA9	IRQ6	—	—
23	SA8	IRQ5	—	—
24	SA7	IRQ4	—	—
25	SA6	IRQ3	—	—
26	SA5	DACK2*	—	—
27	SA4	TC	—	—
28	SA3	Bale	—	—
29	SA2	+5V	—	—
30	SA1	14 MHz	—	—
31	SA0	Gnd	—	—
32	Gnd	Gnd	—	—

* = active low; ** = wait state

Appendix B: Software utilities

Introduction

The CD ROM comes with the utilities listed below. This chapter describes these utilities and their use.

Support commands

- FXCHK.EXE
- FXDOS.SYS
- FXFMT.EXE
- FXINFO.EXE
- FXRECLM.EXE
- FXREMNT.EXE
- GETBIOS.EXE
- GETIMG.EXE
- GETIMGH.EXE
- I17HNDLR.EXE
- LPT1CON.COM
- PGMBIOS.EXE
- PGMIMG.EXE
- PGMIMGH.EXE
- RESET.COM
- SETSSD.EXE
- TESTOEM.EXE

Support device drivers

- HIMEM.SYS
- FXDOS.SYS
- VDISK.SYS

FXCHK.EXE

Purpose

This support command checks the FlashFX flash media for bad spare units.

Syntax

FXCHK *drive*

Parameter

drive specifies the drive letter to check.

Remarks

FlashFX usually keeps at least one spare unit so that if a flash device develops a bad block, the spare unit takes it's place.

FXDOS.SYS

Purpose

An alternate way to access the on–board SSD. By using this driver, it is possible to free up the address area at D8000h–DFFFFh by the “X” jumper.

Syntax

FXDOS.SYS

Remarks

This command will not allow booting from an SSD.

FXFMT.EXE

Purpose

This support command formats a FlashFX flash disk.

Syntax

```
FXFMT <Drive> { Options }
```

Parameter

<*Drive*> Specifies the drive to format. This can be either a drive letter ie. C: (if the drive has already been formatted) or drive descriptor ie. 80 if the drive is not yet formatted.

Drive descriptors are of the format 8x where x is the number of the hard drive beginning with 0. For example 80 is the first hard drive, 81 is the second hard drive.

/C Don't confirm before formatting.
/B Force a system reboot and do NOT confirm.
/M Do NOT write an MBR to the media.
/Px[K] Preserves (x) bytes from start of flash.

Default is current format, Max is 960.

/Sx Format with x number of spare units.

Default is current format.

/Tx[KM] Formats (x) bytes of flash.

Default is current format, Max is media size.

/Qx Reserves (x) percent of the media size for cushion.

Default is 2%. Min is 1, Max is 25.

/Vx Place volume label "x" on disk.
/Dx Where (x) is the number of root directory entries.

Default is 240. Min is 16, Max is 512.

Remarks

The previous the contents of the drive are lost.

After reformatting, the drive can be SYSed to become bootable which requires another boot device with an installed O/S (such as a floppy).

FXFMT will not affect IDE hard drives.

FXFMT uses the current SETSSD information. This defines the order and the amount of space reserved for the BIOS. If the amount of space reserved for the BIOS changes, FXFMT must be re–executed.

See also

SETSSD.EXE.

FXINFO.EXE

Purpose

This support command displays information about a FlashFX disk.

Syntax

FXINFO *drive*

Parameter

drive specifies the drive letter to show.

Remarks

FlashFX keeps a host of information about each drive.

FXRECLM.EXE

Purpose

This support command removes outdated versions of files and reorganizes files on the FlashFX flash media, to improve access time.

Syntax

```
FXRECLM drive count
```

Parameter

drive specifies the drive letter to reclaim.
count specifies the maximum number of operations to perform.

Remarks

FlashFX will reclaim the drive specified, up to the maximum number of operations specified. This may be helpful if files are constantly changed or deleted.

FXREMNT.EXE

Purpose

This support command forces a remount of the FlashFX flash media.

Syntax

```
FXREMNT drive
```

Parameter

drive specifies the drive letter to remount.

Remarks

FlashFX will remount the drive specified. This may be helpful if the

drive was just re–programmed using PGMIMG and when a reboot of the system is not wanted.

GETBIOS.EXE

Purpose

This support command stores the BIOS information in a specific file.

Syntax

GETBIOS *filename*

Parameters

- *filename* specifies the output file for saving or programming.

See also

PGMBIOS.EXE

GETIMG.EXE

Purpose 1

This support command captures an image of an SSD and places it into a local file.

Syntax 1

GETIMG SSD1 *filename*

Purpose 2

This support command captures an image of an SSD and transfers it to a host PC running **GETIMGH**.

Syntax 2

GETIMG SSD1 /COM*x* [**/B***xx*] [**/I***xx*]

Purpose 3

This support command captures an image of an SSD and transfers it to a host PC running GETIMGH. A nonstandard serial port I/O address is used and the IRQ value must be specified. This syntax is required when using the serial console.

Syntax 3

GETIMG SSD1 **/U***xxxx* [**/B***xx*] [**/I***xx*]

Parameters

- **SSD1** specifies the target SSD for file saving or programming.
- *filename* specifies the output file for saving or programming.
- **/COM***x* specifies the PC COM port for serial transfer where *x* represents a value from 1 to 4.
- **/U***xxxx* specifies the UART base address to use for serial transfer. The base address, 100–3FF, is in hexadecimal format.
- **/B***xx* specifies baud rate of transfer where *xx* can be 300, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115 (115200). The default is 38400 when using a **/COM***x* switch.
- **/I***xx* specifies the interrupt to which the UART is connected. The value *x* represents a variable from 3 to 15.

See also

GETIMGH.EXE, PGMIMG.EXE, PGMIMGH.EXE

GETIMGH.EXE

Purpose 1

This support command transfers an SSD image from a target PC running **GETIMG** and saves the image to a host file.

Syntax 1

GETIMGH *filename* **/COM***x* [**/B***xx*] [**/I***xx*]

Purpose 2

This support command transfers an SSD image from a target PC running GETIMG and saves the image to a host file via a serial UART connection. A non-standard serial port address is used and the IRQ value must be specified.

Syntax 2

GETIMGH *filename* **/U***xxx* [**/B***xx*] **/I***xx*

Parameters

- *filename* specifies the output file for saving or programming and it also represents the host filename.

- **/COM***x* specifies the PC COM port for serial transfer. The variable *x* represents a value from 1 to 4.

- **/U***xxx* specifies the UART base address to use for serial transfer. The base address, 100-3FF, is in hexadecimal format.

- **/B***xx* specifies baud rate of transfer where *xx* can be (300, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115 (115200). The default is 38400 when using a **/COM***x* switch. If COM1 is the console, the baud rate defaults to the current console baud rate.

    - **/I***xx* specifies the interrupt to which the UART. The variable *x* represents a value from 3 to 15.

See also

GETIMG.EXE, PGMIMG.EXE, PGMIMGH.EXE

HIMEM.SYS

Purpose

This device driver manages extended memory and the High Memory Area (HMA) in a 286, 386, or greater PS/2 system. HIMEM prevents programs from simultaneously using the same area of memory for two different purposes. HIMEM supports the eXtended Memory Specification (XMS) 2.0. HIMEM is installed as a device driver in CONFIG.SYS.

Syntax

DEVICE=[*d*:] [*path*] **HIMEM.SYS** [/machine:*n*]

Remarks

- The HIMEM driver can be used to allow ROM–DOS to run in High Memory.
- HIMEM supports a default of 32 handles.
- HIMEM should not be used with older versions of VDISK. Current versions of VDISK will use XMS memory if it is available.
- HIMEM recognizes PS/2 style machines A20 line control. HIMEM determines whether to use the PS/2 A20 control or the AT A20 control method automatically by calling INT15h, function C0h (get system configuration).
- The automatic detection can be overridden with the "/Machine:*n*" command line switch. Replacing "*n*" with 1 designates the PC AT A20 control method. Replacing "*n*" with 2 designates the PS/2 method.

Example 1

```
DEVICE=HIMEM.SYS
```

The above command installs the XMS device driver. Once this driver is installed, accessing the HMA and Extended Memory (XMS) areas are legal. The Extended Memory area can contain up to 2 GB of memory. Typical systems have 4, 8, or 16 MB XMS memory installed.

Example 2

```
DEVICE=HIMEM.SYS /machine:1
```

This example forces the use of the AT style A20 line control.

The HIMEM driver will fail to load if either the machine does not have memory above the 1 MB boundary or the BIOS does not provide support for it. It will also fail to load if another XMS manager has been previously installed .

I17HNDLR.EXE

Purpose

This support command is an alternate way to use the INT 17h functions when the extended BIOS area is disabled (i.e., the jumper at the “X” position is removed at W1. Also, use this support command to reprogram the extended BIOS area with another BIOS.

Syntax

```
I17HNDLR
```

Remarks

This command is used if the extended BIOS area (D8000–DFFFF) is not used. The I17HNDLR allows the system to use the INT 17h functions.

LPT1CON.COM

Purpose

This support command redirects the video to the LPT1 port.

Syntax

```
LPT1CON
```

Remarks

If you have a keypad and display board and an LCD display connected to the AUX I/O port, executing the DISPLAY.EXE and LPT1CON.COM programs allow you to use the display as the system console. You must reset your system to change the video to the original parameters.

PGMBIOS.EXE

Purpose

This support command programs a new system BIOS into the 2050.

Syntax

```
PGMBIOS filename [/y] [/?]
```

Parameters

- *filename* specifies the BIOS .DAT file to program into flash.
- **/y** disables “Are you sure?” confirmation message.
- **/?** requests a help menu.

Example 1

To program the BIOSFILE.BIN files into the SSD1 BIOS area, enter:

```
PGMBIOS BIOSFILE.BIN
```

See also

GETBIOS.EXE

PGMIMG.EXE

Purpose 1

This support command programs a local file image to an SSD.

Syntax 1

```
PGMIMG SSD1 filename
```

Purpose 2

This support command programs a local file image to an SSD and transfers it to a host PC running **PGMIMGH**.

Syntax 2

PGMIMG SSD1 /COM*x* [**B***xx*] [**I***xx*]

Purpose 3

This support command programs a local file image to an SSD and transfers it to a host PC running PGMIMGH. A nonstandard serial port I/O address is used and the IRQ value must be specified. This is the required format when using a serial console.

Syntax 3

PGMIMG SSD1 /U*xxxx* [**B***xx*] **/I***xx*

Parameters

- *filename* specifies the input file programming.
- **SSD1** specifies the target SSD for image.
- **/COM***x* specifies the PC COM port for serial transfer. The variable *x* represents a value from 1 to 4.
- **/U***xxx* specifies the UART base address to use for serial transfer. The base address, 100–3FF is in hexadecimal format.
- **/B***xx* specifies baud rate of transfer where *xx* can be 300, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115 (115200). The default is 38400 when using a **/COM***x* switch. If COM1 is the console, the baud rate defaults to the current console baud rate.
- **/I***xx* specifies the interrupt to which the UART base address is connected. The variable *x* represents a value from 3 to 15.

See also

GETIMG.EXE, GETIMGH.EXE, PGMIMGH.EXE

PGMIMGH.EXE

Purpose 1

This support command programs an image file from a target PC running PGMIMG.

Syntax 1

PGMIMGH *filename* /**COM***x* [/**B***xx*] [/**I***xx*]

Purpose 2

This support command transfers an SSD image to a target computer via a serial UART connection and programs the image to an SSD. A non-standard serial port address is used and the IRQ must be specified.

Syntax 2

PGMIMGH *filename* /**U***xxx* [/**B***xx*] /**I***xx*

Parameters

- *filename* specifies the input file for programming and it also represents the host filename.

- **/COM***x* specifies the PC COM port for serial transfer. The variable x represents a value from 1 to 4.

- **/U***xxx* specifies the UART base address to use for serial transfer. The base address, 100-3FF is in hexadecimal format.

- **/B***xx* specifies baud rate of transfer where *b* can be (300, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115 (115200). The default is 38400 when using a **/COM***x* switch. If COM1 is the console, the baud rate defaults to the current console baud rate.

- **/I***xx* specifies the interrupt to which the UART base address is connected. The variable x represents a value from 3 to 15.

See also

GETIMG.EXE, GETIMGH.EXE, PGMIMG.EXE

RESET.COM

Purpose

This support command enables the watchdog timer and allows time–out to expire, thus restarting the system.

Syntax

RESET

Remarks

The RESET command also restarts all the expansion I/O cards on the bus. This differs from a <CTRL><ALT><DEL> reboot of the system which only restarts the system but not the expansion cards. The RESET button on the 2050 also accomplishes the same thing as the RESET command.

SETSSD.EXE

Purpose

This support command sets the arrangement of the SSD FlashFX devices.

Syntax

```
SETSSD [SSD1] [/Before | /After] [/NoSSD] [/?] [/D] [/WP1+ |
/Wp1-]
```

Parameters

SSDspecifies whether SSD1 is enabled.
/After indicates the SSDs are allocated after any hard drives and the hard drive will be the boot device.

/Before indicates the SSDs are allocated before any hard drives and the first SSD specified will be the boot device.
/NoSSD specifies no SSDs are to be allocated (default if either **/A** or **/B** used and no SSDs are specified) .
/D selects default settings. (**SSD1 /After**)
/WP1+ write protects drive on SSD1 (- enables writes)

Remarks

Defines the SSD device order used by the flash file system. If an SSD name is included on the command line, the SSD will be enabled and can therefore be detected and used by FlashFX. If more than one SSD is listed on the command line, the first SSD will be the first drive (usually C:) and the second SSD will be the second (usually D:).

When IDE hard drives are used, the **/BEFORE** and **/AFTER** switches allow the first SSD to appear before or after the hard drives. For example if two IDE drives and one flash device (SSD1) are defined by SETSSD, and the /BEFORE switch is used; SSD1 will be drive C: and the two IDE drives will be D: and E:. However if the /AFTER switch is used, the first IDE drive will be drive C:, SSD1 will be drive D: and the second IDE drive will be E:.

When other devices such as a DOC are used, the DOC extended BIOS may affect the /AFTER and /BEFORE and the "C: ONLY" setup option.

Warning

Use this option with caution!

/WPn can be used to disable writes to the part via the drivers. PGMIMG can still write to these devices.

The SETSSD device order can be overridden by removing the "S" jumper. When removed and when the system boots up, prompts will be asked as to which device is first and which is second. Then a prompt asking whether you wish this information to be saved. If the save prompt is answered "Y" the new information overrides the current SETSSD options. If the save prompt is answered "N" the new information is used but not saved. Since the SETSSD command shows the "saved" options not the temporary "working" options, this may lead to confusion. It is therefore recommended to always "save" your options.

The device order is used by the FlashFX extended BIOS as well as the FXFMT and TESTOEM commands. Other commands such as FXCHK, FXINFO, FXRECLM etc. are indirectly affected as the device order is changed. The BIOS reserve option effects the FXFMT, PGMIMG,

GETIMG commands.

See also

See the *SETSSD* section in the *Setup programs* chapter.

TESTOEM.EXE

Purpose

This support command tests the FlashFX flash media.

Syntax

```
TESTOEM </Dx | /A> [/Cs] [/E] [/X] [/R] [/W] [P]
```

Parameter

/Dx Replace x with the hard drive number, zero based.
/A Tests all drives available in the OEM layer
/C"s" Replace s with command line args to send to the OEM layer during initialization
/E Intensive write/erase tests to the entire media
/X Extensive erase/write/verify tests to the hidden areas
/R Test removable media change detection
/W Test write protect status detection
/P Execute benchmarks on the access routines

Remarks

All data on the flash part drive area will be erased. The current FlashFX drive settings are used. See SETSSD for more details.

Appendix C: Accessories

Table C–1 Cables and terminal board

Product	Description	Octagon part number
MPC-18	Multipurpose cable	5905
Cable, 6225 power cable	Power cable	5848
Null modem adapter	9-pin to 9-pin	2470
CMA-26-12	12" cable	2776
CMA-26-24	24" cable	1257
3.5" hard drive adapter	2mm connector to 2.5", 3.5" hard drive	4080
STB-26	Terminal board, 26-position	2905

Table C–2 LCD displays and keypads

Product	Description	Octagon part number
LCD–4 x 20	LCD display w/cable, 40 character	2783
LCD–4 x 40	LCD display w/cable, 80 character	2784
2010	LCD display/keypad interface	3909
KP-1	Keypad w/cable, 16-key, low cost	1218

Table C–3 Miscellaneous part numbers

Product	Description	Octagon part number
AT battery	Calendar/clock battery backup	3186
CAMBASIC	Multitasking, industrial control programming language	4059

Warranty

Octagon Systems Corporation (Octagon), warrants that its standard hardware products will be free from defects in materials and workmanship under normal use and service for the current established warranty period. Octagon's obligation under this warranty shall not arise until Buyer returns the defective product, freight prepaid to Octagon's facility or another specified location. Octagon's only responsibility under this warranty is, at its option, to replace or repair, free of charge, any defective component part of such products.

Limitations on warranty

The warranty set forth above does not extend to and shall not apply to:

1. Products, including software, which have been repaired or altered by other than Octagon personnel, unless Buyer has properly altered or repaired the products in accordance with procedures previously approved in writing by Octagon.
2. Products which have been subject to power supply reversal, misuse, neglect, accident, or improper installation.
3. The design, capability, capacity, or suitability for use of the Software. Software is licensed on an "AS IS" basis without warranty.

The warranty and remedies set forth above are in lieu of all other warranties expressed or implied, oral or written, either in fact or by operation of law, statutory or otherwise, including warranties of merchantability and fitness for a particular purpose, which Octagon specifically disclaims. Octagon neither assumes nor authorizes any other liability in connection with the sale, installation or use of its products. Octagon shall have no liability for incidental or consequential damages of any kind arising out of the sale, delay in delivery, installation, or use of its products.

Service policy

1. If a product should fail during the warranty period, it will be repaired free of charge. For out of warranty repairs, the customer will be invoiced for repair charges at current standard labor and materials rates.
2. Customers that return products for repairs, within the warranty period, and the product is found to be free of defect, may be liable for the minimum current repair charge.

Returning a product for repair

Upon determining that repair services are required, the customer must:

1. Obtain an RMA (Return Material Authorization) number from the Customer Service Department, 303-430–1500.
2. If the request is for an out of warranty repair, a purchase order number or other acceptable information must be supplied by the customer.
3. Include a list of problems encountered along with your name, address, telephone, and RMA number.
4. Carefully package the product in an antistatic bag. (Failure to package in antistatic material will VOID all warranties.) Then package in a safe container for shipping.
5. Write RMA number on the outside of the box.
6. For products under warranty, the customer pays for shipping to Octagon. Octagon pays for shipping back to customer.
7. Other conditions and limitations may apply to international shipments.

Note PRODUCTS RETURNED TO OCTAGON FREIGHT COLLECT OR WITHOUT AN RMA NUMBER CANNOT BE ACCEPTED AND WILL BE RETURNED FREIGHT COLLECT.

Returns

There will be a 15% restocking charge on returned product that is unopened and unused, if Octagon accepts such a return. Returns will not be accepted 30 days after purchase. Opened and/or used products, non–standard products, software and printed materials are not returnable without prior written agreement.

Governing law

This agreement is made in, governed by and shall be construed in accordance with the laws of the State of Colorado.

The information in this manual is provided for reference only. Octagon does not assume any liability arising out of the application or use of the information or products described in this manual. This manual may contain or reference information and products protected by copyrights or patents. No license is conveyed under the rights of Octagon or others.