Cromemco®

Cromix-Plus Administrator's Guide

Cromix-Plus Administrator's Guide

This manual was produced using a Cromemco System Three computer running under the Cromemco Cromix Operating System. The text was edited with the Cromemco Cromix Screen Editor. The edited text was proofread by the Cromemco SpellMaster Program and formatted by the Cromemco Word Processing System Formatter II. Camera-ready copy was printed on a Cromemco 3355B printer.

The following are registered trademarks of Cromemco, Inc.

C-Net®
Cromemco®
Cromix®
FontMaster®
SlideMaster®
SpellMaster®
System Zero®
System Two®
System Three®
WriteMaster®

The following are trademarks of Cromemco, Inc.

C-10™
CalcMaster™
Cromix-Plus™
DiskMaster™
Maximizer™
System One™
TeleMaster™
System 400™

UNIX is a registered trademark of AT&T Bell Laboratories, Inc.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF TABLES

LIST OF APPENDICES

INTRODUCTION

This manual is written for the Cromix-Plus system administrator. The system administrator is assumed to have overall responsibility for:

1. Installing system hardware and software.

2. Assigning user identification numbers, group identification numbers, and passwords to system users, and safeguarding system security.

3. Regularly copying important user files to a backup storage medium (floppy diskette, another hard disk drive, or magnetic tape).

4. Handling system hardware and software problems as they arise.

The manual presents administrative tasks chronologically. Chapter 1 discusses some fundamental ideas that should clarify why you perform certain steps in later chapters.

In chapter 2, you will start up the system for the first time. This chapter includes procedures for copying the Cromix-Plus software from the factory-shipped system disks to the hard disk, bringing the hard disk on-line, and making copies of the system disks for safekeeping.

In chapter 3, you will modify certain files and programs on the hard disk for compatibility with your system's peripheral hardware. Chapters 4 and 5 describe how to set up and install the standard peripherals and printed circuit boards.

This manual assumes the reader has a basic understanding of programming (any language), files and file operations (creating, editing, copying, moving, deleting), and the basic functions of devices like a CPU, RAM, interface boards, terminals, and the like. Since Cromix-Plus is designed to take advantage of the large storage capacity and high data transfer rates of a hard disk, this manual also assumes that your system includes a hard disk. The operating system can be set up to run from a floppy disk, but this is normally done only to diagnose system faults with the hard disk off-line.

1

For more information on a particular Cromemco product, refer to the appropriate manual listed below. You can obtain copies from your Cromemco dealer.

-- Cromemco Introduction to Cromix-Plus User's Manual, part number 023-5012

-- Cromemco Cromix-Plus User's Reference Manual, part number 023-5013

-- Cromemco Cromix-Plus Programmer's Reference Manual, part number 023-5014

-- Cromemco XPU Instruction Manual, part number 023-2029

-- Cromemco 68000 Board Family Instruction Manual, part number 023-2016

-- Cromemco M-Series Memory Board Family, part number 023-2033

-- Cromemco 256KZ Random Access Memory Instruction Manual, part number 023-2021

-- Cromemco 64FDC Floppy Disk Controller Instruction Manual, part number 023-2022

-- Cromemco 16FDC Disk Controller Instruction Manual, part number 023-2004

-- Cromemco STDC Disk Controller Instruction Manual, part number 023-2031

-- Cromemco Octart Asynchronous Communications Processor, part number 023-2028

-- Cromemco Quadart Serial Interface Instruction Manual, part number 023-2005

-- Cromemco IOP Input/Output Processor Instruction Manual, part number 023-2006

-- Cromemco TU-ART Digital Interface Instruction Manual, part number 023-0011

-- Cromemco PRI Printer Interface Instruction Manual, part number 023-0055

-- Cromemco C-Net User Manual, part number 023-4059

-- Cromemco C-10 Personal Computer User Manual, part number 023-6037

Chapter 1

CROMIX SYSTEM FUNDAMENTALS

This chapter presents fundamental background material for later chapters. It defines frequently used terms (important terms appear **boldfaced** when first introduced), and discusses key concepts common to all Cromix-Plus Systems.

PROGRAM CROMIX.SYS

Program **cromix.sys** is the heart of the Cromix-Plus Operating System, and always resides in system memory. It contains many subprograms which can be grouped into three general categories: the **shell program**, the **kernel**, and the **device drivers.** Brief descriptions of each category follow.

The Shell Program

The **Shell** program acts as an interface between Cromix users and the Cromix **kernel** (the hidden inner workings of the Cromix-Plus Operating System). The Shell program is structured so that several users may run it simultaneously; thus, each user is said to be running a shell or shell process.

When a user logs in, a shell process is automatically attached to the user's terminal. The Shell prompts the user for a **command** with either a **#** (privileged user) or **%** (non-privileged user) character. When you type something in response to the prompt, the Shell stores the command in a shell buffer, and attempts to interpret its instructions. Normally, a command consists of a program name followed by one or more arguments. For example, if you type:

> % **match nuts parts_list**

the Shell parses the command into three parts, **match** (the program name), **nuts** (a string argument), and **parts_list** (a string argument), and places the two string arguments where program **match** can later find them. The Shell then looks for program **match** (the Match utility normally resides on the disk in the **/bin** directory with the name **match.bin**), and if it is found, control passes to the kernel, which schedules, loads, and runs the program. The Shell supplies an error message if the program is not found, and then prompts the user for a new command. While the kernel is running **match**, it puts the shell to **sleep** (temporarily inactivates it); when **match** stops running, the kernel starts the shell running again, and the shell prompts the user for a new command.

In addition to preprocessing commands from user terminals, the shell can also interpret and carry out multiple commands stored in ordinary disk files. Files of this type are called **command files.**

The Shell recognizes and processes several special symbols. There are special symbols for: program input and output redirection (>, >>, <, >*, ><, ><*, >>*), pipes (|, |*), defining labels (%), parameter identification (#), string equality (=), string inequality (!=), sequential processing (;), and detached processing (&).

Most programs are stored as disk files, but some frequently used programs permanently reside in system memory as part of the Shell program itself. These programs are called either **intrinsic commands** (since they are intrinsic to **cromix.sys**), or **Shell commands.** Shell commands run entirely in system memory (they require no user memory).

The Kernel

The second general category of subprograms in **cromix.sys** is the Cromix **kernel.** At this level, programs run automatically, and are not directly accessible to users. The kernel is responsible for loading, scheduling, and executing all programs.

The data needed by the kernel to control a program is stored in that program's **process table.** The kernel assigns each running program, or process, a **process identification number** (PID). At a given time, each process is either running (the system CPU is executing its instructions), sleeping (its instructions are in RAM, but the CPU is not currently executing them), or terminated (the process is finished, but the process which started it has not yet received its termination status). You can view the current state of the process tables by running the **pstat -al** Shell command.

Every 16 milliseconds, the system CPU is interrupted. At these times, the current process momentarily stops running, and the kernel decides whether to put the process to sleep, or run it until the next interrupt occurs. This decision is based on the priorities of all other sleeping processes (if a user does not explicitly assign a priority with the **priority** Shell command, the kernel assigns it a default value). After a process terminates, and the parent process receives its termination status, the kernel eliminates its entry in the process table.

The Device Drivers

A device driver is a program that manages physical input and output for a related class of peripherals. For example, there is one device driver for all Cromix-compatible floppy disk drives, one for hard disk drives, one for printers, and so on. If your system uses an IOP co-processor board to manage peripherals, two device drivers are required: one in **cromix.sys** to communicate with the IOP, and one on the IOP to manage peripheral I/O and communicate with the driver in **cromix.sys**.

The code for the shell and kernel is fixed in each **cromix.sys** program, but the set of device drivers may be varied. The set of device drivers is specified in the **/gen/sysdef** file, and a new **cromix.sys** program is created with the **/gen/crogen** utility.

Default Root Device

The Cromix **root device** is the disk drive that contains the root directory **/** and some, or all, of its descendant files. If the root device is the only on-line disk drive, then it contains the entire file structure. Additional disk drives can be attached to (**mounted**), or disconnected from (**unmounted**), dummy files on the root device to form extended Cromix file structures. In this case, the root device contains only part of the Cromix file structure. If you select a **default root device** in the **/gen/sysdef** file, then the drive you specify is automatically chosen as the root device when you **boot** the new **cromix.sys** program. If you do not select a default root device, the new **cromix.sys** program prompts you to select one each time you boot up.

The variables that make one **cromix.sys** program different from another are defined in the **/gen/sysdef** file. If you change the **/gen/sysdef** file and generate a new **cromix.sys** program, the new version takes effect only **after** the new program is booted.

THE FACTORY-SHIPPED SYSTEM DISKS

Cromix-Plus software is supplied from the factory on three 8" (or six 5-1/4") floppy diskettes called **system disks**. Disk #1, the boot disk, is used to start the system for the first time. The remaining diskettes contain the actual Cromix-Plus release in Ftar archival format. The boot process loads program **cromix.sys** from a floppy disk (initially, a system disk) into system memory, and starts it running.

You can boot the system from any floppy disk drive, as long as you set the switches on the 64FDC/16FDC board correctly (refer to chapter 5).

The initial **cromix.sys** program on the system disk contains the following device drivers: TU-ART console, floppy disk, and hard disks (including STDC, hd, and the removable part of smd0). These are the only peripherals you can use when this initial program is running.

Factory-shipped system disks do not have a default root device defined, so you must respond to the prompts to select one when you boot up your system.

System Disk Files

The **/bin** directory contains the utilities (i.e., general-purpose programs with **.bin** or **.com** filename extensions). Directory **/cmd** contains a group of useful command files. Command files are ordinary text files, named with extension **.cmd**, that contain one or more commands that the Shell can interpret. Like utilities, command files are run by typing the command file's name (minus its extension) in a command line. New command files can be created and edited with the Screen or Ed utility (two names for Cromemco's Screen Editor programs), and then copied to the **/cmd** directory.

Directory **/dev** contains the **device files.** A device file is a special file that associates a specific system peripheral with a corresponding device driver program, but it is **not** itself a device driver program. The association is internally established using three parameters: **device type** (B = block device, C = character device), **major device number**, and **minor device number.** Device files can be changed with utilities Makdev, Maklink, and with the Delete shell command.

Directory **/etc** contains several files which are normally accessed only by system utilities. For example, the operating system reads the **/etc/ttys** file to determine which terminals are on-line, then runs a **login** process for each terminal. The login process reads the **/etc/passwd** file to grant or deny system access.

In the **/etc** directory, files **iostartup.cmd** and **startup.cmd** are run each time you boot the system. File **iostartup.cmd** is used to load device drivers into the IOP and OCTART boards. The **startup.cmd** file is used to set the system time and date, and to set page widths, parities, and so on, of system peripherals. Most files in the **/etc** directory can be modified with the Screen utility, but some are intended to be altered only by specific utilities (for example, the **/etc/passwd** file should only be altered by the Passwd utility). The Passwd utility creates a home directory and writes the home directory name (along with the user name, password, etc.) to the **/etc/passwd** file. Entries in the initial **/etc/passwd** file on system disks allow you to log in as privileged user **system** with no password, and with the root directoy **/** as your home directory.

Directory **/gen** contains the programs and support files for generating a new **cromix.sys** program. Directory **/usr** contains several subdirectories. Some **/usr** directories are used as home directories for Cromix users. A home directory is the current (or working) directory right after a user logs in. Home directories (along with user names, passwords, user identification numbers, group numbers, and starting programs) are assigned and deleted with the Passwd utility.

The **/usr/help** directory contains many text files that collectively form the on-line manual. These files contain descriptions of each Shell command, command file, and utility contained on the system disk. For example, to display the Match utility description enter:

% **help match**

THE BOOT PROCESS

Booting the system when it is first turned on (or reset) is called a **cold boot**. To cold boot, you must have a floppy diskette in your boot drive that satisfies three requirements:

1.   The diskette must be Cromix formatted.

2.   The diskette must contain a version of **cromix.sys** (in the diskette's root directory).

3.   The diskette must contain a **bootstrap program**.

Any diskette which satisfies all three requirements, such as Disk #1 of the system disks, is called a **boot disk**. The cold boot procedure is shown below:

COLD BOOT PROCESS

1.   On power-up or reset, the system CPU aborts all processes (there should be none), and starts running **RDOS** (Resident Disk Operating System), a firmware program on the 64FDC/16FDC board. The CPU, RAM, and 64FDC/16FDC board switches and jumpers must be properly set for this to occur.

2.   If the 64FDC/16FDC switches are properly set, and a boot disk is installed in the boot drive, then RDOS automatically reads the bootstrap program from the boot disk, loads it into system memory, and starts it running.

3.   In turn, the bootstrap program automatically reads program **cromix.sys** from the boot disk, loads it into system memory, and starts it running.

4.   If a default root device was selected when the **cromix.sys** program was generated, that device is automatically brought on-line; otherwise, **cromix.sys** prompts you for a root device.

5.   The system automatically runs programs **/etc/iostartup.cmd** and **/etc/startup.cmd**, in order. **Iostartup.cmd** downloads selected programs (if any) into RAM on one or more IOP, STDC, and/or OCTART boards. **Startup.cmd** prompts you for the system time and date.

6.   The system then prompts you for a login name and an optional password.

At this point, program **cromix.sys** is running in system memory, a root disk drive is on-line, and the Cromix-Plus Operating System is fully functional. The **system console** is the terminal used for all boot dialogue. The first time you start the system, the system console is, by definition, the terminal attached to serial connector J4 on the 64FDC/16FDC board. Any other terminal can later be made to function as the system console. However, only the FDC terminal displays the initial sign-on message, the system memory test, the root device prompt (if no default root device has been specified), and the disk I/O error messages (whenever they occur).

While one **cromix.sys** program is running, you can warm boot (load and run) a new **cromix.sys** program with the Boot utility. Always test a new version of **cromix.sys** with a warm boot <u>before</u> you overwrite the old version.

WARM BOOT PROCESS

1.   You must be logged in as a privileged user to warm boot the system. Since warm booting kills all processes, you should make sure that all other users are logged off, and that the only process running is your shell (run the **pstat -a** shell command).

2.   The new **cromix.sys** program must be located somewhere in the Cromix file structure, but not in the root directory (the old version of **cromix.sys** is in the root directory). The new version may be on the root device or on a mounted disk. The file name must have a **.sys** extension (e.g., **test.sys**, **cromix2.sys**, etc.)

3.   Warm boot the system by running the Boot utility with the pathname of the new **cromix.sys** file, but omit the **.sys** extension. If you do not supply a pathname, the system tries to warm boot the **cromix.sys** program in the root directory. For example, to warm boot program **cromix.sys** in the **/gen** directory, enter:

>   **# boot /gen/cromix**

4.   The Boot utility loads the specified **cromix.sys** program into system memory and starts it running.

5.   From this point, the warm boot procedure is identical to the cold boot procedure, steps 4 through 6, described above.

Stopping the System

Normally, the system is stopped in one of four ways:

1. Running the Shutdown command (privileged users only). All users are given a 5-second warning, then all processes are killed, all buffers are flushed, and all users are logged off. All users should be warned at least a minute beforehand with the Msg utility.

2. Resetting the computer stops the CPU and initiates an attempt to cold boot the system. Never reset the system without first running the Shutdown command or the Kill command (kill -2 1), else you may damage the files being modified at the time. The Kill command is the same as Shutdown, except that it does not issue a warning to the other users.

3. Turning off the system. Observe the same precautions as when resetting the computer, but in addition, remove all floppy diskettes from their drives before turning off the power.

4. An unexpected power failure. You should back up your hard disk periodically (copy hard disk files to another hard disk, to floppy diskette, or to magnetic tape) to avoid loss of data due to power failures.

Chapter 2

INITIAL CROMIX SYSTEM START-UP

This chapter describes how to start up the operating system for the first time, how to build and update the Cromix-Plus system on a hard disk, how to tailor the system to your particular needs, how to create a boot diskette, and how to make backup copies of the release diskettes. For these procedures you will need a set of Cromix-Plus release diskettes (5-1/4" or 8"), all Ftar diskettes, and several extra floppy diskettes (blank or used).

Cromix-Plus release disks are write protected to prevent them from being accidently erased or overwritten. The 5-1/4" release disks should have a silver sticker on the upper right edge; 8" release disks should NOT have a silver sticker on the bottom edge. You can ignore the "Read only file system" messages issued by the Mount, Unmount, and Update1 commands. However, when using the Mount command to list the mounted devices, the Mount utility cannot know about write-protected devices because the mount table (the /etc/mnt file) cannot be written to-it is a write-protected root device.

Use the system console (the terminal connected to J4 on the 64FDC/16FDC board) for all dialogue in the procedures that follow (refer to chapters 4 and 5 for hardware installation). User entries are in boldfaced type, and all entries must be terminated by pressing the RETURN key.

COLD BOOTING CROMIX FROM THE RELEASE DISKETTES

Turn on the system and the system console, insert the Cromix-Plus BOOT disk into drive A, and reset the system (refer to your system manual). If nothing happens in a few seconds, press the console RETURN key several times. You should see the following (A more detailed description of what occurs during the booting process appears in the sections that follow.)

```
Preparing to boot, ESC to abort

Standby
Address:   Memory test by 16K blocks
000000h:   ++++++++++++++++++++++++++++++++++++

Floppy = 1, STDC = 6
Enter major root device number:   1
fda = 0, fdb = 1, fdc = 2, fdd = 3
sfda = 4, sfdb = 5, sfdc = 6, sfdd = 7
```

```
          dfda = 16, dfdb = 17, dfdc = 18, dfdd = 19
          Enter minor root device number:  0 (4 for 5-1/4" disk)

          System initialization complete

          Cromix-Plus Operating System version xx.yy
          Copyright (c) 1985 Cromemco, Inc.
          Boot System

          Read only file system
          Read only file system
          PID = 6
```

This is Phase one of Cromix-Plus installation. Please note the following installation steps.

1.   After you have noted the information on the screen, hold down the CNTRL key and press Q to display the next page of this information.

2.   If the message:

     "Incorrect version of STDC firmware (1.23 or higher required)"

     was displayed during boot procedure, please enter the command "boot" to reboot Cromix-Plus. Please see SUDS note Cromix-Plus-1 for details.

3.   The disk on which you plan to install Cromix-Plus must be initialized and the file structure must be built on it. (See the manuals for the Initstdc, Initsmd, Initflop, Makfs utilities). If your disk already has a file system on it, run the Check utility to verify the file system integrity. If necessary, run the Check -s utility or take any other necessary steps. Do not proceed until Check reports no errors.

4.   Run the Update1 command. The argument to Update1 command must be the name of your future root device. The Update1 command will rename a few files in your future /etc directory (in case they exist already) to names with the extension ".old". If a brand new file structure was generated there will be no ".old" files. The Update1 command will then Cptree the current file structure to the new disk and invoke the Boot command to reboot the system. This time you should designate your new disk as the root device.

```
          Cromix-Plus Operating System version xx.yy
          Copyright (c) 1985 Cromemco, Inc.
          Boot System

          Logged in system Mar-01-1960 hh:mm:ss on tty1

          Message of the day:  Welcome to the Cromix-Plus Operating System
          #
```

EXPLANATORY NOTES ON THE BOOTING PROCESS

When you begin the booting procedure, and the system message,

Preparing to boot, ESC to abort

appears, you may allow the system to boot, or you may press ESCAPE (you have about 4 seconds) and the semicolon (;) prompt of RDOS will appear. You can then boot the system by entering **BX**, where X is the boot drive specifier (A, B, C, or D). Refer to other sections in this chapter for additional details on RDOS.

After the "Standby" message appears, the ESCAPE key will not respond, the system is busy reading the **/cromix.sys** file into memory. During this time, the drive active light on the front of the specified drive will be lit most of the time, indicating that the drive is busy being read.

When the **/cromix.sys** file has been loaded into memory, Cromix-Plus will begin executing a short memory test. Note that even if more memory is present in your system, the boot program will test only 512 Kbyte of memory. When the memory test has been completed, you should see the 32 plus (+) signs, indicating that your system's memory has passed the test. If you do not see all the plus signs, either some of the memory is defective, or you do not have 512 Kbytes of memory installed to test.

In the next step the system prompts you for the root device number. Respond by entering "1" for the major device number, and the appropriate minor device number. DO NOT BOOT THIS VERSION ONTO THE HARD DISK UNTIL THE HARD DISK HAS BEEN CHECKED AND UPDATED as described later in this chapter.

A number of messages will follow. Two "Read only file system" messages will be displayed, originating from the unsuccessful writing to the **/etc/mount** and **/etc/who** files. (This is to be expected because the diskette is write-protected and thus "read-only".) The "PID" message refers to the Flush utility which is running in the background.

The next information that appears is the **startup.msg** file, as displayed in the previous section. If the message,

Incorrect version of STDC firmware (1.23 or higher required)

appears, it means that unless you replace IC26 on the STDC board with EPROM version 1.23 or higher, you will have to repeat the double boot procedure each time you reboot the system. The correct version of STDC firmware has been downloaded from the **/etc/stdcfirm** file, but it appeared too late to be recognized. The system has already recognized that the STDC is inoperative. Upon "warm booting" the correct version of the firmware will be in memory because it was downloaded during the previous "cold boot".

13

The Cromix identification message is then displayed again, and you are logged in automatically as "system". Ignore the March 1, 1960 date shown--you will enter the correct time and date later; however, be sure that the Time command is present in the **/etc/startup.cmd** file so the correct time and date will be prompted for later. Reasonably accurate time and date are required for the Make utility to function properly.

The message of the day is followed by the privileged user prompt (#).

BOOTING CROMIX-PLUS FROM STDC HARD DISKS

Version 03.12 of RDOS, Cromemco's Resident Disk Operating System, allows for booting Cromix-Plus directly to a variety of STDC hard disk partitions or to floppy disk. In addition to RDOS 03.12, STDC firmware 01.23 or higher is required.

Users who wish to boot directly to WDI hard disk drives must still use RDOS version 03.08. It will be necessary to write a boot track to the device, as described below; the respond 8:0 when you are prompted for the boot device.

DETERMINING YOUR VERSION OF RDOS

To determine your version of RDOS, simply reset your system and press ESC when the message:

Preparing to boot, ESC to abort

appears on the screen (you have approximately four seconds). RDOS will display its version number.

RDOS

RDOS is capable of being set to boot in default to a variety of devices and partitions. This is determined by the switch settings on the 64FDC board. Switches 2 thru 5 are used to determine default boot devices. Refer to chapter 5 for information on how to set these switches for all RDOS versions.

THE BOOT TRACK

Cromix-Plus requires a boot program that RDOS can load into memory and begin executing. The boot program in turn loads the Cromix-Plus kernel and begins its execution.

14

The boot program resides on an area of the hard disk or floppy disk that is not part of the normal file system. This area is designated as the first track of a given device—the boot track.

The Cromix-Plus bootstrap must be located on the same partition on which the file **cromix.sys** resides. Depending on how the file **cromix.sys** is generated, it is possible to "root" on another device. In that case you will be so prompted during the boot process.

The boot program is written to this special area by the program **wboot.bin**. The bootstrap program may be located on the first 10 Kbytes of a Cromemco floppy disk (not a uniform floppy) or on the first track (physical track 0) of the entire hard disk or the first (logical) track of any Cromix-Plus partition on a hard disk.

To write the boot program to the appropriate location under Cromix-Plus:

> # **wboot stdX** (or **hdX**, or **fdX**, and so on)

Where X is any Cromix-Plus partition.

NON-DEFAULT BOOTING PROCEDURES

To boot Cromix-Plus on the default partition requires merely pressing RETURN a few times (pressing RETURN is not necessary on Cromemco 3102 terminals). RDOS will execute a simple memory test and then proceed to load the appropriate bootstrap program.

If you wish to boot to other than the default partition (for instance booting alternately between Cromix-plus and UNIX), you must press the ESCAPE (ESC) key before the memory test is completed. This will cause the RDOS prompt (";") to appear (after the memory test). The screen will look something like this:

```
Bank 0 > 0 1 2 3 4 5 6 7 8 9 A B C D E F
        ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^
^[
;
```

At this point you may instruct RDOS to read the bootstrap from any valid partition or device. The syntax for this operation is as follows (Note: by default RDOS accepts input in hexidecimal. To use decimal values, they must be terminated with a period):

Enter:	;bst31.	Read bootstrap from:	STD31
	;bst63.		STD63
	;bst0.		STD0
	;bst2.		STD2
	;bst5.		STD5
	.		.
	.		.
	;ba		FLOPPY A
	;bb		FLOPPY B
	;bc		FLOPPY C
	;bd		FLOPPY D

INITIALIZING THE HARD DISK

This section describes how to format, declare alternate tracks, and set the disk partitions of an SMD or STDC-controlled hard disk drive. This procedure is required only to install a new drive, to reinitialize an existing drive, or to change the partition or alternate track tables. For further information, refer to the description of the Inithard utility in the Cromix-Plus User's Reference Manual.

Caution: This procedure may destroy all data stored on the hard disk.

Boot the system as described in the previous section, and enter:

 # inithard

You should see the following display:

Inithard Version 30.XX

Press:	RETURN to supply default answers
	CTRL-C to abort program
	To only do alternate tracks and partitions answer "x" to first cylinder question.
Warning:	Inithard can destroy all disk data

Disk to initialize (devname)? **std31**

To initialize the entire hard disk, or to change the partition or alternate track tables, enter the device name for the entire disk (e.g., **std31** for STDC drive 0; **std63** for STDC drive 1). To reinitialize a single partition, enter the device name of that partition. If the device is already formatted, the program asks if you want to continue (since initialization erases all of the data on the device).

16

When initializing the entire disk, you will be prompted for the drive specifications: number of data surfaces, number of cylinders, maximum number of alternate tracks, starting cylinder of the disk, and the starting cylinder for write precompensation. Specifications for Cromemco STDC drives are shown in table 2-1; for any other drive, refer to the hard disk documentation.

It is recommended that you set the starting cylinder of the disk to 1. This means that STD0 (or STD32) begins with cylinder 1 rather than cylinder 0. To access cylinder 0, you must refer to STD31 (or STD63). This prevents accidently over-writing the disk label and alternate track information.

You can now declare up to 32 partitions (logical drives) on each hard disk. Each partition must have its own device name in the **/dev** directory, and its own file structure (refer to chapter 3). On the first hard disk, the name of the first partition is **std0**, the second partition is **std1**, and so on up to **std31** (which refers to the whole first drive). On the second hard disk, the name of the first partition is **std32**, the second partition is **std33**, and so on up to **std63** (which refers to the whole second drive).

Each partition starts on a specified cylinder, and ends on the starting cylinder of the next partition, minus one. For example, to create three equal partitions on a Vertex V150 hard disk (which begins on cylinder 0), divide the number of data cylinders by 3. There are a total of 987 cylinders, but the 60 alternate tracks use 60/5 = 12 cylinders. Thus, there are 987 - 12 = 975 data cylinders, and each partition will have 975/3 = 325 cylinders. The cylinders occupied by each partition are:

```
std0 (or std32)        from    0 to 324
std1 (or std33)        from  325 to 649
std2 (or std34)        from  650 to 974
```

When asked for the starting cylinder of each partition, enter 325, 650, and RETURN, respectively.

The next set of prompts defines the cylinder and head numbers of the disk to be initialized. (If you enter **x** for the first cylinder prompt, the initialization process is skipped, and the alternate track table is displayed.) When initializing device **std31** (or **std63**), all of the cylinders can be initialized (counting from zero). When initializing a single partition, only the cylinders belonging to that partition can be initialized (again counting from zero). If a bad track is encountered, the declared alternate track will be initialized instead.

When initializing a single partition, the program will end here. When initializing the entire drive (**std31** or **std63**), the current alternate tracks (if any) are displayed, and you are prompted to retain them and/or to declare new ones. Keep a written record of the declared alternate tracks for future reference. A final prompt allows you to change or delete all of the entries in the alternate track table.

If the drive was not previously initialized (for initialized drives bytes 78H - 7BH contain "CSTD") the utility will display a recommended starting cylinder for alternate tracks. Use the recommended value unless you have a specific reason for changing it.

Table 2-1: HARD DISK DRIVE CAPACITIES

Model	Vertex Model V150	Micropolis Model 1304	Hitachi Model DK511-5	Maxtor Model XT-1140	Control Data Model 94155-57	Maxtor XT-2140
Unformatted Capacity	51.4 Mbytes	51.9 Mbytes	51.0 Mbytes	143.4 Mbytes	57.8 Mbytes	140.2 Mbytes
Write Precomp. Starting Cyl.	986	829*	256	917	0	1224
No. of Platters	3	4	4	8	4	6
Data Surfaces	5	6	7	15	8	11
Data Cylinders	987	830	714	918	925	1225
Data tracks (surf.x cyl.)	4,935	4,980	4,998	13,770	4,950	13,475
Bytes per track	10 Kbytes	10Kbytes	10 Kbytes	10 Kbytes	10 Kbytes	10 Kbytes
Formatted Capacity	49.35 Mbytes	49.8 Mbytes	49.98 Mbytes	138.0 Mbytes	55.5 Mbytes	135.0 Mbytes

* If drive serial number is less than 101242-XX-X, set precompensation to 400.

VERIFYING THE HARD DISK

If you did not initialize the hard disk as described in the previous section, use the Readall utility to verify the integrity of the disk. For example, to read the entire STDC drive 0, enter:

readall -a std31

The -a option skips the tracks declared bad in the alternate track table. If there are no errors, go to the next section; otherwise note the head and cylinder numbers of the bad tracks, and run Readall a few more times. If the bad tracks appear at random spots on the disk, contact your Cromemco dealer, distributor, or authorized service facility. If successive Readall errors stay on the same tracks, reinitialize the bad tracks with the Inithard utility and run Readall again. If you have critical data already stored on the disk, try copying the data to another device, because reinitializing even a few tracks is likely to damage part of the file system.

If the bad tracks persist, add them to the alternate track table; if reinitializing the bad tracks makes them readable, keep a record of them and add them to the alternate track table if they cause trouble in the future.

CREATING AND CHECKING THE FILE STRUCTURE

After initializing and verifying the hard disk, use the Makfs utility to create a file structure on a selected partition (unless one already exists). Beware that using the Makfs utility will destroy the information in a file system if it is already present. For example, to make a file system on the first partition of

18

STDC drive 0, enter:

makfs std0

If the file system already exists, use the Check utility to verify the structure's consistency. This step is extremely important if you reinitialized any tracks. For example, to check the STDC drive 0, enter:

check std0

If there are no errors, go to the next section; otherwise, run the Dcheck and Icheck utilities (in order) with the -s option. For example:

dcheck -s std0
icheck -s std0

If either utility reports an error, consult the Dcheck and Icheck sections of the Cromix-Plus User's Reference Manual for corrective action. Most likely the corrective action will be:

- Mount the hard disk (e.g., **mount std0 /drive**)
- Use the Ncheck utility to find the names of bad inodes
- Delete the bad filenames
- Unmount the hard disk

Run Dcheck and Icheck utilities again, and repeat the corrective process until no errors are reported.

UPDATING THE HARD DISK

After the Check utility verifies the file structure on the hard disk, use the Update1 utility to copy the Cromix-Plus boot diskette in drive A to the appropriate hard disk partition. For example:

update1 std0

If the following files already exist on the hard disk, the Update1 utility renames them by adding a .old filename extension:

```
/etc/group
/etc/iostartup.cmd
/etc/motd
/etc/passwd
/etc/startup.cmd
/etc/startup.msg
/etc/termcaps
/etc/ttys
/gen/sysdef
```

Update1 automatically reboots the system, but when you are prompted for the root device, enter the major and minor root device numbers of the appropriate hard disk partition (e.g., 6 and 0 for std0).

The boot diskette contains only a small portion of the Cromix-Plus Operating System. The remainder of Cromix-Plus must also be added to the hard disk from the rest of the floppies, which are in Ftar format. Note that Ftar formatted floppies cannot be either mounted or booted--the data they contain can only be extracted from the floppies into your root directory using the Ftar utility.

Make the root directory the current directory (enter: d /), replace the boot disk in drive A with Cromix-Plus Diskette #2, and enter:

update2 sfda (use **fda** for 8" drives)

Run the Update2 utility for each of the remaining Cromix-Plus diskettes (in order). After you copy all of the floppies, you have a complete Cromix-Plus system on your hard disk. Then execute the **boot** command. When you are so prompted, enter the hard disk as the root device.

CUSTOMIZING CROMIX-PLUS

This section describes how to generate a new **cromix.sys** program that includes all the device drivers for your particular system, and that automatically selects the hard disk as the default root device when you boot up. First, copy the **/gen/sysdef** file to the file **/gen/newsysdef** for editing (refer to appendix B of the Introduction to Cromix-Plus User's Manual, part number 023-5012, for a description of the Screen Editor program):

screen /gen/sysdef /gen/newsysdef

Refer to the Sysdef section of the Cromix-Plus User's Reference Manual for editing instructions (the complete **/gen/sysdef** file is shown in appendix C). Note that some drivers need arguments to be included in the **sysdef** file.

20

When you finish editing the Sysdef file, enter the next two commands: the first generates a new **cromix.sys** program, and the second tests the new program with a warm boot.

    # /gen/crogen   /gen/cromix   /gen/newsysdef
    # boot   /gen/cromix

If the system doesn't respond, you probably made some incorrect changes to the **/gen/sysdef** file. Cold boot the system with Disk # 1 (selecting the hard disk as the root device), run the Check utility on the hard disk, and review the Sysdef file correcting any errors.

When your new Cromix boots correctly, you may need to edit some or all of the following files:

    /etc/group
    /etc/iostartup.cmd
    /etc/motd
    /etc/passwd
    /etc/startup.cmd
    /etc/startup.msg
    /etc/termcaps
    /etc/ttys

Your previous versions of these files (if you had them) still exist in the **/etc** directory with **.old** filename extensions (**group.old, iostartup.cmd.old,** and so on). To retain any of your old files, simply overwrite the new versions.

The following information will help you determine whether or not you elect to retain old versions of these files. Use the Move utility to move any old files you want to keep in place of their new equivalents.

You will probably want to retain the **/etc/group.old** file; if so, enter:

    move -vf /etc/group.old /etc/group

Otherwise, you can delete the old version:

    del -v /etc/group.old

Compare the new **/etc/iostartup.cmd** file to the old **/etc/iostartup.group.old** file. If necessary, edit the new file to delete percent signs (%) from lines you wish to execute.

The **/etc/motd** file is merely an arbitrary message file. Move it to your new directory if you wish to retain it.

You will probably want to keep your original **/etc/passwd.old** file, if so, move it to your directory.

Compare the new **/etc/startup.cmd** file to your old **/etc/startup.cmd.old** file; the command **mode retype** is no longer supported—the Shell editor will be enabled by either the proper **ttys** file or the Term command.

The **/etc/startup.msg.old** file is another arbitrary message file that you may retain or not.

The **/etc/termcaps** file is new, and describes terminal capabilities.   If necessary, you can add new features and terminal types to it.

Compare the new **/etc/ttys** file to the old file. Edit the new file to enable terminals you wish enabled by changing the terminal type from "dumb" to whatever type it is. (Refer to the **/etc/termcaps** file, described in the Cromix-Plus User's Reference Manual, for the names of terminals supported.)

When you complete these changes, enter

 # boot /gen/cromix

and if the system responds correctly, overwrite the old **cromix.sys** program in the root directory with the new version:

 # move -vf /gen/cromix.sys /cromix.sys

MAKING A NEW BOOT DISK

To create a new boot diskette for the new **cromix.sys** program generated in the previous sections, insert a blank floppy disk (new or used) in drive A. The disk must NOT be write-protected (an 8" disk must have a silver sticker on the bottom edge; a 5-1/4" disk must not have a sticker on its upper right edge). Enter the following command sequence:

```
# initflop -v fda (use sfda for a 5-1/4" disk)
# makfs fda
# wboot fda
# mount fda /drive
# copy -v /cromix.sys /drive
# unmount fda
```

When the drive select light goes off, remove the diskette and protect it against writing. From now on, use this diskette to cold boot the system (use the original boot diskette only if the new one doesn't work). To check the new diskette:

22

- enter **shutdown** (or **kill** -2 -1 if no one is using the system)
- insert the new boot disk in drive A (or whatever)
- reset the system
- press RETURN a few times if necessary

The system should respond correctly and prompt you for the time (if you added the **time -s** command to the **/etc/startup.cmd** file). You should now be rooted on the hard disk.

MAKING BACKUP COPIES OF THE RELEASE DISKS

To make backup copies of the Cromix-Plus release disks, you need several blank disks (new or used) that are not write-protected.

If you have two floppy drives of the appropriate size, put Cromix-Plus Disk #1 in drive A, put a blank disk in drive B, and enter (use **sfda** for 5-1/4" drives):

```
# initflop -v fdb
# rcopy -vtp /dev/fda /dev/fdb
```

Remove the diskettes from both drives, and repeat this procedure for all of the Cromix-Plus disks.

If you have just one floppy disk drive, put disk #1 in drive A and copy it to the **/tmp** directory (or any other empty directory) as follows (use **sfda** for 5-1/4" disks):

```
# d /tmp
# mount -r fda /drive
# cptree -v /drive .
# unmount fda
```

Now replace the disk in drive A with a blank disk, and enter:

```
# initflop -v fda
# makfs fda
# mount fda /drive
# cptree -v . /drive
# unmount fda
```

23

Remove the disk from drive A, install a write-protect sticker, and delete the files in the **/tmp** directory:

 # d /
 # deltree -a /tmp

Go to an empty directory such as **/tmp** and copy each of the remaining Ftar format Cromix-Plus disks from drive A:

 # d /tmp
 # ftar -vx /dev/fda .

When all of the Ftar floppies are stored in the **/tmp** directory (you need about 1.5 Mbytes of disk space), initialize the required number of floppies in advance:

 # initflop -v fda

When the floppies are initialized, insert the first floppy and execute:

 # ftar -vc /dev/fda *

Whenever one floppy is full the Ftar utility will prompt for another. Remove the one already written, insert the next one, and press RETURN. If any problems should occur, the entire procedure must be redone from the beginning.

Chapter 3

CROMIX PERIPHERALS; SOFTWARE CHANGES

This chapter describes the software changes required to add, or remove terminals, modems, and printers. Be sure to make the appropriate software changes in this chapter, and all hardware changes in chapters 4 and 5, before you re-boot the system; otherwise you may activate software changes that are incompatible with the on-line hardware.

SINGLE-USER VERSUS MULTIUSER SYSTEMS

The system set up in chapter 2 is single-user because only one terminal is on-line (the system console connected to the floppy disk controller board). For a multiuser system, install one terminal for each additional user. With the Passwd utility, assign each user a name, password, user identification number (UID), group identification number (GID), and a home directory (refer to the Cromix-Plus User's Reference Manual, part number 023-5013).

One person can log in on several terminals at once, each time using the same user name and password, because the Cromix kernel associates not only a UID with each process, but a terminal number as well.

CREATING AND DELETING DEVICE FILES

When exchanging I/O with an on-line peripheral, the operating system first looks for the peripheral's device file in the /dev directory. The device file defines the peripheral's type (B = block device, C = character device) and major:minor device number. The operating system uses the device type and device numbers to run the correct device driver routine for that peripheral. As shown in appendix A, each peripheral is assigned a unique device filename (e.g., **fda**, **std0**, **lpt1**, and so on).

After editing the **sysdef** file for the device drivers your system requires (refer to the section "Customizing Cromix-Plus" in chapter 2), the next software modification is to create or delete selected device files.

Creating Device Files

The /dev directory on the factory-shipped system disk contains several device files for each device type, but you may wish to create additional device files to avoid I/O-based port conflicts among interface boards, or to assign special names to your peripherals.

25

Follow this procedure to create a new device file in directory **/dev**:

1. Log in as privileged user **system** (only privileged users can alter **/dev** files).

2. Consult appendix A for the major:minor device numbers of the new device.

3. Run the Makdev utility to create the device file as shown below. This sample command specifies, in order, device name = **mtty32**, device type = **c** (for character device; use **b** for block device), major device number = **2**, minor device number = **159**.

   # **makdev mtty32 c 2 159**

4. Run the Chowner utility to change ownership of the device file from privileged user **system** (**system** currently owns it because you, as a privileged user, just created it) to **bin**, as shown below. All device files must be owned by user **bin**.

   # **chowner bin mtty32**

5. Run the Access utility to define the access privileges of the various categories of users. Model the access privileges following other similar device in the **/dev** directory. For example:

   # **access rewa.wa.wa mtty32**

6. Run the List utility to verify that the device file is in directory **/dev**:

   # **ls -l /dev**

   The Cromix OS should now display the complete list of device files in directory **/dev** using **long list** format (because of the **-l** option in the command line). Verify that the device file just created is present, and that it is owned by **bin** (the file owner appears in the field to the far right).

7. If you create a device file for a hard disk partition, be sure to make a file structure with the Makfs utility (e.g., **makfs std2**).

Deleting Device Files

If a device file and device driver are present, but the physical device is not attached to the system, the operating system might hang up, waiting indefinitely for an I/O exchange that never occurs. From directory **/dev**, delete every device file (except the system device files listed in appendix A) that does not have a corresponding I/O device attached to the system:

1. Log in as privileged user **system** (only privileged users can alter **/dev** files).

2. Run the List utility to examine all device files in directory **/dev**:

   # **ls -m /dev**

26

3. Run the Delete Shell command to delete individual device files from directory **/dev.** The command shown below deletes the four 8-inch floppy disk drive device files, and device files for 5-1/4" floppy disk drives **sfdc** and **sfdd.**

    # del   /dev/fd[a-d]   /dev/sfd[cd]

    Continue in this way until all unused device files are deleted.

4. List the **/dev** directory again to verify that all unused device files have been deleted.

CHANGING TERMINALS, PRINTERS, AND MODEMS

Make the following software changes to alter the number and type of peripherals attached to your system:

1. Include device drivers in program **cromix.sys** for all on-line terminals, printers, and modems, and make a boot disk (refer to the section "Customizing Cromix-Plus" in chapter 2).

2. Create or delete device files in the **/dev** directory to reflect your new peripheral configuration; change the owner of created files to user **bin** (refer to the previous section).

3. If necessary, edit the **/etc/iostartup.cmd** file to load the **IOP, OCTART,** or other device drivers.

4. For modems or terminals, edit the **/etc/ttys** file to set the baud rates of the on-line devices.

5. Optionally change the system console and system printer by changing the links to files **/dev/console** and **/dev/prt.**

6. Optionally establish operating modes (parities, margins, column widths, etc.) with commands in the **/etc/startup.cmd** file.

7. Boot the new **cromix.sys** program to activate all software and hardware changes.

Steps 3 through 6 are discussed in the following sections.

CHANGING THE /ETC/TTYS FILE

As part of the boot process, the operating system automatically examines the **/etc/ttys** file to determine which terminals and modems are on-line (users can only log in on terminals and modems which are on-line). You can force the Cromix OS to re-examine this file at any time by running the **kill -1 1** command (the Kill arguments are numbers, not letters).

To change the **/etc/ttys** file, follow these steps:

1.   Edit the **/etc/ttys** file to disable the terminals you wish to change.

2.   Enter the **kill -1 1** command.

3.   Edit the **/etc/ttys** file to change the characteristics of the disabled terminals (and enabling them at the same time, if you wish).

4.   Enter the **kill -1 1** command again to make the changes take place.

You can edit the **/etc/ttys** file by running the Screen utility (e.g., **screen /etc/ttys**). Each entry in the file corresponds to a potential user terminal or modem: **ttys** are terminals connected to 64FDC, 16FDC, or TU-ART boards; **qttys** are terminals connected to OCTART or QUADART boards; and **mttys** are remote terminals connected through modems to QUADART boards. The factory-shipped **/etc/ttys** file is shown in table 3-1.

Table 3-1: THE /ETC/TTYS FILE

FACTORY SHIPPED		QUADART EXAMPLE		TU-ART EXAMPLE	
1:n	:tty1:dumb	0:n	:tty1:dumb	1:9600	:tty1:dumb
0:a	:tty2:dumb	0:a	:tty2:dumb	1:9600	:tty2:dumb
0:a	:tty3:dumb	0:a	:tty3:dumb	1:9600	:tty3:dumb
0:a	:tty4:dumb	0:a	:tty4:dumb	1:9600	:tty4:dumb
0:a	:tty5:dumb	0:a	:tty5:dumb	1:9600	:tty5:dumb
0:a	:tty6:dumb	0:a	:tty6:dumb	0:a	:tty6:dumb
0:a	:tty7:dumb	0:a	:tty7:dumb	0:a	:tty7:dumb
0:a	:tty8:dumb	0:a	:tty8:dumb	0:a	:tty8:dumb
0:a	:tty9:dumb	0:a	:tty9:dumb	0:a	:tty9:dumb
0:a	:qtty1:dumb	1:19200	:qtty1:dumb	0:a	:qtty1:dumb
0:a	:qtty2:dumb	1:19200	:qtty2:dumb	0:a	:qtty2:dumb
0:a	:qtty3:dumb	1:19200	:qtty3:dumb	0:a	:qtty3:dumb
0:a	:qtty4:dumb	0:a	:qtty4:dumb	0:a	:qtty4:dumb
0:a	:qtty5:dumb	0:a	:qtty5:dumb	0:a	:qtty5:dumb
0:a	:qtty6:dumb	0:a	:qtty6:dumb	0:a	:qtty6:dumb
0:a	:qtty7:dumb	0:a	:qtty7:dumb	0:a	:qtty7:dumb
0:a	:qtty8:dumb	0:a	:qtty8:dumb	0:a	:qtty8:dumb
0:a	:mtty1:dumb	0:a	:mtty1:dumb	0:a	:mtty1:dumb
0:a	:mtty2:dumb	0:a	:mtty2:dumb	0:a	:mtty2:dumb
0:a	:mtty3:dumb	0:a	:mtty3:dumb	0:a	:mtty3:dumb
0:a	:mtty4:dumb	1:1200	:mtty4:dumb	0:a	:mtty4:dumb
0:a	:mtty5:dumb	0:a	:mtty5:dumb	0:a	:mtty5:dumb
0:a	:mtty6:dumb	0:a	:mtty6:dumb	0:a	:mtty6:dumb
0:a	:mtty7:dumb	0:a	:mtty7:dumb	0:a	:mtty7:dumb
0:a	:mtty8:dumb	0:a	:mtty8:dumb	0:a	:mtty8:dumb

Each file entry consists of four fields, separated by colons. Reading from left to right:

1. The first field contains either a 0 (zero) or a 1 (one). A **0** takes the device off-line (disconnected); a 1 puts the device on-line (connected).

2. The second field defines the device's baud rate. A number in this field (like 9600) sets the baud rate to the specified value. Permissible values are 110, 300, 600, 1200, 2400, 4800, 9600, and 19200 baud (9600 maximum for **tty1**). An **a** (for auto baud) in this field makes the operating system wait for RETURN characters from the device, and then adjust the channel baud rate to match that of the device. An **n** (for no change) initially appears in this field for device **tty1** only; this makes the Cromix OS use the baud rate determined by program RDOS when **tty1** is initially used as the system console to cold boot the system.

3. The third field contains the name of the device file which must be present in the **/dev** directory.

4. The fourth field is a four-character terminal name, which describes its capabilities, e.g., "dumb", "3102", "C-10", and "C-05". You may use any name you find listed in the **/etc/termcaps** file. If you want to use a terminal not described in the **/etc/termcaps** file, you must first designate the name in the **/etc/termcaps** file, then use the new name for the fourth field of the **/etc/termcaps** file. (Note that the names are case-sensitive.) Setting this field correctly is mandatory to enable some programs to work correctly (the Shell and CE editors, for example). Without defining the terminal type in the **/etc/termcaps** file, the user must use the Term command to define his terminal each time he logs in.

The following two examples illustrate how to change entries in the **/etc/ttys** file.

QUADART/OCTART Example

Assuming that you have three computer terminals and one modem connected to a QUADART or OCTART board, change the **/etc/ttys** file as follows:

1. As privileged user **system**, enter:

    # screen   /etc/ttys

2. Use the Screen X (Xchange) command and take **tty1** off-line by replacing the 1 with a 0 in the first field (see the middle two columns of table 3-1). **Important:** In an all-Quadart or Octart terminal system such as this, one of the Quadart or Octart terminals **must** be linked to the system console (see below), otherwise the system cannot be booted (there is no terminal for the boot dialogue). If you specify a default root device, you do not need a terminal connected to the 64FDC/16FDC. However, only the FDC terminal can display disk I/O error messages.

29

3.   Likewise, put **qtty1, qtty2, qtty3,** and **mtty4** on-line by replacing **0** with **1** four times, as shown in table 3-1.

4.   Specify a baud rate for all channels with a fixed data rate to make the login message appear automatically; otherwise, users have to press the RETURN key to establish the channel data rate. Set terminals and modems to the maximum data rate (19200 baud).

5.   Change the qtty terminal names to the appropriate names describe above. Mtty terminals should remain "dumb" because the terminal type from which a user may call the system is unknown. The user must then uses the Term command when logging in on an mtty terminal to define his terminal type.

6.   Press **E** (Exit) and **U** (Update) to exit the utility and update the **/etc/ttys** file.

TU-ART Example

Assuming that you have one terminal attached to the 64FDC as the system console, and four terminals attached to two TU-ART boards, change the **/etc/ttys** file as follows:

1.   As privileged user system, enter:

          # screen   /etc/ttys

2.   Use the Screen X command and replace the **n** in the baud rate field for **tty1** with **9600** (see the last column of table 3-1).

3.   Put **tty2, tty3, tty4,** and **tty5** on line by replacing **0** with **1** four times as shown in the table 3-1.

4.   Specify a baud rate for all channels with a fixed data rate to make the login message appear automatically; otherwise, users have to press the RETURN key to establish the channel data rate. Set the terminals to 19200 baud (9600 on **tty1**).

5.   Define the terminal type, as described in the previous sections.

6.   Press **E** (for Exit) and **U** (for Update) to exit the utility and update the **/etc/ttys** file.

CHANGING THE SYSTEM CONSOLE

If you list the files in the **/dev** directory by running the List utility (e.g., **ls -m /dev**), you will see the current set of device files. The device file, **/dev/console** is unusual in that it is not associated with a particular device, but instead may be linked to any one of several devices. When the Cromix OS is booted, the operating system automatically exchanges boot dialogue with **/dev/console**, which in turn may be linked (or connected) to any on-line computer terminal.

List file **/dev/console** and you will see the major:minor device number of the terminal that is currently linked to it (**tty1**, with major:minor device numbers 1:0, is linked to **/dev/console** on the factory-shipped system disks). You should also see a link count of 2 for this file.

Run the Maklink utility to link another terminal to **/dev/console**. Use the **-f** (force) option to delete the current link, and replace it with a new one. If QUADART or OCTART boards interface all of your computer terminals to the system, you must link one of the terminals to **/dev/console**. As a privileged user, enter:

```
# maklink  -f  /dev/qtty1  /dev/console
```

List the **/dev** directory; the major:minor device numbers of **/dev/console** and **qtty1** should be the same (2:0).

PRINTERS

The Cromix OS supports three types of printers: dot matrix printers with a parallel Centronics interface (Cromemco Models 3715 and 3703), typewriter parallel printers (Cromemco Model 3355B), and serial printers (Cromemco models 3032CQ and 3038).

Each potential printer connection is associated with a different device file in directory **/dev** (refer to appendix A). All printer devices must be owned by user **bin**. There are four kinds of printer device files, each managed by a different device driver:

1. An **lpt** (line printer) device is a Centronics-compatible dot matrix printer attached to TU-ART parallel connector J2 or J3, or to PRI board connector J1.

2. A **typ** (typewriter) device is a fully formed character printer attached to PRI connector J2.

3. A **qslpt** is a serial printer attached to OCTART connectors J1 or J2, or QUADART connectors J2, J4, J6, or J8. The permissible data rates for **qslpt** devices are the same as those for **qtty** devices (up to 19200 baud for QUADART, and up to 38400 for OCTART).

4.   An **slpt** is a serial printer attached to a TUART serial connector or to the FDC instead of a terminal.

You can have several printers of different types on-line at the same time.

THE SYSTEM PRINTER

The system printer is the device to which all printer output is automatically sent if no other printer is explicitly specified.

You can make any printer function as the system printer by linking it to device file **/dev/prt**. To do this, run the Maklink utility to link any printer to **/dev/prt**. Use the **-f** (force) option to delete the current link, and replace it with a new one. For example, you can define **typl** (the fully formed character printer connected to PRI #1 connector J2) to be the system printer. As a privileged user enter:

        # maklink -f /dev/typl /dev/prt

When you list the **/dev** directory, the major:minor device numbers of **/dev/prt** and **/dev/typl** should be the same (6:5).

SERIAL PRINTERS

Referring to appendix A, notice that each **qslpt** serial printer can be assigned one of two different major:minor device numbers. For example, **qslpt1** can be assigned either 9:0 or 9:128. The serial printer device driver programs key on the assigned minor device number to select the printer's protocol: if the number is less than 128, the device driver assumes DC1/DC3 (or X-ON/X-OFF) protocol; if the number is at least 128, the driver assumes an ETX/ACK protocol. In both protocols, the printer sends the driver program special characters to control the average data rate, and thus prevent printer overruns.

With the DC1/DC3 protocol, the serial printer sends the device driver program a non-printing DC3 character (ASCII 13h = CONTROL-S = X-OFF) when the printer's internal character buffer is nearly full. This causes the driver program to stop sending characters to the printer. The printer signals its readiness to receive further characters by sending back the driver program a DC1 character (ASCII 11h = CONTROL-Q = X-ON), and the driver program starts sending characters to the printer again in response.

With the ETX/ACK protocol, the driver program normally sends 60 characters to the printer, followed by a non-printing ETX character (ASCII 03h = CONTROL-C). The driver program then stops sending characters and waits for the printer to send back an ACK character (ASCII 06h = CONTROL-F). This causes the driver program to send the next 60 characters, and so on.

Some serial printers which use ETX/ACK protocol also interpret certain character sequences sent to them as **command sequences.** These command sequences are used to enable or disable particular printer features (like setting the column width to 132 columns). Their meanings differ for each printer make and model. In such cases, the device driver must not send an ETX character to the printer in the middle of a command sequence (otherwise, the command sequence becomes garbled).

Notice that each **slpt** serial printer, however, can be assigned one of three different major:minor device numbers (refer to appendix A). For example, **slpt2** can be assigned either 7:2, 7:66, or 7:130. The serial printer device driver programs key on the assigned minor device number to select the printer's protocol: if the number is less than 64, the device driver assumes DC1/DC3 (or X-ON/X-OFF) protocol; if the number is between 64 and 127 (inclusive), the device driver assumes the hardware handshake (DTR) protocol; if the number is greater than or equal to 128, the driver assumes ETX/ACK protocol.

The DC1/DC3 protocol works as described above for **qslpt.**

With the hardware handshaking protocol the driver monitors the DTR line and will not send further characters to the printer until the printer signals on DTR line it is ready to receive more characters.

With ETX/ACK protocol driver will send at most BLKsize characters to the printer followed by the ETX (ASCII 03h = CONTROL-C) character. The driver will then wait for the printer to send back an ACK (ASCII 06h = CONTROL-F) character. This causes the driver program to send another block of characters, and so on.

The number BLKsize is defined by the Mode utility. As a rule the full number (BLKsize) of characters will be sent at once. To prevent an Escape sequence begin broken by the ETX character, each block will be terminated prior to full size if an ESC character is to be sent. As the result the ESC character is always the first character of a block which gives room for an Escape sequence up to BLKsize characters in length.

FILES STARTUP.CMD AND IOSTARTUP.CMD

When you boot the Cromix OS, the Shell runs the **iostartup.cmd** file, and the **startup.cmd** file.

File /Etc/iostartup.cmd

When edited correctly, this file loads the device drivers for the OCTART and IOP boards, and tape drive. As a privileged user, run the Screen utility as follows:

    # screen   /etc/iostartup.cmd

You should see the following display:

```
% Download new Std controller firmware (required)
   /etc/stdload /etc/stdcfirm    std0

% Remove comment signs, "%", in order to load I/O processors
% with quadart driver
% /etc/iopload /etc/quadart.iop io1
% /etc/iopload /etc/quadart.iop io2
% /etc/iopload /etc/quadart.iop io3
% /etc/iopload /etc/quadart.iop io4

% Remove comment signs, "%", in order to load IO processors
% with octart driver
% /etc/octload /etc/octart.iop io1
% /etc/octload /etc/octart.iop io2
% /etc/octload /etc/octart.iop io3
% /etc/octload /etc/octart.iop io4

% Remove comment signs, "%", in order to load IO processors
% with the tape driver.  Use tape16.iop for 64K IOP boards.
% /etc/iopload /etc/tape8.iop io1
% /etc/iopload /etc/tape8.iop io2
% /etc/iopload /etc/tape8.iop io3
% /etc/iopload /etc/tape8.iop io4
```

Use the Screen **x** (Xchange) command to remove the % comment character from each command line you want to activate. For example, if in your system IOP1 and IOP2 control QUADART boards, then edit the file to look like this:

```
% Download new Std controller firmware (required)
   /etc/stdload /etc/stdcfirm    std0

% Remove comment signs, "%", in order to load IO processors
% with quadart driver
   /etc/iopload /etc/quadart.iop io1
   /etc/iopload /etc/quadart.iop io2
% /etc/iopload /etc/quadart.iop io3
% /etc/iopload /etc/quadart.iop io4

% Remove comment signs, "%", in order to load IO processors
% with octart driver
% /etc/octload /etc/octart.iop io1
% /etc/octload /etc/octart.iop io2
% /etc/octload /etc/octart.iop io3
% /etc/octload /etc/octart.iop io4

% Remove comment signs, "%", in order to load IO processors
% with the tape driver.  Use tape16.iop for 64K IOP boards.
% /etc/iopload /etc/tape8.iop io1
```

34

```
% /etc/iopload /etc/tape8.iop io1
% /etc/iopload /etc/tape8.iop io2
% /etc/iopload /etc/tape8.iop io3
% /etc/iopload /etc/tape8.iop io4
```

Press **E** (Exit) and **U** (Update) to exit the utility and update the file **/etc/iostartup.cmd.** You may also wish to delete the backup file **/etc/iostartup.bak.**

File /etc/startup.cmd

This file allows you to automatically set the time, display a message, and set the peripheral operating modes each time the system is booted. For example, the **/etc/startup.cmd** file on the factory-shipped system disk contains the following commands:

```
mode tty1 delay 1        % To prevent C-10 missing from characters
flush 120 &              % Or use SYNC occasionally
time -s
if -r /etc/.warning ty /etc/.warning
ren /etc/warning /etc/.warning >*/dev/null
```

The first line (mode tty1 delay 1) is required to set carriage return delays in driver to prevent a C-10 terminal from occasionally losing characters. Systems not using C-10's do not require this line.

The second line (flush 120 &) causes the flush utility to be run in the background flushing the system write buffers every 120 seconds. The **sync** command may be used to flush the buffers immediately.

The third line displays the current time on the system console, and then prompts you to enter the correct time.

The fourth line displays a warning message if the system was not previously shut down by the Shutdown command. The contents of file **/etc/.warning** will be displayed if the file has read access (if it exists, it will have read access). Command file **startup.cmd** always renames the file containing the warning message to **/etc/.warning,** while command file **shutdown.cmd** always renames the same file to **/etc/warning** (without the period). If **/etc/.warning** does not exist when **startup.cmd** is run (because **shutdown.cmd** renamed the file), the warning message is not printed. If **/etc/.warning** does exist when **startup.cmd** is run (because **shutdown.cmd** did not rename the file), then the warning message is printed.

The fifth line changes the name of file **/etc/warning** (if it exists) to **/etc/.warning,** and all dialogue and error messages normally sent to the system console by this command are discarded by redirecting them to **/dev/null,** a null device.

By including commands which run the Mode utility, you can set the operating characteristics of system devices. Note that the operating system assigns a default set of Mode characteristics to each peripheral. You can examine the current set of attributes by running the Mode utility with only the device name as an argument (e.g., **mode   qtty1**).

Run the Screen utility to edit the **/etc/startup.cmd** file. The following command lines are representative entries:

```
logerr 30
mode   qslpt2   baud   9600   tab   ff   -pa
```

The first command turns on MCU/MSU memory error correction and starts a background job to examine the MCU board every 30 seconds. If any error is stored in the board, the error message will be appended to the **/etc/msu_errs** file.

The second command fixes the data rate of serial line printer **qslpt2** at 9600 baud, expands TAB characters (ASCII 09h = CONTROL-I) to the correct number of SPACE characters (ASCII 20h), expands FORMFEED characters (ASCII 0Bh = CONTROL-L) to sufficient newline characters (ASCII 0Ah = CONTROL-J) to bring the carriage to the top of the next page, and disables Pause (so that printing is not stopped after one page of text is sent to the printer).

Chapter 4

INSTALLING TERMINALS, PRINTERS, AND MODEMS

This chapter describes how to install and test terminals, printers, and modems on a Cromix-Plus system. The I/O interface boards that control these peripherals are discussed in chapter 5.

TERMINALS

All terminals used in Cromix systems must exchange ASCII-coded characters, use RS-232C interface circuits (not 20-mA current loop), and have a DB-25 plug on one end of the terminal cable. To use the C-10 computer as a terminal, refer to the Cromemco C-10 Personal Computer User Manual, part number 023-6037.

Configure each terminal as follows:

1. Set the baud rate to 110, 300, 600, 1200, 2400, 4800, 9600, 19200, or 38400, depending on the I/O board used to control the terminal. The maximum baud rates are as follows:

I/O Board	Max. Baud Rate
OCTART	38,400
QUADART	19,200
TU-ART	19,200 (9,600 for tty1)
64FDC	9,600
16FDC	9,600

The baud rate must also be set in the **/etc/ttys** file, as described in chapter 3.

2. Use two stop bits for 110 baud; otherwise, use one stop bit.

3. Full duplex operation.

4. Seven data bits per character, excluding the parity bit.

5. Either space or mark parity (parity bit is reset to logic 0, or set to logic 1, respectively).

6. RETURN as the line termination character.

7. No automatic linefeeds.

37

The MAIN port on most terminals is wired DTE-style, and the AUX port is wired DCE-style (with possibly a fixed baud rate). Install a terminal cable from the MAIN port of each terminal to a DB-25 connector on the system rear panel. Each rear panel connector must be linked internally to the appropriate interface board (refer to the section "I/O Interface Boards" in chapter 5).

The System Console

The system console, the terminal from which you boot the system, is initially connected to the 64FDC/16FDC board. Install the terminal cable from the MAIN port on the system console to the factory-installed rear panel connector coming from J4 on the 64FDC/16FDC board. Set the system console to 9600 baud.

SERIAL PRINTERS

All serial printers exchange ASCII-coded characters, have an RS-232C interface, and have a DB-25 plug on one end of the printer cable. Configure each serial printer as follows:

1. Set the baud rate on a printer as you would for a terminal (refer to the previous section), but check the manufacturer's documentation to avoid exceeding the maximum rate of the printer. The baud rate must also be set with the Mode utility, as described in chapter 3 (the **etc/ttys** file is used only for terminals and modems).

2. Use two stop bits for 110 baud; otherwise, use one stop bit.

3. Seven data bits per character, excluding the parity bit.

4. Either space or mark parity (parity bit is reset to logic 0, or set to logic 1, respectively).

5. No automatic linefeeds.

6. When operating above 300 baud, use either the DC1/DC3 or ETX/ACK protocol. If the data rate is 110 or 300 baud, the serial printer character buffer should never overrun, and no start/stop transmit protocol is required.

7. Circuit DTR (pin 20) strapped ON (spacing high).

8. If the printer has the TOF (Top Of Form) feature, add **mode device-name -ff** to file **/etc/startup.cmd**; if the printer does not have the TOF feature, add **mode device-name ff** to file **/etc/startup.cmd** (refer to the section "Startup.cmd and Iostartup.cmd" in chapter 3).

9. If your printer monitors the CTS (Clear To Send) circuit, install the transmit jumper between CTS and RTS (figure 4-1). Keeping CTS true allows the printer to send DC1 or ACK characters back to the Cromix driver. If the printer does not drive RTS high, strap CTS high by some other means. If the printer does not monitor circuit CTS, the transmit jumper is not needed.

10. If your printer monitors circuits DCD (Data Carrier Detect) and/or DSR (Data Set Ready), install the receive jumper between DCD, DSR, and DTR (figure 4-1). Keeping DCD and DSR true allows the printer to receive characters from the Cromix driver. If the printer does not drive DTR high, strap DCD and DSR high by some other means. If the printer does not monitor either DCD or DSR, the receive jumper is not needed.

Connect the cable from each printer to a DB-25 connector on the system rear panel. Each rear panel connector must be linked internally to the appropriate interface board (refer to the section "I/O Interface Boards" in chapter 5).

Figure 4-1: SERIAL PRINTER CABLE JUMPERS

PARALLEL PRINTERS

All parallel printers must be Centronics-compatible, and have a DB-25 connector on one end of the printer cable. Connect the cable from each printer to a DB-25 connector on the system rear panel. Each rear panel connector must be linked internally to the appropriate interface board (refer to the section "I/O Interface Boards" in chapter 5).
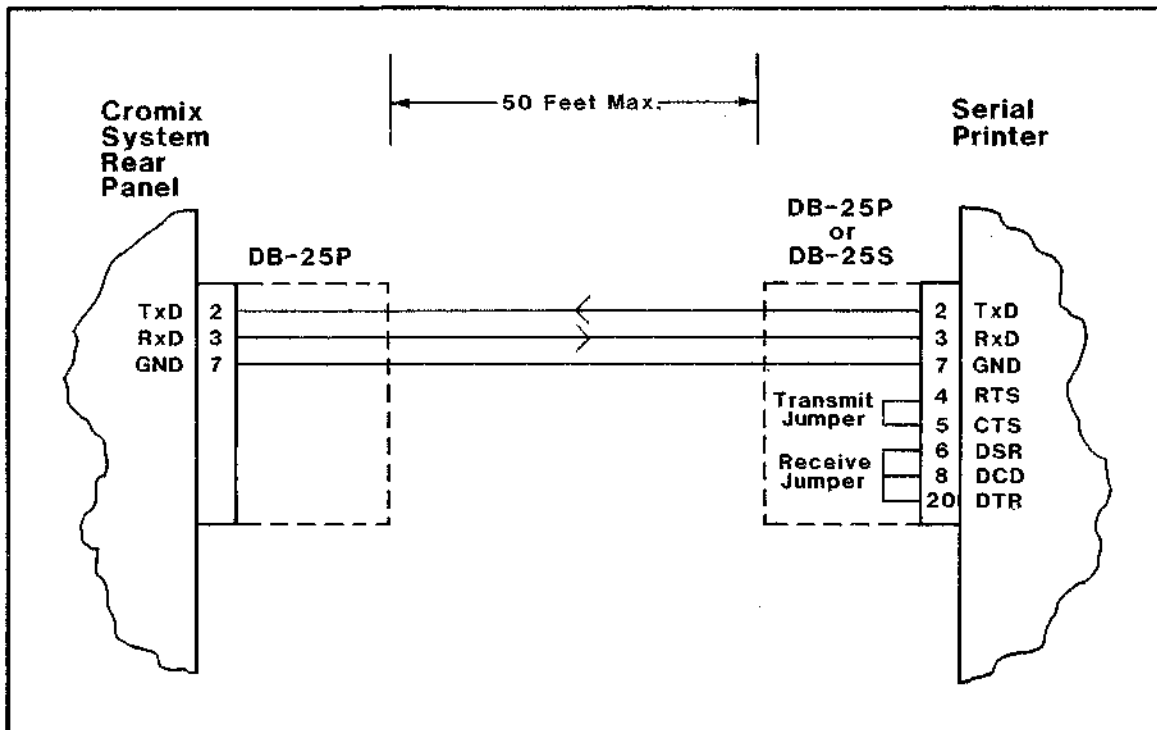
Note: When using a PRI board, cables from other vendors may not connect pin 15 on the PRI to pin 10 on the Cromemco model 3703 and 3715 dot matrix printers.

MODEMS

Quadart

Any asynchronous modem may be used (such as Bell type 103J, 113C, or 212A), provided that the modems at both ends of the link are compatible. Configure the modem per the manufacturer's documentation, and plug the modem cable into a connector on the system rear panel. The rear panel connector must be linked internally to J3, J5, J7, or J9 on the QUADART board.

The following RS-232C circuits are active on the QUADART DCE connectors (J3, J5, J7, and J9): **TxD** (circuit BA) pin 2, **RxD** (BB) pin 3, **RTS** (CA) pin 4, **CTS** (CB) pin 5, **DSR** (CC) pin 6, S-100 Bus Ground (AB) pin 7, **DCD** (CF) pin 8, **TxC** (DB) pin 15, **RxC** (DD) pin 17, **DTR** (CD) pin 20, **RI** (CE) pin 22, and **EXT CK** (DA) pin 24. These connectors also support a special-purpose RS-232C level output line, **CY** pin 11.

If mode attribute **sighup** is on, and the remote modem hangs up, or if either Data Carrier Detect (DCD) or Clear To Send (CTS) are lost before a user hangs up, then a kill signal is sent to all processes started by the user, and the user is automatically logged off. If **sighup** is on and the user logs off normally, circuit DTR is briefly turned off, then turned back on (this hangs up the modem on the Cromix system end, and permits another user to phone in). Strapping DTR high, as allowed by some modems, prevents the driver from hanging up the modem.

Octart

At present, the OCTART board can support an intelligent modem under the following conditions (no device name is required):

1.   The modem-to-OCTART cable (Cromemco part CBL-HAYES P/N 519-0249) must be wired as shown in figure 4-2.

2.   For the Hayes Smartmodem 1200 (and other compatible modems), the user must change configuration switches 1 and 6 to the UP (OFF) position to enable pins 8 and 20; the remaining switches should be in the DOWN (ON) position.

To OCTART
(DB25)

To Modem
(DB25)

Pin		Pin	
2	←——————→	3	Rxt
3	←——————→	2	TxD
4	←——————→	8	DCD
5	←——————→	20	DTR
7	←——————→	7	GND
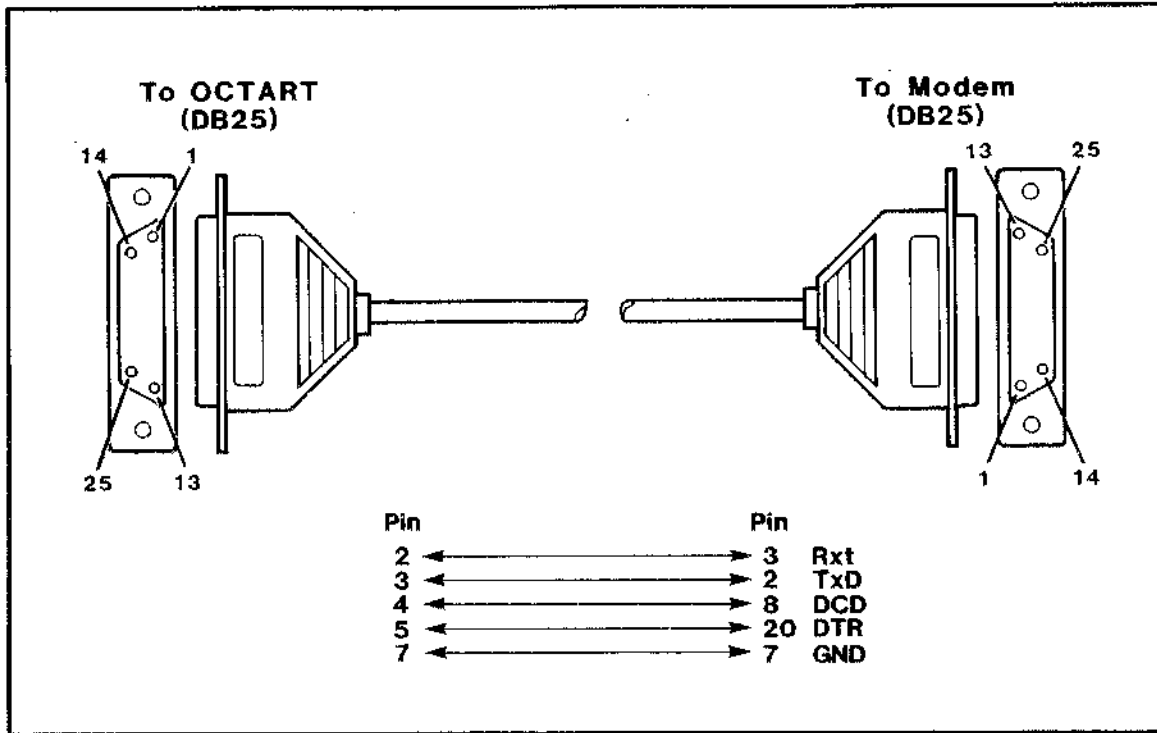
Figure 4-2: MODEM-TO-OCTART CABLE CONNECTION

TESTING PERIPHERAL HARDWARE

If a peripheral does not respond correctly, check the following items (refer to chapters 3 and 5 as necessary):

1. Remove the boot disk and turn off system power. Check all switch settings, jumper options, and cable connections. Be sure that the red cable stripe on all ribbon cables is properly aligned. Be sure that each device is attached to the right connector on the right board (refer to appendix A). If the system does not boot properly, be sure the system console is connected to J4 on the 64FDC/16FDC. If the system detects an error before the drivers are loaded, the diagnostic message is sent only to the 64FDC/16FDC port.

2.   For terminals, check all **/etc/ttys** entries for accuracy. Make sure that a fixed baud rate is specified for the system console (9600 baud maximum for **tty's**, 19200 for **qtty's**). If necessary, modify the file with the Screen utility, and enter the command:

     #   **kill   -1   1**

     to incorporate the change.

3.   Verify that the appropriate device file exists in the **/dev** directory. If not, create one with the Makdev utility.

4.   Check that the system console and system printer are linked to the correct devices for your system.

5.   Verify that the device driver is included in the **cromix.sys** program by entering the command (device **tty1** used as an example):

     # **mode   /dev/tty1**

     If a message reports that there is no device driver, you must generate a new **cromix.sys** program (refer to chapter 3), and reboot the system.

6.   If a device driver is present, the operating modes for the device are displayed. Compare them to the modes selected on the device itself for possible conflicts. If a conflict exists, correct it by changing the options on the device, or by changing the Mode command in the **/etc/startup.cmd** file.

7.   If your system has QUADART or OCTART boards, verify that the percent sign has been removed from the appropriate command line(s) in the **/etc/iostartup.cmd** file. If necessary, use the Screen utility to correct the file, and reboot the system.

If the problem persists, contact your local Cromemco dealer, distributor, or authorized service facility.

Chapter 5

INSTALLING CIRCUIT BOARDS

This chapter describes the cables, switch settings, and jumper-selectable options for all of the Cromemco circuit boards used in standard Cromix-Plus systems.

Refer to your system manual for details on accessing the system card cage, securing connectors to the rear panel, and so on. For more information on a particular board, refer to the appropriate board manual listed in the introduction.

CPU BOARDS

There are two central processor boards to choose from: the 68000(68010)/Z80B-based XPU board and the 68000/Z80A-based DPU board. The XPU is compatible with both the Cromix-Plus and UNIX System V Operating Systems.

The XPU

The XPU board has no switches to set. As shipped, traces at A15 and A14 (figure 5-1) set the power-up and reset address to C000h (the starting address of the ROM-based RDOS program on the 64FDC/16FDC board).

Insert the XPU in any slot in the system card cage. On UNIX systems only, install a 34-conductor cable (part number 519-0062) from the XPU to the XMM board (with the red cable stripe to the left).

The DPU

The DPU board has no switches to set, and no cables to install. As shipped, traces at A15 and A14 (figure 5-2) set the power-up and reset address to C000h (the starting address of the ROM-based RDOS program on the 64FDC/16FDC board). Insert the DPU in any card slot.
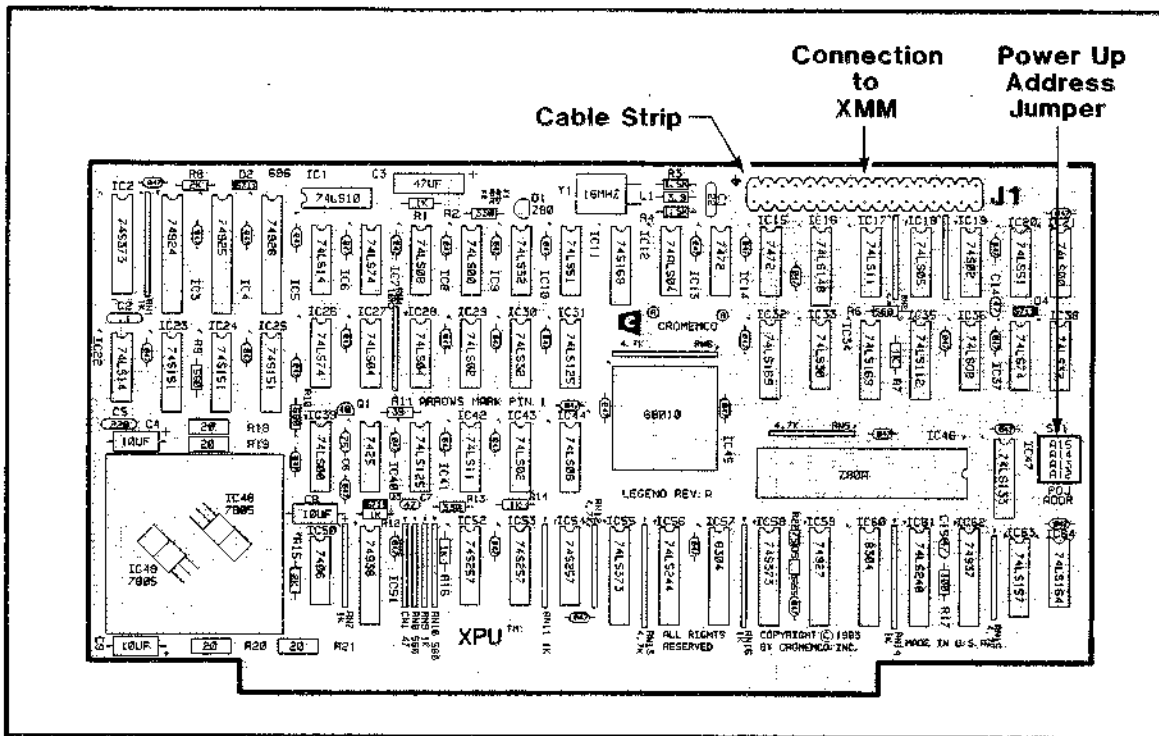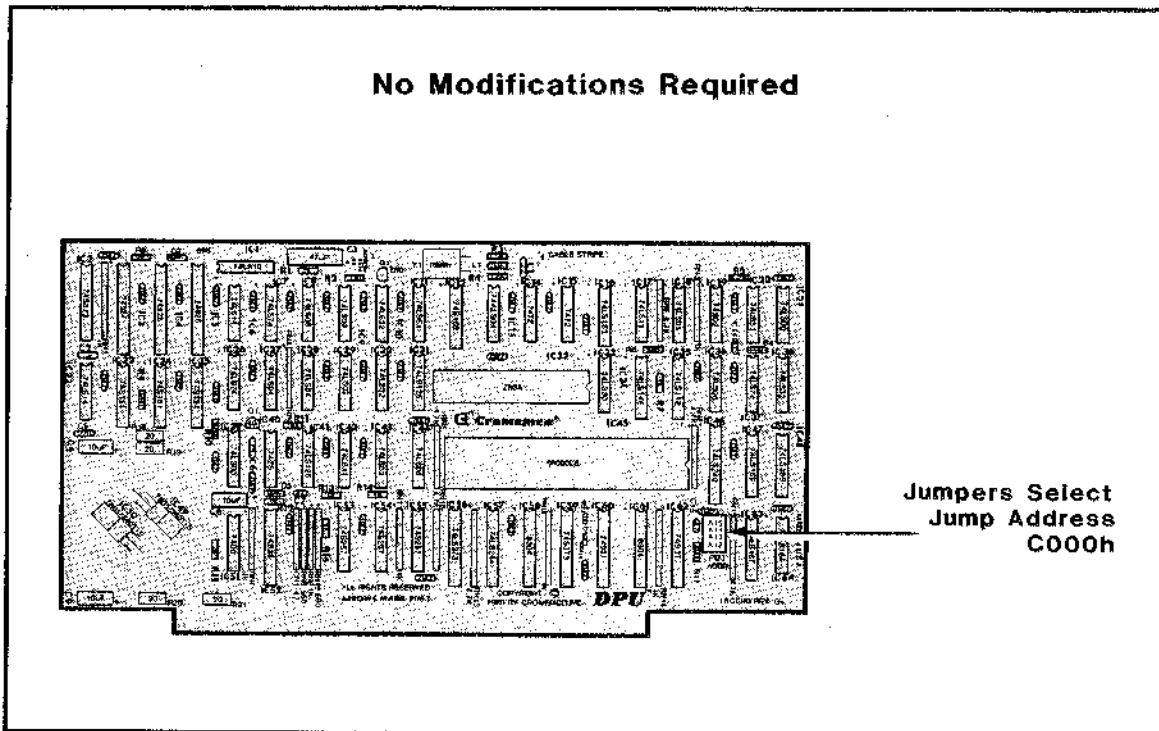
Figure 5-1: THE XPU BOARD

Figure 5-2: THE DPU BOARD

MEMORY BOARDS

There are four memory boards to choose from: the 2048MSU (2 megabytes of RAM), the 512MSU (512 Kbytes of RAM), the 1024KZ (1 megabyte of RAM), and the 256KZ (256 Kbytes of RAM). All are compatible with both the Cromix-Plus and UNIX System V Operating Systems. The 2048MSU and 512MSU are used in conjunction with an MCU board to provide error detection and correction.

The MCU

A revision K or higher MCU controls up to six MSU boards (any combination); earlier revisions of the MCU support only two MSU's. There are no switch settings or jumpers to change unless you have more than one MCU board. For multiple MCU's of revision K or higher, set the I/O port address jumpers as shown in figure 5-3 (the jumpers may be on either side of the board). For multiple MCU's of revision J or lower, refer to Technical Bulletin 023-9131, "Error Correction for Multiple MCU's." A revision K or higher MCU cannot be used with an MCU of revision J or lower.

The 2048MSU

The 2048MSU is configured for use with a revision K or higher MCU board. If you have a revision J or lower MCU board, change the MCU-select jumper as shown in figure 5-4. Set the 2048MSU switches as shown in figure 5-5.

The 512MSU

Set the 512MSU switches as shown in figure 5-6. There are no jumpers to install. A revision B 512MSU board with mod level 2 (or higher) is configured for use with a revision K or higher MCU, and must be modified to work with an MCU of revision J or lower (refer to Technical Bulletin 023-9132, "Using a 512MSU with a Revision J (or Lower) MCU").

The M-Bus Cable

Insert the MCU and MSU boards in adjacent slots of the system card cage. Install an M-bus cable (part number 519-0162 for one MSU, 519-0150 for two MSU's, or 519-0149 for three or four MSU's) from J1 on the MCU board to J1 on the 2048MSU/512MSU (with the red cable stripe to the left). Always attach the first cable connector to the MCU board. On a revision K or higher MCU, ignore the extra reversed connector on the cable; on a revision J or lower MCU, plug an M-bus terminator (part number 520-0128) into the extra connector.

The 1024KZ

The 1024KZ board has no jumpers to set or cables. To install the board, set the 1024KZ switches as shown in figure 5-7.
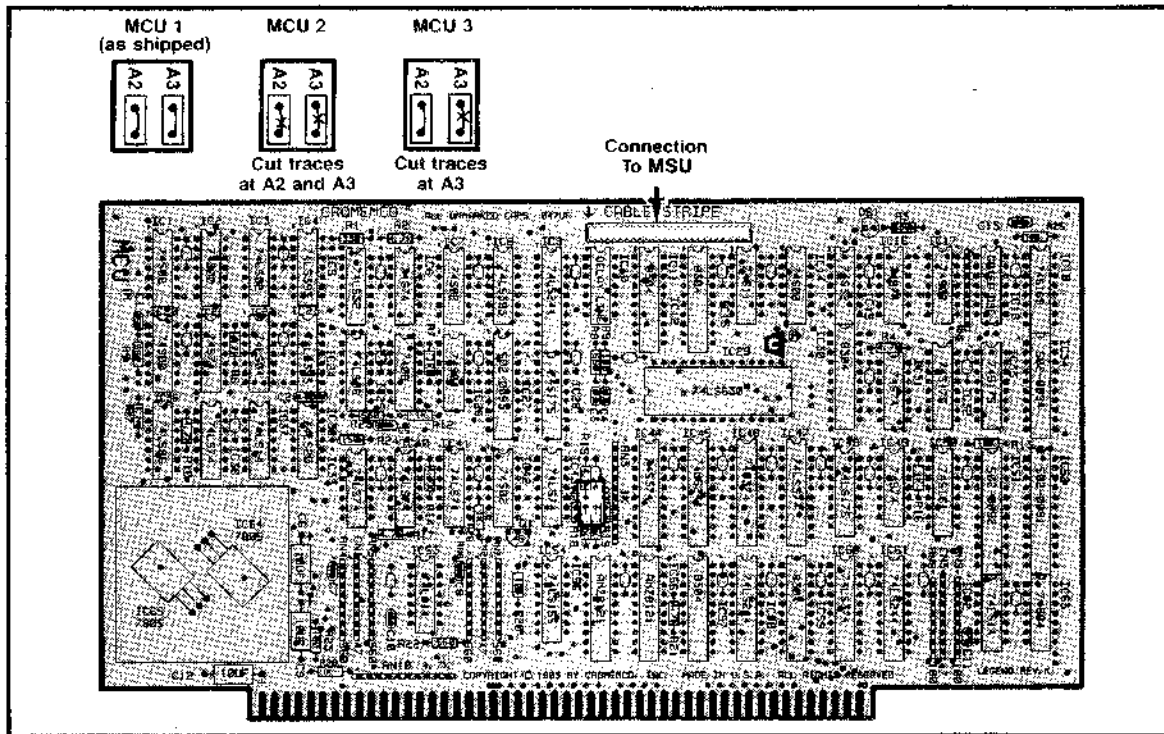
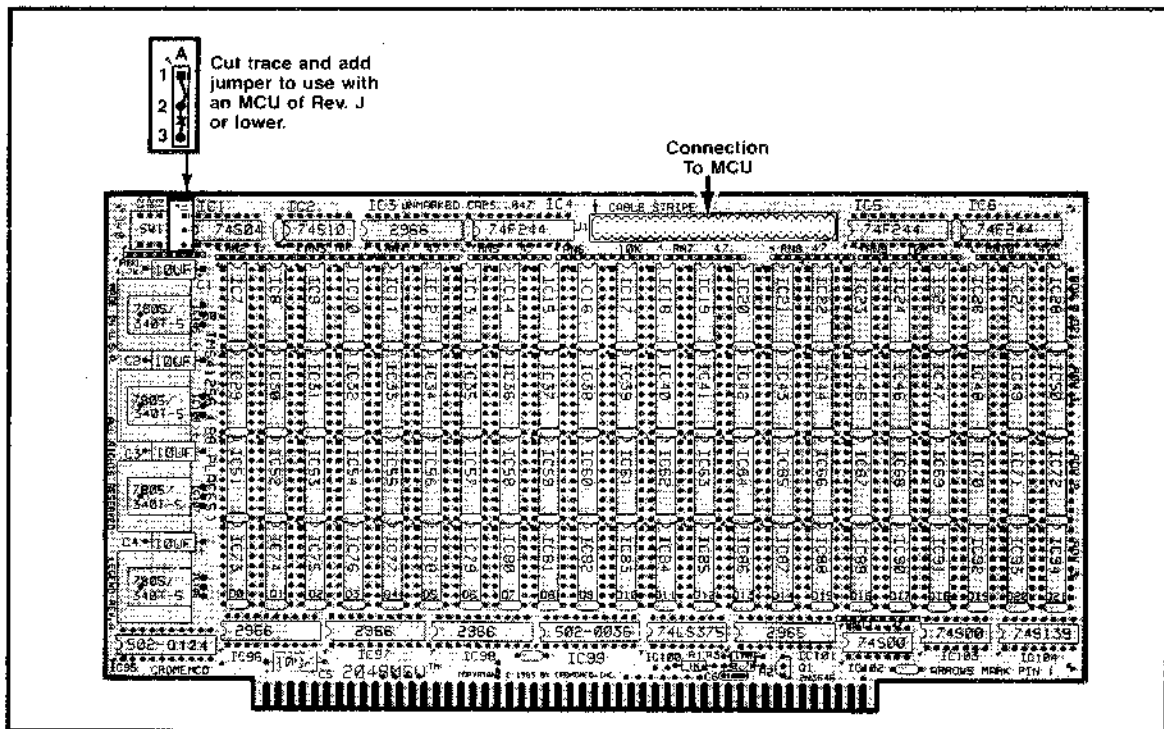**Figure 5-3: REV.K MCU JUMPERS FOR MULTIPLE MCU'S**

Figure 5-4: MCU-SELECT JUMPER ON THE 2048MSU
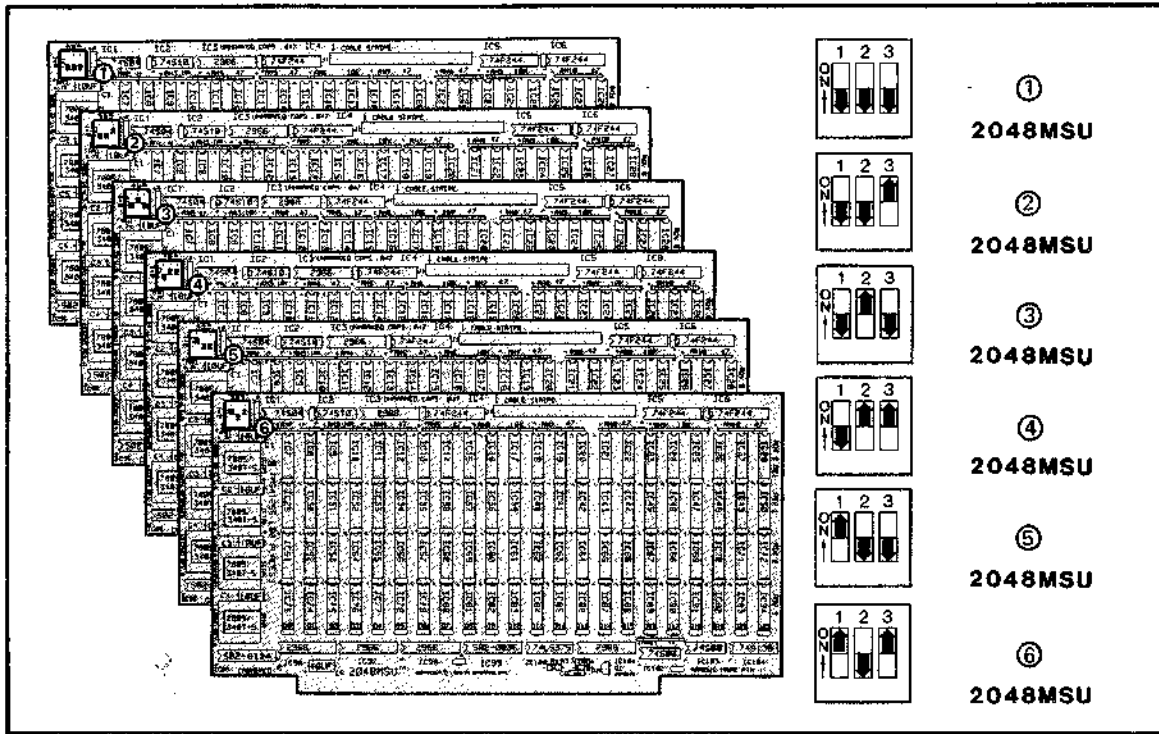
46

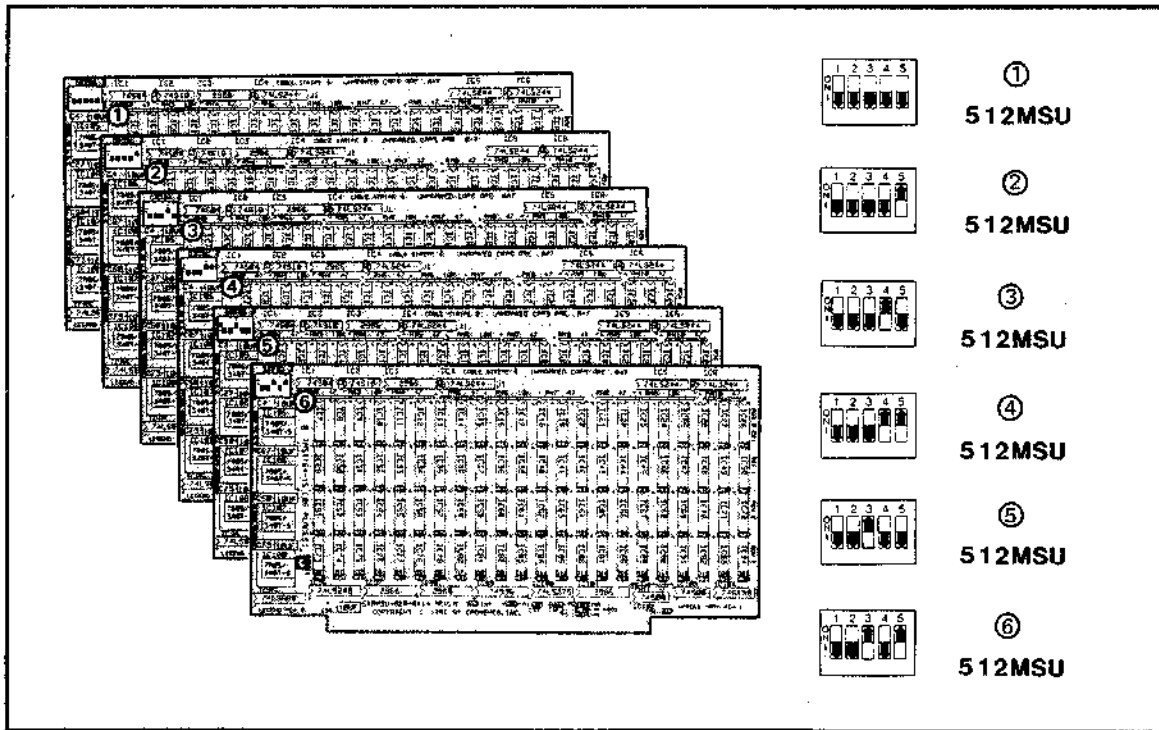**Figure 5-5: 2048MSU SWITCH SETTINGS**

Figure 5-6: 512MSU SWITCH SETTINGS
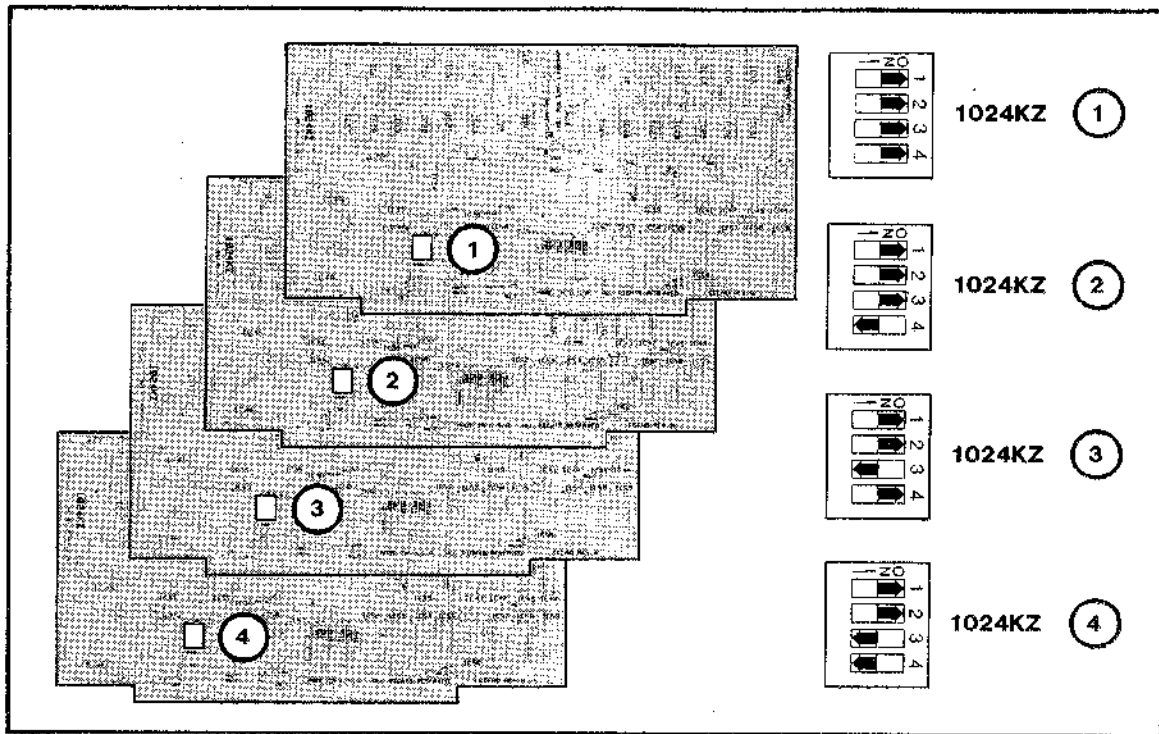
Figure 5-7: 1024KZ SWITCH SETTINGS
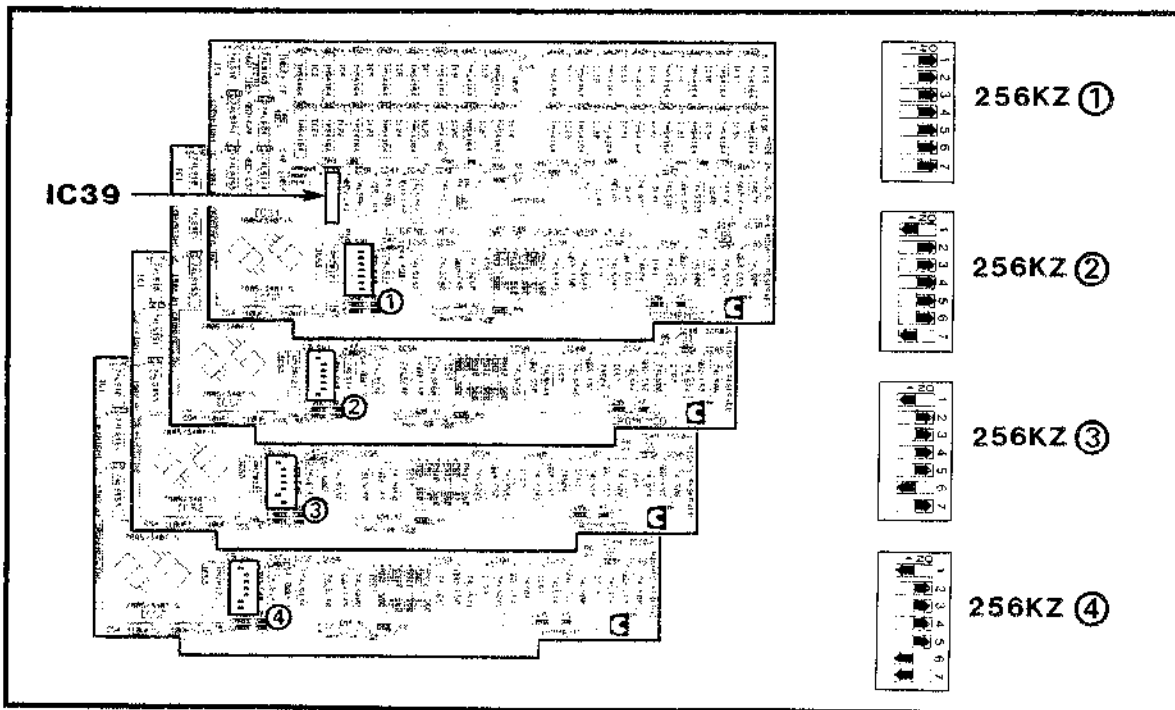
Figure 5-8: 256KZ SWITCH SETTINGS

The 256KZ

The 256KZ board has no jumpers to set, and no cables to install. Set the 256KZ switches as shown in figure 5-8. The PROM at IC39 must be removed from all but the first board.

FLOPPY DISK CONTROLLER BOARDS

Cromemco makes two floppy disk controller boards: the 64FDC and the 16FDC. Both boards use the same cables, and support up to four single- or double-sided, single- or double-density, 5-1/4" or 8" floppy disk drives. Both boards feature the Resident Disk Operating System program (RDOS) and an RS-232C serial channel (initially linked to the system console).

The 64FDC

The RDOS program, stored in ROM on IC25, must be version 02.52 or higher. The standard 64FDC switch settings for RDOS 02.52 through 03.08 (see figure 5-9) configure the board as follows (If your 64FDC has an RDOS ROM version 03.12, use the information that follows these procedures.)

1.   Switch 1 OFF allows RDOS to adjust the serial channel (after receiving a few RETURN characters) to the baud rate of the system console (attached to J4 on the 64FDC). If switch 1 is ON, the baud rate is preset to 300 baud.

2.   Switches 2, 3, and 4 are OFF so that, at power-up or reset, RDOS automatically reads Drive A to find the bootstrap routine. If you want to boot the system from a different drive, use the appropriate switch settings shown below.

Switch 2	Switch 3	Switch 4	
OFF	OFF	OFF	Floppy Disk A
OFF	OFF	ON	Floppy Disk B
OFF	ON	OFF	Floppy Disk C
OFF	ON	ON	Floppy Disk D

3.   Switch 5 OFF prevents RDOS from running the self-test routine on power-up or reset.

The four jumper-selectable options above SW1 (figure 5-9) are factory-set, and should not be changed. They serve the same purpose as the first four switches on the 16FDC board.

RDOS 03.12

For 64FDC's with ROM versions 03.12, use these switch settings:.

	S2	S3	S4	S5
STD31	OFF	OFF	OFF	OFF
STD63	OFF	OFF	ON	OFF
STD0	ON	OFF	OFF	ON
STD1	ON	OFF	ON	ON
STD2	ON	ON	OFF	ON
STD3	ON	ON	ON	ON
STD32	ON	OFF	OFF	OFF
STD33	ON	OFF	ON	OFF
STD34	ON	ON	OFF	OFF
STD35	ON	ON	ON	OFF
FLOP A	OFF	OFF	OFF	ON
FLOP B	OFF	OFF	ON	ON
FLOP C	OFF	ON	OFF	ON
FLOP D	OFF	ON	ON	ON

The 16FDC Board

The RDOS program, stored in ROM on IC25, must be version 02.01 or higher. The standard SW1 switch settings (see figure 5-10) configure the 16FDC board as follows:

1. Switch 1 OFF loads RDOS into memory at address C000h on power-up or reset.

2. Switch 2 ON switches RDOS out of memory after it loads and runs the bootstrap program.

3. Switch 3 ON allows RDOS, on power-up and reset, to automatically load the bootstrap program from the diskette in Drive A. If you do not have RDOS version 3.08 or higher, Drive A is always the boot drive.

4. Switch 4 OFF allows the floppy diskettes to be formatted.

5. Switch 5 OFF allows RDOS to adjust the serial channel (after receiving a few RETURN characters) to the baud rate of the system console (attached to J4 on the 16FDC). If switch 1 is ON, the baud rate is preset to 300 baud.

6. If you have RDOS version 3.08 or higher, switches 6, 7, and 8 serve the same purpose as switches 2, 3, and 4 on the 64FDC board.
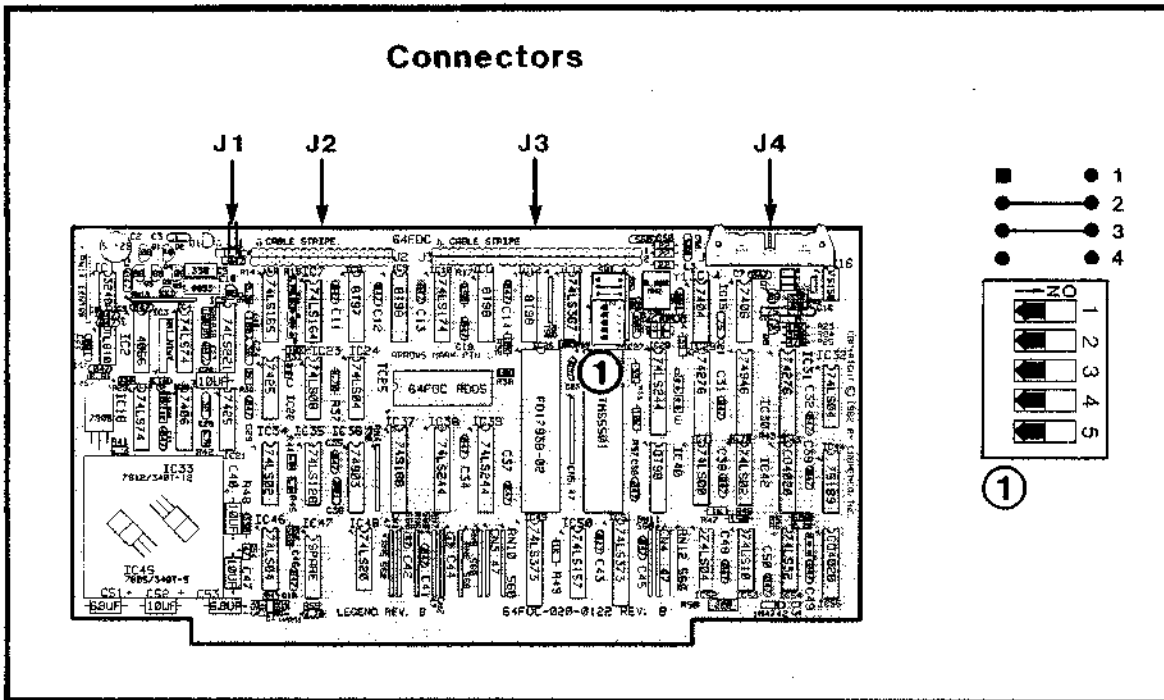
Figure 5-9: 64FDC SWITCHES AND JUMPERS
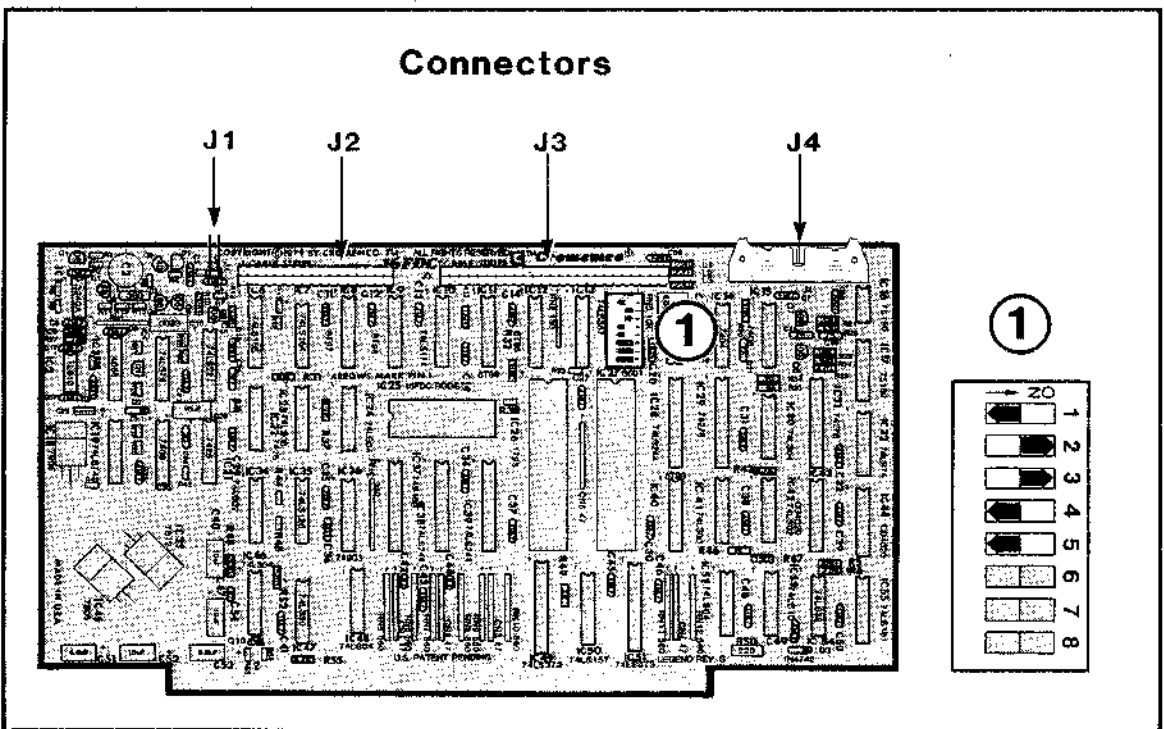
Figure 5-10: 16FDC SWITCH SETTINGS

The 64FDC/16FDC Cables

A 26-conductor cable (part number 519-0086 on CS1 & CS100; 519-0017 on CS2; 519-0008 on CS3 & CS300) is factory installed from J4 on the 64FDC/16FDC (with the red cable stripe on the left) to a rear panel connector slot. The system console plugs into the connector on the rear panel coming from J4 on the 64FDC/16FDC.

On standard configurations, either a 50-conductor cable (part number 519-0135) is installed from J3 to the 8" floppy disk drive(s), or a 34-conductor cable (part number 519-0106 on CS1; 519-0121 on CS1H; 519-0018 on CS2) is installed from J2 to the 5-1/4" floppy disk drive(s).

Connect the priority interrupt cable to J1 on the 64FDC/16FDC (refer to the last section of this chapter).
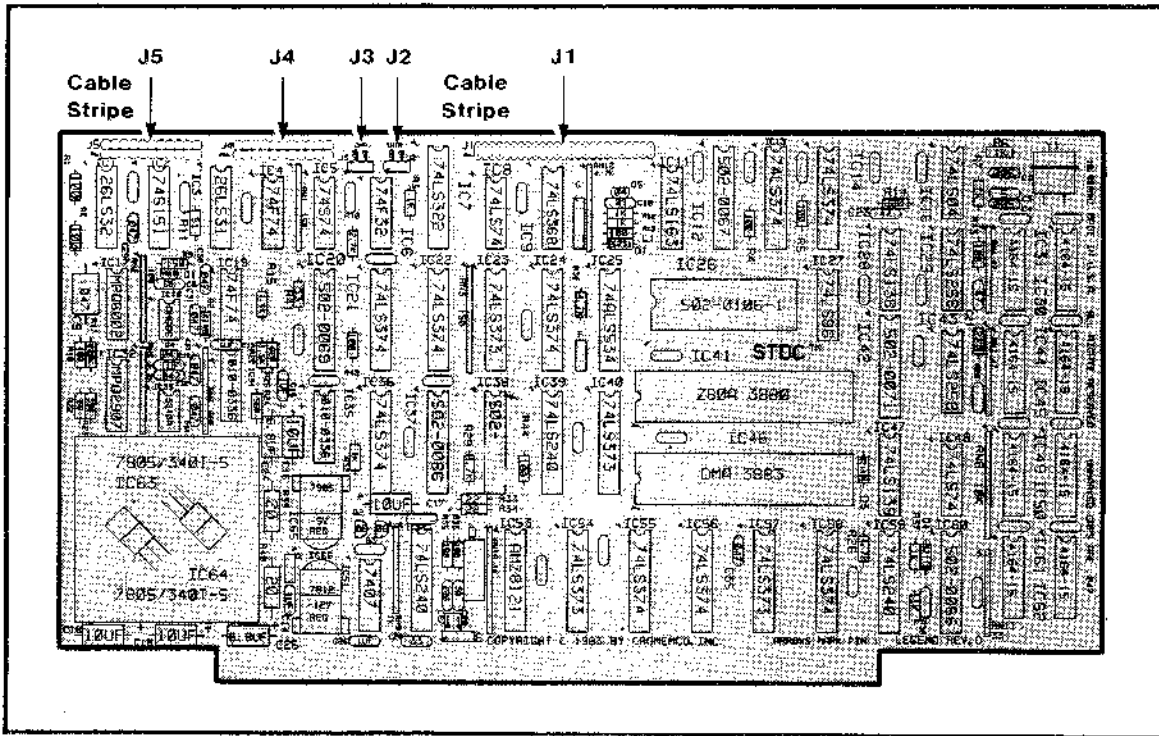
Figure 5-11: THE STDC BOARD

HARD DISK CONTROLLER

The STDC

The STDC hard disk controller board (figure 5-11) uses an ST-506 standard interface to support a wide variety of hard disk drives. There are no switch settings required. When shipped separately, the STDC is configured for use with a DPU or XPU central processor board. The STDC firmware ROM (IC26) should be part number 502-0106-5 or higher.

Install a 34-conductor cable (part number 519-0191 on CS1 and CS100; 519-0193 on CS2; 519-0195 on CS3 and CS300) from J1 on the STDC (with the red cable stripe on the left) to the edge connector on the drive. If you have two hard disks, install a dual-drive control cable (part number 519-0225). Install a 20-conductor cable (part number 519-0190 on CS1 and CS100; 519-0192 on CS2; 519-0194 on CS3 and CS300) from J5 on the STDC to the edge connector on the drive. If you have two hard disk drives, install another data cable from J4 on the STDC to the second hard disk.

Connect the priority interrupt cable to J2 on the STDC (refer to the last section of this chapter).

I/O INTERFACE BOARDS

Cromemco makes several I/O interface boards: OCTART, IOP, QUADART, TU-ART, and PRI. These boards support modems, terminals, and printers. The QUADART is always used in conjunction with an IOP board.

The OCTART

Up to four OCTART boards can be installed in a single system, and one OCTART supports up to eight terminals or serial printers (in any combination).

If you change the standard ROM (IC10) to one with an access time of 150 nSec or less, cut the trace shown in figure 5-12. The OCTART switch settings are shown in figure 5-13. When using OCTART and IOP boards in the same system, each board must have a different base port address (compare figures 5-13 and 5-14).

Insert the OCTART into any slot of the system card cage. Install two 26-conductor cables (part number 519-0086 on CS1 & CS100; 519-0017 on CS2; 519-0008 on CS3 & CS300) from OCTART connectors J1 and J2 to any of the rear panel connector slots. Mark the rear panel to identify the OCTART connectors. To support the full eight terminals or serial printers, plug an OCTART cable (part number 519-0184) into both rear panel connectors, and connect four device cables to each OCTART cable. Without the OCTART cable, you can support two devices (qtty1 and qtty5) by plugging their cables directly into the two rear panel connectors.

Connect the priority interrupt cable to J3 (refer to the last section of this chapter).

Figure 5-12: THE OCTART BOARD

Figure 5-13: OCTART SWITCH SETTINGS

The IOP

Up to four IOP boards can be installed in a single system, and each IOP controls up to four QUADART boards. Be sure that the ROM in IC9, which holds the IOP monitor program (IOPMON), is labeled version 03.00 or higher. Set the IOP switches as shown in figure 5-14. When using OCTART and IOP boards in the same system, each board must have a different base port address (compare figures 5-13 and 5-14).

Insert the IOP in the card cage with at least one empty slot adjacent to it. Install a 2-connector C-bus cable (part number 519-0100) from J1 on the IOP (with the red cable stripe on the left) to the 50-pin connector on the QUADART. If you have two QUADART boards, use the 3-connector cable (part number 519-0181); if you have three or four QUADART boards, use the 5-connector cable (part number 519-0101). Always attach the first cable connector to the IOP board.

Connect the priority interrupt cable to J2 on the IOP (refer to the last section of this chapter).

The QUADART

Up to sixteen QUADART boards can be installed in a single system (four QUADART's for each IOP), and each QUADART supports up to four modems, terminals or serial printers (in any combination).

Set the QUADART switches as shown in figure 5-15. If you have multiple QUADART boards, change the jumpers on the plug in IC28 (see figure 5-15). For QUADART 1, 5, 9, and 13, the plug is correct as shipped; for QUADART 4, 8, 12, and 16, the plug must be removed.

Insert up to four QUADART boards in successive card slots next to each IOP board, and install the C-bus cable from J1 on the IOP (with the red cable stripe on the left) to J10 on each QUADART. Always attach the first cable connector to the IOP board. If you have four QUADART's controlled by one IOP, install a C-bus priority cable (part number 519-0029) from J1 on QUADART 3, 7, 11 or 15 to J1 on QUADART 4, 8, 12 or 16, respectively (see figure 5-15).

Install up to four 26-conductor cables (part number 519-0086 on CS1 & CS100; 519-0017 on CS2; 519-0008 on CS3 & CS300) from QUADART connectors J2 through J9 to any of the rear-panel connector slots. Plug the terminals or serial printers into the rear panel connectors coming from J2, J4, J6, and J8; plug any modems into connectors from J3, J5, J7, and J9. If you use J2, you cannot use J3 (and vice-versa); if you use J4, you cannot use J5, and so on. Mark the rear panel to indicate the device associated with each connector (refer to appendix A).
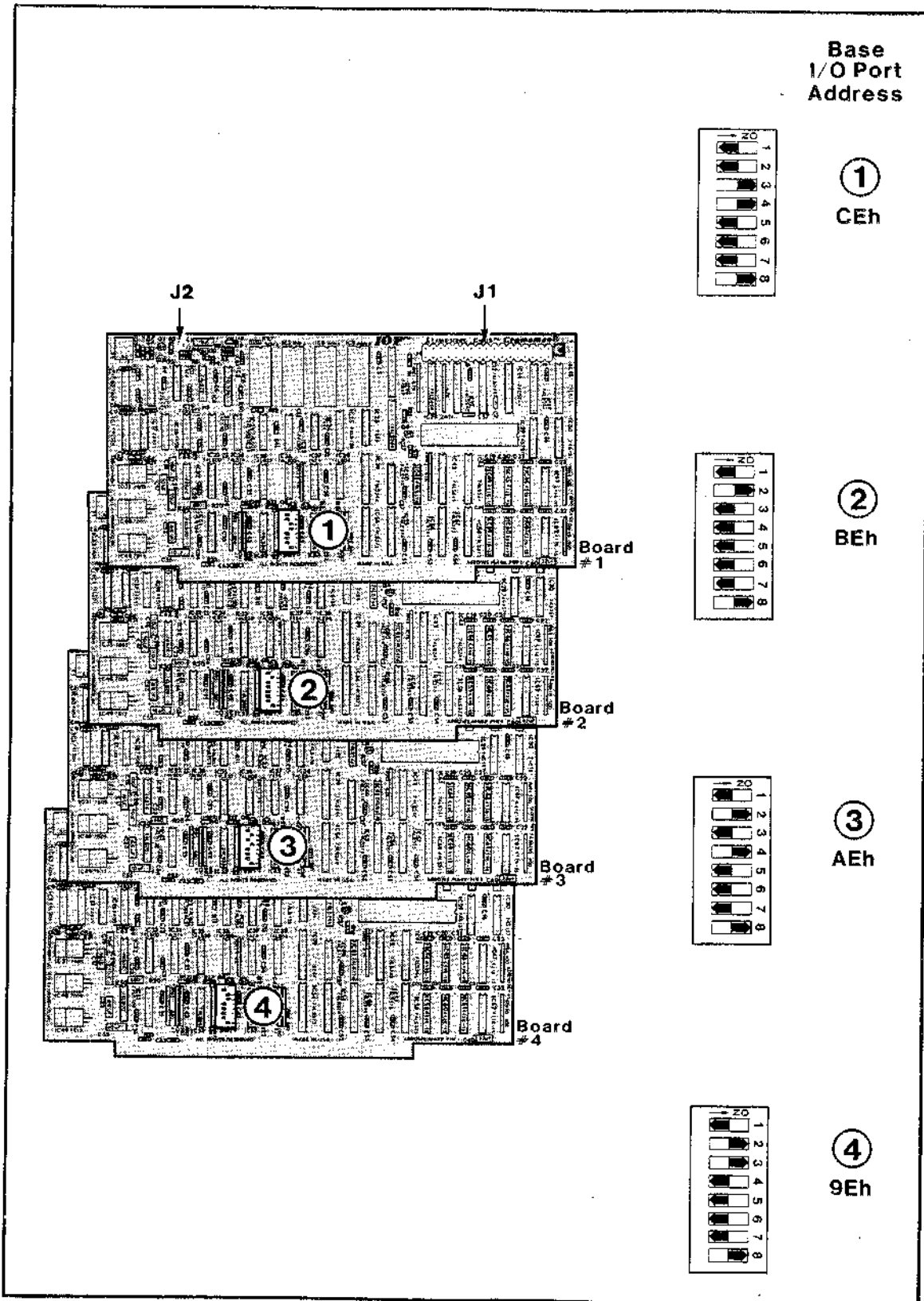
Figure 5-14: IOP SWITCH SETTINGS
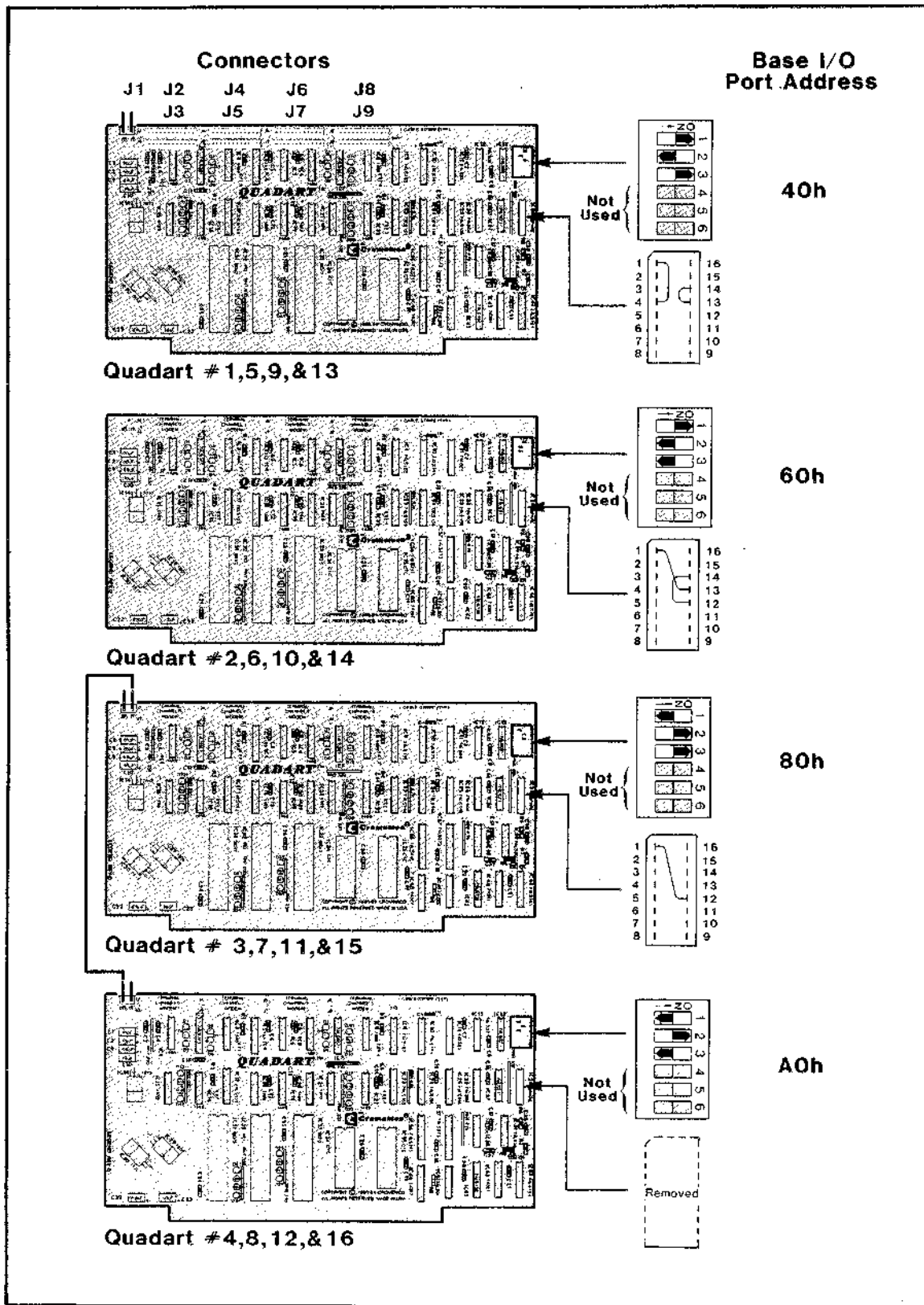
Figure 5-15: QUADART SWITCHES AND JUMPERS

57

The TU-ART

Up to four TU-ART boards can be installed in a single system, and each TU-ART supports two terminals, and two Centronics-style parallel printers. Set the TU-ART switches as shown in figure 5-16.

Insert the TU-ART into any slot in the system card cage, and install up to four 26-conductor cables (part number 519-0086 on CS1 & CS100; 519-0017 on CS2; 519-0008 on CS3 & CS300) from connectors J2 through J5 on the TU-ART board (with the red cable stripe on the left) to any of the rear panel connector slots. Plug the parallel printer cables into the rear panel connectors coming from J2 or J3 on the TU-ART; plug the terminal cables into the rear panel connectors coming from J4 or J5. Mark the rear panel to indicate the device associated with each connector (refer to appendix A).

Connect the priority interrupt cable to J1 on the TU-ART (refer to the last section of this chapter).

The PRI

Two PRI boards can be installed in a single system, and each PRI supports one dot matrix printer and one typewriter printer (both Centronics-style parallel devices). Set the PRI switches and jumpers as shown in figure 5-17. When using a TU-ART and a PRI together, assign a different address to each board (compare figures 5-16 and 5-17).

Insert the PRI into any card slot in the system card cage, and install two 26-conductor cables (part number 519-0086 on CS1 & CS100; 519-0017 on CS2; 519-0008 on CS3 & CS300) from connectors J1 and J2 on the PRI board (with the red cable stripe on the left) to any of the rear panel connector slots. The dot matrix printer cable plugs into the rear panel connector coming from J1 on the PRI; the typewriter printer cable plugs into the rear panel connector coming from J2. Mark the rear panel to indicate the device associated with each connector (refer to appendix A).

Connect the priority interrupt cable to J3 on the PRI (refer to the next section).

Connectors

Base I/O Port Address

① TUART #1

Channel A, 20h
Channel B, 50h

② TUART #2

Channel A, 60h
Channel B, 70h

③ TUART #3

Channel A, 80h
Channel B, 90h

④ TUART #4

Channel A, A0h
Channel B, B0h

Board #1

Board #2

Board #3

Board #4
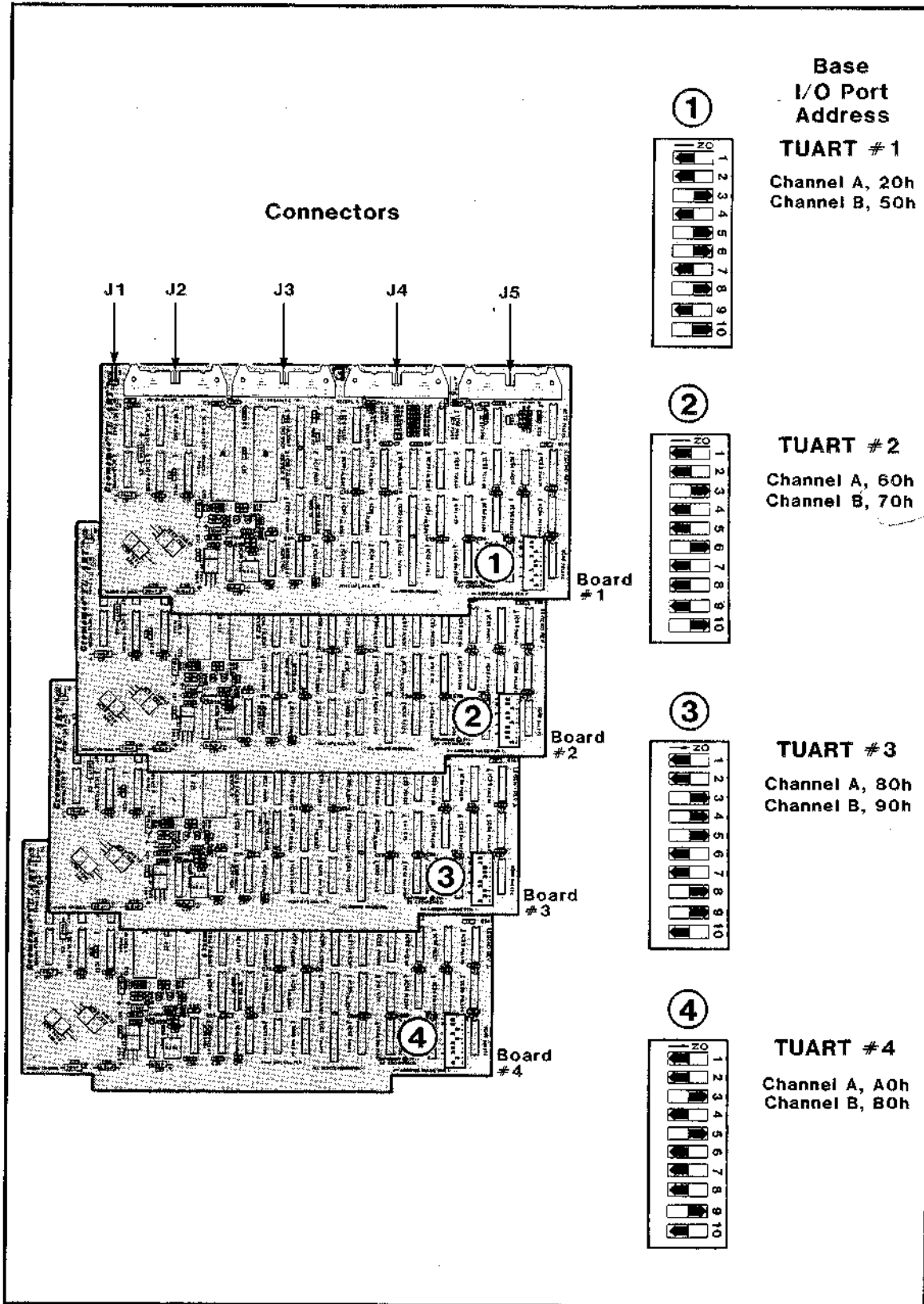
Figure 5-16: TU-ART SWITCH SETTINGS
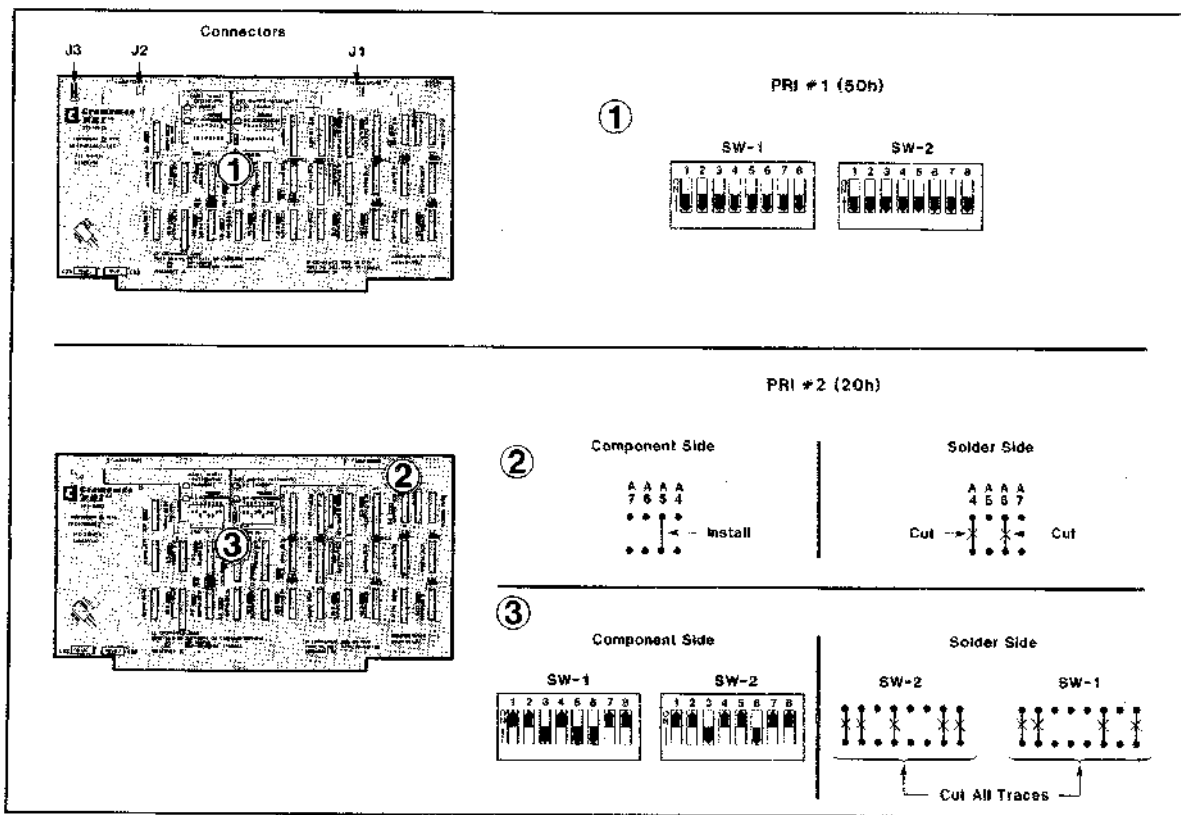
Figure 5-17: PRI SWITCHES AND JUMPERS

THE PRIORITY INTERRUPT CABLE

The 6-connector priority interrupt cable (part number 519-0029) determines the order in which the host processor services conflicting interrupt requests. Attach the first connector to J1 on the 64FDC/16FDC board by aligning the blue dot on the connector with the blue dot on the plug (figure 5-18). Align the yellow dots on the remaining plugs and connectors, and attach the second connector to the next highest priority board, the third connector to the next highest, and so on.

The suggested order of board priorities is: 64FDC/16FDC, OCTART, TU-ART, IOP, BIART, GPIB, CTI, PRI, and STDC. The IN pin of the highest priority board (the 64FDC/16FDC) is not connected, nor is the OUT pin of the lowest priority board (the last board in the chain). The order of the boards is not critical, as long as the 64FDC/16FDC is first and the STDC is last. No boards or connectors should be skipped, and any unused connectors must be at the end of the cable farthest from the 64FDC/16FDC. Do NOT connect the priority cable to the WDI-II or Maximizer boards.

If the priority connectors are not color coded, install the cable so that the OUT pin of the higher priority board is linked to the IN pin of the next highest board (the IN pin is on the right on all boards except 64FDC/16FDC, revision C of the STDC, and CTI).
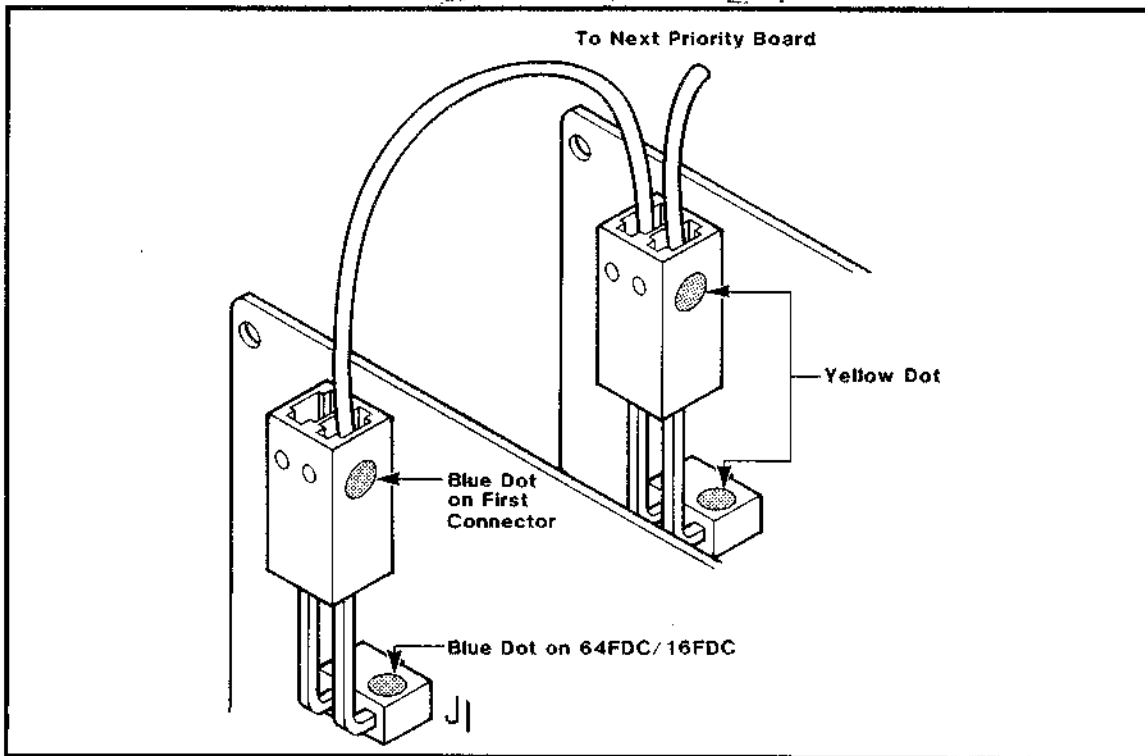
Figure 5-18: INSTALLING THE PRIORITY CABLE

Appendix A

DEVICE FILE DEFINITIONS

This appendix lists all the device files that may appear in the /dev directory. Each entry consists of a device name, the type of board or boards that control the physical device, the board's jumper- or switch-assigned base I/O port address (e.g., OCTART #1 @ CEh means OCTART board number 1 with a base I/O port address of CEh), the major:minor device numbers assigned to the device, and the board connector where the physical device is attached. The block devices appear first, followed by the character devices.

BLOCK DEVICE FILES

System Block Devices

Device Name	Board Type @ Base Port	Device Number Major:Minor	Board Connector
root	---	0:0	--
amem	System RAM	3:0	--

8" Floppy Disk Drives

Device Name	Board Type(s) @ Base Port	Device Number Major:Minor	64FDC Connector
fda	64FDC @ 00h	1:0	J3
fdb	64FDC @ 00h	1:1	J3
fdc	64FDC @ 00h	1:2	J3
fdd	64FDC @ 00h	1:3	J3

8" Persci Drives

Device Name	Board Type(s) @ Base Port	Device Number Major:Minor	16FDC Connector
dfda	16FDC @ 00h	1:16	J3
dfdb	16FDC @ 00h	1:17	J3
dfdc	16FDC @ 00h	1:18	J3
dfdd	16FDC @ 00h	1:19	J3

63

5-1/4" Floppy Disk Drives

Device Name	Board Type(s) @ Base Port	Device Number Major:Minor	64FDC/16FDC Connector
sfda	64FDC @ 00h or 16FDC @ 00h	1:4	J2
sfdb	64FDC @ 00h or 16FDC @ 00h	1:5	J2
sfdc	64FDC @ 00h or 16FDC @ 00h	1:6	J2
sfdd	64FDC @ 00h or 16FDC @ 00h	1:7	J2

Uniform Format Floppies

Uniform format floppies are floppies that have all tracks in the same format and all sectors the same size. The sector size may be 128, 256, 512, or 1024 bytes. (1024-byte sectors can only be read by special utility programs such as rcopy.bin and readall.bin.) The minor device number describes the physical characteristics of the device using the following scheme:

Minor device = unit + small + dtrack + dual + sside + sdens

Where:

    unit = 0, 1, 2, or 3 for A, B, C, or D, respectively
    small = 4 for 5-1/4" diskettes; 0 for 8" diskettes
    dtrack = 8 for double tracked (not supported); 0 otherwise
    dual = 16 for drives in pairs (e.g., Persci); 0 otherwise
    ssode = 32 for single-sided; 0 for double-sided
    sdens = 64 for single-density; 0 for double-density

Examples:

Listed below are two of the most common uniform floppy types encountered. For double-sided, double-density 8" diskettes:

Device Name	Device Number Major:Minor	64FDC/16FDC Connector
ufda	2:0	J3
ufdb	2:1	J3
ufdc	2:2	J3
ufdd	2:3	J3

For double-sided, double-density 5-1/4" diskettes:

Device Name	Device Number Major:Minor	64FDC/16FDC Connector
usfda	2:4	J2
usfdb	2:5	J2
usfdc	2:6	J2
usfdd	2:7	J2

64

64FDC Cartridge Tape Drives

Device Name	Board Type(s) @ Base Port	Device Number Major:Minor	64FDC Connector
ftab	64FDC @ 00h	4:0	J2 or J3
ftab0	64FDC @ 00h	4:1	J2 or J3
ftab1	64FDC @ 00h	4:2	J2 or J3
ftab2	64FDC @ 00h	4:3	J2 or J3
ftcd	64FDC @ 00h	4:4	J2 or J3
ftcd0	64FDC @ 00h	4:5	J2 or J3
ftcd1	64FDC @ 00h	4:6	J2 or J3
ftcd2	64FDC @ 00h	4:7	J2 or J3
fftab	64FDC @ 00h	4:8	J2 or J3
fftab0	64FDC @ 00h	4:9	J2 or J3
fftab1	64FDC @ 00h	4:10	J2 or J3
fftab2	64FDC @ 00h	4:11	J2 or J3
fftcd	64FDC @ 00h	4:12	J2 or J3
fftcd0	64FDC @ 00h	4:13	J2 or J3
fftcd1	64FDC @ 00h	4:14	J2 or J3
fftcd2	64FDC @ 00h	4:15	J2 or J3

STDC Hard Disk Drives

Device Name	Board Type @ Base Port	Device Number Major:Minor	STDC Connector
std0	STDC @ F8h	6:0	J5
std1	STDC @ F8h	6:1	J5
std2	STDC @ F8h	6:2	J5
std3	STDC @ F8h	6:3	J5
std4	STDC @ F8h	6:4	J5
std5	STDC @ F8h	6:5	J5
std6	STDC @ F8h	6:6	J5
std7	STDC @ F8h	6:7	J5
std8	STDC @ F8h	6:8	J5
std9	STDC @ F8h	6:9	J5
std10	STDC @ F8h	6:10	J5
std11	STDC @ F8h	6:11	J5
std12	STDC @ F8h	6:12	J5
std13	STDC @ F8h	6:13	J5
std14	STDC @ F8h	6:14	J5
std15	STDC @ F8h	6:15	J5
std16	STDC @ F8h	6:16	J5
std17	STDC @ F8h	6:17	J5
std18	STDC @ F8h	6:18	J5
std19	STDC @ F8h	6:19	J5
std20	STDC @ F8h	6:20	J5
std21	STDC @ F8h	6:21	J5
std22	STDC @ F8h	6:22	J5
std23	STDC @ F8h	6:23	J5
std24	STDC @ F8h	6:24	J5

std25	STDC @ F8h	6:25	J5
std26	STDC @ F8h	6:26	J5
std27	STDC @ F8h	6:27	J5
std28	STDC @ F8h	6:28	J5
std29	STDC @ F8h	6:29	J5
std30	STDC @ F8h	6:30	J5
std31	STDC @ F8h	6:31	J5
std32	STDC @ F8h	6:32	J4
std33	STDC @ F8h	6:33	J4
std34	STDC @ F8h	6:34	J4
std35	STDC @ F8h	6:35	J4
std36	STDC @ F8h	6:36	J4
std37	STDC @ F8h	6:37	J4
std38	STDC @ F8h	6:38	J4
std39	STDC @ F8h	6:39	J4
std40	STDC @ F8h	6:40	J4
std41	STDC @ F8h	6:41	J4
std42	STDC @ F8h	6:42	J4
std43	STDC @ F8h	6:43	J4
std44	STDC @ F8h	6:44	J4
std45	STDC @ F8h	6:45	J4
std46	STDC @ F8h	6:46	J4
std47	STDC @ F8h	6:47	J4
std48	STDC @ F8h	6:48	J4
std49	STDC @ F8h	6:49	J4
std50	STDC @ F8h	6:50	J4
std51	STDC @ F8h	6:51	J4
std52	STDC @ F8h	6:52	J4
std53	STDC @ F8h	6:53	J4
std54	STDC @ F8h	6:54	J4
std55	STDC @ F8h	6:55	J4
std56	STDC @ F8h	6:56	J4
std57	STDC @ F8h	6:57	J4
std58	STDC @ F8h	6:58	J4
std59	STDC @ F8h	6:59	J4
std60	STDC @ F8h	6:60	J4
std61	STDC @ F8h	6:61	J4
std62	STDC @ F8h	6:62	J4
std63	STDC @ F8h	6:63	J4

64FDC/16FDC and TU-ART Serial Printers

Device Name	Device Number Major:Minor	64FDC/16FDC/TU-ART Connector
slpt1	7:0 or 7:64 or 7:128	J4
slpt2	7:2 or 7:66 or 7:130	J4
slpt3	7:5 or 7:69 or 7:133	J5
splt4	7:6 or 7:70 or 7:134	J4
slpt5	7:7 or 7:71 or 7:135	J5
slpt6	7:8 or 7:72 or 7:136	J4
slpt7	7:9 or 7:73 or 7:137	J5
slpt8	7:10 or 7:74 or 7:138	J4
slpt9	7:11 or 7:75 or 7:139	J5

IMI Hard Disk Drive

Device Name	Board Type(s) @ Base Port	Device Number Major:Minor	WDI-II Connector
hd0	WDI-II @ E0h	8:0	J2
hd1	WDI-II @ E0h	8:1	J2
hd2	WDI-II @ E0h	8:2	J2
hd3	WDI-II @ E0h	8:3	J2

CHARACTER DEVICE FILES

System Character Devices

Device Name	Board Type(s) @ Base Port	Device Number Major:Minor	Board Connector
null	Throwaway Output	3:0	--
timer	64FDC/16FDC Timer	4:0	--

64FDC/16FDC and TU-ART Terminals

Device Name	Board Type(s) @ Base Port	Device Number Major:Minor	64FDC/16FDC/TU-ART Connector
tty1	64FDC @ 00h or 16FDC @ 00h	1:0	J4
tty2	TU-ART #1A @ 20h	1:2	J4
tty3	TU-ART #1B @ 50h	1:5	J5
tty4	TU-ART #2A @ 60h	1:6	J4
tty5	TU-ART #2B @ 70h	1:7	J5
tty6	TU-ART #3A @ 80h	1:8	J4
tty7	TU-ART #3B @ 90h	1:9	J5
tty8	TU-ART #4A @ A0h	1:10	J4
tty9	TU-ART #4B @ B0h	1:11	J5

PRI Typewriter (Parallel) Printers

Device Name	Board Type @ Base Port	Device Number Major:Minor	PRI Connector
typ1	PRI #1 @ 50h	6:5	J2
typ2	PRI #2 @ 20h	6:2	J2

PRI and TU-ART Dot Matrix (Parallel) Printers

Device Name	Board Type(s) @ Base Port	Device Number Major:Minor	PRI/TU-ART Connector
lpt1	PRI #1 @ 50h, or	5:5	J1
	TU-ART #1B @ 50h		J3
lpt2	PRI #2 @ 20h, or	5:2	J1
	TU-ART #1A @ 20h		J2
lpt3	TU-ART #2A @ 60h	5:6	J2
lpt4	TU-ART #2B @ 70h	5:7	J3
lpt5	TU-ART #3A @ 80h	5:8	J2
lpt6	TU-ART #3B @ 90h	5:9	J3
lpt7	TU-ART #4A @ A0h	5:10	J2
lpt8	TU-ART #4B @ B0h	5:11	J3

OCTART Serial Printers

Device Name	Board Type @ Base Port	Device Number Major:Minor	OCTART Connector
qslpt1	OCTART #1 @ CEh	9:0 or 9:128	J1
qslpt2	OCTART #1 @ CEh	9:1 or 9:129	J1
qslpt3	OCTART #1 @ CEh	9:2 or 9:130	J1
qslpt4	OCTART #1 @ CEh	9:3 or 9:131	J1
qslpt5	OCTART #1 @ CEh	9:4 or 9:132	J2
qslpt6	OCTART #1 @ CEh	9:5 or 9:133	J2
qslpt7	OCTART #1 @ CEh	9:6 or 9:134	J2
qslpt8	OCTART #1 @ CEh	9:7 or 9:135	J2
qslpt17	OCTART #2 @ BEh	9:16 or 9:144	J1
qslpt18	OCTART #2 @ BEh	9:17 or 9:145	J1
qslpt19	OCTART #2 @ BEh	9:18 or 9:146	J1
qslpt20	OCTART #2 @ BEh	9:19 or 9:147	J1
qslpt21	OCTART #2 @ BEh	9:20 or 9:148	J2
qslpt22	OCTART #2 @ BEh	9:21 or 9:149	J2
qslpt23	OCTART #2 @ BEh	9:22 or 9:148	J2
qslpt24	OCTART #2 @ BEh	9:23 or 9:149	J2
qslpt33	OCTART #3 @ AEh	9:32 or 9:160	J1
qslpt34	OCTART #3 @ AEh	9:33 or 9:161	J1
qslpt35	OCTART #3 @ AEh	9:34 or 9:162	J1
qslpt36	OCTART #3 @ AEh	9:35 or 9:163	J1
qslpt37	OCTART #3 @ AEh	9:36 or 9:164	J2
qslpt38	OCTART #3 @ AEh	9:37 or 9:165	J2
qslpt39	OCTART #3 @ AEh	9:38 or 9:166	J2
qslpt40	OCTART #3 @ AEh	9:39 or 9:167	J2
qslpt49	OCTART #4 @ 9Eh	9:48 or 9:176	J1
qslpt50	OCTART #4 @ 9Eh	9:49 or 9:177	J1
qslpt51	OCTART #4 @ 9Eh	9:50 or 9:178	J1
qslpt52	OCTART #4 @ 9Eh	9:51 or 9:179	J1
qslpt53	OCTART #4 @ 9Eh	9:52 or 9:180	J2
qslpt54	OCTART #4 @ 9Eh	9:53 or 9:181	J2
qslpt55	OCTART #4 @ 9Eh	9:54 or 9:182	J2
qslpt56	OCTART #4 @ 9Eh	9:55 or 9:183	J2

OCTART Terminals

Device Name	Board Type @ Base Port	Device Number Major:Minor	OCTART Connector
qtty1	OCTART #1 @ CEh	2:0	J1
qtty2	OCTART #1 @ CEh	2:1	J1
qtty3	OCTART #1 @ CEh	2:2	J1
qtty4	OCTART #1 @ CEh	2:3	J1
qtty5	OCTART #1 @ CEh	2:4	J2
qtty6	OCTART #1 @ CEh	2:5	J2
qtty7	OCTART #1 @ CEh	2:6	J2
qtty8	OCTART #1 @ CEh	2:7	J2
qtty17	OCTART #2 @ BEh	2:16	J1
qtty18	OCTART #2 @ BEh	2:17	J1
qtty19	OCTART #2 @ BEh	2:18	J1
qtty20	OCTART #2 @ BEh	2:19	J1
qtty21	OCTART #2 @ BEh	2:20	J2
qtty22	OCTART #2 @ BEh	2:21	J2
qtty23	OCTART #2 @ BEh	2:22	J2
qtty24	OCTART #2 @ BEh	2:23	J2
qtty33	OCTART #3 @ AEh	2:32	J1
qtty34	OCTART #3 @ AEh	2:33	J1
qtty35	OCTART #3 @ AEh	2:34	J1
qtty36	OCTART #3 @ AEh	2:35	J1
qtty37	OCTART #3 @ AEh	2:36	J2
qtty38	OCTART #3 @ AEh	2:37	J2
qtty39	OCTART #3 @ AEh	2:38	J2
qtty40	OCTART #3 @ AEh	2:39	J2
qtty49	OCTART #4 @ 9Eh	2:48	J1
qtty50	OCTART #4 @ 9Eh	2:49	J1
qtty51	OCTART #4 @ 9Eh	2:50	J1
qtty52	OCTART #4 @ 9Eh	2:51	J1
qtty53	OCTART #4 @ 9Eh	2:52	J2
qtty54	OCTART #4 @ 9Eh	2:53	J2
qtty55	OCTART #4 @ 9Eh	2:54	J2
qtty56	OCTART #4 @ 9Eh	2:55	J2

IOP/QUADART Serial Printers

Device Name	Board Type(s) @ Base Port	Device Number Major:Minor	Quadart Connector
qslpt1	IOP #1 @ CEh, Quadart #1 @ 40h	9:0 or 9:128	J2
qslpt2	IOP #1 @ CEh, Quadart #1 @ 40h	9:1 or 9:129	J4
qslpt3	IOP #1 @ CEh, Quadart #1 @ 40h	9:2 or 9:130	J6
qslpt4	IOP #1 @ CEh, Quadart #1 @ 40h	9:3 or 9:131	J8
qslpt5	IOP #1 @ CEh, Quadart #2 @ 60h	9:4 or 9:132	J2
qslpt6	IOP #1 @ CEh, Quadart #2 @ 60h	9:5 or 9:133	J4
qslpt7	IOP #1 @ CEh, Quadart #2 @ 60h	9:6 or 9:134	J6
qslpt8	IOP #1 @ CEh, Quadart #2 @ 60h	9:7 or 9:135	J8
qslpt9	IOP #1 @ CEh, Quadart #3 @ 80h	9:8 or 9:136	J2
qslpt10	IOP #1 @ CEh, Quadart #3 @ 80h	9:9 or 9:137	J4
qslpt11	IOP #1 @ CEh, Quadart #3 @ 80h	9:10 or 9:138	J6
qslpt12	IOP #1 @ CEh, Quadart #3 @ 80h	9:11 or 9:139	J8
qslpt13	IOP #1 @ CEh, Quadart #4 @ A0h	9:12 or 9:140	J2
qslpt14	IOP #1 @ CEh, Quadart #4 @ A0h	9:13 or 9:141	J4
qslpt15	IOP #1 @ CEh, Quadart #4 @ A0h	9:14 or 9:142	J6
qslpt16	IOP #1 @ CEh, Quadart #4 @ A0h	9:15 or 9:143	J8
qslpt17	IOP #2 @ BEh, Quadart #5 @ 40h	9:16 or 9:144	J2
qslpt18	IOP #2 @ BEh, Quadart #5 @ 40h	9:17 or 9:145	J4
qslpt19	IOP #2 @ BEh, Quadart #5 @ 40h	9:18 or 9:146	J6
qslpt20	IOP #2 @ BEh, Quadart #5 @ 40h	9:19 or 9:147	J8
qslpt21	IOP #2 @ BEh, Quadart #6 @ 60h	9:20 or 9:148	J2
qslpt22	IOP #2 @ BEh, Quadart #6 @ 60h	9:21 or 9:149	J4
qslpt23	IOP #2 @ BEh, Quadart #6 @ 60h	9:22 or 9:150	J6
qslpt24	IOP #2 @ BEh, Quadart #6 @ 60h	9:23 or 9:151	J8
qslpt25	IOP #2 @ BEh, Quadart #7 @ 80h	9:24 or 9:152	J2
qslpt26	IOP #2 @ BEh, Quadart #7 @ 80h	9:25 or 9:153	J4
qslpt27	IOP #2 @ BEh, Quadart #7 @ 80h	9:26 or 9:154	J6
qslpt28	IOP #2 @ BEh, Quadart #7 @ 80h	9:27 or 9:155	J8
qslpt29	IOP #2 @ BEh, Quadart #8 @ A0h	9:28 or 9:156	J2
qslpt30	IOP #2 @ BEh, Quadart #8 @ A0h	9:29 or 9:157	J4
qslpt31	IOP #2 @ BEh, Quadart #8 @ A0h	9:30 or 9:158	J6
qslpt32	IOP #2 @ BEh, Quadart #8 @ A0h	9:31 or 9:159	J8
qslpt33	IOP #3 @ AEh, Quadart #9 @ 40h	9:32 or 9:160	J2
qslpt34	IOP #3 @ AEh, Quadart #9 @ 40h	9:33 or 9:161	J4
qslpt35	IOP #3 @ AEh, Quadart #9 @ 40h	9:34 or 9:162	J6
qslpt36	IOP #3 @ AEh, Quadart #9 @ 40h	9:35 or 9:163	J8
qslpt37	IOP #3 @ AEh, Quadart #10 @ 60h	9:36 or 9:164	J2
qslpt38	IOP #3 @ AEh, Quadart #10 @ 60h	9:37 or 9:165	J4
qslpt39	IOP #3 @ AEh, Quadart #10 @ 60h	9:38 or 9:166	J6
qslpt40	IOP #3 @ AEh, Quadart #10 @ 60h	9:39 or 9:167	J8
qslpt41	IOP #3 @ AEh, Quadart #11 @ 80h	9:40 or 9:168	J2
qslpt42	IOP #3 @ AEh, Quadart #11 @ 80h	9:41 or 9:169	J4
qslpt43	IOP #3 @ AEh, Quadart #11 @ 80h	9:42 or 9:170	J6
qslpt44	IOP #3 @ AEh, Quadart #11 @ 80h	9:43 or 9:171	J8
qslpt45	IOP #3 @ AEh, Quadart #12 @ A0h	9:44 or 9:172	J2
qslpt46	IOP #3 @ AEh, Quadart #12 @ A0h	9:45 or 9:173	J4

qslpt47	IOP #3 @ AEh, Quadart #12 @ A0h	9:46 or 9:174	J6
qslpt48	IOP #3 @ AEh, Quadart #12 @ A0h	9:47 or 9:175	J8
qslpt49	IOP #4 @ 9Eh, Quadart #13 @ 40h	9:48 or 9:176	J2
qslpt50	IOP #4 @ 9Eh, Quadart #13 @ 40h	9:49 or 9:177	J4
qslpt51	IOP #4 @ 9Eh, Quadart #13 @ 40h	9:50 or 9:178	J6
qslpt52	IOP #4 @ 9Eh, Quadart #13 @ 40h	9:51 or 9:179	J8
qslpt53	IOP #4 @ 9Eh, Quadart #14 @ 60h	9:52 or 9:180	J2
qslpt54	IOP #4 @ 9Eh, Quadart #14 @ 60h	9:53 or 9:181	J4
qslpt55	IOP #4 @ 9Eh, Quadart #14 @ 60h	9:54 or 9:182	J6
qslpt56	IOP #4 @ 9Eh, Quadart #14 @ 60h	9:55 or 9:183	J8
qslpt57	IOP #4 @ 9Eh, Quadart #15 @ 80h	9:56 or 9:184	J2
qslpt58	IOP #4 @ 9Eh, Quadart #15 @ 80h	9:57 or 9:185	J4
qslpt59	IOP #4 @ 9Eh, Quadart #15 @ 80h	9:58 or 9:186	J6
qslpt60	IOP #4 @ 9Eh, Quadart #15 @ 80h	9:59 or 9:187	J8
qslpt61	IOP #4 @ 9Eh, Quadart #16 @ A0h	9:60 or 9:188	J2
qslpt62	IOP #4 @ 9Eh, Quadart #16 @ A0h	9:61 or 9:189	J4
qslpt63	IOP #4 @ 9Eh, Quadart #16 @ A0h	9:62 or 9:190	J6
qslpt64	IOP #4 @ 9Eh, Quadart #16 @ A0h	9:63 or 9:191	J8

IOP/QUADART Terminals

Device Name	Board Type(s) @ Base Port	Device Number Major:Minor	Quadart Connector
qtty1	IOP #1 @ CEh, Quadart #1 @ 40h	2:0	J2 or J3
qtty2	IOP #1 @ CEh, Quadart #1 @ 40h	2:1	J4 or J5
qtty3	IOP #1 @ CEh, Quadart #1 @ 40h	2:2	J6 or J7
qtty4	IOP #1 @ CEh, Quadart #1 @ 40h	2:3	J8 or J9
qtty5	IOP #1 @ CEh, Quadart #2 @ 60h	2:4	J2 or J3
qtty6	IOP #1 @ CEh, Quadart #2 @ 60h	2:5	J4 or J5
qtty7	IOP #1 @ CEh, Quadart #2 @ 60h	2:6	J6 or J7
qtty8	IOP #1 @ CEh, Quadart #2 @ 60h	2:7	J8 or J9
qtty9	IOP #1 @ CEh, Quadart #3 @ 80h	2:8	J2 or J3
qtty10	IOP #1 @ CEh, Quadart #3 @ 80h	2:9	J4 or J5
qtty11	IOP #1 @ CEh, Quadart #3 @ 80h	2:10	J6 or J7
qtty12	IOP #1 @ CEh, Quadart #3 @ 80h	2:11	J8 or J9
qtty13	IOP #1 @ CEh, Quadart #4 @ A0h	2:12	J2 or J3
qtty14	IOP #1 @ CEh, Quadart #4 @ A0h	2:13	J4 or J5
qtty15	IOP #1 @ CEh, Quadart #4 @ A0h	2:14	J6 or J7
qtty16	IOP #1 @ CEh, Quadart #4 @ A0h	2:15	J8 or J9
qtty17	IOP #2 @ BEh, Quadart #5 @ 40h	2:16	J2 or J3
qtty18	IOP #2 @ BEh, Quadart #5 @ 40h	2:17	J4 or J5
qtty19	IOP #2 @ BEh, Quadart #5 @ 40h	2:18	J6 or J7
qtty20	IOP #2 @ BEh, Quadart #5 @ 40h	2:19	J8 or J9
qtty21	IOP #2 @ BEh, Quadart #6 @ 60h	2:20	J2 or J3
qtty22	IOP #2 @ BEh, Quadart #6 @ 60h	2:21	J4 or J5
qtty23	IOP #2 @ BEh, Quadart #6 @ 60h	2:22	J6 or J7
qtty24	IOP #2 @ BEh, Quadart #6 @ 60h	2:23	J8 or J9
qtty25	IOP #2 @ BEh, Quadart #7 @ 80h	2:24	J2 or J3
qtty26	IOP #2 @ BEh, Quadart #7 @ 80h	2:25	J4 or J5
qtty27	IOP #2 @ BEh, Quadart #7 @ 80h	2:26	J6 or J7

qtty28	IOP #2 @ BEh, Quadart #7 @ 80h	2:27	J8 or J9
qtty29	IOP #2 @ BEh, Quadart #8 @ A0h	2:28	J2 or J3
qtty30	IOP #2 @ BEh, Quadart #8 @ A0h	2:29	J4 or J5
qtty31	IOP #2 @ BEh, Quadart #8 @ A0h	2:30	J6 or J7
qtty32	IOP #2 @ BEh, Quadart #8 @ A0h	2:31	J8 or J9
qtty33	IOP #3 @ AEh, Quadart #9 @ 40h	2:32	J2 or J3
qtty34	IOP #3 @ AEh, Quadart #9 @ 40h	2:33	J4 or J5
qtty35	IOP #3 @ AEh, Quadart #9 @ 40h	2:34	J6 or J7
qtty36	IOP #3 @ AEh, Quadart #9 @ 40h	2:35	J8 or J9
qtty37	IOP #3 @ AEh, Quadart #10 @ 60h	2:36	J2 or J3
qtty38	IOP #3 @ AEh, Quadart #10 @ 60h	2:37	J4 or J5
qtty39	IOP #3 @ AEh, Quadart #10 @ 60h	2:38	J6 or J7
qtty40	IOP #3 @ AEh, Quadart #10 @ 60h	2:39	J8 or J9
qtty41	IOP #3 @ AEh, Quadart #11 @ 80h	2:40	J2 or J3
qtty42	IOP #3 @ AEh, Quadart #11 @ 80h	2:41	J4 or J5
qtty43	IOP #3 @ AEh, Quadart #11 @ 80h	2:42	J6 or J7
qtty44	IOP #3 @ AEh, Quadart #11 @ 80h	2:43	J8 or J9
qtty45	IOP #3 @ AEh, Quadart #12 @ A0h	2:44	J2 or J3
qtty46	IOP #3 @ AEh, Quadart #12 @ A0h	2:45	J4 or J5
qtty47	IOP #3 @ AEh, Quadart #12 @ A0h	2:46	J6 or J7
qtty48	IOP #3 @ AEh, Quadart #12 @ A0h	2:47	J8 or J9
qtty49	IOP #4 @ 9Eh, Quadart #13 @ 40h	2:48	J2 or J3
qtty50	IOP #4 @ 9Eh, Quadart #13 @ 40h	2:49	J4 or J5
qtty51	IOP #4 @ 9Eh, Quadart #13 @ 40h	2:50	J6 or J7
qtty52	IOP #4 @ 9Eh, Quadart #13 @ 40h	2:51	J8 or J9
qtty53	IOP #4 @ 9Eh, Quadart #14 @ 60h	2:52	J2 or J3
qtty54	IOP #4 @ 9Eh, Quadart #14 @ 60h	2:53	J4 or J5
qtty55	IOP #4 @ 9Eh, Quadart #14 @ 60h	2:54	J6 or J7
qtty56	IOP #4 @ 9Eh, Quadart #14 @ 60h	2:55	J8 or J9
qtty57	IOP #4 @ 9Eh, Quadart #15 @ 80h	2:56	J2 or J3
qtty58	IOP #4 @ 9Eh, Quadart #15 @ 80h	2:57	J4 or J5
qtty59	IOP #4 @ 9Eh, Quadart #15 @ 80h	2:58	J6 or J7
qtty60	IOP #4 @ 9Eh, Quadart #15 @ 80h	2:59	J8 or J9
qtty61	IOP #4 @ 9Eh, Quadart #16 @ A0h	2:60	J2 or J3
qtty62	IOP #4 @ 9Eh, Quadart #16 @ A0h	2:61	J4 or J5
qtty63	IOP #4 @ 9Eh, Quadart #16 @ A0h	2:62	J6 or J7
qtty64	IOP #4 @ 9Eh, Quadart #16 @ A0h	2:63	J8 or J9

IOP/QUADART Modems

Device Name	Board Type(s) @ Base Port	Device Number Major:Minor	Quadart Connector
mtty1	IOP #1 @ CEh, Quadart #1 @ 40h	2:128	J3
mtty2	IOP #1 @ CEh, Quadart #1 @ 40h	2:129	J5
mtty3	IOP #1 @ CEh, Quadart #1 @ 40h	2:130	J7
mtty4	IOP #1 @ CEh, Quadart #1 @ 40h	2:131	J9
mtty5	IOP #1 @ CEh, Quadart #2 @ 60h	2:132	J3
mtty6	IOP #1 @ CEh, Quadart #2 @ 60h	2:133	J5
mtty7	IOP #1 @ CEh, Quadart #2 @ 60h	2:134	J7
mtty8	IOP #1 @ CEh, Quadart #2 @ 60h	2:135	J9

mtty9	IOP #1 @ CEh, Quadart #3 @ 80h	2:136	J3
mtty10	IOP #1 @ CEh, Quadart #3 @ 80h	2:137	J5
mtty11	IOP #1 @ CEh, Quadart #3 @ 80h	2:138	J7
mtty12	IOP #1 @ CEh, Quadart #4 @ 80h	2:139	J9
mtty13	IOP #1 @ CEh, Quadart #4 @ A0h	2:140	J3
mtty14	IOP #1 @ CEh, Quadart #4 @ A0h	2:141	J5
mtty15	IOP #1 @ CEh, Quadart #4 @ A0h	2:142	J7
mtty16	IOP #1 @ CEh, Quadart #4 @ A0h	2:143	J9
mtty17	IOP #2 @ BEh, Quadart #5 @ 40h	2:144	J3
mtty18	IOP #2 @ BEh, Quadart #5 @ 40h	2:145	J5
mtty19	IOP #2 @ BEh, Quadart #5 @ 40h	2:146	J7
mtty20	IOP #2 @ BEh, Quadart #5 @ 40h	2:147	J9
mtty21	IOP #2 @ BEh, Quadart #6 @ 60h	2:148	J3
mtty22	IOP #2 @ BEh, Quadart #6 @ 60h	2:149	J5
mtty23	IOP #2 @ BEh, Quadart #6 @ 60h	2:150	J7
mtty24	IOP #2 @ BEh, Quadart #6 @ 60h	2:151	J9
mtty25	IOP #2 @ BEh, Quadart #7 @ 80h	2:152	J3
mtty26	IOP #2 @ BEh, Quadart #7 @ 80h	2:153	J5
mtty27	IOP #2 @ BEh, Quadart #7 @ 80h	2:154	J7
mtty28	IOP #2 @ BEh, Quadart #7 @ 80h	2:155	J9
mtty29	IOP #2 @ BEh, Quadart #8 @ A0h	2:156	J3
mtty30	IOP #2 @ BEh, Quadart #8 @ A0h	2:157	J5
mtty31	IOP #2 @ BEh, Quadart #8 @ A0h	2:158	J7
mtty32	IOP #2 @ BEh, Quadart #8 @ A0h	2:159	J9
mtty33	IOP #3 @ AEh, Quadart #9 @ 40h	2:160	J3
mtty34	IOP #3 @ AEh, Quadart #9 @ 40h	2:161	J5
mtty35	IOP #3 @ AEh, Quadart #9 @ 40h	2:162	J7
mtty36	IOP #3 @ AEh, Quadart #9 @ 40h	2:163	J9
mtty37	IOP #3 @ AEh, Quadart #10 @ 60h	2:164	J3
mtty38	IOP #3 @ AEh, Quadart #10 @ 60h	2:165	J5
mtty39	IOP #3 @ AEh, Quadart #10 @ 60h	2:166	J7
mtty40	IOP #3 @ AEh, Quadart #10 @ 60h	2:167	J9
mtty41	IOP #3 @ AEh, Quadart #11 @ 80h	2:168	J3
mtty42	IOP #3 @ AEh, Quadart #11 @ 80h	2:169	J5
mtty43	IOP #3 @ AEh, Quadart #11 @ 80h	2:170	J7
mtty44	IOP #3 @ AEh, Quadart #11 @ 80h	2:171	J9
mtty45	IOP #3 @ AEh, Quadart #12 @ A0h	2:172	J3
mtty46	IOP #3 @ AEh, Quadart #12 @ A0h	2:173	J5
mtty47	IOP #3 @ AEh, Quadart #12 @ A0h	2:174	J7
mtty48	IOP #3 @ AEh, Quadart #12 @ A0h	2:175	J9
mtty49	IOP #4 @ 9Eh, Quadart #13 @ 40h	2:176	J3
mtty50	IOP #4 @ 9Eh, Quadart #13 @ 40h	2:177	J5
mtty51	IOP #4 @ 9Eh, Quadart #13 @ 40h	2:178	J7
mtty52	IOP #4 @ 9Eh, Quadart #13 @ 40h	2:179	J9
mtty53	IOP #4 @ 9Eh, Quadart #14 @ 60h	2:180	J3
mtty54	IOP #4 @ 9Eh, Quadart #14 @ 60h	2:181	J5
mtty55	IOP #4 @ 9Eh, Quadart #14 @ 60h	2:182	J7
mtty56	IOP #4 @ 9Eh, Quadart #14 @ 60h	2:183	J9
mtty57	IOP #4 @ 9Eh, Quadart #15 @ 80h	2:184	J3
mtty58	IOP #4 @ 9Eh, Quadart #15 @ 80h	2:185	J5
mtty59	IOP #4 @ 9Eh, Quadart #15 @ 80h	2:186	J7

mtty60	IOP #4 @ 9Eh, Quadart #15 @ 80h	2:187	J9
mtty61	IOP #4 @ 9Eh, Quadart #16 @ A0h	2:188	J3
mtty62	IOP #4 @ 9Eh, Quadart #16 @ A0h	2:189	J5
mtty63	IOP #4 @ 9Eh, Quadart #16 @ A0h	2:190	J7
mtty64	IOP #4 @ 9Eh, Quadart #16 @ A0h	2:191	J9

C-Net Sockets

Device Name	Board Type(s) @ Base Port	Device Number Major:Minor	CNI Connector
socket1	IOP #2 @ BEh, CNI @ F0h	12:1	J4
socket2	IOP #2 @ BEh, CNI @ F0h	12:2	J4
socket3	IOP #2 @ BEh, CNI @ F0h	12:3	J4
socket4	IOP #2 @ BEh, CNI @ F0h	12:4	J4
socket5	IOP #2 @ BEh, CNI @ F0h	12:5	J4
socket6	IOP #2 @ BEh, CNI @ F0h	12:6	J4
socket7	IOP #2 @ BEh, CNI @ F0h	12:7	J4
socket8	IOP #2 @ BEh, CNI @ F0h	12:8	J4
socket9	IOP #2 @ BEh, CNI @ F0h	12:9	J4
socket10	IOP #2 @ BEh, CNI @ F0h	12:10	J4
socket11	IOP #2 @ BEh, CNI @ F0h	12:11	J4
socket12	IOP #2 @ BEh, CNI @ F0h	12:12	J4
socket13	IOP #1 @ BEh, CNI @ F0h	12:13	J4
socket14	IOP #1 @ BEh, CNI @ F0h	12:14	J4
socket15	IOP #1 @ BEh, CNI @ F0h	12:15	J4

Appendix B

DISK ERROR MESSAGES

In the event of a disk error, the Cromix-Plus Operating System displays an error message to aid in the diagnosis and correction of the problem.

FLOPPY DISK ERROR MESSAGES

When the operating system cannot access a diskette, an error message is displayed in the following format:

<cflop, uflop, or tflop> mode error: Unit **uu**, Side **xx**, Track **cc**, Sector **ss**, Status **ee**

where:

Mode   stands for one of the following words:

    Select   Error occurred in selecting the disk.

    Seek   Error occurred in seeking a track on the disk.

    Read   Error occurred during a read from the disk.

    Write   Error occurred during a write to the disk.

    Home   Error occurred in seeking track 0 on the disk.

    Preread   Error occurred during preread.

    Read Address   Error occurred during a read address operation.

    Write Track   Error occurred during a write track operation.

uu   is the unit number (0-3).

xx   is the side number.

cc   is the track (in decimal) where the error occurred.

ss   is the sector number (in decimal) where the error occurred.

ff   is an 8-bit byte displayed in hexadecimal indicating the disk flags when the error occurred.

ee   is the 8 bit status byte displayed in hexadecimal which describes the error and the conditions at the time the error occurred.

The status byte is a hexadecimal number that is either one of the hex values in the table below or the combination or two or more of those hex values. The bits which correspond to those hex values describe the reasons or the error.

DISK FLAGS:

Flag bits set:

D7 DRO
D6 BOOT*
D5 HEADLOAD
D4 INHIBIT INIT*

D3 MOTOR ON
D2 MOTOR TIMEOUT
D1 AUTOWAIT TIMEOUT
D0 EOJ

D7   Data Request (DRQ)

A high in bit 7 indicates the 16FDC has a byte from the disk or needs a byte for the disk according to the current operation.

D6   Boot*

A low in bit 6 indicates that SW3 is set to BOOT*. A high in bit 6 indicates SW3 is set to MON.

D5   Headload

A one in bit 5 indicates the 1793 is requesting the head to load. A zero in bit 5 indicates the 1793 is not asking the head to load.

D4   Inhibit Init*

A zero in bit 4 indicates that switch 4, INHIBIT* INIT*, is ON. A one in bit 4 indicates that switch 4 is off.

D3   Motor On

A one in bit three indicates that the 16FDC is requesting the drive motors to turn on. A zero in bit three indicates that the 16FDC is no longer requesting the drive motors to turn.

D2 Motor Timeout

A one in bit 2 indicates that the motors have been turned off. The motors will turn off about 8 seconds after the last disk operation. A zero in bit 2 indicates the motors have not been turned off.

D1 Autowait Timeout

A one in bit 1 indicates that the autowait circuit has been turned off by the timer. This will occur about 4 seconds after autowait is turned on. A zero in bit 1 indicates that the autowait circuit has not timed out.

D0 End of Job (EOJ)

A one in bit 0 indicates the command has finished (end of job).

Status Bits Set and
Corresponding Hexadecimal Values

Bits	7	6	5	4	3	2	1	0
Hex value	80	40	20	10	8	4	2	1

If the status byte were 0B, the bits set would be 3, 1, and 0 because the only combination of corresponding hexadecimal values that add up to 0B are the ones which correspond to bits 3, 1, and 0.

The following table describes the malfunctions corresponding to the bits set in the status byte.

Status Bits Set	Seek/Home	Read/Preread	Write
7	not ready	not ready	not ready
6	write protect*	0	write protect
5	head engaged*	record type*	0
4	seek error	record not found	record not found
3	crc error	crc error	crc error
2	track 0*	lost data	lost data
1	index*	data request*	data request*
0	busy	busy	busy

Status Bits Set	Read Address	Write Track
7	not ready	not ready
6	0	write protect
5	0	0
4	record not found	0
3	crc error	0
2	lost data	lost data
1	data request*	data request*
0	busy	busy

The asterisk (*) in the table above indicates that the condition is not the cause of the error message, but that it was present when the error occurred. For example, if the status byte was 30h during a Seek error, bits 4 and 5 are set (=1). This is a Seek error and the head is engaged. The head is supposed to be engaged during a seek. Therefore, this condition is not an error, and is marked with an asterisk. CRC stands for Cyclic Redundancy Check. It is a verification done after a Read operation. A CRC error indicates that an error occurred when the data was transferred.

During a Read operation, status code 10 or 08 indicates the data is not readable. This may be caused by bringing the disk close to a magnetic source or by scratching or otherwise mishandling the disk.

HARD-DISK ERROR MESSAGES

If the Cromix Operating System encounters an error when accessing a hard disk drive, it displays the error in the following format:

STDC **mode** error: Unit **uu**, Block d, Head **h**, Cyl **cc**, Stat **xx**, Error **ffss**

where:

mode	is either Read, Write, Verify, Home, or Seek.
uu	is the minor device number.
d	is the block number in (decimal).
h	is the head number (decimal).
cc	is the cylinder number (decimal).
xx	is the status byte in hexadecimal (indicates type of error).
ffss	is the error number in hexadecimal. The first two digits give the fatal disk error and the last two give the system disk error.

80

If bit 0 of the status byte is set to 1, the error numbers refer to the following error codes. If bit 1 is set to 1, the fatal error number (ff) refers to the Cromix errors in the **/equ/jsysequ.asm** file (refer to appendix A of the Cromix-Plus Programmer's Reference Manual, part number 023-5014).

STDC Hard-Disk Fatal Errors

The following error codes are displayed when a fatal disk error occurs:

00   Failed to Seek & Read Header during R/W

An error occurred during an attempt to seek & read the header preceding a read/write operation.

01   Failed to Seek - Timeout

The seek did not complete within a specified time. Check the drive electronics.

02   Fault Occurred during Seek

During the seek, a fault error occurred within the drive, as reported by the drive. This may be any of several errors.

03   Failed to Seek to Correct Track

The sector header as read off the disk is not what the drivers expected, thus the current disk location is incorrect.

04   Failed to Read CRC of Header

The CRC for the header as read from the disk is incorrect; it is different than what was expected. Most likely, the current disk location is incorrect or the media surface is damaged.

05   Failed to Rezero - Timeout

A rezero command did not complete within a specified time. Check the drive electronics.

06   Fault Occurred after Rezeroing

A fault error occurred within the drive after a rezero command was executed. This may be any of several errors.

07   Drive not Ready

The ready signal from the drive is not active. Make sure the drive is connected properly.

08   Failed to Write - Fault Error

During the write, a fault error occurred within the drive, as reported by the drive. This may be any of several errors.

09   Failed to Verify after Write

After data is written to the disk, it is read back and verified. This error occurs if the data cannot be properly verified.

0A   Failed to Read - Fault Error

During the read, a fault error occurred within the drive, as reported by the drive. This may be any of several errors.

0B   Failed to Read - CRC Error

The CRC read from the disk is incorrect; it is different than the expected CRC. This error usually means that the data just read is incorrect.

0C   Failed to Read - Cannot Locate Sector

The sector cannot be found on the current track. This error occurs if the media surface is damaged or if the controller electronics are not functioning properly.

0D   Surface is Write Protected

The surface selected for the current write command is write protected and cannot be written to.

0E   Failed to Select Unit

There was an attempt to select a drive that was not present, or the controller or drive malfunctioned.

0F   Failed to Select Head

The drive has returned a fault error on attempting to select a non-existent head.

10   Index Pulse Timeout

Index pulses were not being received properly.

11   Seek Range Error

There was an attempt to access a non-existent track.

12   Buffer not Available

Error occurred while trying to flush write buffers.

STDC Hard Disk System Errors

The following error codes are displayed when a system disk error occurs:

00   No Acknowledge Received from Drive

The drive did not acknowledge a command sent to it. Make sure the drive is connected properly.

01   Drive Remains BUSY – Acknowledge Stuck Low

The acknowledge signal from the drive did not go high again after the command strobe went inactive.

02   Timeout Occurred during Rezeroing

A rezero command did not complete within a specified time. Check the drive electronics.

03   Fault Condition Reported by Drive

A fault condition occurred within the drive, as reported by the drive. This may be any of several errors.

04   Failed to Read – CRC Error

The CRC just read from the disk is incorrect; it is different than the expected CRC. This error usually means the data just read is incorrect.

05   Header Off the Disk Does Not Compare with Expected Header

The sector header as read from the disk is not what the drivers expected. Thus, the current disk location is incorrect.

06   Failed to Verify after Write Operation

After data is written to the disk, it is read back and verified. This error occurs if the data cannot be properly verified.

07   Header or Trailer Error

The format of the data header or trailer just read is incorrect.

08   Track Header Error

Track address header is incorrect.

hd Error Messages

If the Cromix-Plus Operating System encounters an error when accessing a hard disk drive, it displays the error in the following format:

Hd **mode** error: Unit **uu**, Head **h**, Cyl **cc**, Sector **s**, Status **ffss**
Status decode: **Status**

Where:

mode	is either Select, Home, Read Address, Seek, Preread, Read, Write or Write Track.
uu	is the minor device number.
h	is the head number.
cc	is the cylinder number (in decimal).
s	is the sector number (in decimal).
ffss	is the error number in hexadecimal. The first two digits give the fatal disk error (same as **mode**) and the last two give the system disk error (same as **status**).
status	is one of the following: No Acknowledge Received From Drive; Timeout; Fault Error; CRC Error; Header Did Not Compare; Header Compare Failed After Seek; Write Data Did Not Verify; Non-Supported Drive Type; Bad Label Format; or No Unit Ready From Drive.

Appendix C

THE SYSDEF FILE

The **sysdef** file must be edited for your particular system configuration (refer to chapter 2). The entire file is shown below:

```
%       Cromix 30 System Generation file
%
%       Sep 20, 1985
```

% Device driver names should be entered on appropriate row. A current
% list of devices supported and their driver names can be found at
% the end of this file. Each driver can have a number of integer
% arguments. Those arguments, if any, should follow the driver name.
% The arguments must be separated by white space. The number of arguments
% and their meaning depend on the particular driver. See description
% at the end for the arguments a driver might require.

% System memory size:

```
        maxmem  2                % Amount of supported memory expressed
                                 % in 256K units.
```

% Character devices:

```
        CDEV    01      tty 0           % FDC terminal only
        CDEV    02      qtty            % Suggested qtty
        CDEV    03      sysdev          % System driver (required)
        CDEV    04      timer           % Timer driver (required)
        CDEV    05                      % Suggested lpt
        CDEV    06                      % Suggested typ
        CDEV    07                      % Suggested slpt
        CDEV    08                      % Not used
        CDEV    09      qslpt           % Suggested qslpt
        CDEV    10                      % Suggested ffp
        CDEV    11                      % Suggested tape
        CDEV    12                      % Suggested cnet
```

% Block devices:

```
        BDEV    01      cflop           % Cromemco floppy driver
        BDEV    02      uflop           % Suggested uflop
        BDEV    03      allmem          % Amem driver (required)
        BDEV    04                      % Suggested tflop
```

85

```
BDEV    05                          % Suggested ramdsk
BDEV    06    stdc                  % STDC driver
BDEV    07    smd 0            .    % Removable part of SMD 0
BDEV    08    hd   d y c           % IMI hard disk
```

% Primitive terminal device:

```
RAW     raw_fdc                    % FDC primitive terminal driver
```

% Root device:

```
ROOT    none          % ROOT none    (Means: Ask the operator)
                      % ROOT boot    (Means: Same as boot disk)
                      % ROOT 6 0     (Means: Use device 6:0)
```

% Automatic login name:

```
LOGIN   system        % LOGIN system  (For Boot System)
                      % LOGIN         (For no auto login)
```

% Customized logon message:

```
LOGMSG  Boot System                % Any message can be here
```

% Default access:

```
ACCESS  rewa.re.re    % Files created will have this access
                      % unless it is changed here
```

```
%                      SYSTEM PARAMETERS
%
% NOTE: Be sure you are aware of the ramifications of altering
%       these values prior to changing them.  See Cromix-Plus User's
%       Reference Manual 023-5013 for details.

   bufcnt   30         % Number of memory resident data blocks

   inocnt   30         % Number of memory resident inodes

   filcnt   80         % Number of files which can be opened
                         simultaneously

   usrcnt   24         % Number of process tables

   logcnt   24         % Number of users who can be logged in at the same
```

                              time

   ß   mntcnt 8   % Number of devices that can be mounted at any time

   16   lckcnt 16   % Number of locks that can be installed

        freecnt 2048   % Number of bytes in the system memory pool

     END

Character device drivers

tty   Supports terminals on FDC and on TUARTs. The driver name must be followed by a list of minor device numbers supported. Applicable minor device numbers are 0, 2, 5, 6, 7, 8, 9, 10, 11

qtty   Supports both octart and IOP terminals. Ensure that the IOPs and/or octarts will have the code downloaded (See iostartup.cmd). Arguments are IOP/OCTART numbers (1, 2, 3, or 4) which have the quadart.iop or octart.iop code downloaded.

sysdev   This driver must be present. It provides null device. No arguments.

timer   This driver must be present. It supports the timer and the real time clock. No arguments.

lpt   This driver supports parallel printers. Arguments are minor device numbers supported. Available values are 2, 5, 6, 7, 8, 9, 10, 11.

typ   This driver support up to two fully formed character printers (spinwriter). Arguments are the supported minor device numbers. Available values are 5 and 2. See help mode.

slpt   Supports serial printers on FDC and on TUARTs. The driver name must be followed by a list of minor device numbers supported. Applicable minor device numbers are 0, 2, 5, 6, 7, 8, 9, 10, 11 with possible communication protocol offsets. Use above numbers for XON/XOFF protocol, add 64 for CLQ type printers, add 128 for ETX/ACK protocol.

qslpt   Supports both octart and IOP serial printers. Ensure that the IOPs and/or octarts will have the code downloaded (See iostartup.cmd). Arguments are IOP/OCTART numbers (1, 2, 3, or 4) which have the quadart.iop or octart.iop code downloaded.

ffp   Supports FFP processor driver. No arguments.

tape   Supports up to four nine track tape units. The IOPs must be loaded with the tape8.iop or with the tape16.iop driver. The tape16.iop driver can be used only with a 64K IOP board and it allows block sizes up to 16K. Arguments are IO numbers (1, 2, 3, or 4). Minor device numbers corresponding to these IO numbers are 0, 1, 2, or 3.

cnet   Supports CNET hardware. No arguments.

Block device drivers
=======================

cflop   Supports Tandon or PERSCI drives. No arguments. The minor device
number is defined as
     unit + small + dual
where
     unit   =   0, 1, 2, or 3 for A, B, C, D
     small   =   4 if 5", zero if 8"
     dual   = 16 if drives in pairs (PERSCI), zero if not

uflop   Supports Tandon or PERSCI driver in uniform format. No arguments.
In uniform format all tracks are in the same format, all sectors are the
same size, sector size might be 128, 256, or 512 bytes. Minor device
number describes the physical characteristics of the device. Compute
the minor device number as
     unit + small + dtrack + dual + sside + sdens
where
     unit   =   0, 1, 2, or 3 for A, B, C, D
     small   =   4 if 5", zero if 8"
     dtrack =   8 if double tracked (not supported), zero otherwise
     dual   = 16 if drives in pairs (PERSCI), zero if not
     sside   = 32 if single sided, zero if double sided
     sdens   = 64 if single density, zero if double density

allmem   Supports access to all of system memory (amem). No arguments.

tflop   Supports up to two floppy tapes. Minor device numbers are
     unit + drive + full
where
     unit   = 0   whole tape (6 streams)
             1   streams 0 and 1 only
             2   streams 2 and 3 only
             3   streams 4 and 5 only
     drive   = 0   AB
           = 4   CD
     full   = 0   245 segments per stream
          = 8   255 segments per stream (we cannot initialize it)

ramdsk   Supports 4 Ram disks. Use Ramdisk utility to allocate room. No
arguments.

stdc   Supports up to two STDC hard disks. No arguments.

smd   Supports up to four SMD hard disks (two drives on each of two
controllers). Arguments specify beginning head numbers of drives to
be included. Arguments are created as follows:
     controller + drive + fixed + head
where
     controller   = 0   controller at base port 0x38

		128	controller at base port 0xe8
drive	= 0		for controller drive 0
		64	for controller drive 1
fixed	= 0		for removable part of drive
		32	for fixed part of drive
head	= 0-31		beginning head number

hd   Supports up to 4 hard disks on WDI-II board. No arguments.

Primitive terminal drivers
==========================

raw_fdc   Supports primitive character I/O on the terminal connected to 16FDC
          or 64FDC board. No arguments.

Reader Responses To This Documentation

Dear Reader,

We have made a sincere effort to provide you with the information you need in this manual. If you should find the documentation deficient or in error, let us know so we can correct it. We appreciate and value your response; it will be useful in improving the documentation. Please detach and use the Reader Response Card below to send us your comments.

Thank you for your time and interest in Cromemco products.

Sincerely,

Winthrop A. Stiles III

Technical Publications Manager

(Detach Here)

Cromemco®

Reader Response Card

To: Winthrop A. Stiles III,
    Technical Publications Manager

Re (Manual title):_____

My System is (Specify configuration):_____

The following information is incorrect (Please specify page number):_____

_____

_____

_____

_____

_____

_____

(Detach Here)

(Fold Here)

The following additional information would be helpful:_____

_____

What general suggestions do you have for improving this manual?_____

_____

If you need a response from Cromemco, please print your name, mailing address, and telephone number:

Name:_____

Address:_____

_____

Telephone: (_____) _____

(Fold as indicated and tape the edge)

Cromemco®