MARK 2000

Business System Installation/ Operation Guide

POINT DATA CORPORATION

POINT 4 DATA CORPORATION

MARK 2000

BUSINESS SYSTEM INSTALLATION/ OPERATION MANUAL

Revision 01

NOTICE

Every attempt has been made to make this manual complete, accurate and up-to-date. However, all information herein is subject to change due to updates. All inquiries concerning this manual should be directed to POINT 4 Data Corporation.

Document Order Number: HTP0084

POINT 4 Data Corporation
15442 Del Amo Avenue
Tustin, CA 92680
(714) 259-0777

REVISION RECORD

PUBLICATION NUMBER: HM-250-0084

Revision	Description	Date
01	Initial customer release	06/01/89

LIST OF EFFECTIVE PAGES

Changes, additions, and deletions to information in this manual are indicated by vertical bars in the margins or by a dot near the page number if the entire page is affected. A vertical bar by the page number indicates pagination rather than content has changed. The effective revision for each page is shown below.

Page	Rev	Page	Rev	Page	Rev
Cover	-				
Title thru x	01				
1-1 thru 1-7	01				
2-1 thru 2-5	01				
3-1 thru 3-11	01				
4-1 thru 4-26	01				
5-1 thru 5-9	01				
6-1 thru 6-12	01				
7-1 thru 7-3	01				
A-1 thru A-12	01				
B-1 thru B-4	01				
Comment Sheet	01				
Mailer	-				
Back Cover	-				

PREFACE

This manual covers topics the user needs, to install, initialize and maintain the Mark 2000 System.

For a more in-depth understanding of the system software refer to the related documents listed below.

Related Documents

Title	Document Order No.
RISC/os Programmer's Reference Manual	UTP2001
RISC/os User's Reference Manual	UTP2002
RISC/os System Administration Reference Manual	UTP2003
RISC/os System Administrator's Guide	UTP2004
RISC/os Programmer's Guide	UTP2005
RISC/os User's Guide	UTP2006
RISC/os Languages Programmer's Guide	UTP2007
RISC/os Software Release Notes	UTP2008

CONTENTS

Appendices

Figures

Tables

Section 1
GETTING TO KNOW YOUR COMPUTER

1.1 MARK 2000 SYSTEM FEATURES

* MIPS RISC R2000 CPU chip technology
* 16.7MHz operation
* 8 to 48MB of RAM, 32-bit bus
* RISC/os (UNIX System V + BSD 4.2)
* MIPS R2010 floating point processor
* 702MB, 376MB, 182MB hard disk drives
* 150MB cartridge tape drives
* SCSI port
* Ethernet port (IEEE 802.3 Standard)
* Battery-backed calendar/clock
* Boot PROM monitor & power-up diagnostics
* 132 RS232 ports total
* 4 hard disk drives total
* Half-inch tape drive
* 2.3GB Helical Scan tape drive
* 4 AT-bus slots
* Front panel keylock/switch

1.1.1 Motherboard

The Motherboard contains the control logic that interfaces the CPU Board, up to six memory boards, and up to four AT-bus-compatible boards.

The Motherboard contains control logic to handle four RS232 serial ports, one Ethernet port, and a SCSI port. The disk and tape drives are controlled by the SCSI port.

A boot PROM with power-on diagnostics and a battery-backed clock/calendar are also contained on the Motherboard.

1.1.2 CPU Board

The CPU consists of the MIPS RISC Technology R2000 CPU chip, the R2010 Floating Point Processor chip, and the R2020 Write Buffer chip.

There are two 64KB caches for the instructions and data. The CPU board plugs into the Motherboard.

1.1.3 Memory Board

Each memory board has 8MB of memory. Up to six boards can be plugged into the Motherboard for a total of 48MB of memory.

1.1.4 Front Panel and Controls

The Front Panel consists of the LED indicators and the Keyswitch. One of the LEDs indicates that power is on, the other four LEDs indicate disk-busy status (see Figure 1-1).

The Keyswitch has 3 positions: Lock, Unlock, and Reset.

Lock - The lock position
The key can be removed in this position. The system cannot be booted if the power is on and the system is at the PROM Monitor. This position also prevents the system from being shutdown and places restraints on operating system Run state transitions. For additional information refer to "telinit" (1M).

Unlock - The unlock position
The key can be removed in this position. The system can be booted when the power is on and the system is at the PROM Monitor prompt. The system can also be shut down when the key is in the unlock position and root or the superuser is the initiator of the shutdown.

Reset - The reset position is a momentary position and causes the system to be re-booted without executing the power-on diagnostics.

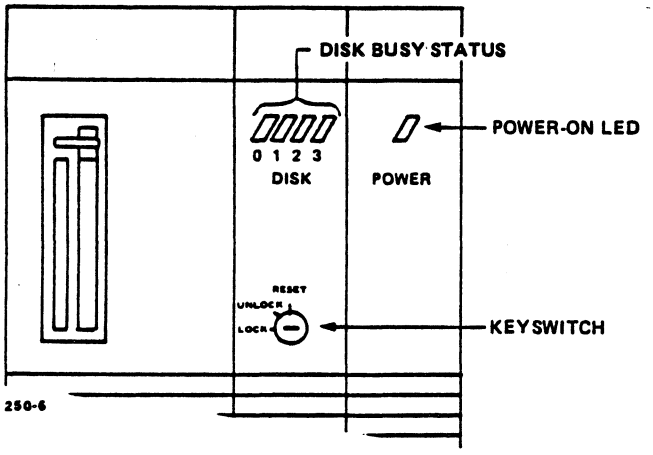


Figure 1-1. Front Panel

1.2 SOFTWARE

1.2.1 RISC/os Documentation

RISC/os is a fully featured, high performance implementation of the UNIX operating system. The documentation for RISC/os, and its associated subpackages, is available from POINT 4; it consists of a series of guides and reference manuals. An important distinction is made between a "guide" and a "reference" manual. Guides are tutorial, in that they provide step-by-step examples of various system functions and commands; the reference documents contain detailed information for each function or feature. Reference manuals are factual in their presentation style and are best suited to the more knowledgeable user. Every topic supported by RISC/os is covered in the reference manuals, whereas only the major features are covered in the guides.

RISC/os has many on-line help facilities. A very important feature of RISC/os is that the reference documentation is available on-line. For any command supported, type "man" and the command name; the system will respond with the appropriate pages of the reference manual.

In addition, there is an on-line glossary, with usage description and subject locate commands. This on-line concept provides a beginners start-up tutorial, which should be reviewed by new users. To "walk-through" these facilities the user should type "help" for access and directions for use.

The references and guides are divided into three main topic areas:

System Administration
Programming
Usage

The system administration documents address topics such as creating users, managing hardware (i.e., disks and terminals), and backups. They describe the functions necessary for the proper management of a multiuser computer system.

The programming documents describe the programming functions provided in the base RISC/os release (not the optional languages). The majority of the documentation is about the "C" programming language, which is the heartbeat of any UNIX-based system.

The user's documentation focuses on the usage of the system, not on the administration or programming. All user functions and commands are discussed in these documents.

When a question arises, the category of the question will guide the user to the appropriate manual.

Within the guides and references you will find chapter designators, for example:

cat(1)

The number in parentheses denotes the chapter in the reference manuals that contains documentation for that command. The organization of the chapters is as follows:

	chapter
User's Reference Manual	
commands	1
System Administrator's Reference Manual	
administration commands	1M
special files	7
Programmer's Reference Manual	
system calls	2
library subroutines	3
file formats	4
miscellaneous	5

You will find a complete listing of manuals available, as well as ordering procedures, in the POINT 4 Products and Services Catalog.

1.2.2 RISC/os Distribution Package

RISC/os is distributed by POINT 4 Data Corporation on a QIC-150 tape. The tape contains all the necessary programs and procedures to install an operating system from scratch, update an existing operating system, or perform maintenance functions such as the formatting of disks. When a new system arrives from POINT 4 Data Corporation, the RISC/os is already loaded. These tapes should be kept as backup in case it ever becomes necessary to reload the RISC/os. When a new release of the RISC/os is ordered from POINT 4 for an existing system, it is distributed in this same format along with the RISC/os Software Release Notes for that release. The release notes describe the installation and upgrade processes as well as details of the features and fixes of the particular release.

1.2.3 Starting RISC/os

When you receive your MARK 2000 system, it has been preloaded with RISC/os and is capable of full operation from the console port (Port 0 of the system motherboard). The console port of the system must have a terminal attached that has the following characteristics:

9600 baud
8 data bits
No parity
Character mode (not block mode)
XON/XOFF operation
Stop bit

When the system is powered-up, notifications of system activity are displayed through the console. When the system is "alive", the system can be configured for your particular requirements.

1.2.4 SYSADM - Systems Administration

An automated facility for administration of your system is provided. By logging in as **adm** and entering **sysadm** at the # prompt.

The systems administrator then provides you with menu selections and on-line help text to "walk" you through the configuration of your system. The following are the major categories available through the system administrator:

- File management - provides for system backups, disk usage, file age, file size, restore and storing of files on removable media

- Machine management - provides for power-down, reboot and who's on the system functions

- System setup - used to set administration passwords, date and time, node name setting, first-time setup, and the assignment of system passwords

- TTY (TTYs in UNIX terminology refer to Asynchronous Serial Ports) management - administers the hunt sequences for TTY lines, creates new hunt sequences and line settings, and allows viewing and, optionally, setting of TTY lines

- User management - used to add a group, add a user, delete a group, delete a user, list groups, lists users, and modify defaults used by adduser

The system administrator functions are accessed through a menu; this provides a simple and direct approach for configuring and maintaining your system. These tasks can also be done manually, i.e. editing script files, with much more work. It is suggested that you utilize sysadm until you are in complete command of the various system relationships that must be maintained in RISC/os; then you may find other methods of accomplishing your system administration needs.

1.3 HARD DISK DRIVE

The system can support up to four SCSI hard disk drives, which interface to the Motherboard SCSI port. Drives are available in capacities of 182MB, 376MB, and 702MB unformatted.

The drives are supplied already formatted (512 bytes/sector).

1.4 CARTRIDGE TAPE DRIVES

The 1/4" cartridge tape drive has a capacity of 150MB. It is used to back up and restore data to the hard disk. To utilize the drive's full 150MB capacity, the DC600XTD tapes must be used. The drive interface is SCSI and is controlled through the Motherboard SCSI port.

The helical scan tape drive uses an 8MM tape cartridge and has a capacity of 2.3 G Bytes. This drive also has an SCSI interface.

1.5 HALF-INCH TAPE DRIVES

Half-inch tape is an optional feature of the system. It requires that an adapter board and cable assembly be installed. A tape drive with a Pertec interface is then connected to the rear panel of the system. The adapter board is capable of supporting tape densities of 1600, 3200, 6250 BPI.

1.6 VIDEO DISPLAYS

POINT 4 offers optional video display terminals. The terminals are connected either through the RS232 ports (DB25 connectors) of the Motherboard or through the RS232 ports (RJ11 connectors) of the Terminal Concentrators. Up to 132 RS232 ports may be configured on this system.

1.7 PRINTERS

Printers are connected either through the RS232 serial ports of the Motherboard or through the Terminal Concentrators.

Section 2
SETTING UP YOUR COMPUTER

This section provides you with all of the information necessary to set up your system hardware.

2.1 CONNECTING THE POWER CORD

WARNING!

Be sure the CPU power cord is not plugged into the wall outlet before you try to connect it to the computer.

CAUTION

Be sure that the CPU power switch is off before attempting to connect the power cord.

Verify that the computer power matches the wall power, 115VAC or 220VAC.

Connect the female end of the power cord to the 3-prong male outlet on the rear panel of the computer. See Fig. 2-1. Connect the male end of the power cord to the wall outlet.

2.2 CHECKING THE CABLE CONNECTIONS

Display Terminals and Printers:

Serial devices such as display terminals or printers are connected to the RS232 ports of the Motherboard or Terminal Concentrator. The Motherboard port uses a DB25 connector. The Terminal Concentrator uses a RJ11 connector. Individual power cords are required for the display terminals and printers.

Terminal Concentrator:

For each Terminal Concentrator, a Mux-8tc Cable is required to be connected between the Mux Board in the computer and the Terminal Concentrator (see Figure 2-1). The Mux-8tc Cable has 8-pin DIN (round) connectors at each end. This cable is 10 feet long and also carries the power connection to the Terminal Concentrator board. See Section 4.4 for more information.

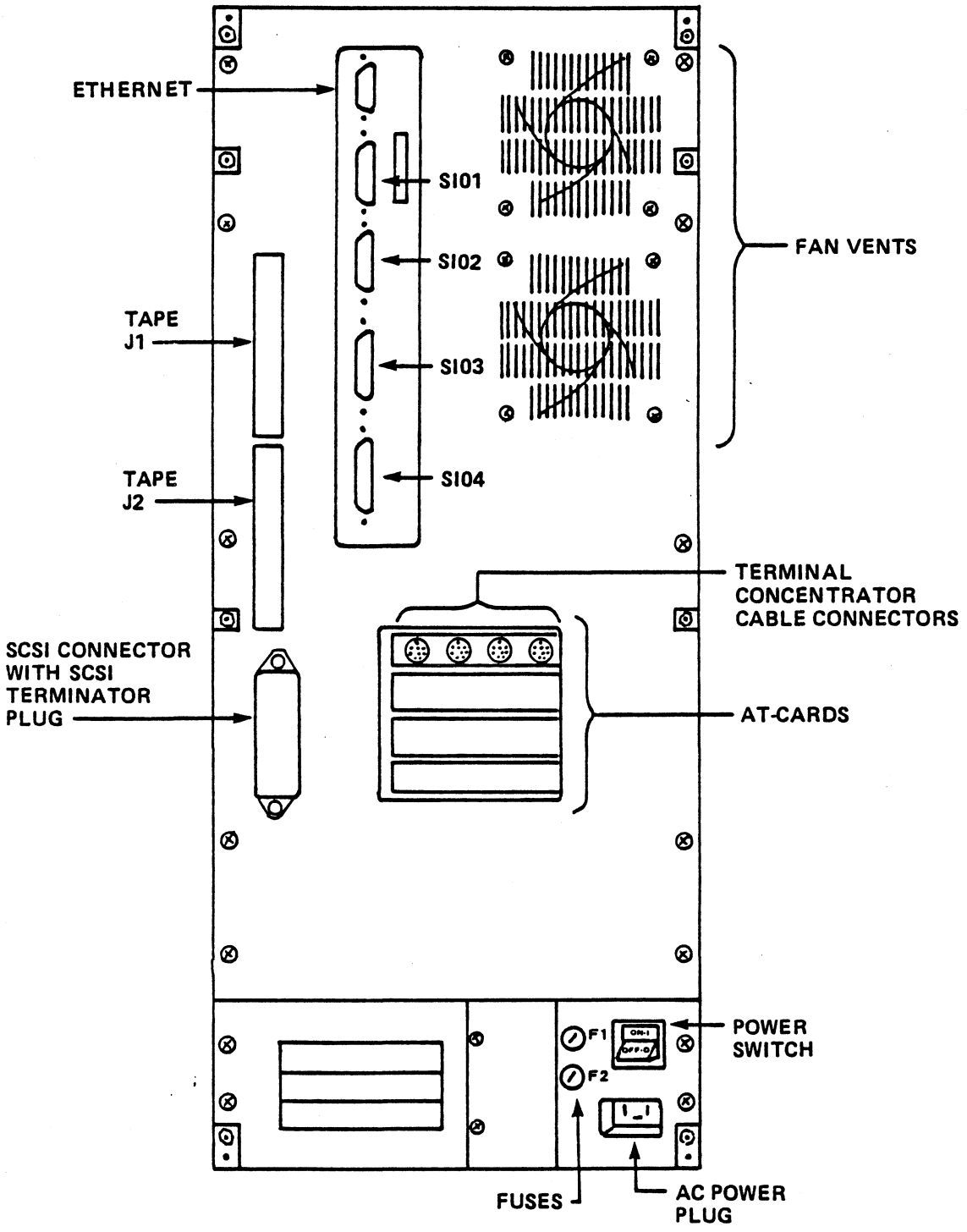


Figure 2-1. MARK 2000 Rear View

CAUTION

If the Mux-8TC cable is longer than 10 feet, a separate power supply is required to be plugged into the terminal concentrator.

1/2-Inch Tape Drive:

There are two 50-pin connectors on the computer rear panel for interfacing to the Pertec-compatible 1/2-inch Tape Drive. Both cables are identical. J1 on the computer is connected to P1 of the drive. J2 on the computer is connected to P2 of the drive. A separate power cord is required for the drive.

SCSI Port:

The SCSI port on the computer rear panel has a terminator plugged into the connector. This is left on at all times if there are no external SCSI devices connected to the port. If an external SCSI device is to be used, the terminator is removed and a cable is plugged into the connector; the external SCSI device must then have terminators. The SCSI connector is an Amphenol 57FE or equivalent.

Ethernet Port:

There is one Ethernet port on the computer rear panel. A DB15 connector is used.

2.3 POWER-UP SYSTEM AND LOG-ON

This procedure steps you through turning on the system and bring it up to the multi-user mode.

The first step is to turn on the console and wait for the cursor.

The second step is to press the power button on the back of the system to the on position. This starts the system bootstrap. On the console, the system identifies itself and the amount of memory available to the system.

The third step is to respond to the >> prompt with auto.

```
>> auto
```

This starts the process of loading the system into memory. As it is being loaded into memory, the software reports the status of the system and its status in loading the system. Any failures are noted to the console. You will be asked the question how you want the file system checked. To get a complete file system check, you do not respond to the message. You should normally do a complete file system check.

```
***** To fsck only dirty ones, type 'yes' within 5 seconds:
```

After the system software has been loaded, it displays a ready message:

```
The system is ready.
```

The fourth step is to log on to the system by responding to the following message with your log-on and password:

```
"... console login:"
```

If it is the first time that the system is being logged-on, use **root** and a carriage return password. The software then displays the copyright information and prompts you for your next command. If this is your first time you should now proceed with the setup of your user accounts, terminal definitions, user directories, etc. For information on these items refer to the sections in the System Administrator's Reference Manual on system security, TTY management, file system, etc.

2.4 SHUT-DOWN PROCEDURE

The system should NEVER be POWERED-OFF without going through an orderly system shutdown. Powering the system down without an orderly shutdown may damage your file system.

2.4.1 Multi-User Power-Down

To do a system power-down from the multi-user mode use the sysadm command **sysadm powerdown.** This command flushes the system buffers, closes open files, stops all user or daemon processes, unmounts the file system, and puts the system in monitor mode so it can be safely powered-off.

The first step in the power-down sequence is to determine who is on the system. This is done by using the command **who.** This command displays who is logged on to the system, their identification and terminal number.

The second step is to notify the users that the system is to be powered-down. This is done using the **wall** command. This command sends a message to all logged on users.

The third step after notifying the user of the pending shutdown is to issue the command **sysadm powerdown.** This command puts the system in a state where nothing is running. As this command is being processed, it asks a couple questions; answer them appropriately. When the system is completely down it is at the >> prompt waiting for instructions.

ONLY at this point can the power be turned off without damage to the file system.

2.4.2 Single-User Power-Down

If you are in the single user mode, the **shutdown** command is used before powering down the system. As this command is being processed, it asks some questions; answer the questions appropriately. When the system is completely down, it is at the >> prompt waiting for instructions.

ONLY at this point can the power be turned off without damage to the file system.

For more information on these commands and other see the System Administrator's Reference Manual.

Section 3
INSTALLING SOFTWARE AND CONFIGURING THE SYSTEM

Your MARK 2000 is shipped from POINT 4 Data Corporation with the RISC/os already installed and minimally configured. The procedures provided in this section are required only in the event of a catastrophic system failure or when a disk drive is added.

3.1 DISK FORMATTING

3.1.1 Overview

The MARK 2000 employs SCSI disk drives. Most SCSI disks are preformatted by the drive vendor and thus do not require formatting. They do, however, require that a Volume Header Record be written to the disk. If a disk drive is added to the system, it will be necessary to write the Volume Header Record. In the event of a catastrophic disk failure, it will be necessary to format the drive. It is important to note that the writing of Volume Header Record is not a full format; a full format should be done only if you suspect that something is physically wrong with the disk.

3.1.2 Writing a Volume Header

Power-on the system and insert the first of the two RISC/os tapes. At the >> prompt of the Prommonitor (the startup program stored in programmable Read-Only Memory) respond as shown in the following example:

```
M/120 MIPS Monitor Version 4.10 MIPS OPT Tue Nov. 15 12:41:31 PDT 1988 root

Memory size:  8388608 (0x800000) bytes
Icache size:  16384 (0x4000) bytes
Dcache size:  8192 (0x2000) bytes
>> boot -f tqis(,,2)format
145488+49664+200656 entry: 0x80020000

MIPS Format Utility
Version 4.10  Tue Nov. 15 11:01:11 PDT 1988 root

name of device? dkis
LUN number? 0
target id? 0

choose new drive parameters (y if yes)? y
```

When initializing a drive, you must answer yes to this question. The current drive parameters are not available because the Volume Header has not been initialized. The following menu is displayed. Currently only the CDC drives are supported.

```
device parameters are known for:
       (9) fuji 2246sa (140Meg SCSI)
      (10) cdc 94161 (160Meg SCSI)
      (11) cdc 94171 (328Meg SCSI)
      (12) fuji 2249sa (325Meg SCSI)
      (16) cdc 94181 (600Meg SCSI)
      (17) cdc 94351-172 (172Meg SCSI 3.5")
enter number for one of the above? 10

The Unix file system partitions may be either BSD or System V
do you desire BSD file system partitions (y if yes)? y
```

You must answer yes to this question.

```
dump device parameters (y if yes)? n
modify device parameters (y if yes)? n
```

Typically the default parameters should be used. You may modify them if you need something special.

```
dump partition table (y if yes)? y
```

If you wish to use the default partition scheme, you do not have to answer yes to this question.

```
Root partition is entry #0
Swap partition is entry #1
Default boot file is /vmunix

entry  type           #blks   #cyls  cg(mod)  1st_lbn  1st_cyl  num_bytes
0-a    BSD file sys   44415   141    8(13)    2205     7        22740480
1-b    BSD file sys   39375   125    7(13)    265230   842      20160000
2-c    BSD file sys   302400  960    60(0)    2205     7        154828800
3-d    BSD file sys   218610  694    43(6)    46620    148      111928320
4-e    BSD file sys   116235  369    23(1)    46620    148      59512320
5-f    BSD file sys   58275   185    11(9)    162855   517      29836800
6-g    BSD file sys   174510  554    34(10)   46620    148      89349120
7-h    BSD file sys   44100   140    8(12)    221130   702      22579200
8      volume header  2205    7      0(7)     0        0        11289600
partition 9 size == 0
10     entire volume  304605  967    60(7)    0        0        155957760
11-i   BSD file sys   160650  510    31(14)   2205     7        82252800
12-j   BSD file sys   141750  450    28(2)    162855   517      72576000
13-k   BSD file sys   83475   265    16(9)    221130   702      42739200
14-l   BSD file sys   80325   255    15(15)   2205     7        41126400
15-m   BSD file sys   80325   255    15(15)   82530    262      41126400

modify partition table (y if yes)? n
```

If you do not wish to make any changes, answer no. However, you may want to change the Default boot file from /vmunix to /unix. This is the entry that is used for autobooting.

```
formatting destroys ALL SCSI disk data, perform format (y if yes)? n
```

Do not answer yes! Formatting should be done on a SCSI disk only when you suspect that something is physically wrong with the disk.

```
formatting wasn't done, perform scan anyway (y if yes)? n
```

Do not answer yes! There is no need to scan the disk unless the disk has just been reformatted or you suspect that something may be physically wrong with the disk.

```
SCSI defect list manipulation, when prompted choose one of (list,
add, delete, quit) command? quit
```

Enter quit. The other three commands are used only after a complete format has been done.

```
write new volume header? (y if yes)? y

writing volume header...
exit(0) called
```

The Volume Header has been written to disk. If an error message occurs, check that the correct drive type was entered. If correct, ensure that the disk has been cabled correctly.

```
M/120 MIPS Monitor Version 4.10 MIPS OPT Tue  Nov 15 12:41:31 PDT 1988 root
Memory size: 8388608 (0x800000) bytes
Icache size: 16384 (0x4000) bytes
Dcache size: 8192 (0x2000) bytes
>>
```

3.1.3 Formatting the Disk

If for some reason you must format the drive, follow the same procedure as in Section 3.1.2, up to the point where you are asked to perform a format. Answer "Yes".

This will start the format operation. When completed, you will be asked if you wish to perform a scan. Again, answer "Yes" and specify at least "2" for the number of scans. This operation determines where the bad sectors are located and enters them in the defect list.

After the scan, the Volume Header is written and the formatting operation is completed.

3.1.4 Clean Disk

Once you have formatted the system disk and written the Volume Header on it, you are in a state of a "clean disk" with no operating system. Now it is necessary to do a full load of RISC/os.

3.2 INSTALLING RISC/os

The RISC/os installation instruction may change from released to release, so it is necessary to use the RISC/os Software Release Notes for the particular release you are installing.

3.3 BACKUP AND RESTORE PROCEDURES

Offline backup protects RISC/os fixed-disk files from the unexpected. Backup provides copies of files and file systems against accident, carelessness, and technical mishap.

RISC/os provides a number of programs to create archive copies of files and file systems on tape. This subsection introduces the available programs and gives a brief overview of which to use in different situations.

3.3.1 Choosing a Copy Program

The RISC/os provides a variety of copy programs for backup and other archive functions. (See your RISC/os user manual for complete information about each program.) The advantages and disadvantages of each program are summarized in Table 3-1.

TABLE 3-1. RISC/OS COPY PROGRAMS SUMMARY

Program	Advantages for Backup	Disadvantages for Backup
cpio	Used with find to descend directory hierarchy; a good all-purpose program for total and incremental backups and restores; prompts at end of tape	You must unmount file systems you do not want to copy if they are lower in the tree than filesystems you are copying; in general, not as easy to copy single file systems
DD	Fast; good for single file systems; disk partitions make convenient units of files for backup; arbitrary block sizes available; the only program that can be used to back up a non-UNIX file system (data base)	Difficult to recover individual files (entire archive must be copied back to disk); only one file (partition) per archive; not useful for incremental backup
dump	Recommended backup program can be used for both a full backup or an incremental backup; individual files can be restored from the backup. Used in conjunction with restore	
tar	Used with find, same advantages as cpio, recursively descends directory tree without find; can move directory hierarchies; the most portable copy program among UNIX systems	Same disadvantages as cpio, must be used with multivol for multitape backups

3.3.2 File System Backup and Restore

The importance of establishing and following a file system backup plan is too often not appreciated until data is lost and cannot be recovered. Backing up a file system takes time. Trying to recover lost or damaged data from paper records and best-guess-work takes even more time. The value of an effective system backup plan lies in the ability to recover lost or damaged data easily and reliably.

The dump and restore commands are the recommended method for backing up entire file systems in a consistent manner. Dump supports "dump levels" which can be used to back up only those files which have changed since the previous dump.

The capability to copy selected directories and files to tape is provided by using the find(1) command with cpio(1). The directories and files are also read back to the hard disk by using the appropriate cpio command. Another command that can be used to archive files is tar(1). Both cpio and tar may be combined with the multivol(1) command to handle backups that exceed the capacity of a single tape. Note that backups made with the multivol command must be restored with the multivol command.

The backup plan can include any or all of these methods. This plan should be reevaluated as the use of the machine changes.

For details of further options and features for each of the backup and restore capabilities, refer to the appropriate sections of the Systems Administrator's Reference Manual.

3.3.3 Complete Backup

A complete backup of a file system (such as "/" or "/usr") may be obtained by performing a level 0 dump of the specific file system. It is necessary to do this periodically to ensure a complete backup of the file system exists.

Before beginning a backup session, the system should be taken down to single user mode. To do this, enter:

```
#telinit s
```

from the root user account. After a short while, the system will indicate that the run level is being changed to single user mode. This will insure that the file system is not altered while the backup is being performed.

To backup an entire file system enter:

```
#dump 0u  {filesystem}
```

where {filesystem} might be "/usr".

This command specifies that level 0 (all) of the specified file system is to be dumped to tape, and that the /etc/dumpdates file is to be updated (u) upon completion of the dump process.

If a tape has not been inserted, the system informs the user that it is unable to write the tape and asks if the user wants to try again.

3.3.4 Incremental Backup

An incremental dump backs up those files that have changed since the last lower-level dump. When dump is run with the u (update) and dump-level keys specified, the file /etc/dumpdates is checked to determine when the last lower-level dump occurred. All files with a later modification time in the specified file system are then backed up.

The following is an example of an /etc/dumpdates file:

```
/dev/usr          0 Sat Jun  4 07:46:11 1988
/dev/usr          1 Sat Jun 18 06:42:38 1988
/dev/root         1 Sat Jun 18 06:41:55 1988
/dev/root         0 Sat Jun  4 07:23:06 1988
```

If a level 1 dump of /usr is given, e.g.

```
# dump 1u /usr
```

Any file that has been modified since June 18, 1988 will be written to disk.

Note: Any file backed up with the dump command can be recovered only by use of the restore command (see Section 3.3.5).

3.3.5 restore Command

The restore(1M) command is used to read files from a tape created by dump and write them to disk. This command must be used **cautiously!** A complete restore of a file system could be made by entering the following command in the appropriate directory:

```
# restore r
```

This command would cause the files on the dump tape to be written to the current directory (and any subdirectories contained on the tape). If a file (or files) on the tape has the same name as a file (or files) in the current directory, the latter would be overwritten! This form of the command should be used only on a newly-created file system. A complete multilevel series of incremental dumps can restore the most current backups of the complete file system by beginning with a level 0 dump and then inserting progressively higher numbered dump tapes until the most recent dump is reached.

The restore command also permits selective restores by use of the x key. The name of the desired file(s) is supplied on the command line, for example:

```
# restore x /usr/danc/.cshrc
```

This command would restore the file /usr/danc.cshrc from the dump tape to the current directory.

A useful interface to the restore command is provided by the "-i", or interactive, option. Used with "-i", restore reads in the directory information from tape and then provides a standard shell-like interface that permits the user to move around the directory tree to extract files. Commands supported include ls, cd, pwd, and add (to add files to the extraction list).

Section 4
INSTALLING HARDWARE

Refer to Figures 4-1 through 4-3 to identify components within the computer.

4.1 REMOVING AND REPLACING ENCLOSURES

WARNING!

Make sure the system power switch is off and the power cord is disconnected from the wall outlet before attempting to remove the enclosure.

Removal Procedure: See Figure 4-1 for screw locations.

a. Snap off the front panel by pulling forward on the panel. There are snap off studs at the 4 corners of the panel.

b. Remove the 6 screws which hold each side. There are 3 screws along the side toward the front frame rail and 3 screws through the rear panel. Slide the side panel back about an inch and then lift upward and out.

c. Remove the 2 screws in the rear of the top cover. Slide back and lift the top cover off.

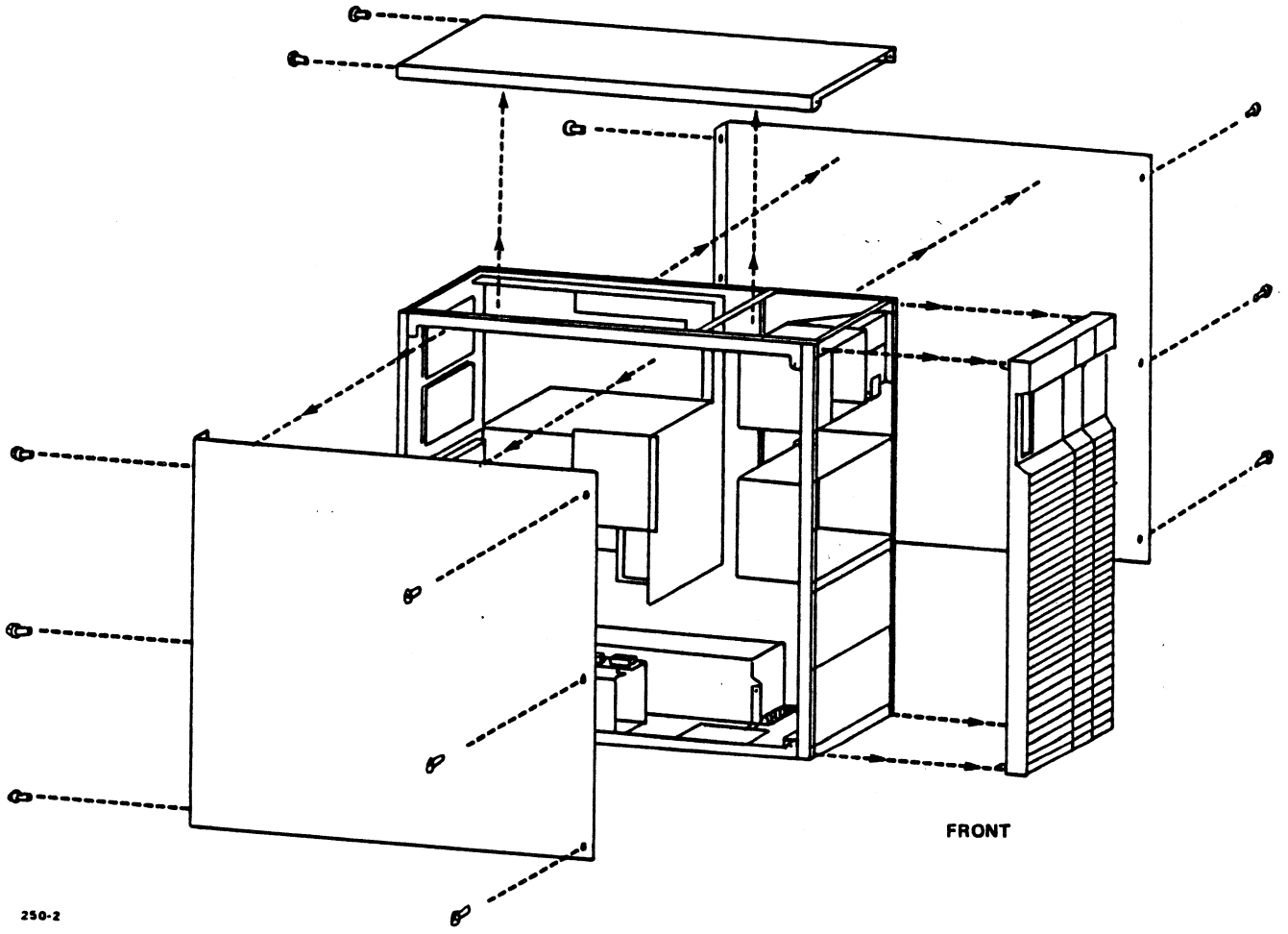


Figure 4-1. MARK 2000 Top Assembly

4.2 ADDING MEMORY BOARDS

There are 6 connectors designated JX300 through JX305 on the Motherboard. The connectors are located in the middle of the board (see Figure 4-2).

The restraining bracket over the PCBAs must be removed first, by removing the two screws and one nut.

JX300 is the first memory board connector. The boards are inserted with the components facing up. Each additional memory board is inserted in the next lower position.

There are no switch settings on the memory boards. The system will sense the presence of the boards during power-up self-test. Re-install the restraining bracket after the memory boards have been installed.

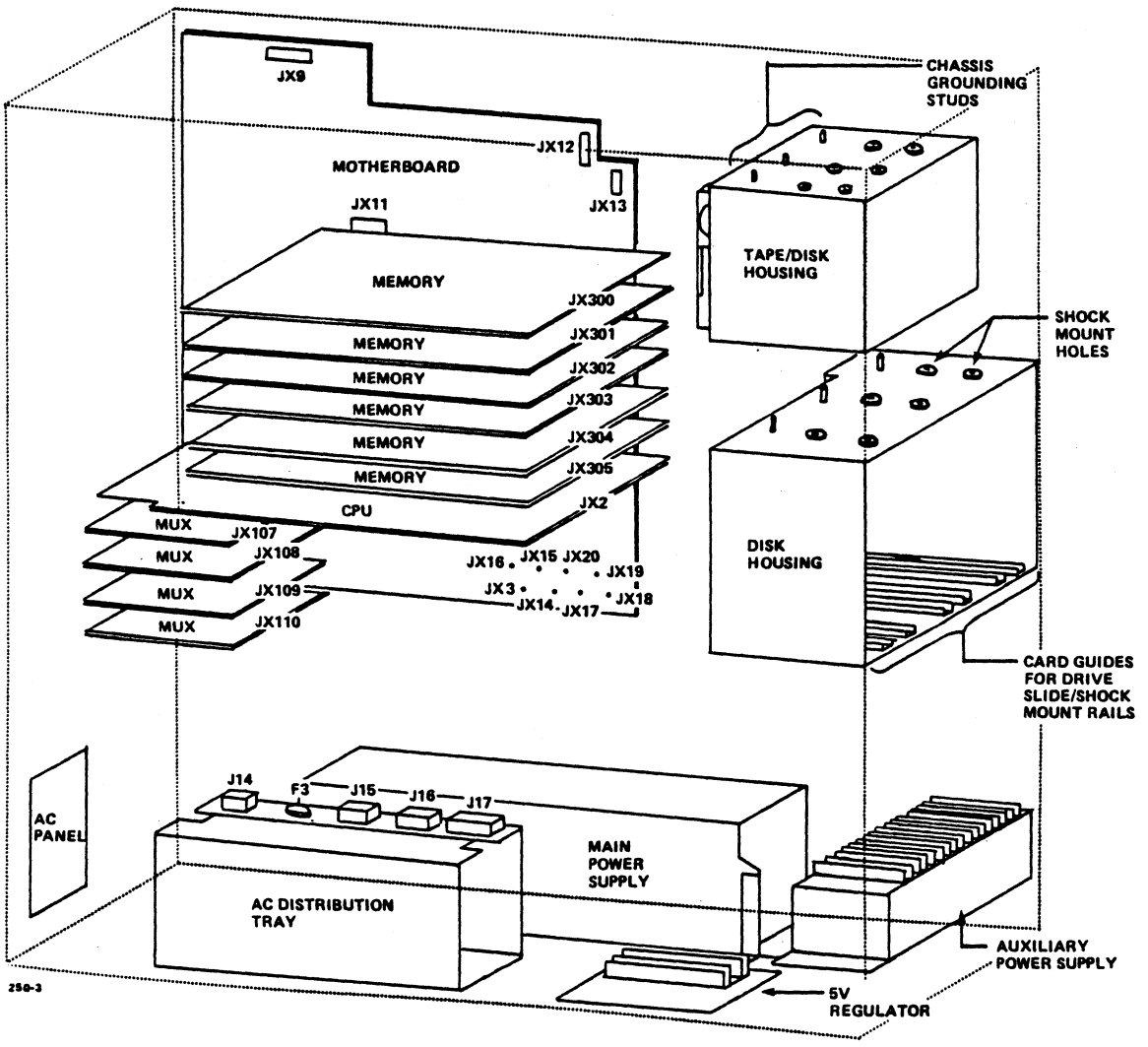


Figure 4-2. MARK 2000 Internal Components

4.3 ADDING DISK DRIVES OR REPLACING DISK DRIVES

4.3.1 Hardware Installation

The disk drive jumper settings, terminator locations, connector locations, and chassis grounds are different for each type of drive. Refer to Figures 4-3 through 4-10.

Drive Location (see Figure 4-3)

The drives are located at the front of the computer. Viewing from the front, drive 0 is on the far right side, and is the first drive. Moving to the left, the next drive is 1, then drive 2. Drive 3 is next to the cartridge tape drive.

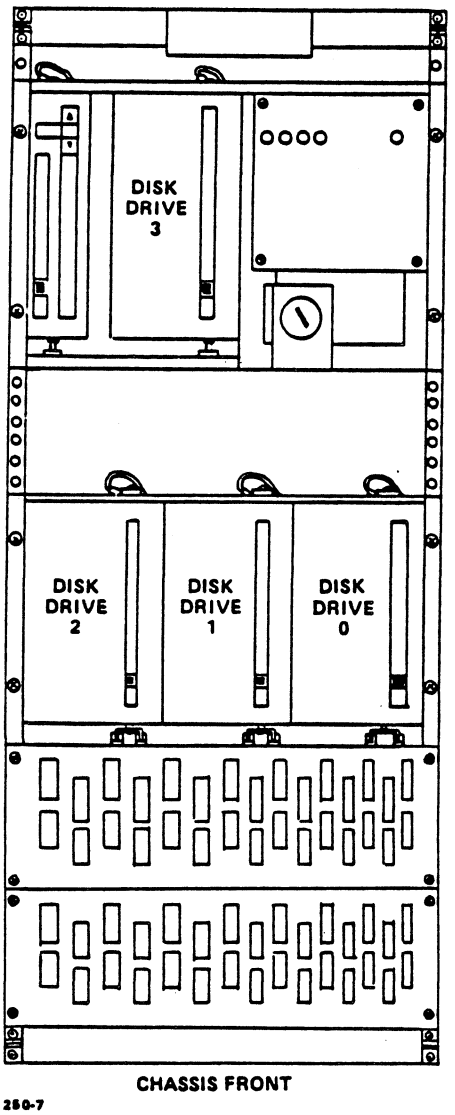


Figure 4-3. Location of Disk/Tape Drives

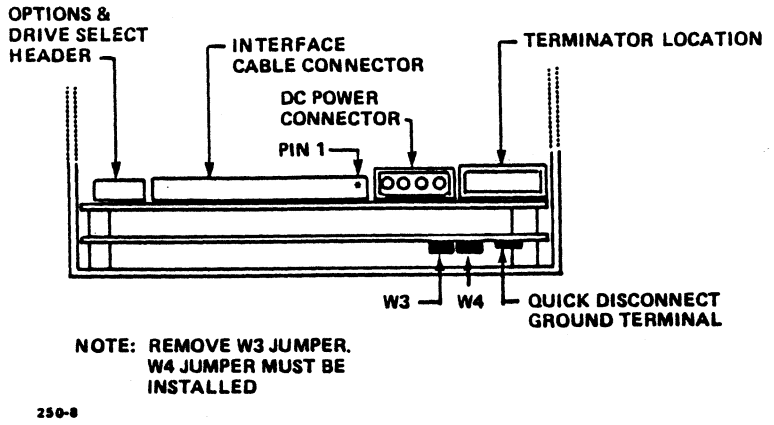


Figure 4-4. CDC Model 94161 Disk Drive

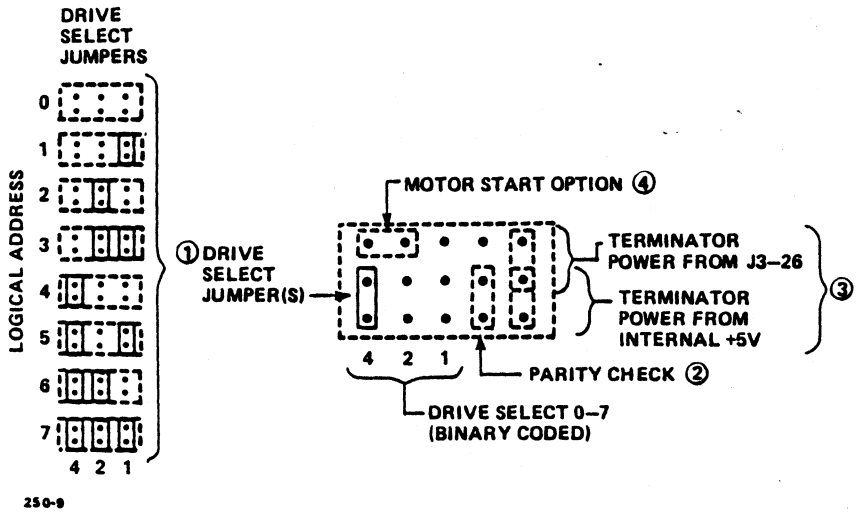


Figure 4-5. CDC Model 94161 Configuration Settings

1. Drive ID is binary coded jumper position (most significant bit on left). Jumper positions and associated drive logical address are shown at the left part of the figure.

2. Jumper plug installed means parity checking by the WREN III is enabled.

3. Jumper in lower position means terminator power (+5V) is from WREN III power connector. Jumper in upper position means terminator power is taken from interface cable.

   If unit is not terminated, TP jumper is to be left off.

   When a jumper is installed, it must only be in one of the positions shown. Two jumpers must not be installed in the "TP" positions at the same time.

4. Jumper plug installed enables the Motor Start Option. In this mode of operation, the drive will wait for a Start Unit command from the Host before starting the motor. If the jumper plug is not installed, the motor will start as soon as DC power is applied to the unit.

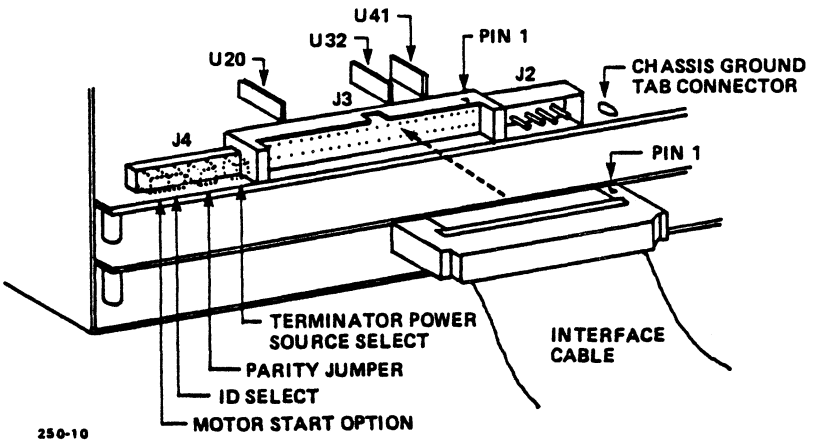


Figure 4-6. Early CDC Model 94161 Disk Drive

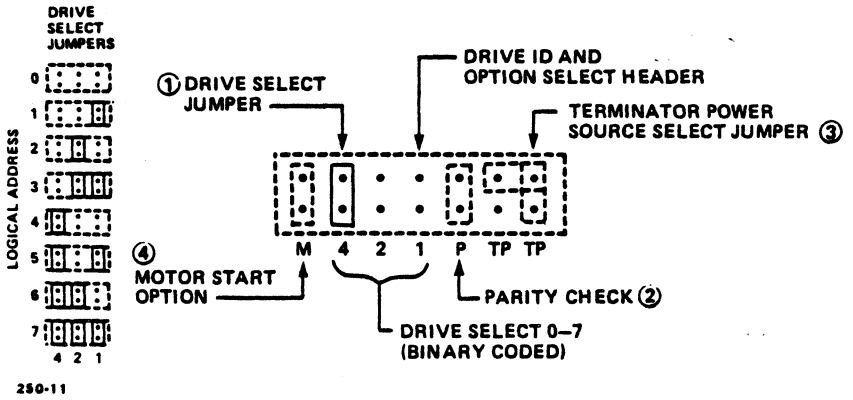


Figure 4-7. Early CDC Model 94161 Configuration Settings

1. Drive ID is binary coded by jumper position (most significant bit on left), ie., jumper in position 4 would be Drive ID 4, no jumpers means ID 0.

2. Jumper plug installed means parity checking by the WREN III is enabled.

3. Jumper in vertical position means terminator power (+5V) is from WREN III power connector. Jumper in horizontal position means terminator power is taken from interface cable. If interface cable is removed when terminators are in, the unit will not power-up, unless jumper is in vertical position.

   If unit is not to be terminated, remove the terminator resistor packs. Removal of terminator power source select jumper does not disconnect the terminator resistors from the circuit.

   When a jumper is installed, it must only be in one of the positions shown. Two jumpers must not be installed in the "TP" positions at the same time.

4. Jumper plug installed enables the Motor Start Option. In this mode of operation, the drive will wait for a Start Unit command from the host before starting the motor. If the jumper plug is not installed, the motor will start as soon as DC power is applied to the unit.

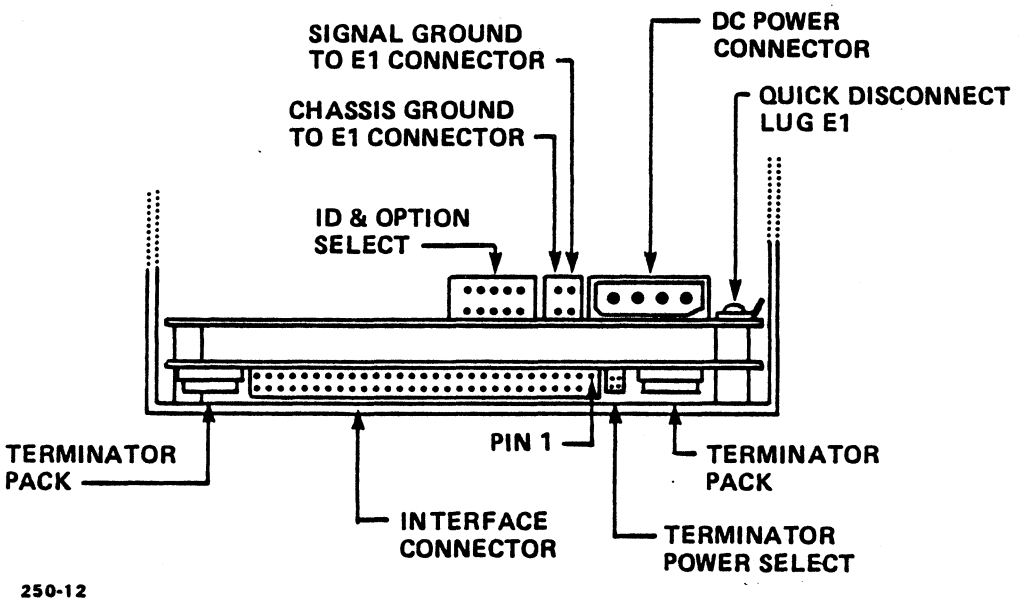


Figure 4-8. CDC Model 94171 Disk Drive

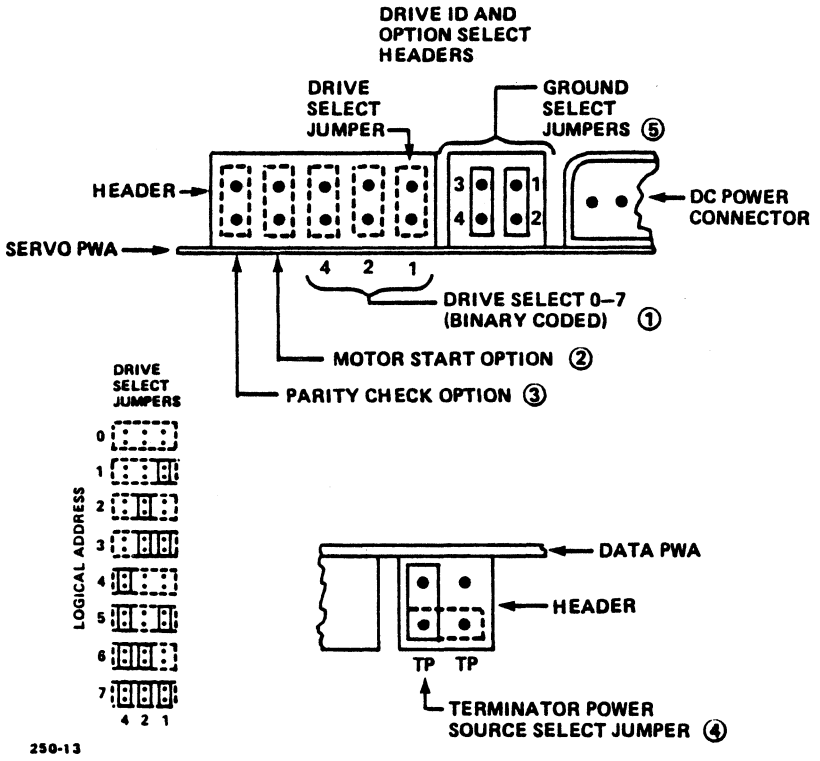


Figure 4-9. CDC Model 94171 Configuration Settings

1. Drive ID is binary coded jumper position (most significant bit on left), ie., jumper in position 0 would be Drive ID 1, no jumpers means ID 0.

2. Jumper plug installed enables the Motor Start Option. In this mode of operation, the drive will wait for a Start Unit Command from the Host before starting the Motor. If the jumper plug is not installed, the Motor will start as soon as DC power is applied to the unit.

3. Jumper plug installed means parity checking by the WREN IV is enabled.

4. If the unit is not to be terminated, remove terminator resistor DIPs. If installed, the "TP" jumper must be installed only in one of the two positions shown with dotted lines in Figure 4-9.

   Jumper in vertical position means terminator power (+5V) is from WREN IV power connector. Jumper in horizontal position means terminator power is taken from interface cable.

   If the drive is not terminated, the "TP" jumper need not be installed.

5. 3, 4 - Jumper installed; 1, 2 - jumper removed.

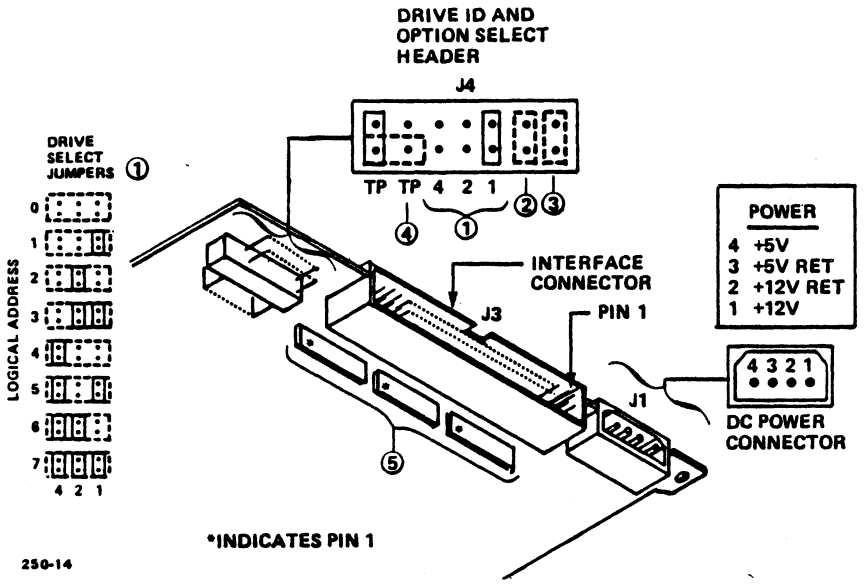


Figure 4-10. CDC 94181 Drive Configuration Settings

1. Drive ID is binary coded jumper position (most significant bit on left), ie., jumper in position 0 would be Drive ID 1, no jumpers means ID 0.

2. Jumper plug installed enables the Motor Start Option. In this mode of operation, the drive will wait for a Start Unit Command from the Host before starting the Motor. If the jumper plug is not installed, the Motor will start as soon as DC power is applied to the unit.

3. Jumper plug installed means parity checking by the WREN V is enabled.

4. If the unit is not to be terminated, remove terminator resistor DIPs [5]. If installed, the "TP" jumper must be installed only in one of the two positions shown with dotted lines in Figure 4-10.

   Jumper in vertical position means terminator power (+5V) is from WREN V power connector. Jumper in horizontal position means terminator power is taken from interface cable.

   If the drive is not terminated, the "TP" jumper need not be installed.

5. Removable terminator SIP resistor modules.

Accessing Drives:

To get access to the drives, the vent panel just below the cartridge tape drive must be removed. This will allow access to the top shock mounts of drives 0, 1, and 2, and the bottom shock mounts of drive 3. Drives 0, 1, and 2 have the top shock mounts attached to the housing; the bottom shock mounts are attached to a sliding metal rail. Drive 3 has both the top and bottom shock mounts attached to the housing.

Drive Settings:

Several items should be set on the disk before the drive is mounted into the computer. Refer Figures 4-4, 4-6, and 4-8.

SCSI ID - the drive ID should be set with jumpers according to its location 0, 1, 2 or 3.

Parity - parity checking is enabled.

Motor Start - jumper left off so motor starts with DC power.

Terminator Power - set such that terminator power is provided by the interface cable.

Terminator Resistors - the terminator resistors on the drives should be removed if they have not been removed already. The SCSI bus is terminated by the Motherboard and the SCSI plug-in terminator on the rear panel.

Chassis Ground - the disk drives chassis and signal grounds must be isolated. There are jumper options to configure the ground isolation.

Drive Mounting:

After the drive settings have been done, the shock-mounted slide rails are attached if the drive is for positions 0, 1, or 2. The drive is now placed in position and the SCSI Cable (flat cable with 50 pin connector) is plugged in. The SCSI Cable is the daisy-chained cable from the Motherboard that attaches to the disk drives and cartridge tape drive. Make sure the cable connector and the drive connector pin 1s are aligned.

The DC Power Cable (molex 4 pin) is plugged in. This cable comes from the Power Distribution Board that is mounted on the side opposite the Motherboard. The connectors for each drive is marked on the Power Distribution Board.

Chassis ground wire is provided with each drive and must be connected between the drive chassis (nearest shock mount screw) and the stud on the housing (near the shock mount holes).

The drive top shock mounts are aligned with the holes in the housing and attached to the drive. On drive 3, the bottom shock mounts are attached in a similar fashion.

A cable for the drive LED is included with the drive. This cable is attached to the 2-pin connector at the front of the drive. The other end of the LED Cable is attached to the designated position on the Front Control Panel which is located above the keyswitch.

Auxiliary Power Supply:

When more than 2 drives (disc or 8MM tape) are added to the computer, an Auxiliary Power Supply is required. The Auxiliary Power Supply is located below the disk housing, and attaches to a front panel with 2 screws.

There are 2 cables that come off the Auxiliary Power Supply. The cable with 5 wires connects to the TB1 Main Power Supply. The cable with 8 wires connects to the Power Distribution Board at J18 (see Figure 4-11).

Re-install vent panels after all the drives and cables have been installed.

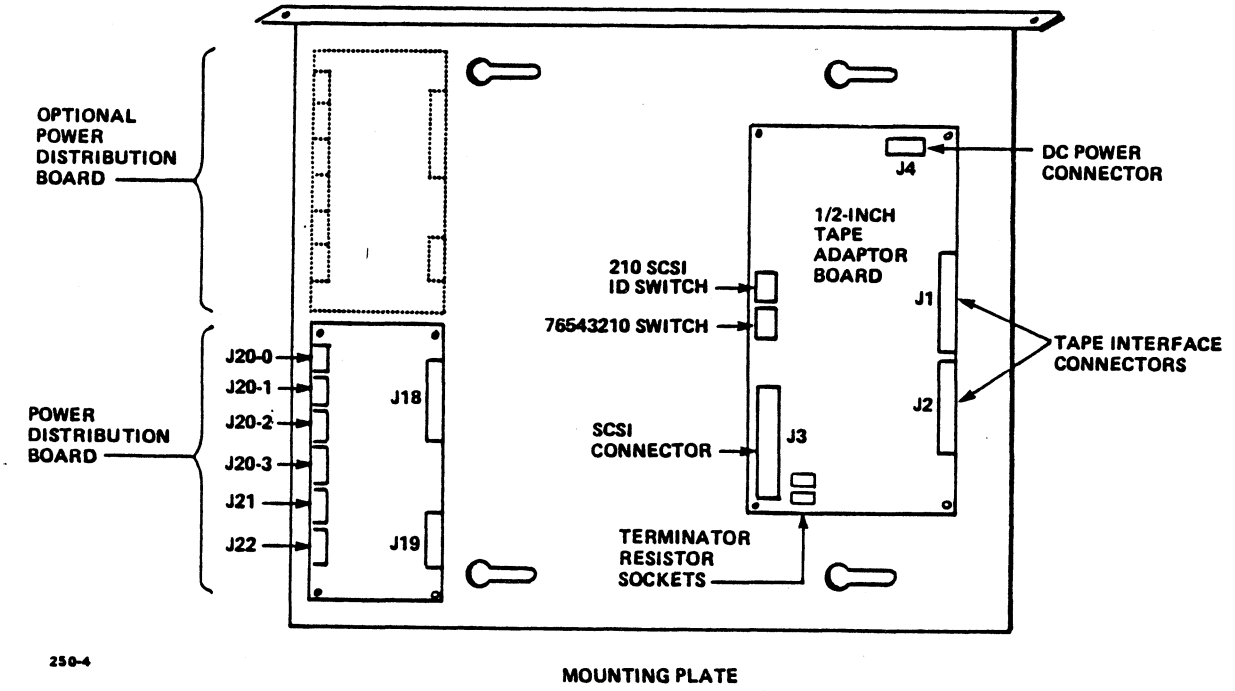


Figure 4-11. Mounting Plate

4.4 ADDING 8x4 MUX BOARDS AND 8tc TERMINAL CONCENTRATORS

8x4 Mux Board:
There are 4 connectors designated JX107 through JX110 on the Motherboard. The connectors are located at the lower left side of the board.

The 8x4 Mux boards have an attached bracket that screws into the back side of the mounting chassis. Boards can be plugged into any AT-bus connector. The boards are inserted with the components facing up.

There are address jumpers (S1,S2) on the 8x4 Mux Board. Each 8x4 Mux Board must be a different address.

Jumper S1	S2	Memory Address Range
in	in	D00000 - D7FFFF
out	in	D80000 - DFFFFF
in	out	E00000 - E7FFFF
out	out	E80000 - EFFFFF

See Figure 4-12 for the jumper location. The ports on the board are designated A, B, C, and D. The first port is "A" and is located nearest the plastic handle. The 8x4 Mux is connected to the 8tc through an 8-pin DIN connector, see Figure 4-13 for the cable connections.

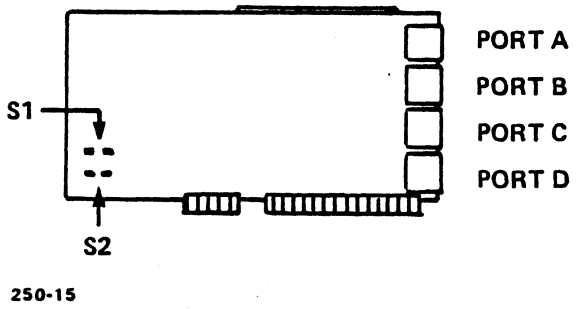
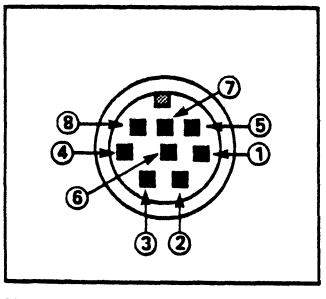


Figure 4-12. (8x4) Mux Board



8X4 MUX END	CONCENTRATOR END	SIGNAL NAME
1	1	DATA+
2	2	DATA-
3	3	+5
4	4	GND
5	5	GND
6	6	+5
7	7	+5
8	8	GND

Figure 4-13. Mux to 8t/c Connection

8tc Terminal Concentrator:

The 8tc Terminal Concentrator is placed outside the computer. A Mux-8tc Cable is used to connect the 8tc Terminal Concentrator to an 8x4 Mux Board. Up to 4 Terminal Concentrators can be connected to one Mux Board.

Serial devices such as video display terminals or printers are connected to the 8tc Terminal Concentrator using an 8tc-serial Cable. See Figure 4-14 for the wiring connections for either RJ11 4-pin or 6-pin.

Note: If the 8tc Terminal Concentrator is to be located more than 10 feet from the computer, a separate power supply is required. Also the Mux-8tc Cable must not have the +5 volt and ground signals connected when the external power supply is used.

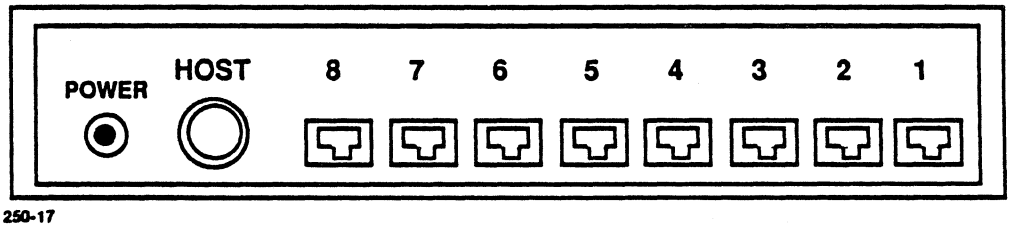
There are no switch settings for the 8tc Terminal Concentrator.

Note: If the 8tc Terminal Concentrator enclosure is required to be opened, the top can be removed by pressing on the side while pulling apart the bottom. This must be done for each side since there is a latch mechanism molded into the enclosure on each side. Connector locations are shown in Figure 4-15 for the 8 serial ports, one host port, and one power plug.

MODULAR JACK (4-PIN)	(6-PIN)	DB25	TERMINAL SIGNAL
	1	1	SHIELD GROUND
1	2	2	TRANSMIT DATA
2	3	3	RECEIVED DATA
3	4	7	CIRCUIT GROUND
4	5	4,5	RTS, CTS
	6	...	NO CONNECT
	...	6,20	DSR, DTR (CONNECTED)

250-18

Figure 4-14. 8t/c Port Wiring



250-17

Figure 4-15. 8t/c Backpanel

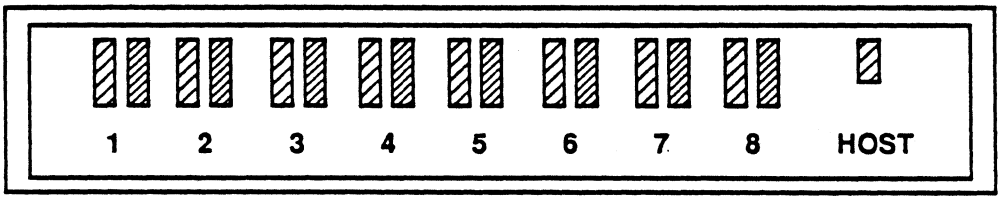
Front Panel:

There is one host LED and 16 port LEDs on the front panel. Each port has 2 LEDs, a yellow and a green. See Figure 4-16.

Host LED - green color.
- blinking if the unit has power but is not being polled from the host.
- steady-light when the host and the 8/tc are communicating.

Green Port LED - blinking or lit steadily indicates that the channel has Data Set Ready asserted.

Yellow Port LED - blinking when Receive Data is seen. This indication is independent of whether Data Set Ready is asserted.



250-10

Figure 4-16. 8t/c Front Panel

Connecting Serial Printers:

Printers as well as terminals and other RS232 interfaced serial devices can be connected to the 8 ports on the concentrator. Flow control must be handled by Xon/Xoff protocol.

Chassis Ground:

The chassis ground of the 8tc is isolated from the DC ground.

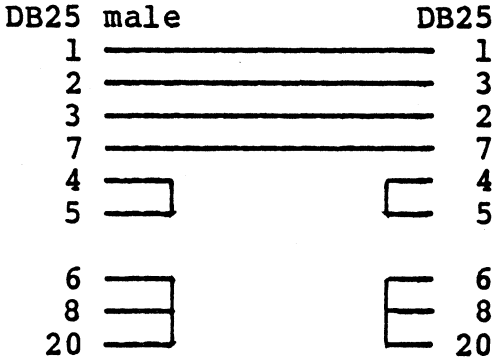
External Power Supply:

The connector marked "power" on the 8tc is for an external regulated +5VDC. This is used when the 8x4 Mux cable is greater than 10 feet. The power plug is a 2.5 millimeter center-pin connector with ground on the outer contact and +5VDC on the center contact.

4.5 ADDING VIDEO DISPLAY TERMINALS

Video Display Terminals (VDT) are connected to the RS232 ports of the Motherboard or from the 8tc Terminal Concentrator. The Motherboard ports use DB25 connectors; the 8tc Terminal Concentrators use RJ11 Connectors (see Figure 4-14).

Cable connections are as shown in the tables below.



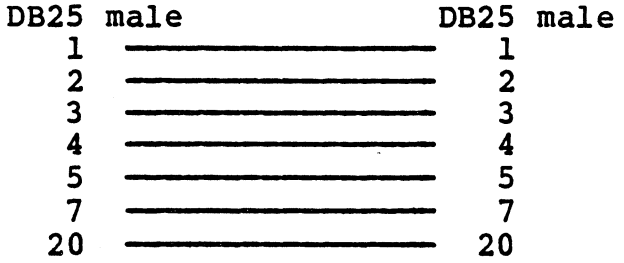
NOTE:
Set VDT to the following:
bits per character=8
Protocol=XON/XOFF
Parity=disabled
Mode=full duplex, character
Stop bits=1

4.6 ADDING PRINTERS

Printers, using Xon/Xoff flow control only, are connected to the RS232 ports of the Motherboard or the 8tc Terminal Concentrator. Refer to the diagrams in adding VDTs for cable connections and connector types.

4.7 ADDING MODEMS

Modems are added to RS232 ports 1 and 3 of the Motherboard for full modem control.



For modems that can operate with Xon/Xoff flow control, all RS232 ports of the Motherboard or the 8tc Terminal Concentrator ports can be used.

4.8 ADDING 1/2-INCH TAPE ADAPTER AND TAPE DRIVE

1/2-Inch Tape Adapter Board:

The Tape Adapter Board is mounted on the opposite side of the vertical plate holding the Motherboard. There are 4 standoffs to screw in the board. The board must be oriented with the two 50-pin connectors (J1, J2) towards the rear panel. The Peripheral Power Cable is connected from J4 of the Tape Adapter Board to J22 of the Power Distribution PCBA. The internal SCSI Cable is connected to J3 of the Tape Adapter Board.

Two identical cables supplied with the Tape Adapter Board, are connected from J1, J2 to the rear panel. The Plastic covers on the rear panel must be removed as the cable is attached to the rear panel. See Tables 4-3 and 4-4 for the pin outs.

There are two sockets at location 2A and 2B on the adapter board; there should not be any resistor terminators in these sockets. If there are any in these 2 sockets, remove them.

Switches:

There are two 8-position switches to select options. One is labeled 210SCSIID (1N location), and the other is labeled 76543210 (1K location). Set switches to the normal setting as indicated in the following tables. The remaining switches are not used and should remain closed.

Table 4-1. 76543210 Switch Setting Options for 1/2" Tape Adapter

Switch Label	Setting		Result
7	Open Closed		Disable arbitration Normal (enable arbitration)
6	Open Closed		Disable SCSI parity check Normal (enable SCSI parity check)
5,4	Closed Closed Open Open	Closed Open Closed Open	F880 Normal (MUX block size 64K) M890 (max block size 32K) M990 (max block size 32K) M990 (cache GCR 64K block size)
3,2	Closed Closed Open Open	Closed Open Closed Open	Normal (diagnostics) Loop on power-up routines Loop on RAM data dump Allow command execute after power-up failure
1,0	Closed Closed Open Open	Closed Open Closed Open	1 drive on controller, Normal 2 drives on controller 3 drives on controller 4 drives on controller

(Location 1k)

Table 4-2. 210SCSCIID Switch Settings for 1/2" Tape Adapter

Switch Label	Settings			Result
	2	1	0	
2,1,0	Open	Open	Open	0 SCSI ID
	Open	Open	Closed	1 SCSI ID
	Open	Closed	Open	2 SCSI ID
	Open	Closed	Closed	3 SCSI ID
	Closed	Open	Open	4 SCSI ID
	Closed	Open	Closed	5 SCSI ID Normal
	Closed	Closed	Open	6 SCSI ID
	Closed	Closed	Closed	7 SCSI ID
S	Open			Normal operation position
	Closed			Resets board and cycles thru diagnostic
The remaining switches are not used.				

(Location 1n)

Tape Drive:

The Tape Drive is connected to the computer with two 50-pin flat shielded ribbon cables. The tape interface connectors are on the computer rear panel located on the left and are the top 2 connectors, designated Tape J1 and Tape J2. On most drives the corresponding connector designations are P1 and P2. See Figure 4-3 for the 1/2" Tape Pertec Interface.

Two 50-pin connectors, J1 and J2, connects the controller to the tape drive. The following tables contain the pin assignments for J1 and J2. The first table is for J1, all odd numbered lines are connected to ground and pin 14 is not connected (nc).

TABLE 4-3 PERTEC 1/2" TAPE INTERFACE JI (1 of 2)

J1 CONNECTOR		
Pin Number	Signal Name	Direction (at controller)
2	FBUSY/	IN
4	LWD/	OUT
6	WD4/	OUT
8	GO/	OUT
10	WDO/	OUT
12	WD1/	OUT
14	nc	-
16	'ON-LINE/	OUT
18	REVERSE/	OUT
20	REWIND/	OUT
22	WDP/	OUT
24	WD7/	OUT
26	WD3/	OUT
28	WD6/	OUT
30	WD2/	OUT
32	WD5/	OUT
34	WRITE/	OUT
36	'LONGGAP/	OUT
38	EDIT/	OUT
40	ERASE/	OUT
42	WRITEFMK/	OUT
44	'THRESHOLD/	OUT
46	TADRO/	OUT
48	RD2/	IN
50	RD3/	IN

'Connected at the controller, but not used by the current firmware.

This table is for connector J2. Pin 26, GCR/ is connected at the controller but the firmware does not use the line. All pins not shown in the table below are connected to ground.

TABLE 4-3 PERTEC 1/2" TAPE INTERFACE J2 (2 of 2)

J2 CONNECTOR		
Pin Number	Signal Name	Direction (at controller)
1	RDP/	IN
2	RD0/	IN
3	RD1/	IN
4	LOAD POINT/	IN
6	RD4/	IN
8	RD7/	IN
10	RD6/	IN
12	HARD ERROR/	IN
14	FILEMARK/	IN
16	ID/	IN
18	TENABLE/	OUT
20	RD5/	IN
22	EOT/	IN
24	RWD UNLOAD/	OUT
26	'GCR/	IN
28	READY/	IN
30	REWINDING/	IN
32	PROTECTED/	IN
34	RSTR/	IN
36	WSTR/	IN
38	DBUSY/	IN
40	HSPEED STAT/	IN
42	COR ERROR/	IN
44	ON LINE/	IN
46	TADR1/	OUT
48	EFAD/	-
50	SEL HSPEED/	OUT

'Connected at the controller, but not used by the current firmware.

4.9 REPLACING CPU BOARD

The CPU Board is plugged into the Motherboard at location JX2. This is located next to the 6 memory board connectors and is the long connector with 150 pins.

The Restraining Bracket for the PCBAs must be removed first by unscrewing the two screws and one nut.

The board is inserted into the Motherboard connector with the components facing up. A bracket is attached to the CPU board and this is screwed into the chassis similar to the AT-board mounting.

The Restraining Bracket is re-installed after replacing the CPU Board. Make sure the card guide on the Restraining Bracket aligns properly against the cards to prevent movement.

4.10 REPLACING THE MOTHERBOARD

Location:

The Motherboard is mounted vertically against the chassis.

Disconnecting cables:

All I/O cables must be disconnected from the Motherboard as well as from the 8x4 Mux Boards. All the boards plugged into the Motherboard must be removed. The SCSI Cable must be disconnected from the Motherboard; the connector JX9 is located at the top edge of the board. The Main DC Power to the Motherboard must be disconnected; a total of 8 connections are located at JX3, 14, 15, 16, 17, 18, 19, 20. The Motherboard Auxiliary Power must be disconnected from JX11; this is a 6-pin molex connector located just left of the first memory board. The cable at JX13 for the front panel must be removed. The cable at JX12 for the Keyswitch must be removed.

Removing the Motherboard:

There are 4 screws holding the Motherboard and chassis to a vertical Mounting Plate (see Figure 4-17). Loosen the 4 screws several turns and slide the Motherboard toward the front of the computer about 2 inches. The Motherboard and chassis can now be lifted slightly and pulled away from vertical mounting plate. Place the chassis assembly on a flat work surface. There are 8 screws that hold the Motherboard in place. There are also 2 screws holding each I/O connector to the chassis that must be removed. Remove the screws and lift the Motherboard out.

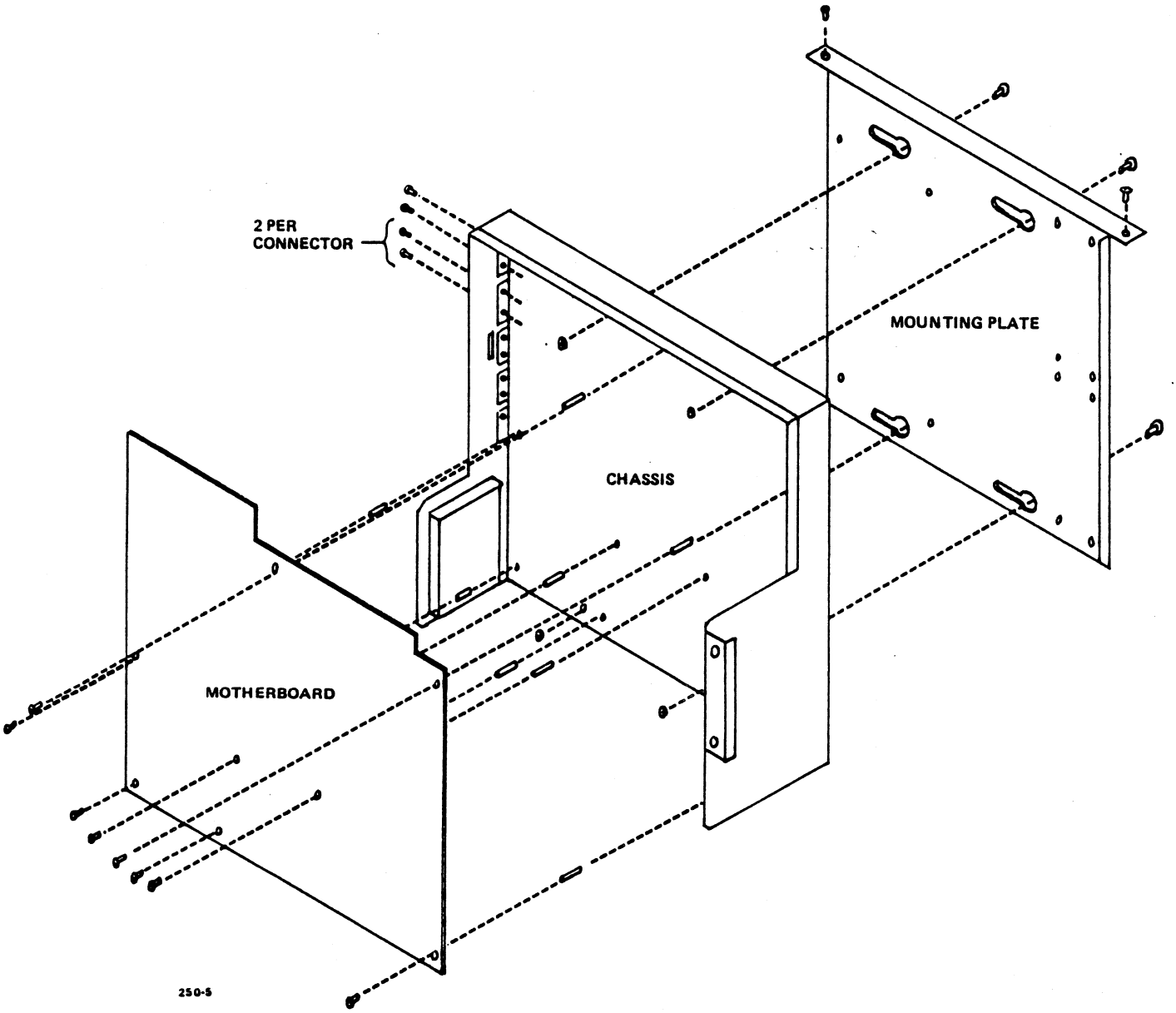


Figure 4-17. Motherboard/Chassis Mounting

Jumper settings:
There are jumpers located adjacent to the CPU Board Connector.

Verify the following jumper settings:

1. Check that pin 4 and pin 13 are not connected on JPD8.
2. Check that pin 5 and pin 12 are not connected on JPF9.
3. Check that pin 4 of JPD8 is connected to pin 12 of JPF9.

Installing the Motherboard:
Reverse the procedure outlined above to re-install the board. Care should be taken as boards and cables are re-installed to assure good connections and to avoid any damage to the connectors or pins.

NOTE: There is one long threaded rod that is used to hold the restraining bracket for the PCBAs. This should be removed when removing the Motherboard.

4.11 REPLACING A CARTRIDGE TAPE DRIVE

The Cartridge Tape Drive is located at the top left-side of the computer.

Drive Removal:

Remove the vent panel just below the tape drive. Remove the 4 screws and shock mounts on the tape drive; there are 2 shock mounts on each side of the tape drive. Remove the chassis ground wire that is attached to the drive housing. Disconnect the SCSI flat ribbon cable from the drive. Disconnect the DC Power Cable from the drive. Slide the tape drive forward and out.

Setting Jumpers: See Figures 4-18 and 4-19

Before installing the drive, the jumper settings must be verified. The jumper block is located just below the DC power connector. The SCSI ID must be set to ID=6. The Disconnect Size is set to 16K. Parity Enable jumper must be in place.

Installing the Drive:

Use procedures above in reverse order to install the replacement drive.

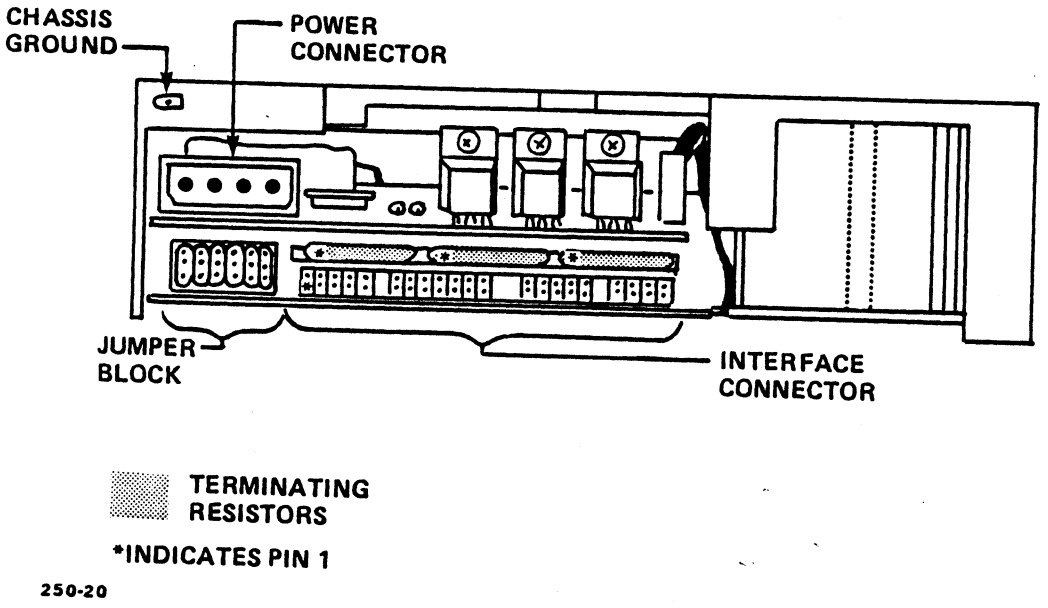


Figure 18. 1/4-Inch Cartridge Tape Rear View

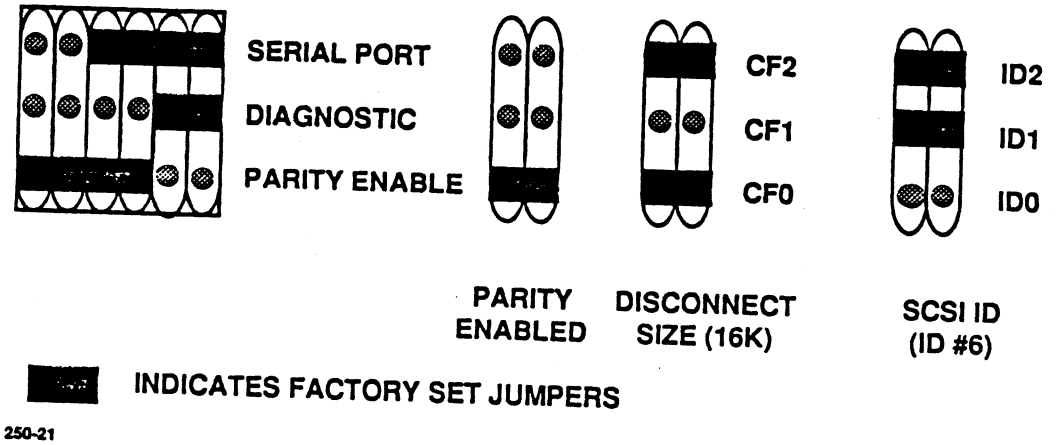


Figure 19. 1/4-Inch Cartridge Tape Jumper Block

4.12 INSTALLING AN 8mm HELICAL SCAN TAPE DRIVE

The 8mm Helical Scan Tape Drive is placed next to the Cartridge Tape Drive.

Install the 4 metal 1/4-inch standoffs on the drive. Slide the drive into the housing slot and secure the standoffs to the housing chassis.

Connect the SCSI flat ribbon cable and the DC Power cable. Attach the chassis ground wire to the housing chassis. Note: any of the disk drive DC power cables can be used to power the 8mm tape drive.

The following jumper definitions apply only to the MX card of the Helical Scan Tape Drive. The MX card is used to set the default (reset) conditions of the following options. The MX card is the second board from the rear of the drive. There are typically 5 screws holding the cover of the drive.

JUMPER CONFIGURATION

J1	L--M	Bypass Memory Test, 8 Second Power Up
	M--R	Run Memory Test, 65 Second Power Up
J2	L--M	Parity Enable
	M--R	Parity Disable
J3	L--M	Even Byte Disconnect
	M--R	Odd or Even Byte Disconnect
J4	L--M	No Busy Enable
	M--R	Report Busy Status
J5	L--M	P6 Cartridge Type - Domestic
	M--R	P5 Cartridge Type - European
J6	L--M	Reserved for Future Use
J7	L--M	Reserved for Future Use
J8	L--M	Fixed Block Mode on Power Up
	M--R	Variable Block Mode on Power Up

See Figure 4-20 for the location of these jumpers, and there settings for the MARK 2000. There is a 3 position dip switch at the rear of the drive; switch #1 and #2 should be OFF and switch #3 should be ON.

There is a write-protect tab on the tape cartridge (see Figure 4-21).

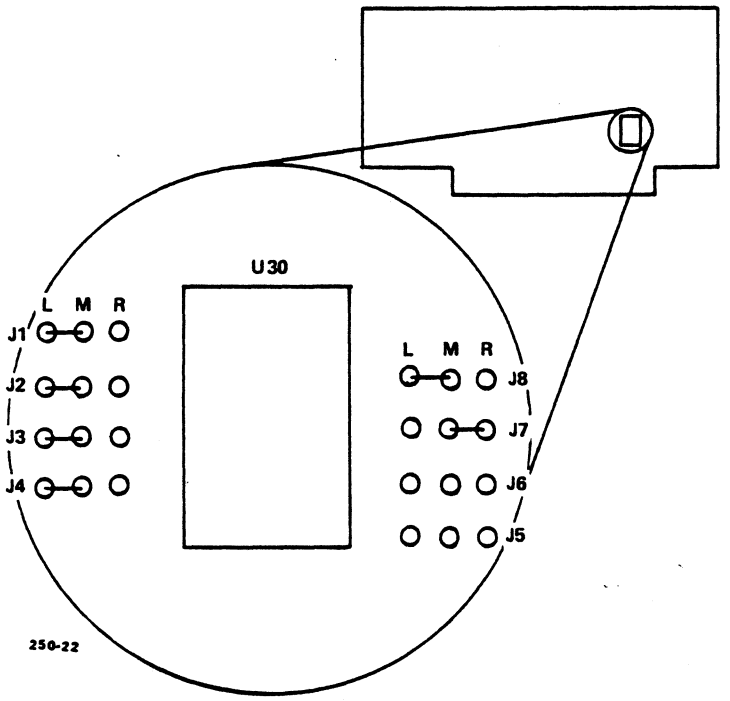


Figure 4-20. MX Jumper Location, Helical Scan Tape Drive

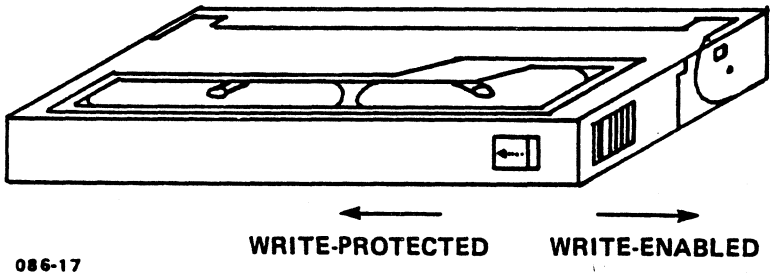


Figure 4-21. Write-Protecting an 8mm Tape Cartridge

4.13 REPLACING THE MAIN POWER SUPPLY

WARNING!

Disconnect the AC power cord before beginning to remove cables and wires connected from the main power supply.

Refer to Appendix B for the System Wiring Diagram.

a. Remove the wires from terminal block TB1, this cable comes from the AC panel with power switch.

b. Remove all wires connected to terminal block TB2.

c. Remove all wires connected to the +5VDC and ground terminals E1 and E2.

d. Disconnect the cable plugged into the power sense connector J6.

e. Remove the 3 screws from the mounting brackets holding the power supply to the base of the frame.

f. Remove the power supply and install the replacement supply. Make sure the replacement supply has the same AC power input rating (115VAC or 220VAC)

g. Install the replacement supply in the reverse order of the above steps.

4.14 ETHERNET

The Ethernet port connections are as show below. DB15 male connectors are used for the cable.

DB15 MALE	DB15 MALE	
1	1	Shield
2	2	Collision Presence +
3	3	Transmit +
4	4	Ground
5	5	Receive +
6	6	Power Return (Ground)
7	7	Reserved
8	8	Reserved
9	9	Collision Presence -
10	10	Transmit -
11	11	Reserved
12	12	Receive -
13	13	Power +12v (fused)
14	14	Reserved
15	15	Reserved

Cable between MARK 2000 and the standard Ethernet transceiver.

Section 5
MAINTENANCE

The computer system requires periodic maintenance.

5.1 CLEANING THE INTERIOR

WARNING!

Make sure the system power is off and the power cord is disconnected from the wall outlet before attempting to clean.

Using an air hose or vacuum, clean any dust that may have collected near the vents and fans. There are vents in the rear where the fans mount. There are vents in the front bezel as well as a filter. There is a fan and several vents on the main power supply located at the bottom of the computer. Any other collection of dust within the computer should be cleaned.

5.2 CARING FOR TAPES AND CARTRIDGES

Tapes are media on which computer programs and information are stored. Tapes, and the cartridges in which they are housed, are sensitive and need to be handled carefully to ensure that the tape drive can read and write reliably. This section contains information and instructions about the proper care and handling of 1/4-inch streaming tape and 8mm tape and their cartridges.

5.2.1 1/4-Inch Tape Cartridge

With proper care, the typical life of a tape cartridge is up to 5000 track passes. The following information and instructions will help to ensure reliable operation and long life.

5.2.1.1 Specifications

The 1/4-inch streaming tape (QIC-02) drive has 125/150MB capacity. Table 5-1 lists order numbers, cartridge types, and the functions supported for these tape capacities.

TABLE 5-1. CARTRIDGE TYPES AND FUNCTIONS

Drive Capacity	POINT 4 Order No.	Cartridge Type	Functions Supported
125/150MB	QCT0270	DC600XTD	Read/Write QIC-150 Read only QIC-120 Read only QIC-24 Read only QIC-11
	QCT0250	DC600A	Read/Write QIC-120 Read only QIC-24 Read only QIC-11
	QCT0200	DC300XLP	Read only QIC-24 Read only QIC-11

5.2.1.2 Labeling

Each tape cartridge should be labeled with the date of creation, a description of the tape contents, and any other relevant information. The label should be placed on the plastic top of the cartridge and not on the metal bottom plate.

5.2.1.3 Storage

When not in use, the 1/4-inch tape cartridge should always be returned to its protective case and stored in a cool, dry place.

Before using, allow the tape cartridge to acclimate to the operating environment for 24 hours or for the amount of time it has been exposed to a dissimilar environment, whichever is less.

5.2.1.4 Handling a 1/4-Inch Tape Cartridge

Incorrect handling can adversely affect tape performance.

Avoid the following: touching the recording media, having the cartridge close to magnetic fields or magnetic materials, dust, sunlight, and moisture.

Faulty tape tension can cause reading problems. To help ensure correct tape tension, retension the tape before each use.

5.2.1.5 Write-Protecting a 1/4-Inch Tape Cartridge

The 1/4-inch tape cartridge has a write-protect plug, which can be rotated before the cartridge is inserted into the drive to either permit or prohibit writing to the tape. If the user is allowed to read from and write to the tape, use a screwdriver to rotate the plug so that the arrow points away from SAFE. If the user is allowed only to read from the tape, make certain the arrow of the write protect plug points to SAFE (see Figure 5-1).

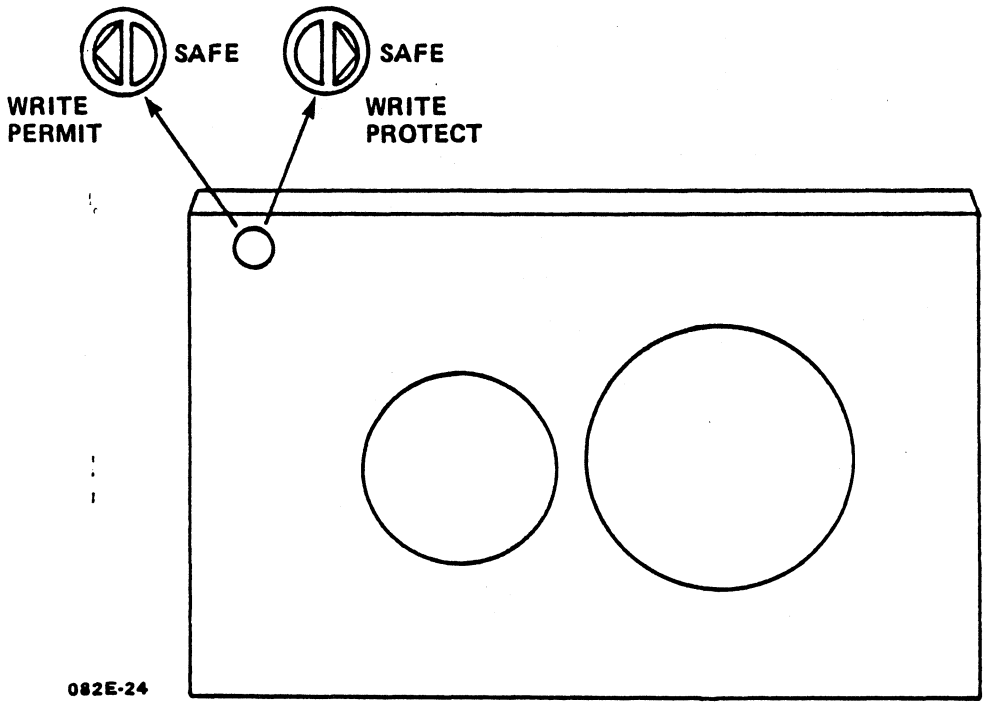


Figure 5-1. Write-Protecting a 1/4-Inch Tape Cartridge

5.2.1.6 Inserting and Removing a 1/4-Inch Tape Cartridge

Once the tape cartridge has been removed from its protective case and the write-protect plug set as appropriate, the tape cartridge can be inserted in the drive as follows (see Figure 5-2):

1. Insert the cartridge into the tape drive opening and push it in until it stops.

   The drive is designed so that the tape cartridge can be inserted only in the correct way.

2. Move the loading lever toward the cartridge to lock the cartridge into operating position.

To remove the tape cartridge from the tape drive:

1. Push the loading lever away from the cartridge.
2. When the tape cartridge ejects, remove it from the tape drive.
3. Return the tape cartridge to its protective case.

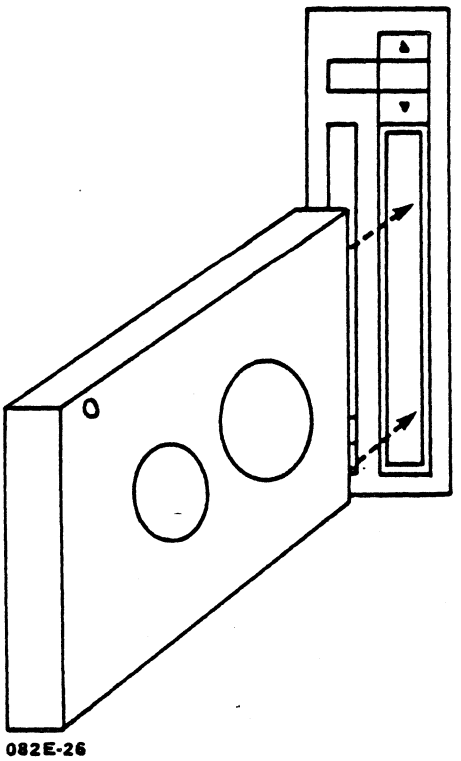


Figure 5-2. Inserting a 1/4-Inch Tape Cartridge

5.2.1.7 Cleaning a 1/4-Inch Tape Drive

To prevent tape problems from occurring, establish a regular cleaning routine. The read/write/erase heads of the tape drive should be cleaned after each initial pass of a new tape cartridge and after every eight hours of normal use. The sensor openings and cartridge cavity should be cleaned whenever dust or dirt are visible. **Before doing any cleaning, turn off power to the tape drive.**

Clean heads with a lint free swab and Freon TF. Do not use solutions that contain alcohol or water. Cartridge cleaning kits can be ordered from POINT 4.

Clean the sensor heads and cartridge cavity by cvarefully blowing out dust or dirt with low pressure air from an aerosol can of dry air.

5.2.2 8mm Tape Cartridge

5.2.2.1 Specifications

The 8mm helical scan tape drive uses standard 8mm tape cartridges (3.7 x 2.5 x 0.6 inches).

5.2.2.2 Labeling

Each tape cartridge should be labeled with the date of creation, a description of the tape contents, and any other relevant information. The label should be placed on the plastic top of the cartridge.

5.2.2.3 Storage

When not in use the 8mm tape cartridge should be returned to its protective case and stored vertically in a cool, dry place.

Before using, allow the tape cartridge to acclimate to the operating environment for 24 hours or for the amount of time it has been exposed to a dissimilar environment, whichever is less.

5.2.2.4 Handling an 8mm Tape Cartridge

Incorrect handling can adversely affect tape performance.

Avoid the following: touching the recording media, having the cartridge close to magnetic fields and magnetic materials, dust, sunlight, and moisture.

5.2.2.5 Write-Protecting an 8mm Tape Cartridge

The tape cartridge has a write-protect tab, which can be set before the cartridge is inserted into the drive to either permit or prohibit writing to tape. If the user is allowed to read from and write to the tape, make certain the tab is moved to the right so that the red portion of the tab is not visible. If the user is allowed only to read from the tape, use a ball-point pen or other similar instrument to move the tab to the left so that the red portion of the tab is visible (see Figure 5-3).

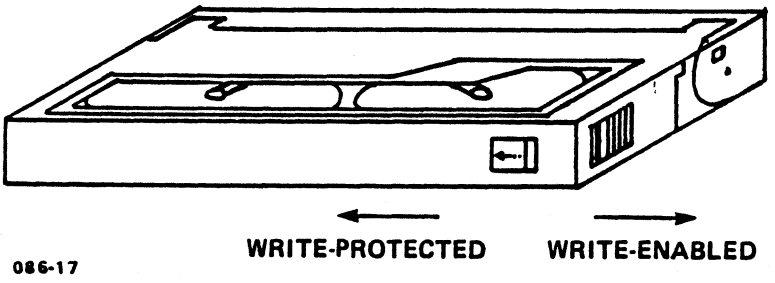


Figure 5-3. Write-Protecting an 8mm Tape Cartridge

5.2.2.6 Inserting and Removing an 8mm Tape Cartridge

Once the tape cartridge has been removed from its protective case and the write-protect tab set as appropriate, the tape cartridge can be inserted in the tape drive as follows (see Figure 5-4):

1. If the tape drive door is closed, press the UNLOAD button to open the drive door.

2. Insert the tape cartridge with the label side facing up and the cartridge lid facing the drive.

3. Gently close the drive door; the tape cartridge automatically loads.

To remove the tape cartridge from the tape drive:

1. Press the UNLOAD button on the drive.*

2. When the tape cartridge ejects, remove it from the tape drive.

3. Return the tape cartridge to its protective case.

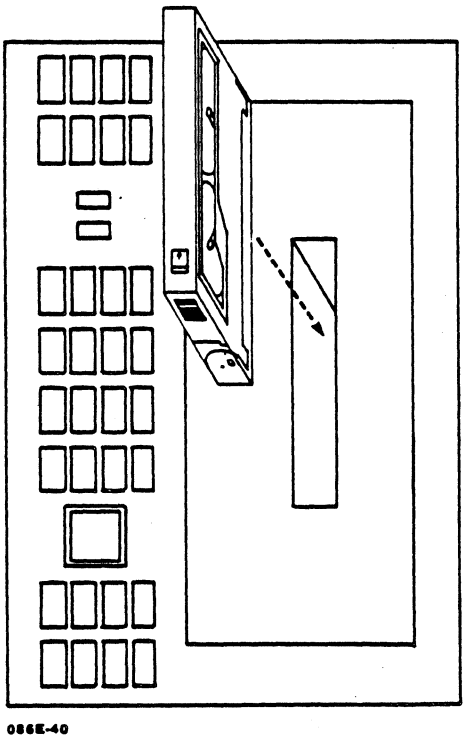


Figure 5-4. Inserting the 8mm Tape Cartridge

*If power is lost, the tape cartridge will not eject.

5.3 FUSES

Main AC fuse:

This is located on the rear panel at the bottom next to the AC cord plug. The fuse holders are marked F1 and F2. Fuse F1 is for AC Power. Fuse F2 is not used in this system.

Ethernet 12V Fuse:

This is located on the Motherboard at location A35B behind the first DB25 Connector next to the Ethernet connector. It looks like a resistor and is about 1/4" long. The fuse rating is 1A-125V.

SCSI Terminator Fuse:

This is located on the Motherboard at location C41 next to the SCSI connector JX9. It looks like a resistor and is about 1/4" long. The fuse rating is 1A-125V.

Auxiliary Power Supply Fuse:

This fuse is located on the Auxiliary Power Supply in a fuse-clip-holder.

Note: There is a fuse in the AC Distribution Tray designated F3 but this circuitry is associated with J14 and is not used in this system.

5.4 CLOCK/CALENDAR CHIP WITH BATTERY

There is a Clock/Calendar chip with a built-in battery on the Motherboard located at M29. This is a socketed device. The part is the Mostek MK48T02.

Section 6
TROUBLESHOOTING

6.1 TROUBLESHOOTING GUIDE

6.2 POWER-ON DIAGNOSTIC

The Power-on Diagnostic checks the integrity of the Point 4 MARK 2000 system. The Power-on diagnostics are activated whenever the system is powered on, or the system is RESET via the keyswitch. When either of the above actions occurs, the following sequence of events is initiated:

1. The CPU Cause and Status registers are initialized.

2. The bootmode variable is checked for **"d"**; if a **"d"** does not exist, the Power-on diagnostics are run and system memory is cleared. If it does exist, the diagnostics are not run and memory is not cleared.

3. The Power-on diagnostics are executed.

These diagnostics are go, no-go tests. They start testing from the level of IC devices and system registers on upward. The two LED blocks on the Motherboard are used by the test. The tests display their own unique identification code on the LEDs, which consist of at least two LEDs on, so they are distinguishable from other tests. If the test finds an error, it flashes the error code using the LEDs. The LED is one vertically oriented with the least significant bit (LSD) at the top.

The following pages give a brief test description and LED messages from the Power-on diagnostic.

LEDs

This test does a walking ones pattern on the LEDs and verifies that each can be turned on independently.

This test does not return a message.

Duart

This test transmits 20 different data patterns and verifies that the data received is correct. The Duart can fail in one of three ways: 1 - if the data transmitted is not the same as the data received; 2 - if timeout occurs while waiting for the transmitter to become ready; 3 - if timeout occurs while waiting for the receiver to become ready.

If the Duart fails, the motherboard LEDs will flash in the following configuration:

MSB ¦ ¦ ¦ ¦ ¦ ¦ ¦ ¦*¦*¦ LSB

Banner

The module displays the following message on the console:

```
RUNNING POWER-ON DIAGNOSTIC...
```

Cache1

This test checks the following parts of the cache: that the cache is mapping correctly; that the processor accesses either main memory or cache memory depending on the location of the data to be accessed; that writes do write to memory; and that the valid bit logic is correct.

If this test fails, the following message is displayed:

```
FAILED Cache Test #1
```

The LEDs flash the following:

MSB ¦ ¦ ¦ ¦ ¦ ¦ ¦*¦ ¦*¦ LSB

Cache2

This test checks the following items: that the instruction cache is loaded on instruction fetches; that the instruction cache is utilized; and that the system executes one instruction per clock when the instruction cache is valid.

If the test fails, the following message is displayed:

FAILED Cache Test #2

The LEDs flash the following:

MSB ! ! ! ! ! ! !*!*! ! LSB

Cache3

This test does a modified algorithmic test sequence on the data cache checking for address and data stuck-at faults.

If the test fails, the following message is displayed:

FAILED Data Cache MATS+ Test

The LEDs flash the following:

MSB ! ! ! ! ! ! !*!*!*! LSB

Cache4

This diagnostic does a modified algorithmic test sequence on the instruction cache checking for address and data struck-at faults.

If the test fails, the following message is displayed:

FAILED Instruction Cache MATS+ Test

The LEDs flash the following:

MSB ! ! ! ! ! !*! ! !*! LSB

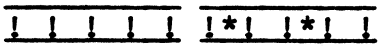
IdProm

This test verifies the checksum in the ID PROM.

If the test fails, the following message is displayed:

    FAILED ID PROM Test

The LEDs flash the following:



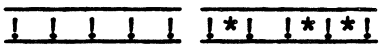
WB

This diagnostic verifies the byte, half-word, tri-byte, and word write operations, using the R2020 Write Buffers.

If the test fails, the following message is displayed:

    FAILED Write Buffer Test

The LEDs flash the following:



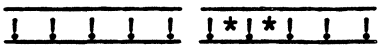
Memory

Using the memory cache, this diagnostic determines the system memory size and checks memory by looking for addressing and data stuck-at faults.

If the test fails, the following message is displayed:

    FAILED Memory Test

The LEDs flash the following:



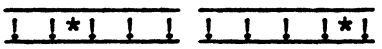
Scr

This diagnostic checks the writable/readable bits in the System Configuration Register.

If the test fails, the following message is displayed:

FAILED SCR Test

The LEDs flash the following:



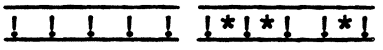
VM

This diagnostic checks the R2000 Translation Lookaside Buffer (TLB) for the following: that all undefined bits are read back as zero; that all TLB slots respond upon address matches; that the virtual and physical addresses match; that the TBL entry is marked as valid when the valid bit is not set and an exception occurs; that page modification can be controlled; that the global bit causes the PID match requirement to be ignored for valid translation; that the R2000 accesses memory when the non-cacheable bit is set.

If the test fails, the following message is displayed:

FAILED TLB Test

The LEDs flash the following:



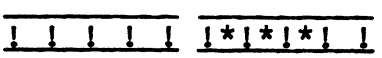
Allexc

This diagnostic tests exception handling. It tests both the generation of an exception condition and the handling of the exception.

If the test fails, the following message is displayed:

FAILED All Exception Test

The LEDs flash the following:



Parity

This diagnostic causes a byte parity error. The test then reads the memory location and checks if the data was detected correctly. The test conditions are: that no parity error is detected if parity checking is disabled; that if parity is enabled and an error is detected reading from the bad location, an interrupt occurs and the error information is put in the Fault Address Register and the Fault ID Register.

If the test fails, the following message is displayed:

```
FAILED Parity Test
```

The LEDs flash the following:

```
MSB ¦ ¦*¦ ¦ ¦ ¦ ¦ ¦*¦ ¦ LSB
```

NVram

This diagnostic does a non-destructive write/read test of the non-volatile system RAM.

If the test fails, one or both of the following messages are displayed:

```
FAILED NVRAM Test
FAILED Battery TEST
```

The LEDs flash the following:

```
MSB ¦ ¦ ¦ ¦ ¦ ¦*¦*¦*¦*¦ LSB
```

Timers

This diagnostic checks the counter/timer and the time-of-day clock devices.

If the test fails, one or both of the following messages are displayed:

```
FAILED 8254 Timer Test
FAILED Time-of-Day Clock Test
```

The LEDs flash the following:

```
          Timer                                  Time-of-Day
MSB ¦ ¦ ¦ ¦*¦  ¦ ¦ ¦ ¦*¦ LSB MSB ¦ ¦ ¦ ¦*¦  ¦ ¦ ¦*¦ ¦ LSB
```

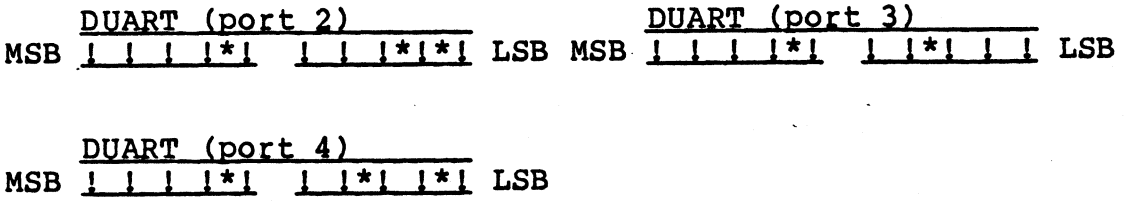
Duarts

This diagnostic is similar to the **Duart** test, except that it checks three DUART ports at various baud rates. The test is aborted after the first reported error.

If any of the tests fail one or more of the following messages are displayed:

```
FAILED DUART 1 Channel B Test
FAILED DUART 2 Channel A Test
FAILED DUART 2 Channel B Test
```

The LEDs flash the following:



Imr

This diagnostic does a write/read test of the Interrupt Mask Register.

If the test fails, the following message is displayed:

```
FAILED IMR TEST
```

The LEDs flash the following:

Fp1 and Fp2

This diagnostic verifies the basic floating point operations of the Floating Point Accelerator.

If the Fp1 test fails, the following message is displayed:

```
FAILED FP TEST #1
```

The LEDs flash the following:

```
MSB ¦ ¦ ¦ ¦*¦ ¦ ¦*¦*¦ ¦ LSB
```

If the Fp2 test fails, the following message is displayed:

```
FAILED FP TEST #2
```

The LEDs flash the following:

```
MSB ¦ ¦ ¦ ¦*¦ ¦ ¦*¦*¦*¦ LSB
```

Udc Slave

This diagnostic is a write/read test of the DMA controller.

If the test fails, the following message is displayed:

```
FAILED UDC Slave Test
```

The LEDs flash the following:

```
MSB ¦ ¦*¦ ¦ ¦ ¦ ¦*¦ ¦ ¦ LSB
```

Chain1

This diagnostic verifies that the Universal DMA controller can become the bus master and read from main memory.

If the test fails, the following message is displayed:

```
FAILED UDC Channel 1 Chain Test
```

The LEDs flash the following:

```
MSB ¦ ¦*¦ ¦ ¦ ¦ ¦*¦ ¦*¦ LSB
```

Chain2

This is the same as **Chainl** except it uses channel 2. It verifies that the Universal DMA controller can become the bus master and read from main memory.

If the test fails, the following message is displayed:

FAILED UDC Channel 2 Chain Test

The LEDs flash the following:

MSB ¦ ¦*¦ ¦ ¦ ¦ ¦*¦*¦ ¦ LSB

Scsi Slave

This diagnostic test is a write/read test of the writable/ readable register of the SCSI Protocol Controller.

If the test fails, the following message is displayed:

FAILED SCSI Slave Test

The LEDs flash the following:

MSB ¦ ¦*¦ ¦ ¦ ¦ ¦*¦*¦*¦ LSB

Scsi Master

This diagnostic does write/read test to the SCSI disk on the system. The data written is compared to verify that it is correct. This test also searches for the presence of other devices on the SCSI bus.

If the test fails, the following message is displayed:

FAILED SCSI Master Test

The LEDs flash the following:

MSB ¦ ¦*¦ ¦ ¦ ¦*¦ ¦ ¦ ¦ LSB

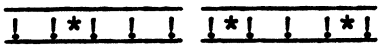
EnetProm

This diagnostic computes and verifies the checksum of the Ethernet PROM.

If the test fails, the following message is displayed:

FAILED ETHERNET ID PROM

The LEDs flash the following:



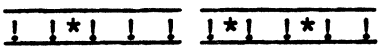
Lance Slave

This diagnostic is a write/read test of the Lance registers.

If the test fails, the following message is displayed:

FAILED Lance Slave Register Test

The LEDs flash the following:



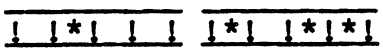
Lance Master

This diagnostic initializes the Lance and sets up transmit and receive descriptor rings to communicate with the Lance. The Lance is tested in a loopback mode, comparing the data sent and received.

If the test fails, the following message is displayed:

FAILED Lance Master Test

The LEDs flash the following:



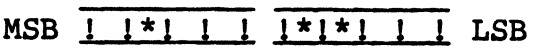
Atreg

This is a diagnostic write/read test of the PCAT register.

If the test fails, the following message is displayed:

FAILED AT Register Test

The LEDs flash the following:



6.3 VOLTAGE MEASUREMENT POINTS

Voltage	Location	
+5VDC	Motherboard	Jx15, Jx16, Jx19, Jx20
+5V Return	Motherboard	Jx3, Jx14, Jx17, Jx18
+12VDC	Motherboard	Pin 6, Jx11
-12VDC	Motherboard	Pin 5, Jx11
-5VDC	Motherboard	Pin 4, Jx11
+5VDC	Power Distributor Board	Pin 5, Pin 6 of Jx19
+12VDC	Power Distributor Board	Pin 1, Pin 2 of Jx19
Ground	Power Distributor Board	Pin 4 of Jx19
+12V Return	Power Distributor Board	Pin 3 of Jx19
+12VDC Aux	Power Distributor Board	Pin 2, Pin 3 of Jx18
+12V Aux Return	Power Distributor Board	Pin 6, Pin 7 of Jx18

Section 7
TECHNICAL SPECIFICATIONS

MOTHERBOARD

Card Slots	1 CPU Slot 6 Memory Slots 4 AT-bus Slots
I/O Ports	4 RS232 Async Ports 1 Ethernet Port 1 SCSI Port
Functions	Memory Control Logic CPU Interface Boot PROM with Power-on Diagnostics Async Port Control Ethernet Port Control SCSI Port Control Time-of-day Clock Non Volatile RAM AT-bus Control
Size	401mm (15.8 inches) x 460mm (18.1 inches)

CPU BOARD

Processor	MIPS R2000 RISC Chip Set
Floating Point	MIPS R2010
Clock Frequency	16.7MHz
Cycle Time	60 nsec
Cache Memory	64KB Instruction 64KB Data
Size	140.3mm (5.5 inches) x 396mm (15.6 inches)

MEMORY BOARDS

Capacity	8MB Dynamic RAM
Parity	Byte Parity
Data Bus	32 Bits
Size	140mm (5.5 inches) x 320mm (12.6 inches)

MUX BOARD

Bus	AT-bus
Processor	80188 at 8MHZ
Memory	64KB
Ports	4 RS422
Protocol	Synchronous SDLC-like point-to-point
Data Rate	230.4 KHz
Size	4.8 inches x 7.0 inches
Interrupt	IRQ10,11,12,15 used and are software programmable

8/tc TERMINAL CONCENTRATOR

Processor	CMOS HC64180 at 6.144 MHZ

RS232 Ports

Number Ports	8
Protocol	Async
Connector Type	RJ11 4-conductor
Flow Control	Xon/Xoff
Line Speed	300 - 19.2K Baud
Control Signal	DSR
Cable Length	50 feet max for RS232
Data Transfer	9600 baud continuous full duplex, on all 8 ports.
LED Indicators	2 per RS232 port, Green LED indicates data terminal ready, Yellow LED indicates 8/tc is receiving data from a serial port. These LEDs are independent of any host communication.

Host Port

Interface	One RS422 Port
Protocol	SDLC-like point-to-point
Connector Type	8 pin DIN
Cable Length	1000 feet max. (power module required beyond 10 feet)
Data Rate	230.4KHZ
LED Indicator	There is one LED indicating Host communication. This LED blinks after unit has passed power-up self test and is not yet being accessed by the 8x4 Mux.
Size	8.2 inches x 5.8 inches x 1.6 inches

1/2-INCH TAPE ADAPTER BOARD

Interface	SCSI at host-side, Pertec at tape drive.
Density	1600, 3200, 6250 BPI
Size	6.3 inches x 11 inches

HARD DISK DRIVES

Capacities	182MB, 376MB, 702MB
Size	5-1/4 inches
Interface	SCSI
Avg. Seek Time	16ms

CARTRIDGE TAPE DRIVE

Capacity	150MB
Size	5-1/4 inches
Interface	SCSI
Media	DC-600XTD for 150MB read/write DC-600A for 125MB read/write DC-300XL for 60MB read only

POWER SUPPLY MAIN

Power Rating	500 Watts
Input Voltage	90-130VAC or 180-260VAC, 47-63Hz
Outputs:	+5VDC 75A max, 90A peak +15VDC 6A max, 7A peak -15VDC 6A max, 7A peak +12VDC 12A max, 18A peak -12VDC 6A max, 7A peak
Fuse Rating	7AMP (115VAC), 3AMP (220 VAC)

POWER SUPPLY AUX.

Power Rating	144 Watts
Input Voltage	90-130VAC or 180-260VAC, 47-63HZ
Outputs	+12VDC 12A max, 15A peak

CABINET

Size	29 inches H x 12 inches W x 29 inches D
Weight	100 lb. typical

HELICAL SCAN TAPE DRIVE

Capacity	2.3 G Bytes
Size	5 1/4"
Interface	SCSI
Accesstimes	900 usec read, 950 usec write
Media	8mm, 3.7 x 2.5 x 0.6 inches

Appendix A
GLOSSARY

Address - a number, label, or name that indicates the location of information in the computer's memory.

A.out - the default name of a freshly compiled object file, pronounced 'A-dot-out'; historically a.out signified assembler output.

Application program - an application program is a working program in a system. Such programs are usually unique to one type of users' work, although some application programs can be used in a variety of business situations. An accounting application, for example, may well be applicable to many different businesses.

Archive - 1. a collection of data gathered from several files into one file.
2. especially, such a collection gathered by **ar(1)** for use as a library.

Argument - an argument is additional information that is passed to a command or a function. On a command line, an argument is a character string or number that follows the command name and is separated from it by a space. There are two types of command-line arguments: options and operands. Options are immediately preceded by a minus sign (-) and change the execution or output of the command. Some options can themselves take arguments. Operands are preceded by a space and specify files or directories that will be operated on by the command. For example, in the command

pr -t -h Heading file

all of the elements after the **pr** are arguments. **-t** and **-h** are options, **Heading** is an argument to the **-h** option, and **file** is an operand.

For a function, arguments are enclosed within a pair of parentheses immediately following the function name. The number of arguments can be zero or more; if more than two are present they are separated by commas and the whole list enclosed by the parentheses. The formal definition of a function, found in Section 3 of the Programmer's Reference Manual, describes the number and data type of argument(s) expected by the function.

Automatic calling unit - a hardware device used to dial stored telephone numbers; allows the system to contact another system over phone lines without manual intervention.

Bad block - a section of a storage medium which cannot store data reliably. This is a 512-byte disk block, otherwise known as a sector.

Block - the basic unit of buffering in the kernel, 8192 bytes; see indirect, logical, and physical blocks and sectors.

Block device - a device upon which a file system can be mounted, typically a permanent storage device such as a tape or disk drive, so called because data transfers to the device occur by blocks; cf. character device.

Boot - to start the operating system, so called because the kernel must bootstrap itself from secondary storage into an empty machine. No login or process persists across a boot.

Boot program - loads the operating system into ram.

Buffer - 1. a staging area for input-output where arbitrary-length transactions are collected into convenient units for system operations; the file system uses buffers, as does stdio.
2. to use buffers.

Buffer pool - a region of store available to the file system for holding blocks; all but raw input-output for block devices goes through the buffer pool so read and write operations may be independent of device blocks.

Cartridge tape - a storage medium that consists of a magnetic tape wound on spools housed in a plastic container.

Character device - a device upon which a file system cannot be mounted such as a terminal or the null device.

Child process - see fork.

Client - a host that has mounted an exported file system from an NFS server.

Command - a command is the term commonly used to refer to an instruction that a user types at a computer terminal keyboard. It can be the name of a file that contains an executable program or a shell script that can be processed or executed by the computer on request. A command is composed of a word or string of letters and/or special characters that can continue for several (terminal) lines, up to 256 characters. A command name is sometimes used interchangeably with a program name.

Command file - same as shell script.

Command line - a command line is composed of the command name followed by any argument(s) required by the command or optionally included by the user. The manual page for a command includes a command line synopsis in a notation designed to show the correct way to type in a command, with or without options and arguments.

C compiler - The C compiler converts C programs into assembly language programs that are eventually translated into object files by the assembler.

Configuration - the arrangement of the software or hardware of a system, peripheral, or network as defined by the nature, number, and chief characteristics of its functional units.

Controller - a device that directs the transmission of data over the data links of a network.

Core file - a core image of a terminated process saved for debugging; a core file is created under the name 'core' in the current directory of the process.

Core image - a copy of all the segments of a running or terminated program; the copy may exist in main storage, in the swap area, or in a core file.

Crash - If a hardware or software error condition develops that the system can't handle, it takes itself out of service, or crashes. Such conditions occur when the system can't allocate resources, manage processes respond to requests for system functions, or when the electrical power is unstable.

Cron - a command which creates a daemon that invokes commands at specified dates and times.

Cylinder - the set of all tracks on a disk which are the same distance from the axis about which the disk rotates.

Daemon - a background process, often perpetual, that performs a system-wide public function, e.g. **calendar(1)** and **cron(1M)**; the affected spelling is an ancient legacy.

Destination - the remote system that will ultimately receive a file transferred over a network.

Devise - 1. a file that is not a plain file or a directory, such as a tape drive, or the null device; a special file.
2. a physical input-output unit.

Diagnostic - a message printed at your terminal that identifies and isolates program errors.

Directory - a directory is a type of file used to group and organize other files or directories. A directory consists of entries that specify further files (including directories) and constitutes a node of the file system. A subdirectory is a directory that is pointed to by a directory one level above it in the file system organization.

The **ls(1)** command is used to list the contents of a directory. When you first log onto the system, you are in your home directory ($HOME). You can move to another directory by using the **cd(1)** command and you can print the name of the current directory by using the **pwd(1)** command. You can also create new directories with the **mkdir(1)** command and remove empty directories with **rmdir(1)**.

Directory entry, entry - 1. an association of a name with an inode number appearing as an element of a directory. 2. the name part of such an association.

Directory hierarchy - the tree of all directories, in which each is reachable from the root via a chain of subdirectories.

Directory tree - same as directory hierarchy.

Disk - a platter coated with magnetic material on which data can be stored.

Drive - the hardware device that holds magnetic disks and tapes while they are in use.

Dvh - disk volume header. A volume header (or "volume table of contents") at the beginning of each disk contains information regarding the physical device and the logical partitions. It is manipulated by the standalone format and the UMIPS **dvhtool(1)** commands and can be viewed by using **prtvtoc(1M)**. It is designated in a device name as the "vh" partition, for example: **ipc0d0svh.**

Executable file - 1. an object file that is ready to be copied into the address space of a process to run as the code of that process.
2. a file that has execute permission, either an executable file or a shell script.

FIFO - a named permanent pipe which allows two uunrelated processes to exchange information using a pipe connection.

File - 1. in general, a potential source of input or destination for output.
2. most specifically, an inode and/or associated contents, i.e. a plain file, a special file, or a directory.
3. a directory entry; several directory entries may name the same file.
4. most loosely, a plain file.

Filename - 1. a pathname.
2. the last component name in a pathname.

File system - a UNIX file system is a hierarchical collection of directories and other files that are organized in a tree structure. The base of the structure is the root (/) directory; other directories, all subordinate to the root, are branches. The collection of files can be mounted on a block special file. Each file of a file system appears exactly once in the inode list of the file system and is accessible via a single, unique path from the root directory of the file system.

Flush - to empty a buffer, for example to throw away unwanted input-output upon interrupt or to release output form the clutches of stdio.

Fork - to split one process into two, the **parent process** and **child process**, with separate, but initially identical, text, data, and stack segments.

Formatting - the process of imposing an addressing of a dvh and the mapping of the disk into tracks and sectors.

Getty - one of a series of processes which connect the user to the UNIX system. Getty is invoked by init, and in turn invokes login.

Group - 1. a set of permissions alternative to owner permissions for access to a file.
2. a set of userids that may assume the privileges of a group.
3. the groupid of a file.

Groupid - an integer value, usually associated with one or more login names; as the userid of a process becomes the owner of files created by the process, so the groupid of a process becomes the group of such files.

Init - a general process spawner which is invoked as the last step in the boot procedure; it regularly checks a table that defines what processes should run at what run level.

Inode - an element of a file system; an inode specifies all properties of a particular file and locates the file's contents, if any.

Interpreted language - An interpreted language is a high-level language that is not translated by a compilation system and stored in an executable object file. The statements of a program in an interpreted language are translated each time the program is executed.

I/O (Input/Output) - I/O is the process by which information enters (input) and leaves (output) the computer system.

Integrity - in a file system, the quality of being without errors due to bad blocks.

Interface programs - Shell scripts and programs furnished with the LP spooling software which interface between the user and the printer.

Interrupt - 1. a signal that normally terminates a process, caused by a break or an iterrupt character.
2. a signal generated by a hardware condition or a peripheral device.
3. loosely, any signal.

Kernel - the UNIX system proper; resident code that implements the system calls.

Kernel address space - a portion of memory used for data and code addressable only by the kernel.

Line discipline - a module to handle protocol or data conversion for a stream. A line discipline, unlike a filter, is part of the kernel.

Load device - designates the physical device from which a program will be loaded into main memory.

Log files - contain records of transactions that occur on the system; software that spools, for example, generates various log files.

Logical block - a unit of data as it is handled by the software; the RISC/os handles data in 8192-byte logical blocks.

Login - 1. the program that controls logging in.
2. the act of logging in.
3. by extension, the computing session that follows a login.

Manual page - a manual page, or "man page" in UNIX system jargon, is the repository for the detailed description of a command, a system call, subroutine or other UNIX system component.

Memory - 1. same as memory image.
2. physical memory represents the available space in main memory; programs are either swapped or paged into physical memory for execution.
3. virtual memory management techniques permit programs to treat disk storage as an extension of main memory.

Mode,, file mode - the permissions of a file; colloquially referred to by a 3-digit octal number, e.g. 'a 755 file'; see **chmod(1)**.

Mount - to extend the directory hierarchy by associating the root of a file system with a directory entry in an already mounted file system; converse is unmount, spelled 'umount'.

Network - the hardware and software that constitute the interconnections between computer systems, permitting electronic communication between the systems and associated peripherals.

Object file - a file of machine language code and data; object files are produced form source programs by compilers and from other object files and libraries by the link editor; an object file that is ready to run is an executable file.

Operating system - the program for managing the resources of the computer. It takes care of such things as imput/output procedures, process scheduling, the file system, removing this burden from user programs.

Open file - 1. the destination for input or output obtained by opening a file or creating a pipe; a file descriptor; open files are shared across forks and persist across executes.
2. loosely, a file that has been opened, however an open file need not exist in a file system, and a file may be the destination of several open files simultaneously.

Option - an option is an argument used in a command line to modify program output by modifying the execution of a command. An option is usually one character preceded by a hyphen (-). When you do not specify any options, the command will execute according to its default options. For example, in the command line

ls -a -l directory

-a and **-l** are the options that modify the **ls(1)** command to list all **directory** entries, including entries, whose names begin with a period (.), in the long format (including permissions, size, and date).

Page - a fixed length, 1024-byte block that has a virtual address, and that can be transferred between main and secondary storage.

Paging - the process by which programs are truncated into pages and transferred between main and secondary storage by the virtual handler (or paging daemon).

Path name - a path name is a way of designating the exact location of a file in a file system. It is made up of a series of directory names that proceed down the hierarchical path of the file system. The directory names are separated by a slash character (/). The last name in the path is either a file or another directory. If the path name begins with a slash, it is called a full path name; the initial slash means that the path begins at the root directory.

A path name that does not begin with a slash is known as a relative path name, meaning relative to the present working directory. A relative path name may begin either with a directory name or with two dots followed by a slash (../). One that begins with a directory name indicates that the ultimate file or directory is below the present working directory in the hierarchy. One that begins with ../ indicates that the path first proceeds up the hierarchy; ../ is the parent of the present working directory.

Permission - a right to access a file in a particular way; read, write, execute (or look up in, if a directory); permissions are granted separately to owner, group, and others. Permission bit a permission, so called because each permission is encoded into one bit in an inode.

Physical block - a unit of data as it is actually stored and manipulated; the RISC/os handle data in 512-byte physical blocks, or "sectors".

Pipe - a direct stream connection between processes, whereby data written on an open file in one process becomes available for reading in another.

Pipeline - a sequence of programs connected by pipes.

Ports - the point of physical connection between a peripheral device (such as a terminal or a printer) and the device controller (ports board), which is part of the computer hardware.

Process - a process is a program that is at some stage of execution. In the UNIX system, it also refers to the execution of a computer environment, including contents of memory, register values, name of the current working directory, status of files, information recorded at login time, etc. Every time you type the name of a file that contains an executable program, you initiate a new process. Shell programs can cause the initiation of many processes because they can contain many command lines.

The process id is an unique system-wide identification number that identifies an active process. The process status command, **ps(1)**, prints the process ids of the processes that belong to you.

Process id - an integer that identies a process.

Profile - 1. an optional shell script, '.profile', or '.cshrc' and/or '.login', conventionally used by the shell upon logging in to establish the environment and other working conditions customary to a particular user.

2. to collect a histogram of values of the instruction location counter of a process.

Program - a program is a sequence of instructions or commands that cause the computer to perform a specific task, for example, changing text, making a calculation, or reporting on the status of the system. A subprogram is part of a larger program and can be compiled independently.

Root - a distinguished directory that constitutes the origin of the directory hierarchy in a file system; specifically, the origin for the file system, with the conventional pathname "/"; the origin of the directory hierarchy in a file system.

Shell - 1. the program **sh(1)**, which causes other programs to be executed on command; the shell is usually started on a user's behalf when the user logs in.
2. by analogy, any program started upon logging in.

Shell script - an executable file of commands taken as input to the shell.

Single-user - a state of the operating system in which only one user is supported.

Spool - (simultaneous peripheral operations on line) to collect and serialize output form multiple processes competing for a single output service.

Stack - a segment of the address space into which automatic data and subroutine linkage information is allocated in last-in-first-out fashion; the stack occupies the largest data addresses and grows downward towards static data.

Standard error - Standard error is an output stream from a program. It is normally used to convey error messages. In the UNIX system, the default case is to associate standard error with the user's terminal.

Standard input - Standard input is an input stream to a program. In the UNIX system, the default case is to associate standard input with the user's terminal.

Standard output - Standard output is an output stream from a program. In the UNIX system, the default case is to associate standard output with the user's terminal.

Stdio: standard input-output - **stdio(3S)** is a collection of functions for formatted and character-by-character input-output at a higher level than the basic read, write, and open operations.

Stream - 1. A stream is an open file with buffering provided by the stdio package.
2. A stream is a full duplex, processing and data transfer path in the kernel. It implements a connection between a driver in kernel space and a process in user space, providing a general character input/output interface for the user processes.

String - A string is a contiguous sequence of characters treated as a unit. Strings are normally bounded by white space(s), tab(s), or a character designated as a separator. A string value is a specified group of characters symbolized to the shell by a variable.

Subroutine - A subroutine is a program that defines desired operations and may be used in another program to produce the desired operations. A subroutine can be arranged so that control may be transferred to it from a master routine and so that, at the conclusion of the subroutine, control reverts to the master routine. Such a subroutine is usually called a closed subroutine. A single routine may be simultaneously a subroutine with respect to another routine and a master routine with respect to a third.

Symbol table - A symbol table describes describes information in an object file about the names and functions in that file. The symbol table and relocation bits are used by the link editor and by the debuggers.

Syntax - 1. Command syntax is the order in which command names, options option arguments, and operands are put together to form a command on the command line. The command name is first, followed by options and operands. The order of the options and the operands varies form command to command.
2. Language syntax is the set of rules that describe how the elements of a programming language may legally be used.

System Administration - when capitalized, refers to the package of screens and interactive prompts, invoked through the **sysadm(1)** command, that help you accomplish most system administration tasks.

System calls - the set of system primitive functions through which all system operations are allocated, initiated, monitored, manipulated, and terminated; the system primitives invoked by user processes for system-dependent functions, such as I/O, process creation, etc.

System console - the directly connected terminal used for communication between the operator and the computer.

TCP/IP (Transmission Control Protocol/Internetwork Protocol) - TCP/IP is a connection-oriented, end-to-end reliable protocol designed to fit into a layered hierarchy of protocols that support multi-network applications. It is the Department of Defense standard in packet networks.

Terminal definition - a terminal definition is an entry in the **terminfo(4)** data base that describes the characteristics of a terminal. See **terminfo(4)** and **curses(3X)** in the Programmer's Reference Manual.

Terminfo - a group of routines within the curses library that handle certain terminal capabilities. For example, if your terminal has programmable function keys, you can use these routines to program the keys.

A data base containing the compiled descriptions of many terminals that can be used with **curses(3X)** screen management programs. These descriptions specify the capabilities of a terminal and how it performs various operations, for example, how many lines and columns it has and how its control characters are interpreted. A **curses(3X)** program refers to the data base at run time to obtain the information that it needs about the terminal being used.

See **curses(3X)** in the Programmer's Reference Manual. **terminfo(4)** routines can be used in shell programs, as well as C programs.

Text file, ASCII - a file, the bytes of which are understood to be in ASCII code.

Track - an addressable ring of sections on a disk; each disk has a predefined number of concentric tracks, which allows the disk head to properly access sections of data.

Tunable parameters - variables used to set the sizes and thresholds of the various control structures of the operating system.

Tuning - 1. modifying the tunable parameters so as to improve system performance.
2. the reconfiguration of the operating system to incorporate the modifications into executable version of the system.

UNIX operating system - the UNIX operating system is a general-purpose, multiuser, interactive, time-sharing operating system developed by AT&T. An operating system is the software on the computer, under which all other software runs. The UNIX operating system has two basic parts:

- The kernel is the program that is responsible for most operating system functions. It schedules and manages all the work done by the computer and maintains the file system. It is always running and is invisible to users.

- The shell is the program responsible for handling all interaction between users and the computer. It includes a powerful command language called shell language.

The utility programs or UNIX system commands are executed using the shell, and allow users to communicate with each other, edit and manipulate files, and write and execute programs in several programming languages.

Userid - an integer value, usually associated with a login name; the userid of a process becomes the owner of files created by the process and descendent (forked) processes.

Utility - a utility is a standard, permanently available program used to perform routine functions or to assist a programmer in the diagnosis of hardware and software errors, for example, a loader, editor, debugging, or diagnostics package.

Variable - 1. A variable in a computer program is an object whose value may change during the execution of the program, or from one execution to the next.

2. A variable in the shell is a name representing a sting of characters (a string value).

3. A variable normally set only on a command line is called a parameter (positional parameter and keyword parameter).

4. A variable may be simply a name to which the user (user-defined variable) or the shell itself may assign string values.

White space - White space is one or more spaces, tabs or newline characters. White space is normally used to sparate strings of characters and is required to separate the command from its arguments on a command line.

Window - A window is a screen within your terminal screen that is set off from the rest of the screen. If you have two windows on your screen, they are independent of each other and the rest of the screen.

The most common way to create windows on a UNIX system is by using the layers capability of the TELETYPE 5620 Dot-mapped Display. Each window you create with this program has a separate shell running it. Each one of these shells is called a layer.

If you do not have this facility, the **shl(1)** command, which stands for shell layer, offers a function similar to the layers program. You cannot create windows using shl(1), but you can start different shells that are independent of each other. Each of the shells you create with **shl(1)** is called a layer.

Word - A word is a unit of storage in a computer that is composed of bytes of informantion. The number of bytes in a word depends on the computer you are using. The MARK 2000 for example, have 32 bits or 4 bytes per word, and 16 bits or 2 bytes per half word.

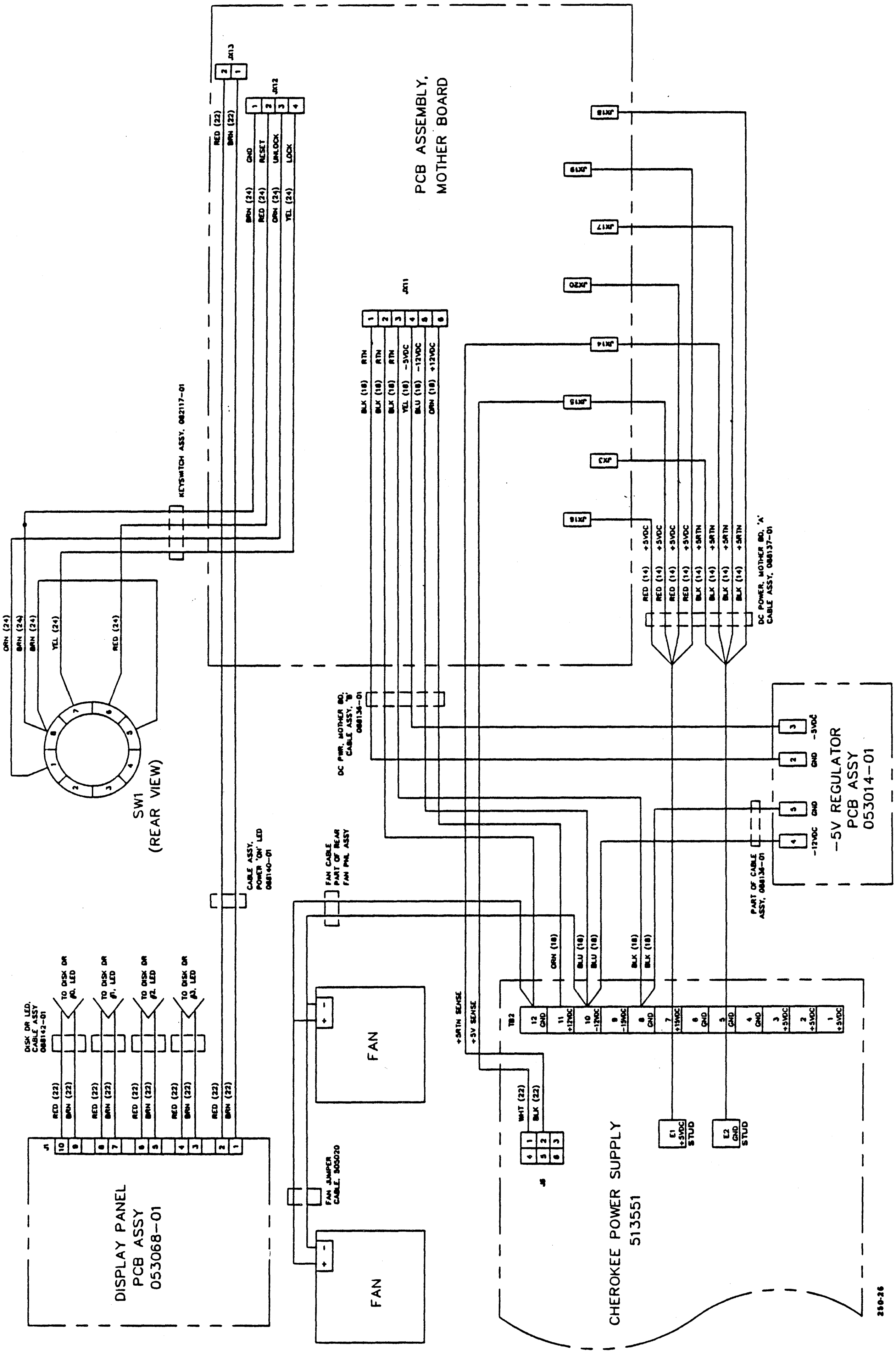
Appendix B
SYSTEM WIRING DIAGRAMS

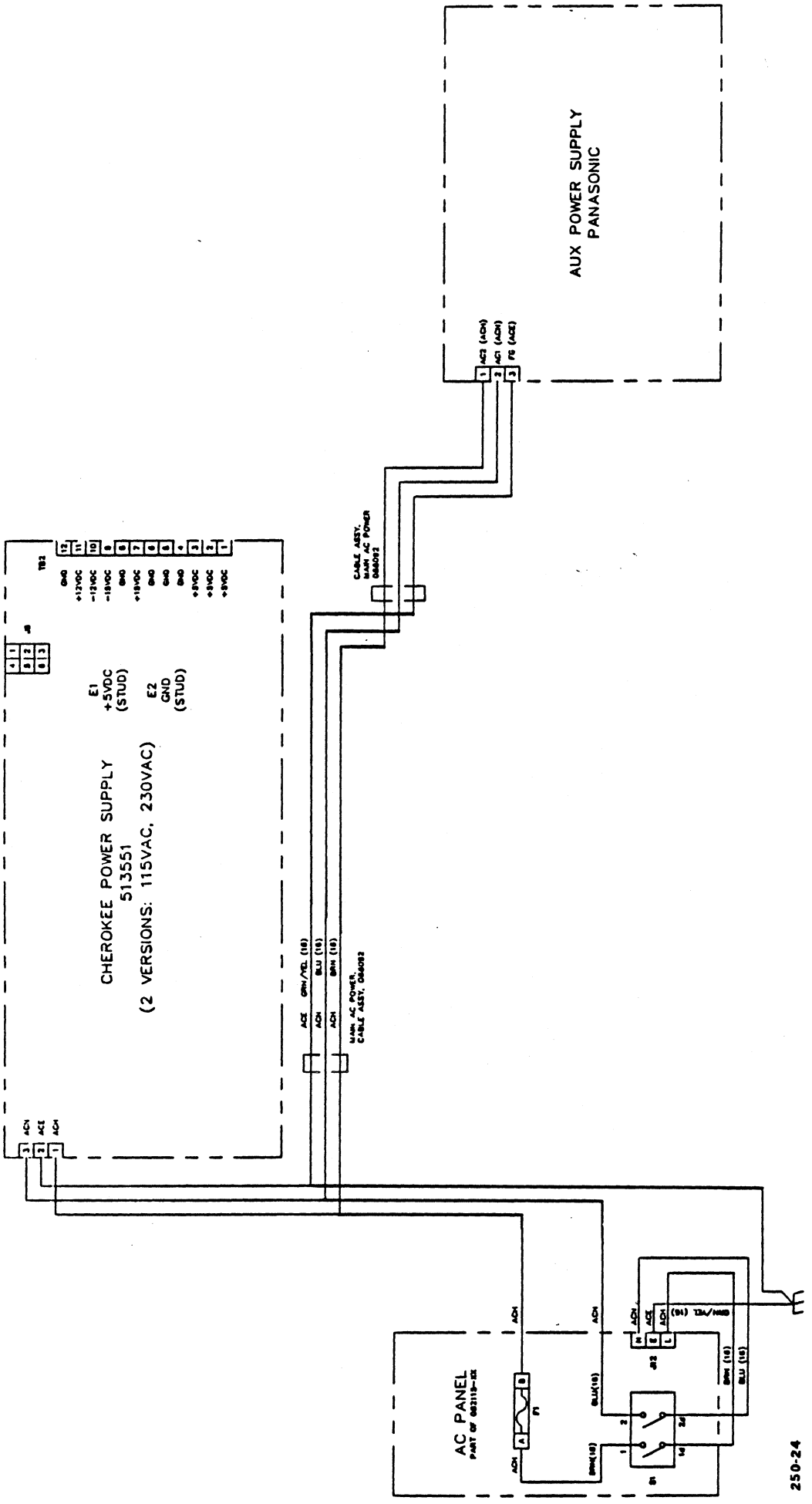
This appendix contains the following system wiring diagrams for the MARK 2000:

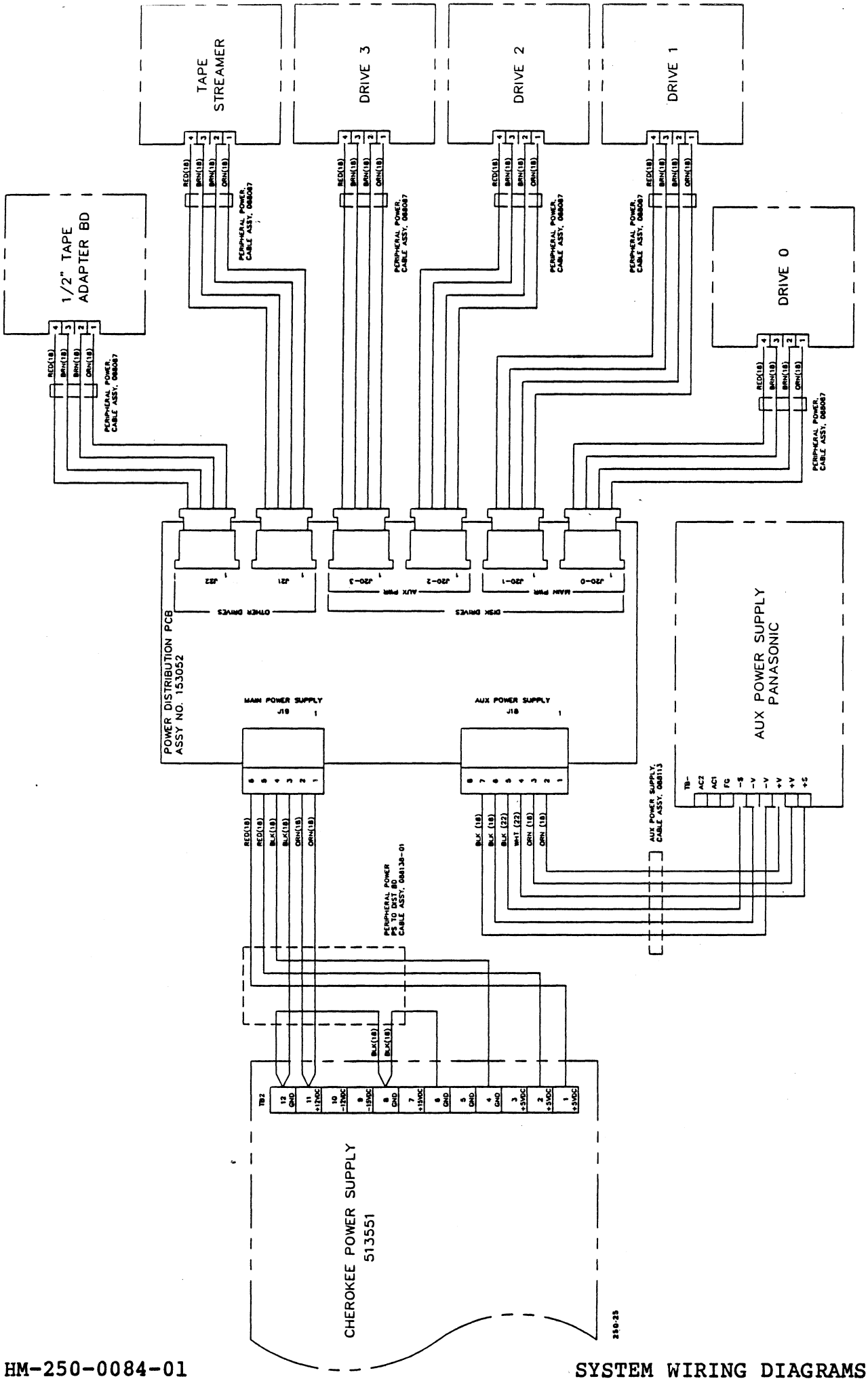
AC/DC Power Distribution

Front Panel Cabling

Internal Cabling for Peripherals


PCB ASSEMBLY, MOTHER BOARD
SW1 (REAR VIEW)
-5V REGULATOR PCB ASSY 053014-01
DISPLAY PANEL PCB ASSY 053068-01
FAN
FAN
CHEROKEE POWER SUPPLY 513551


AUX POWER SUPPLY
PANASONIC
CHEROKEE POWER SUPPLY
513551
(2 VERSIONS: 115VAC, 230VAC)
E1
+5VDC
(STUD)
E2
GND
(STUD)
AC PANEL
250-24


TAPE STREAMER
DRIVE 3
DRIVE 2
DRIVE 1
DRIVE 0
1/2" TAPE ADAPTER BD
PERIPHERAL POWER CABLE ASSY, 068067
POWER DISTRIBUTION PCB ASSY NO. 153052
J22
J21
J20-3
J20-2
J20-1
J20-0
OTHER DRIVES
AUX PWM
MAIN PWM
DISK DRIVES
MAIN POWER SUPPLY J18
AUX POWER SUPPLY J18
AUX POWER SUPPLY PANASONIC
AUX POWER SUPPLY CABLE ASSY, 068113
PERIPHERAL POWER PS TO DST BD CABLE ASSY, 068136-01
TB2
CHEROKEE POWER SUPPLY 513551

Comment Sheet

MANUAL TITLE: MARK 2000 System Manual

PUBLICATION NO. HM-250-0084 REVISION 01

FROM: NAME/COMPANY: _____

BUSINESS ADDRESS: _____

CITY/STATE/ZIP: _____

COMMENTS: Your evaluation of this manual will be appreciated by POINT 4 Data Corporation. Notation of any errors, suggested additions or deletions, or general comments may be made below. Please include page number references where appropriate.

NO POSTAGE NECESSARY IF MAILED IN U.S.A.
Fold on Dotted Lines and Tape

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

CUT ON THIS LINE

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 1458 TUSTIN, CA

POSTAGE WILL BE PAID BY ADDRESSEE

POINT 4 Data Corporation
PUBLICATIONS DEPARTMENT
15442 Del Amo Avenue
Tustin, CA 92680

15442 Del Amo Avenue
Tustin, CA 92680
(714) 259-0777