TEK INSTRUCTION MANUAL

Part No. 061-2880-00
Product Group 16

4170

LOCAL GRAPHICS PROCESSING UNIT

First Printing SEP 1983
Revised NOV 1984

Tektronix
COMMITTED TO EXCELLENCE

WARNING

This equipment generates, uses, and can radiate radio frequency energy and if not installed and used in accordance with the instruction manual, may cause interference to radio communications. It has been tested and found to comply with the limits for Class A computing devices pursuant to Subpart J of Part 15 of FCC Rules, which are designed to provide reasonable protection against such interference when operated in a commercial environment. Operation of this equipment in a residential area is likely to cause interference in which case the users at their own expense will be required to take whatever measures may be required to correct the interference.

MANUAL REVISION STATUS

PRODUCT: 4170 Local Graphics Processing Unit

This manual supports the following versions of this product: B010100 and up.

REV DATE	DESCRIPTION
SEP 1983	Original Issue
OCT 1983	Revised: pages 1-1, 4-39, 4-40, 5-5, 6-4, 8-2, 8-3, 8-5, 8-21, 8-22, 8-35, 8-37, 8-42, 8-43, 9-5, 9-6, 9-10, 9-47, 9-48, 9-49, 9-50, 9-51, 9-52. Added: pages 8-2A, 8-22A, 9-6A, 9-48A, 9-52A, 9-52B, 9-52C, 9-52D.
DEC 1983	Revised: pages 5-5, 6-1, 9-6, 9-6A, 9-10. Added: page 9-10A.
JAN 1984	Revised: pages 9-20 and 9-21. Added: page 9-20A.
MAR 1984	Pages replaced, changed, or deleted to correct miscellaneous errors and to support Version 1.2 software. Contents list and Sections 1, 4, 5, 6, and 7 replaced. Section 9, Appendix B, and 4110 Series Direct Terminal Interface Programmers Reference Manual (070-4534-01) deleted. Pages 2-4 and 3-7 changed.
APR 1984	Revised: pages iv, 4-1, 4-11, 4-13, 4-16, 4-17, 4-20, 4-22, 4-29, 4-33, 4-36, 4-39, 4-41, 5-4, 5-5, 5-24, 5-28, 5-29, 5-30, 7-35, 7-37, 7-39, 7-40, 7-41, 7-42, 7-43, 7-47, 7-52, 7-53, 7-55, 7-75.
NOV 1984	Revised: page 7-23.

CONTENTS

Section 4 INSTALLATION PROCEDURES

Section 5 GETTING STARTED

WARNING

The following servicing instructions are for qualified personnel only. To avoid personal injury, do not perform any servicing other than that described in the operating instructions unless you are qualified to do so.

ILLUSTRATIONS

TABLES

OPERATORS SAFETY SUMMARY

This general safety information is for both operating and servicing personnel. Specific warnings and cautions will be found throughout the manual where they apply, but may not appear in this summary.

TERMS

IN THIS MANUAL

CAUTION statements identify conditions or practices that can result in damage to the equipment or other property.

WARNING statements identify conditions or practices that can result in personal injury or loss of life.

AS MARKED ON EQUIPMENT

CAUTION indicates a personal injury hazard not immediately accessible as one reads the marking, or a hazard to property including the equipment itself.

DANGER indicates a personal injury hazard immediately accessible as one reads the marking.

SYMBOLS

IN THIS MANUAL


This symbol indicates where applicable cautionary or other information is to be found.

As Marked on Equipment


DANGER high voltage.


Protective ground (earth) terminal.


ATTENTION — refer to manual.


Refer to manual.

POWER SOURCE

This product is designed to operate from a power source that does not apply more than 250 volts rms between the supply conductors or between either supply conductor and ground. A protective ground connection by way of the grounding conductor in the power cord is essential for safe operation.

GROUNDING THE PRODUCT

This product is grounded through the grounding conductor of the power cord. To avoid electrical shock, plug the power cord into a properly wired receptacle before connecting to the power input or output terminals. A protective ground connection by way of the grounding conductor in the power cord is essential for safe operation.

DANGER ARISING FROM LOSS OF GROUND

Upon loss of the protective-ground connection, all accessible conductive parts (including knobs and controls that may appear to be insulating) can render an electric shock.

USE THE PROPER POWER CORD

Use only the power cord and connector specified for your product.

Use only a power cord that is in good condition.

Refer cord and connector changes to qualified service personnel.

USE THE PROPER FUSE

To avoid fire hazard, use only the fuse specified in the parts list for your product, and which is identical in type, voltage rating, and current rating.

Refer fuse replacement to qualified service personnel.

DO NOT OPERATE IN EXPLOSIVE ATMOSPHERES

To avoid explosion, do not operate this product in an atmosphere of explosive gases unless it has been specifically certified for such operation.

DO NOT REMOVE COVERS OR PANELS

To avoid personal injury, do not remove the product covers or panels. Do not operate the product without the covers and panels properly installed.

Figure 1-1. 4170 Local Graphics Processing Unit.

Section 1

INTRODUCTION

DO YOU NEED TO READ THIS MANUAL?

Yes, you need to read this manual to install the 4170 and to learn how to use it. But you probably don't need to read all of this manual -- perhaps only certain parts of it. The Suggested Reading Paths that follow will help to introduce you to the 4170 quickly, with as few problems as possible.

SUGGESTED READING PATHS

First Time Use

First, to install the 4170, use the instructions in Section 4 of this manual. Section 4 also contains procedures for connecting a terminal to the 4170. After completing the procedures in Section 4, then go to Section 5, and follow the instructions there for making a backup copy of the CP/M-86 Operating System diskette. (It is a good idea to make a backup of the Operating System diskette before you begin using the system; this way, if the diskette is accidentally damaged, you'll still have a copy of it.) In Section 5, you'll also learn how to get started using the optional hard disk, host communications, and peripherals.

After Installing the 4170

After you have made a backup of the Operating System diskette, you are ready to begin using your 4170. If you are new to the CP/M-86 Operating System, read Section 6 and Digital Research's CP/M-86 Operating System User's Guide to learn how to use some other CP/M-86 Operating System commands. (Knowing the CP/M-86 commands will make it easier for you to run specific applications programs later.) At this point, you are ready to begin using a specific applications program, such as Wordstar or SuperCalc. Follow the instructions that came with the application program for getting started.

If you are an experienced CP/M-86 user, to refresh your memory about CP/M-86 commands, refer to the CP/M-86 Operating System User's Guide or the CP/M-86 Operating System Command Summary. Then proceed with using a specific applications program.

If You Are An Experienced Programmer

After installing the 4170 and making backup copies of the diskettes, experienced programmers may skip to Section 7, which contains complete descriptions of the Tektronix-supplied utilities that supplement the CP/M-86 commands. Section 7 also gives further information on the host communications utility, MODEM86. Finally, Section 7 describes the applications software -- FORTRAN, IGL, GSX, and DTI -- supplied with the 4170. Depending on your specific application, after reading Section 7 you may want to refer to other documentation supplied with the 4170 (see the list of manuals supplied with the 4170, later in this section).

Servicing the 4170

If the 4170 requires service, contact your local Tektronix field office. (This manual includes service and maintenance information that the Tektronix service technician may use; see About This Manual, which follows.)

ABOUT THIS MANUAL

This instruction manual contains:

- Section 1, Introduction (this section). Introduces the 4170, its features and options, and the manuals and diskettes supplied with it.
- Section 2, Specifications. Provides specifications for the 4170.
- Section 3, Controls, Indicators, and Connectors. Shows the locations of the controls, light indicators, and connectors on the 4170's front and rear panels.
- Section 4, Installation. Gives the procedures for installing the 4170 and connecting a terminal to it.
- Section 5, Getting Started. Contains instructions for the first-time users, and examples of using host communications and peripherals.
- Section 6, Operating Information. Contains exercises to help you become familiar with some of the CP/M-86 commands.
- Section 7, Programming Information. Contains reference information on the applications software.
- Section 8, Self-Test. Describes how to run the adjustment self-test.
- (Section 9 has been deleted. The information that it previously contained now appears in other sections.)
- Section 10, Glossary. A list of terms used in this manual.
- Section 11, Service Safety Summary. Safety considerations for those performing service on this instrument.

- Section 12, Theory of Operation. Describes how the 4170's circuitry works.

- Section 13, Checks and Adjustments. Contains service procedures for qualified service technicians.

- Section 14, Maintenance. Contains disassembly and reassembly procedures and preventive maintenance procedures.

- Section 15, Replaceable Electrical Parts; Section 16, Diagrams; Section 17, Replaceable Mechanical Parts. Parts lists and schematics for the 4170.

- Appendix A, Strap Information. Information on strap settings.

DO YOU NEED TO READ OTHER MANUALS?

The 4170 comes with 11 manuals. Some are Digital Research manuals and document the CP/M-86 Operating System; others are Intel manuals and cover Intel utilities. Finally, Tektronix supplies this manual, the 4170 Instruction Manual, as well as manuals on IGL.

What other manuals do you need to read? That depends on what you intend to do. If you'll be writing programs in FORTRAN, you'll probably want to refer to the Intel FORTRAN-86 manuals; otherwise, you may not need them.

Here's a list of the manuals that come with the 4170, with a brief description of what each manual contains and an indication of the manual's intended audience:

- Tektronix 4170 Instruction Manual (this manual). Contains installation instructions and operating information for users; describes the Tektronix commands that supplement CP/M-86; provides introductory information on applications software provided.

- Tektronix 4010C01 PLOT 10 Interactive Graphics Library Users Manual and Reference Guide. The users manual documents the IGL library of graphics subroutines. (A subset of IGL subroutines is included with the 4170; expanded IGL capabilities are available as options.) The reference guide is a quick-reference summary for experienced IGL programmers.

- 4110 Series Direct Terminal Interface Programmers Reference Manual. Documents the DTI routines for Local Programmability.

- CP/M-86 Operating System User's Guide (1). A manual for beginning users of CP/M-86. Describes in detail the CP/M-86 Operating System, the CP/M-86 commands, and the CP/M-86 line editor commands.

- CP/M-86 Operating System Command Summary (1). A quick-reference card that summarizes the purpose and syntax of CP/M-86 commands. Intended as a refresher for those familiar with CP/M-86.

---

(1) Published by Digital Research, Inc.

- CP/M-86 Operating System, System Guide (1). A system programmers' manual that clarifies the differences between CP/M-86 and CP/M-80.

- CP/M-86 Operating System Programmer's Guide (1). Explains ASM-86 operation for assembly language programmers.

- FORTRAN-86 Users Guide and Pocket Reference (2). Provides information on the language, the compiler, and the execution of FORTRAN-86 programs. The Pocket Reference summarizes the information for experienced FORTRAN-86 programmers.

- iAPX86,88 Family Utilities Users Guide and Pocket Reference (2). Describes the iAPX Family utilities used by ASM-86 and FORTRAN-86 programmers.

- GSX-86 Graphics Extension User's Guide (1). Describes how to set up your system to run GSX-86 programs (this is the Graphics System Extension for the CP/M-86 operating system); includes information on devices supported by GSX.

- GSX Graphics Extension Programmer's Guide (1). Provides the information necessary for programmers to adapt GSX for other devices.

If you purchase CP/M-86-compatible software, you'll probably get additional manuals. For example, if you purchase Wordstar (a text-editing applications program) from Tektronix, you'll receive instructions on how to get the software running and also instructions on how to use the text-editing commands.

---

(1) Published by Digital Research, Inc.

(2) Published by Intel, Inc.

ABOUT THE 4170

With a TEKTRONIX 4170 Local Graphics Processing Unit, you can run most applications software packages that are compatible with CP/M-86. The 4170 is often used with a Tektronix 4105, 4107, 4109, or 4014 terminal because it gives you the processing power you need for writing, editing, compiling, linking, and debugging graphics programs for those terminals. The 4170 provides this graphics processing power locally; that is, when you connect your terminal to the 4170, the 4170 acts as the host computer. If you desire, you can also connect the 4170 to a host mainframe, and run applications programs either from the host or from the 4170.

PRODUCT DESCRIPTION

The 4170 contains:

- A processing unit and local memory.
- Two flexible-disk drive units, which provide additional storage for data and programs.
- Interfaces for connecting the 4170 to a terminal, a host, and peripherals.

The operating system for the 4170 is the CP/M-86 Operating System, which is loaded from a diskette in one of the two flexible disk drives or from the optional hard disk. The operating system gives you a set of commands that allow you to control the transfer of information between various parts of the 4170 system; the operating system enables you to copy files, for example, from a flexible disk drive to a printer.

An important feature of the 4170 is its simple design -- you, the user, can install the 4170 and its options, and you can service most of the instrument. Therefore, this manual contains the information you'll need for installation and service, as well as operating and programming information.

4170 SYSTEM CONFIGURATIONS

A terminal connects to one of the three peripheral ports on the back of the 4170. The two remaining ports connect to peripherals such as a plotter, a printer, or a graphics tablet. You can use the Option 09 parallel interface to connect a printer that requires a Centronix-style parallel interface.

The 4170 and its Terminal

The 4170 must be connected to an RS-232 terminal. This allows you to control the 4170 system.

The terminal's keyboard is the main input device for sending information to the 4170; the terminal's display screen is the main output device for receiving data from the 4170. Read your terminal operator's manual for a complete description of its controls and indicators.

When you connect a terminal to the 4170, you have a local processing system. The 4170, with operating system and applications software, performs the functions of a host computer.

The 4170 and a Host System

When connected to a host system, the 4170 acts as an interface between the host and the terminal. An applications program may run on the host while the 4170 performs the local processing tasks as directed by the host software. The processing is then "shared" by the host and the 4170.

Section 7 provides programming information that you may need to interface your host to the 4170.

4170 SOFTWARE

Your 4170 comes with 10 diskettes. The software on the 10 diskettes is organized to reduce the number of times that you must change diskettes.

The 10 diskettes supplied with your 4170 are:

- 4100P01 CP/M-86 Operating System
- 4100P01 MODEM-86
- 4100P01 FORTRAN (2 diskettes)
- 4100P73 PLOT 10 Interactive Graphics Library Fundamental Support for 4170 (2 diskettes)
- 4100P01 GSX Graphics System Extension (2 diskettes)
- 4100P01 DTI and BIOS Source
- IDD Program Exchange (unsupported software)

The following paragraphs introduce you to the operating system and the software. (For detailed descriptions about using the software provided, see Section 7, Programming Information.)

The CP/M-86 Operating System

The operating system for the 4170 is CP/M-86. CP/M-86 is a single-user, single-tasking operating system that is most useful in interactive applications such as program development and debugging, small business computing, and word processing. The CP/M-86 operating system manages the system memory, the disk drives, and peripheral devices (such as terminals, printers, tablets, and plotters) connected to the 3PPI and Option 09 ports. Refer to the CP/M-86 Operating System System Guide, a Digital Research publication, for a more detailed description of the operating system.

Experienced system programmers may want to modify the BIOS portion (the Basic Input/Output System) of the 4170 operating system. To do this, refer to the documentation file "readme.doc" on the FDTI and BIOS operating system diskette for more information.

Additional Operating System Utilities

Tektronix has written and included seven operating system utilities: AR, CONFIG, DEL, FORMAT, RUN, SD, and SETDEV. The AR utility makes archive copies of files. The CONFIG utility defines the logical-to-physical device assignment and sets RS-232 communication parameters for the 4170 peripheral ports. DEL erases a file. FORMAT defines the physical and logical organization of data located on the disk. The RUN utility loads and executes load-time-locatable programs. The SD utility sorts a directory alphabetically. The SETDEV utility is similar to CONFIG but easier to use; SETDEV allows you to check and selectively change logical-to-physical device assignments, device communications parameters, and the default and boot drive specifications.

MODEM86

MODEM86 provides communications between the 4170 and the host computer. A description of how to start host communications with MODEM86 is given in Section 5; however, for a more complete explanation, refer to Section 7, Programming Information.

FORTRAN-86

FORTRAN-86 is an extended version of the FORTRAN 77 subset as defined by the American National Standards Institute (ANSI). It is supplied to you on two diskettes. Refer to Section 7 and the Intel FORTRAN-86 User's Guide for details about using FORTRAN-86.

IGL

The Interactive Graphics Library (IGL) provides a library of FORTRAN-callable, device-independent subroutines that provide high-level graphics functions. IGL should be used when working with applications programs that require device and host independence. IGL follows the SIGGRAPH CORE proposed standard. Usually these programs will be uploaded and/or downloaded from a host mainframe. The supplied routines include the IGL primary command set and the device drivers for the 4105, 4014, and the 4662 Option 31. Refer to Section 7, Programming Information, for more details about IGL.

GSX-86

GSX-86 is a device-independent graphics system for use with CP/M-86. GSX-86 should be used when the applications program will be transported to a non-Tektronix graphics device. The GSX-86 Graphics Extension Programmer's Guide and the GSX-86 Graphics Extension User's Guide, written by Digital Research, are also supplied with the 4170. Additional information, including information about Tektronix extensions, is in Section 7.

DTI

DTI is a library of FORTRAN-callable subroutines (written by Tektronix) for the 4100 Series terminals. DTI controls or emulates a 4100 Series terminal's graphics features to simplify programming support for the terminal. Section 7 contains information on using DTI with the 4170; it also highlights the differences between using the 4170 DTI and the DTI supplied with 4100 Series Local Programmability. (Included at the end of this manual is the 4100 Series DTI Programmers Reference Manual, which describes the use of the DTI supplied with 4100 Series Local Programmability.)

ACCESSORIES

The following accessories are standard equipment for the 4170:

- Manuals -- see the list of manuals earlier in this section.
- Cables and connectors:
  - 4170 power cord
  - 4170-to-terminal power cord
  - RS-232 host port cable
  - Self Test loop-back connector
- Box of 10 software diskettes
- Box of 10 blank diskettes

Optional accessories include:

- Alignment diskette
- Extender board
- Head cleaning kit (for flexible disk drive units)

In addition, you can order these manuals as optional accessories:

- ECC (Error Correcting Code) RAM Service Manual
- 4110 Series Mass Storage Service Manual
- 3PPI Instruction Manual
- 5-1/4 Inch Disk-Drive Service Manual (by Shugart Associates, Inc.)
- Hard Disk Drive Service Manual (by Seagate, Inc.)

To order any of these items see Replaceable Mechanical Parts, Section 13, for part numbers.

Section 2

SPECIFICATIONS

This section lists two different types of specifications: those that cannot be verified by the user (environmental, physical, or static) and those that can be verified as actual operational parameters. The user-verifiable specifications can be verified through the adjustment procedure located in Section 13. (User-verifiable specifications are listed only in Table 2-11 and parts of Tables 2-5 to 2-7.)

The following tables contain specifications or characteristics for the 4170.

Table	Description
2-1	Physical Characteristics
2-2	Environmental Characteristics
2-3	Electrical Characteristics
2-4	Installation Requirements
2-5	General Functional Characteristics
2-6	Host Interface Specification
2-7	Peripheral Interface Specification
2-8	Software Specifications
2-9	Diskette Media Characteristics
2-10	Diskette Drive Unit Characteristics
2-11	User-Verifiable Drive Unit Specifications

PERFORMANCE CONDITIONS

To ensure proper performance, the following conditions must be met. The specifications are valid only under these conditions:

- The 4170 must be operating at an ambient temperature of 50 to 104<degrees> F (10 to 40<degrees> C).
- The 4170 must be operated from an AC line supply with:
  - Frequency of 48 to 66 Hz
  - Voltage of 87 to 128 V (or 174 to 250 V)
  - Voltage crest factor of between 1.20 and 1.414
- The maximum allowable line voltage is 250 VAC.
- The 4170 must operate on a single-phase power source with its neutral conductor at or near ground potential.

CAUTION

Do not operate from two phases of a multi-phase system.

PHYSICAL CHARACTERISTICS

Physical characteristics pertain to the mechanical dimensions and weight of the product. Table 2-1 lists these characteristics. Figure 2-1 is a dimensional drawing of the product.

Table 2-1

PHYSICAL CHARACTERISTICS

Characteristics	Specification
Weight	49 lbs
Width	8.75 in
Height	23.50 in
Depth	24.75 in

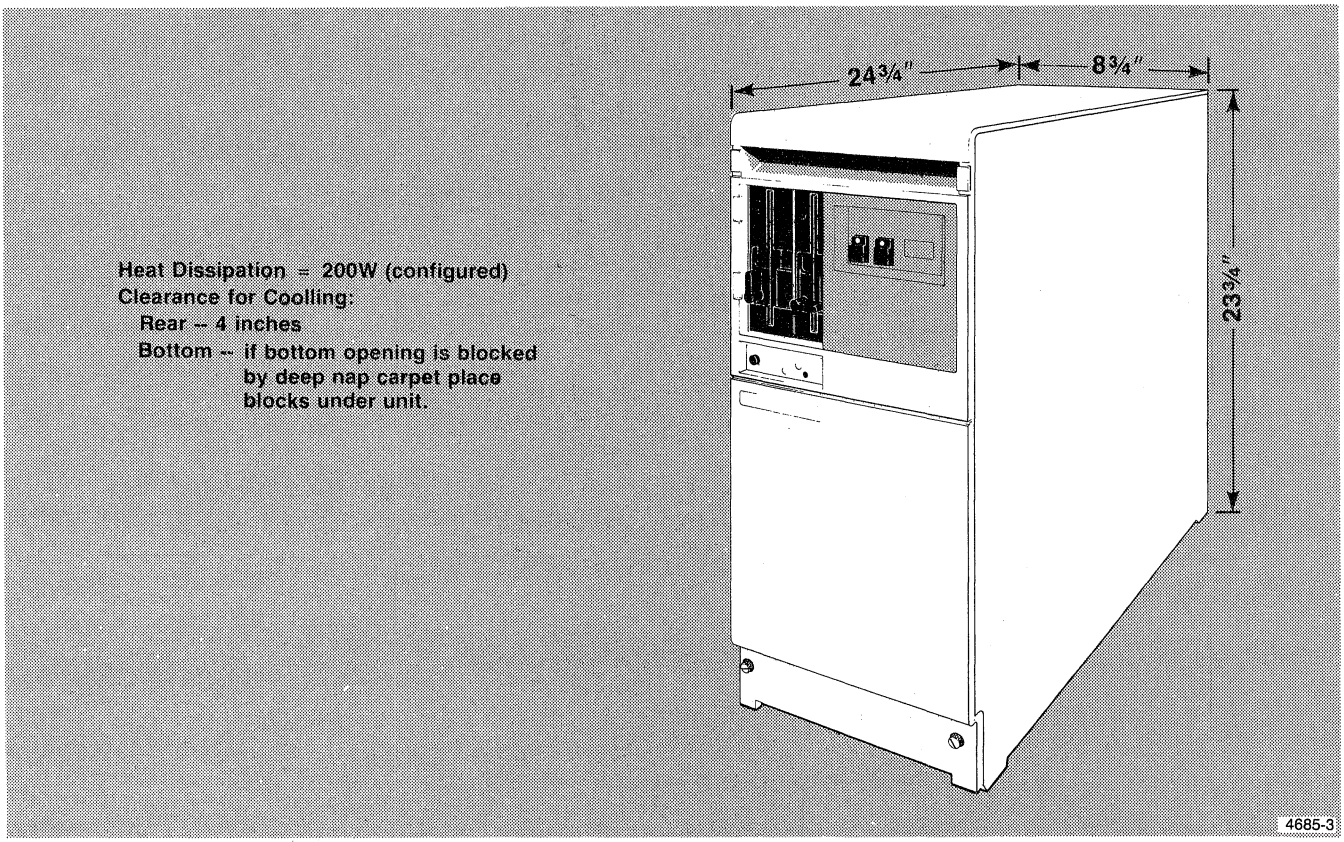


Figure 2-1. 4170 Physical Dimensions.

ENVIRONMENTAL CONDITIONS

Table 2-2 lists the operating environment characteristics that are required for proper operation. The electrical requirements listed under Performance Conditions (at the beginning of this section) must also be met.

Table 2-2

ENVIRONMENTAL CHARACTERISTICS

Condition	Specification
Temperature	
Operating	50 to 105<degrees> F (10 to 40<degrees> C)(a)
Nonoperating	-40 to +125<degrees> F (-40 to +50<degrees> C)
Humidity	
Operating	20 to 80% relative humidity (noncondensing)
Nonoperating	8 to 80% relative humidity (noncondensing)
Altitude	
Operating	To 15,000 ft (4572 m) (b)
Nonoperating	To 30,000 ft (9144 m)
Vibration --type, sine wave --duration, 15 min/axis	
Operating	
5-20 Hz	0.01 in p-p
20-55 Hz	0.25 g (peak)
55-20 Hz	0.25 g (peak)
20-5 Hz	0.01 in p-p
Nonoperating	
5-20 Hz	0.04 in p-p
20-55 Hz	1.0 g (peak)
55-20 Hz	1.0 g (peak)
20-5 Hz	0.04 in p-p

Table 2-2 (cont)

ENVIRONMENTAL CHARACTERISTICS

Condition	Specification
Shock (half sine wave, once in either direction of three orthogonal axes)	
Operating, with no physical damage	10 g, 11 ms, half sine
Operating, with no irrecoverable hard errors	4 g, 11 ms, half sine
Nonoperating	20 g, 11 ms, half sine
Desk handling Operating	The 4170 will continue operating under these conditions: Tilt 4170 on one bottom edge 10<degrees> from upright, then drop to supporting surface. Repeat using all four bottom edges of the 4170 cabinet. The 4170 sustains no damage, its operation is not interrupted, it loses no data, and does not change its operating mode as a result of this test.

Table 2-2 (cont)

ENVIRONMENTAL CHARACTERISTICS

Condition	Specification
Electrostatic immunity	
Operating	15 kV, 500 pF in series with 1000 <ohms>
Nonoperating	20 kV
Electromagnetic compatibility (related to line transients)(c)	
Oscillatory surge	
susceptibility	3.0 kV
vulnerability	3.5 kV
Unidirectional surge	
susceptibility	1.5 kV
vulnerability	2.0 kV

(a)Measured at 4170's rear panel.

(b)Derate the maximum operating temperature -1<degree>C for each 1,000 ft (300 m) above 5000 ft (1.5 km).

(c)This instrument qualifies under FCC Part 15 Subpart J Class A computing devices and VDE 0871/6.78 with respect to radiated and conducted emissions.

ELECTRICAL CHARACTERISTICS

These characteristics and specifications pertain to the 4170's power supply outputs and its specifications.

Table 2-3

ELECTRICAL CHARACTERISTICS

Characteristic	Specification
Fuse	Internal 5 A
Power supply input voltage requirements	87 to 128 Vac or 174 to 250 Vac

INSTALLATION REQUIREMENTS

Table 2-4 lists the factors (and values) to consider when selecting the operation site for the 4170. Also, check the dimensional requirements (in Table 2-1), and the environmental factors (in Table 2-2).

Table 2-4

INSTALLATION REQUIREMENTS

Characteristic	Specification
Heat dissipation	682 Btu (200 W). Includes Options 03,09, 10, 31, 45.
Surge current	34 A (typical)
Cooling clearance	4 in min at rear fan exhaust. 3 in min at floor level, perpendicular to front and rear panel. If carpet nap exceeds 1/2 in, provide increased air flow by supporting the 4170 to regain min clearance.

FUNCTIONAL CHARACTERISTICS

The functional characteristics are grouped in three categories: general characteristics (including 4170-to-terminal interfacing information), specifications for the host interface, and finally, specifications for the peripheral interface.

Table 2-5

GENERAL FUNCTIONAL CHARACTERISTICS

Characteristic	Specification
Memory size RAM (std) (optional) Disks (std) (optional)	 256 K 512 K (Option 29) 768 K (Option 30) 896 K (Option 31) 327.6 K (each) unformatted 10 M (Option 03)
Operating system	CP/M-86
Available languages	FORTRAN-86, ASSEMBLER
I/O ports	Host -- RS-232-C (1 ea) Terminal -- RS-232-C (1 ea) Peripherals -- RS-232-C (2 ea) Copier I/F port (option 9) 3 additional RS-232-C Ports (Optional)
Interface specification for primary terminal (4105, 4107, etc.) Modes Data rate Data protocol	 Full duplex (simultaneous transmit and receive) Up to 19.2k baud (sustained) Asynchronous

Table 2-6

HOST INTERFACE SPECIFICATIONS

Characteristic	**Specification**
Operational modes	Full duplex
Data rates	50, 75, 110, 134.5, 150, 300, 600, 1200, 1800, 2400, 4800, 9600, 19200 baud
Data protocol	Asynchronous
Protocols	Full duplex data communications DC1/DC3 flagging, for both host-to-4170 and 4170-to-host data flow DTR (data terminal ready) flagging for host-to-4170 data flow CTS (clear to send) flagging for 4170-to-host data flow

Table 2-7

PERIPHERAL INTERFACE SPECIFICATION

Characteristic	**Specification**
Operational modes	Simplex: one-way data transmission Full duplex: simultaneous transmit and receive
Data rates	50, 75, 110, 134.5, 150, 300, 600, 1200, 1800, 2000, 2400, 3600, 4800, 7200, 9600, 19200 baud
Data protocol	Asynchronous

Table 2-8

SOFTWARE SPECIFICATIONS

Characteristic	**Specification**
Operating system	CP/M-86
Compiler	FORTRAN-86
Software support package	Local version of Tektronix PLOT-10 IGL (primary command set and some device drivers)

Table 2-9

DISKETTE MEDIA CHARACTERISTICS

Characteristic	**Value**
Type Soft-sectored	5-1/4" flexible diskettes: double-sided double-density 48 tracks per inch
Storage Environment Temperature Humidity	 +50 to +122<degrees>F (+10 to +50<degrees>C) 8 to 80%
Media lifetime Passes per track Insertions	 3 million 30,000 +

Table 2-10

DISKETTE DRIVE UNIT CHARACTERISTICS

Characteristic	**Double Density**
Format data capacity Per disk	327K bytes (formatted) 500K bytes (unformatted)
Per track	6,250 bytes
Transfer rate	250 kb/s
Access time (average)	93 ms
Recording density (inside track)	5876 b/in
Flux density	5,876 fcin
Track density	48 t/in
Number of tracks	40 tracks

Table 2-11

USER-VERIFIABLE DRIVE UNIT SPECIFICATIONS

Characteristic	**Value**	**Tolerance**
Rotational speed	300 r/min	<+/->3 r/min
Period between index pulses	200 ms	
Seek access time Track-to-track Settling time	6 ms 15 ms	
Motor start time	500 ms maximum	

Section 3

CONTROLS, INDICATORS, AND CONNECTORS

This section describes the controls, indicators, connectors, and other features on the front and rear of the 4170.

FRONT FEATURES

The front of the instrument contains the following features (see Figure 3-1):

- Front control panel (controls and indicators)
- Two disk drive units
- Cue card
- Lifting handle

The front panel is recessed where it meets the top panel, making a handle. Use this and the rear handle when moving the 4170.

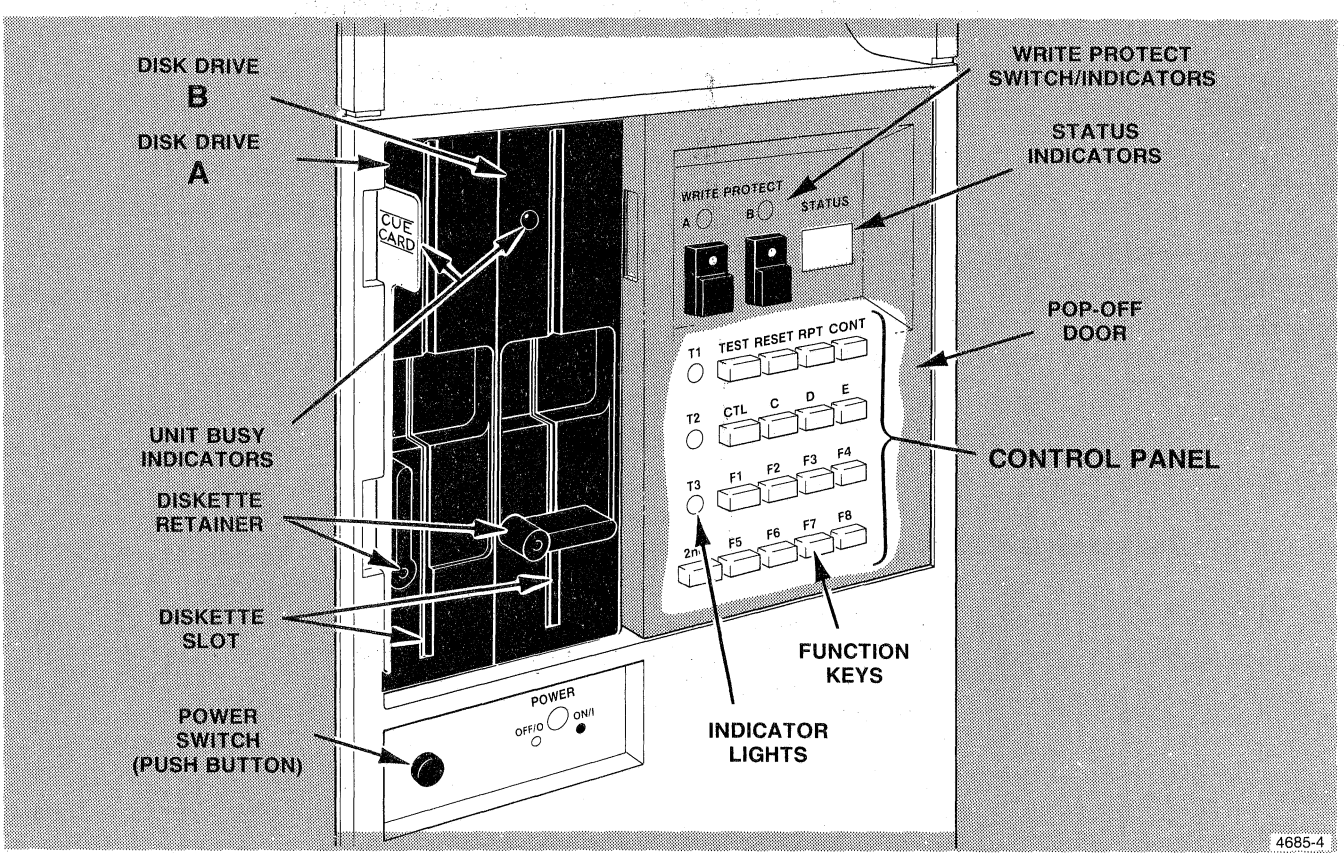


Figure 3-1. 4170 Front Panel Features.

CONTROL PANEL (EXPOSED ITEMS)

The control panel contains the following exposed items:

- **POWER** switch -- This switch turns on the 4170.
- **POWER ON** indicator -- Indicates the power is on by displaying a green light.
- **STATUS** -- Two digit display that indicates status codes during Self Test. Each display can only display the 16 hexadecimal digits. When Option 03 is installed, these displays indicates which track number the hard disk is seeking. That track number will not change until the hard disk is seeking another track.
- **WRITE PROTECT A** and **B** -- These two switches set write-protection for their respective flexible drive units. The lights indicate which drive units are currently write-protected.
- Disk Drives **A** and **B** -- Drive A is the drive located closest to the left-hand edge of the 4170. Drive B is the remaining drive. Each drive has a unit busy indicator and a diskette retainer. The unit busy indicator is located above the diskette retainer and lights when the drive is reading or writing on its diskette.

CONTROL PANEL DOOR

The control panel contains a small door that shields most of the control buttons and several indicators. This door prevents an operator's knee from bumping the control buttons. Remove this door by pulling out on its left edge (use the finger notch shown in Figure 3-2). This exposes the following controls and indicators:

- Three LED indicators:

  T1, T2, T3

  These indicators operate with Self Test to indicate what is happening during the diagnostic program. If you are not familiar with Tektronix terminals of the type that use these lights, see Section 8 for a complete explanation.

- Seventeen push buttons:

  TEST -- Starts Self Test when used with RESET.

  RESET -- Resets the 4170; also used

  with the TEST button to start Self Test.

  RPT -- Repeats current part of Self Test.

  CTL -- When pressed with other keys, sends control codes to processor. Same as the control key on terminals.

  C, D, E -- When used with the CTL button, sends control codes to the processor.

  F1 through F8 -- Performs the same functions as the special function keys on your Tektronix terminal keyboard.

  2nd -- Activates a function key's alternate function when pressed with one of buttons F1 through F8.

  CONT -- Continues running Self Test after it is halted.

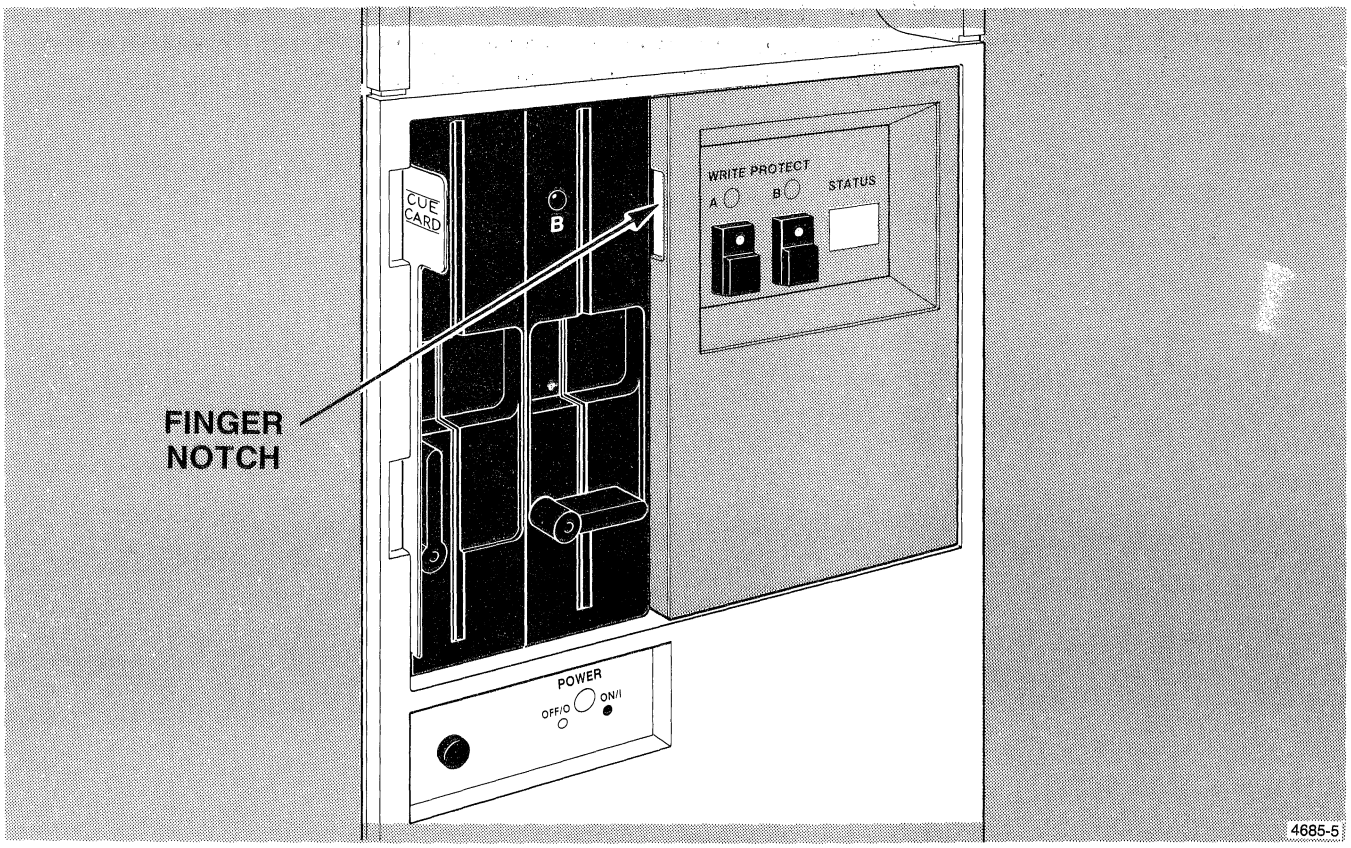


Figure 3-2. Removing the Pop-off Door.

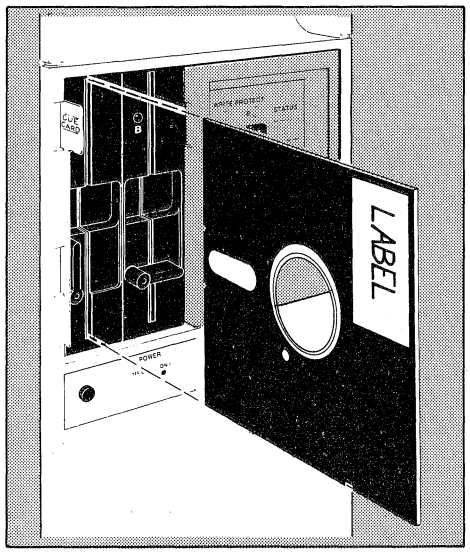
CUE CARD

The slot at the left of Disk Drive A contains a cue card. This card pulls directly out and contains abbreviated instructions for starting the 4170. It also contains some Self Test information. See Figure 3-3.

CUE CARD

POWER-UP AND OPERATING INFORMATION — SIDE 1

1. Insert the diskette containing CP/M-86 Operating System in drive A and close the diskette latch.
2. Set the connected terminal baud rate to 2400 and connect terminal to Terminal Port on 4170.
3. Press the POWER button on the 4170.
4. Wait for the 4170 Power-Up test to run. If the test is successful, the operating system message and prompt **A>** appears on the terminal screen. The system is now operational and you may enter commands.
   (NOTE: $^{C}_{R}$ = press Return key.)
   - To list files in your directory enter: **DIR** $^{C}_{R}$. Your files list (on disk drive A) is displayed on the terminal screen.
   - To format a new diskette, first insert a new diskette in drive B, then enter: **FORMAT** $^{C}_{R}$. When a format question appears, answer: **B** $^{C}_{R}$.
   - Learn more about the system by using the help feature, enter: **HELP** $^{C}_{R}$
5. Refer to the *4170 Instruction Manual* for more extensive and detailed operating information.



Tektronix

SELF-TEST DIAGNOSTIC INFORMATION

The Self-Test program may be used to diagnose system problems. If a hardware malfunction occurs, this program helps locate the problem, and may print an error message on the terminal screen. Section 8 in the *4170 Instruction Manual* includes a full explanation of Self-Test and its error messages. To run Self-Test:

1. Attach terminal to TERMINAL port of the 4170. Terminal baud rate must be 2400.
2. Open the door around the two WRITE-PROTECT switches on the 4170 front panel.
3. Simultaneously press the **TEST** and **RESET** buttons on the 4170 front panel. Hold down **TEST** for a few seconds after releasing **RESET**.
4. The first test performed is the keyboard test. Upon completion the bell rings once, then do one of the following:
   a. Press any key to resume Self-Test.
   b. Press **CNTRL D** to skip lengthy memory test.
   c. Press **CNTRL C** to start the adjustment portion of Self-Test.
5. If an error occurs a bell rings 2 or 3 times and an error code is displayed on the 4170 front panel. If possible, a message is also sent to the terminal.
6. When Self-Test is finished the operating system message and prompt **A>** appears on the terminal screen.

SUMMARY OF COMMANDS — SIDE 2

CUE CARD

Refer to the *CP/M-86 Command Reference Manual* for complete descriptions of each CP/M command – this is only a summary.

* **AR**	Archive a file
ASM86	8086 assembler
* **CONFIG**	Sets 4170 configuration
COPYDISK	Disk-to-disk copy/backup
DDT86	Dynamic (software) Debugging Tool
* **DEL**	Delete a file
DIR	Lists the directory of disk files
DIRS	Lists the system files on the disk
ED	Character file editor
ERA	Erases a filename from the directory
* **FORMAT**	Formats a new disk
GENCMD	Generates a command (executable) file
HELP	Help/information command
PIP	Peripheral Interchange Program
REN	Rename a file
* **RUN**	Execute a UDI program
* **SD**	Print an alphabetically sorted directory
STAT	Set/get status
SUBMIT	Execute a command file
TYPE	Copy a file to the terminal screen
USER	Set/get your user number

* Tektronix Utility - see *4170 Instruction Manual* for complete descriptions.

4685-6A

Figure 3-3. Cue Card.

DISK DRIVE FEATURES

The flexible-disk drive units each have a diskette loading slot. After a diskette is inserted, rotate the wing-type disk retainer clockwise. This holds the diskette in place.

WARNING

Do not attempt to remove a diskette while the unit-busy indicator is on. Attempting to remove a diskette when the unit-busy indicator is on will cause data loss.

Each drive unit has a unit-busy indicator that lights when the drive is writing or reading data to or from its diskette.

REAR PANEL FEATURES

The following connectors and switch are mounted on the 4170's rear panel (see Figure 3-4).

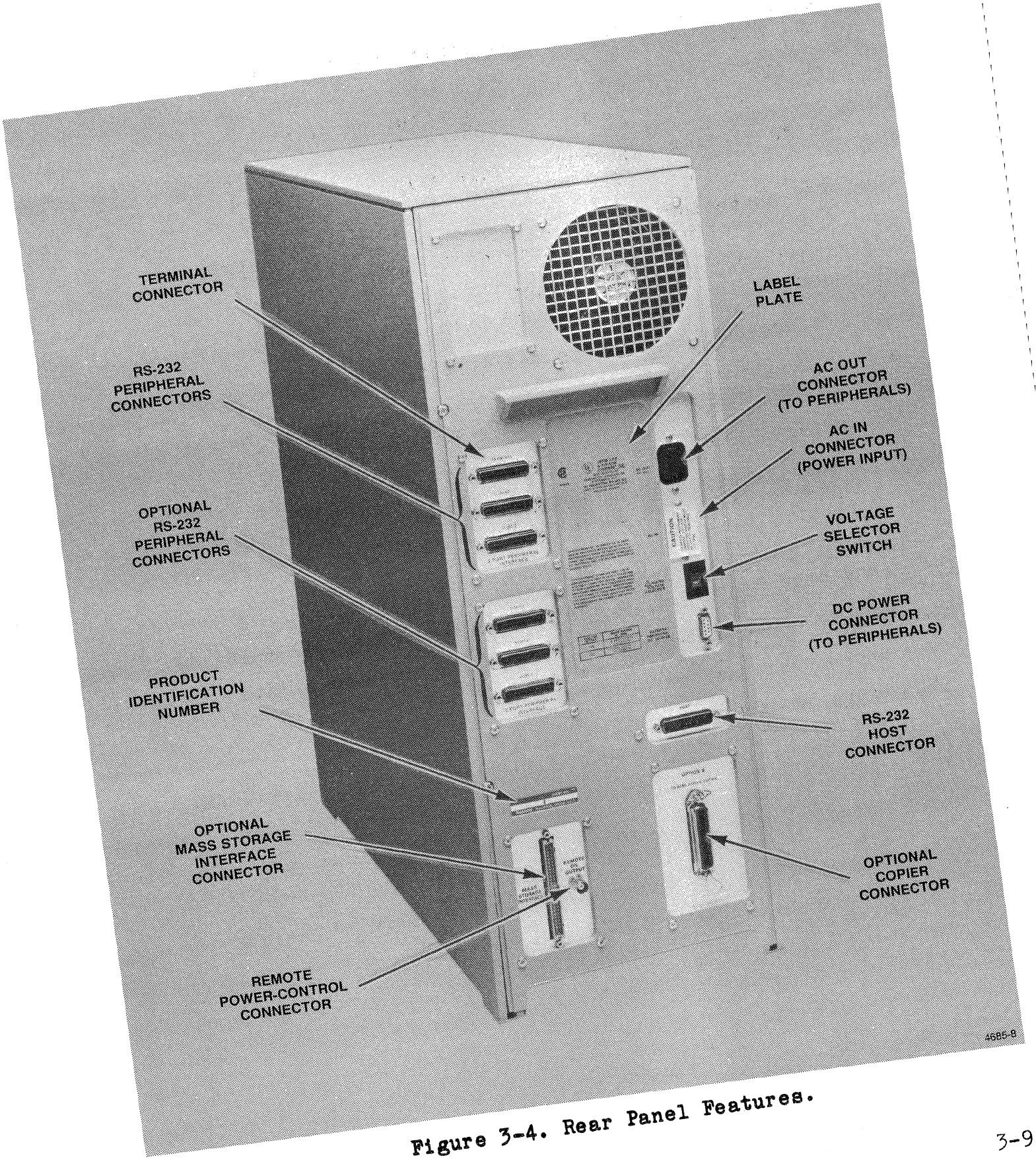


Figure 3-4. Rear Panel Features.

- **AC IN** -- This male IEC connector accepts a standard ac power cord.
- **AC OUT** -- This female IEC connector provides ac power for the terminal or other peripherals.
- **MAINS VOLTAGE SELECTOR** -- This switch sets the 4170 to receive either 115V or 230V ac power.
- **HOST connector** -- This 25-pin male connector supplies RS-232 communications to a remote host.
- **3-PORT PERIPHERAL INTERFACE** -- These 25-pin female connectors pass control and data to and from the 4170's peripherals
  - **Terminal** -- Connects the 4170 to its main terminal.
  - **Port 1** -- Connects the 4170 to a peripheral device or secondary terminal.
  - **Port 2** -- (same as Port 1).
- **EXTERNAL PERIPHERAL DC SOURCE** -- This 9-pin female connector supplies dc power to a peripheral such as a graphics tablet.
- **OPTION 9** -- This optional connector passes control and data signals to a Tektronix 4690 Series color copier.
- **MASS STORAGE INTERFACE** -- This connector passes data and control between the 4170's optional mass storage interface and an external mass storage device.
- **REMOTE ON OUTPUT** -- The circuitry behind this connector enables the 4170 to remotely operate optional disk units.

Section 4

INSTALLATION PROCEDURES

This section contains the procedures required to install your 4170. A repackaging procedure is also included in case you ship your 4170 to another location.

Tektronix has made these procedures as simple as possible. Persons totally new to the 4170 have successfully installed it using these procedures; have confidence that you can too. But before you begin, some words of caution: **READ EACH STEP THOROUGHLY AND LOOK AT EACH ACCOMPANYING ILLUSTRATION BEFORE ATTEMPTING THE STEP.**

Should you encounter a problem, please call your local Tektronix field office for assistance. Chances are that they can correct your problem over the phone.

SELECTING A SITE

Figure 4-1 shows the 4170 dimensions and clearances. Consider this information, potential cable routing problems, and air circulation problems when selecting an installation site.

NOTE

During the installation procedure you will need access to all sides of the 4170. Make sure that you have enough room before you begin.

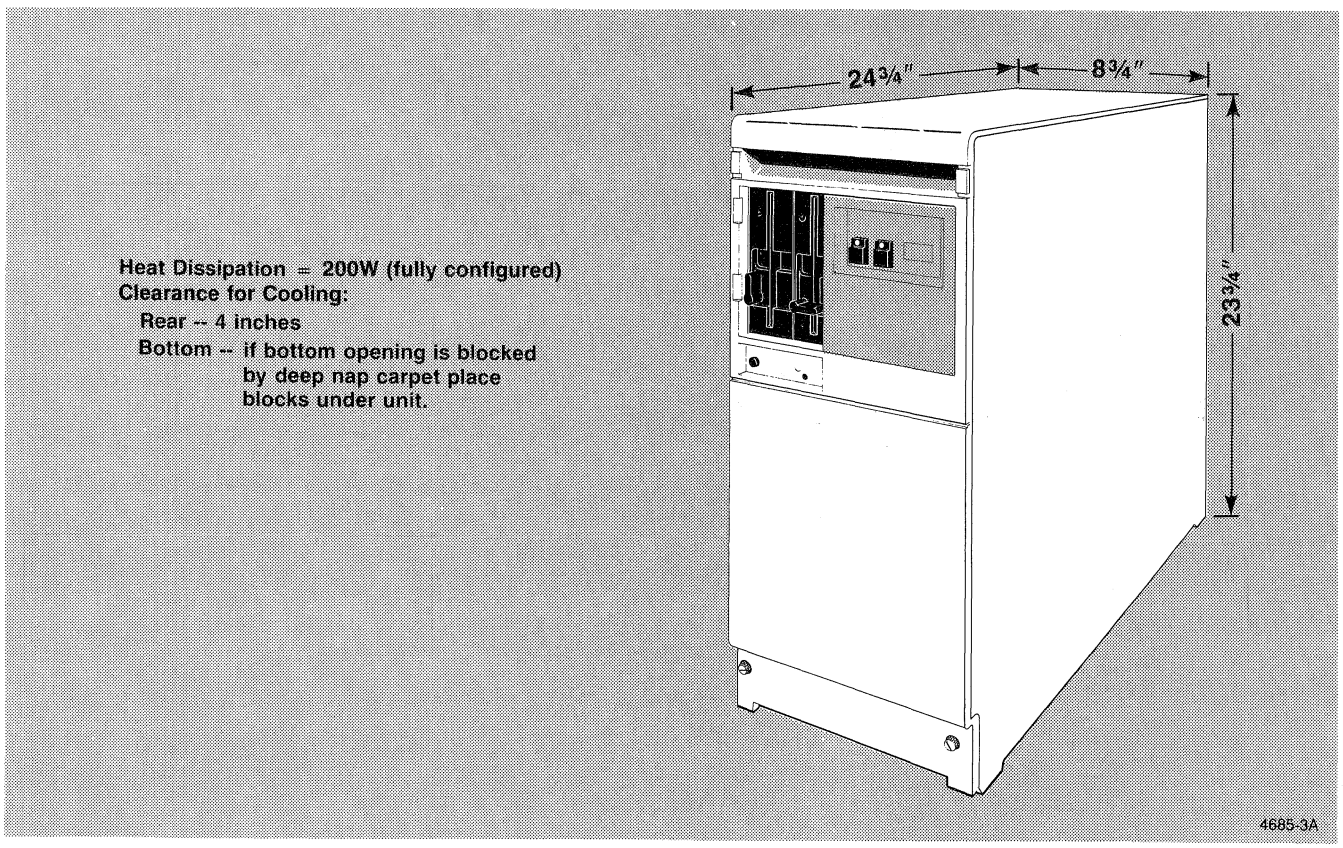


Figure 4-1. 4170 Dimensions.

INSTALLATION GUIDELINES

NOTE

DO NOT LAY THE 4170 ON ITS SIDE UNTIL INSTRUCTED TO IN THE PROCEDURE.

Most of the installation is done with the 4170 lying on its side. It is easier to get to the components when the unit is in this position. However, the circuit boards must be installed (with the board components down) as shown in Figure 4-2.

In addition, when installing a board, turn board ejectors so that they are parallel with card guides, push board into place, then lock board ejectors into place. Doing this will ensure that the board is seated in its connector at the rear of the card cage.

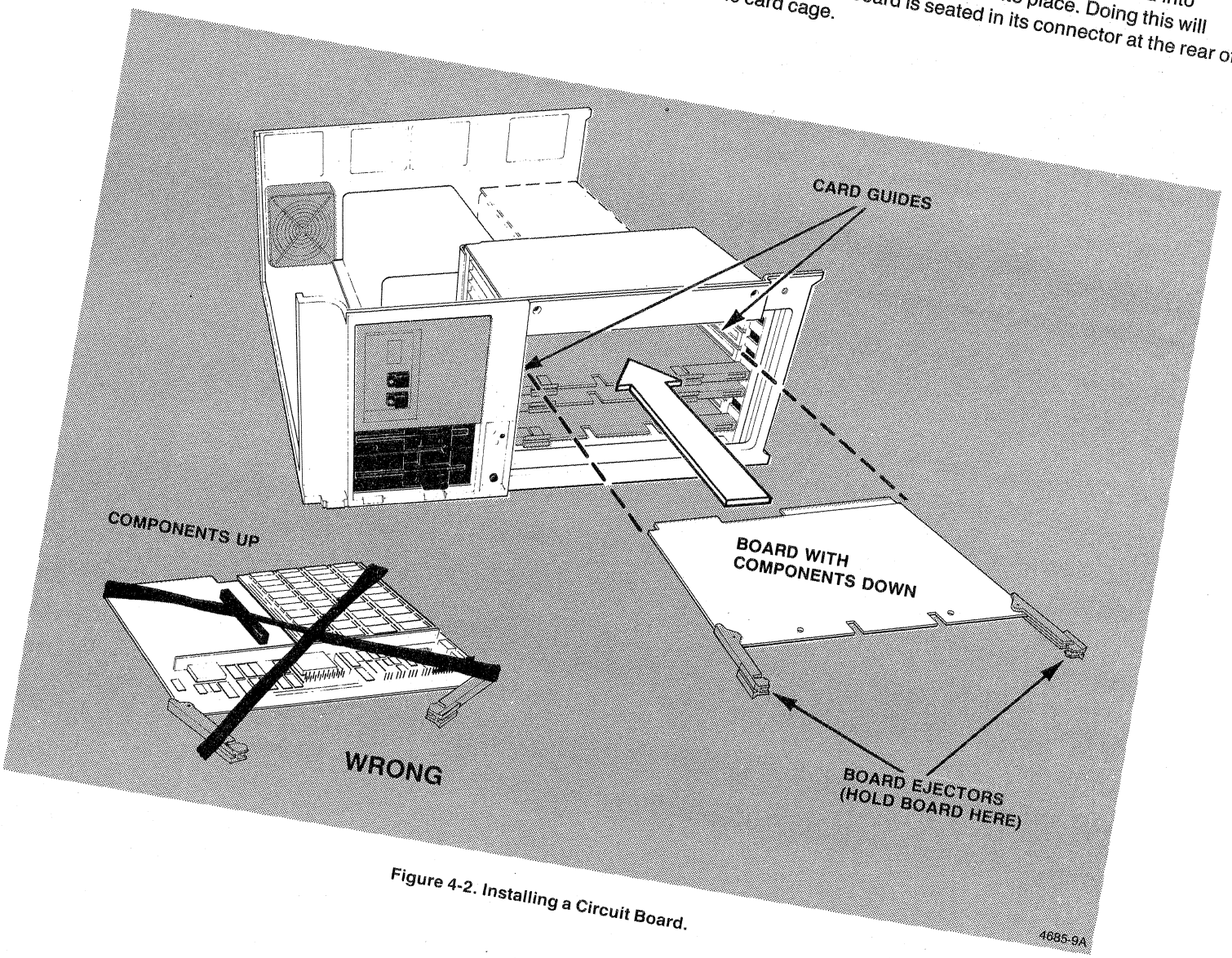


Figure 4-2. Installing a Circuit Board.

What To Install

Table 4-1 lists all the procedures required to install your 4170. The table also has two columns that you need to complete: the TO DO column and the DONE column. Look at the table now. As you can see, some of the procedures already have a check mark in the TO DO column. These are the procedures that must be done for all 4170 installations. To determine if you need to perform additional procedures:

1. Find the list on the box(es) your 4170 is shipped in to see what options were shipped with your 4170. (4170F03 is shipped in a separate box.)

2. Place a check mark in the TO DO column of Table 4-1 opposite each option shipped with your 4170.

The TO DO column of Table 4-1 now lists all the installation procedures that you must do to install your 4170. Review this list and temporarily remove this page from the manual. With this page removed, you can easily refer to Table 4-1 to find out what you must do next.

Table 4-1

INITIAL INSTALLATION PROCEDURES

Procedure	Shipped Option	To Do	Done
1. Unpacking	—	✔	
2. Voltage Selection/Checkout	—	✔	
3. Removing Front Cover	—	✔	
4. Removing Side Cover	—	✔	
5. Installing Memory Options	4170F30, 4170F31, 4170F32, 4170F51		
6. Installing Hard Disk	4170F03		
7. Installing Disk Interface	4170F44, 4170F45		
8. Installing Peripheral Interface	4170F10		
9. Installing Color Copier Interface	4170F09		
10. Replacing Side Cover	—	✔	
11. Replacing Front Cover	—	✔	
12. Connecting a Terminal	—	✔	
13. Verifying Operation		✔	

Before You Begin

There are 13 possible procedures for installing a 4170. These procedures begin immediately after this explanation and are a series of lettered steps. As you read each step, you will notice a small box before each step. Place a check mark in that box when you have completed the step.

When you complete the entire procedure, put a check mark in that procedure's DONE column in Table 4-1. This gives you a record of the completed procedure and guides you to the next procedure.

Begin the installation with Procedure 1 (unpacking).

IF YOU ADD AN OPTION LATER

Table 4-2 identifies the procedures that you will need to do if an option is added after the 4170 has already been installed. You don't need to go through the entire installation procedure, just those steps listed in this table.

Table 4-2

FUTURE OPTION INSTALLATION PROCEDURES

Option	Installation Procedure
4170F03	3, 4, 6, 10, 11, 13
4170F09	3, 4, 9, 10, 11, 13
4170F10	3, 4, 8, 10, 11, 13
4170F30, F31, F32, F51	3, 5, 11, 13
4170F45	3, 4, 7, 10, 11, 13

1. UNPACKING

NOTE

Save all packing material and cartons for repackaging the 4170 in case you need to ship it to another location.

A. ☐ Before unpacking the unit, be sure to visually inspect the shipping container for damage. Any damage should be reported to the carrier immediately.

NOTE

You can remove the 4170 from the shipping carton by lifting it out of the box (it weighs about 40 pounds) or by cutting the end of the box and sliding the 4170 out. Study the pictures and decide which you want to do. If you decide to lift it out start with step B. If you decide to cut the carton, start with step F.

B. ☐ Lift the protective packing off the top of the 4170 and set it aside.

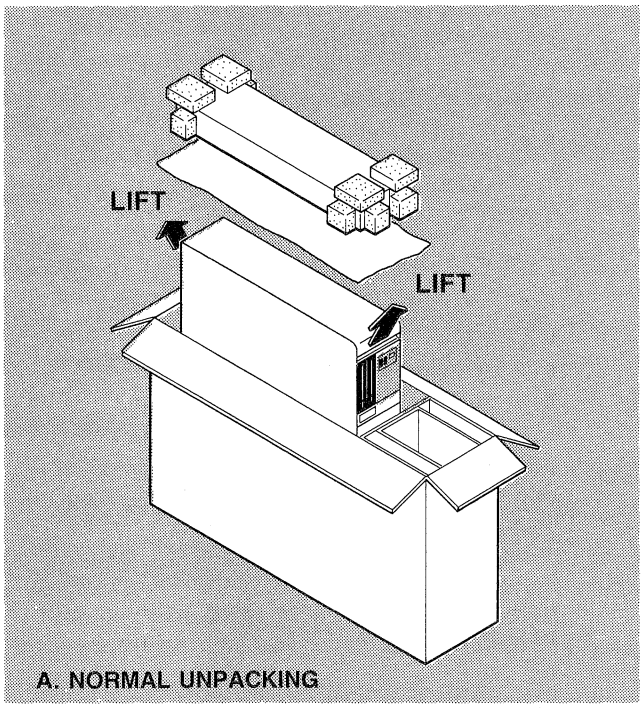
NOTE

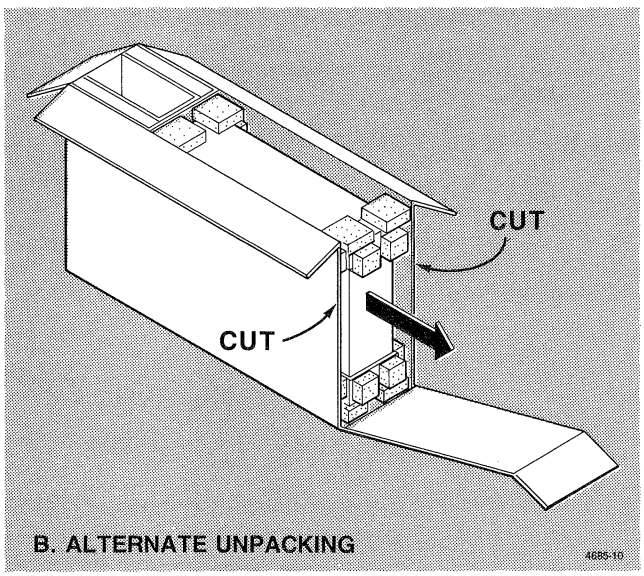
It is easier to complete the installation of the options if you set the 4170 on a desk or table where you can walk around it.

C. ☐ Remove the protective plastic cover from the 4170.

D. ☐ Lift the 4170 out of the carton using the two lifting handles, just under the top front edge and about four inches down from the back edge (indicated by arrows). BE CAREFUL WHEN LIFTING TO AVOID HURTING YOUR BACK.

E. ☐ Set the 4170 upright on a desk or table and skip to step J.





F. ☐ Cut the end of the carton as shown and fold the end of the carton down.

G. ☐ Slide the 4170 out of the carton.

H. ☐ Remove the packing material and plastic sheet from the top of the 4170. Set the packing material aside.

NOTE

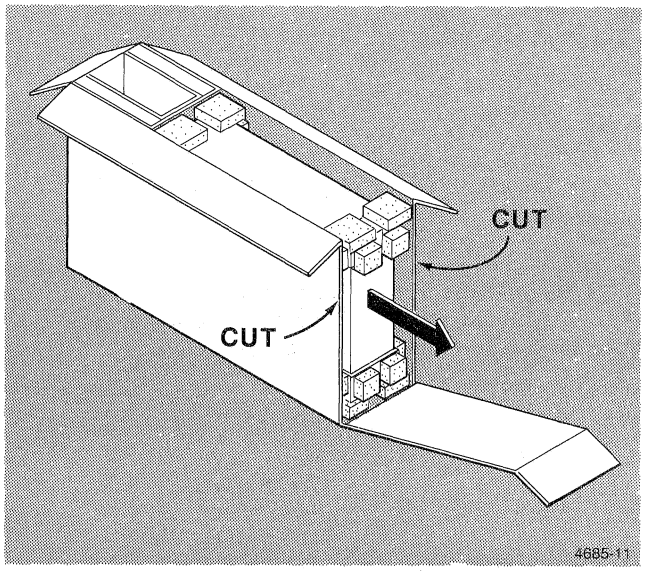
It is easier to install options in the 4170 if you set it on a desk or table where you can walk around it.

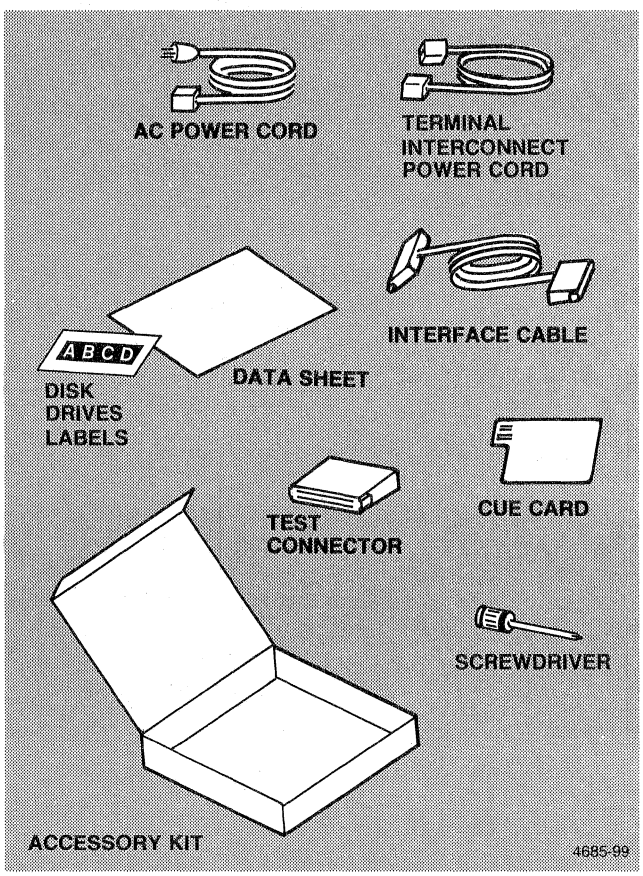
I. ☐ Carefully pick the 4170 up, lifting it off the bottom packing material, and set it upright on a desk or table.

J. ☐ Remove, BUT DO NOT OPEN, the remaining boxes, containing the 4170 software and optional F-kits from the carton and set them aside until later.

K. ☐ Check that accessories in box with the 4170 manual are all present.

L. ☐ You are done with Procedure 1. Check it off Table 4-1 and go to Procedure 2.





2. VOLTAGE SELECTION/CHECKOUT

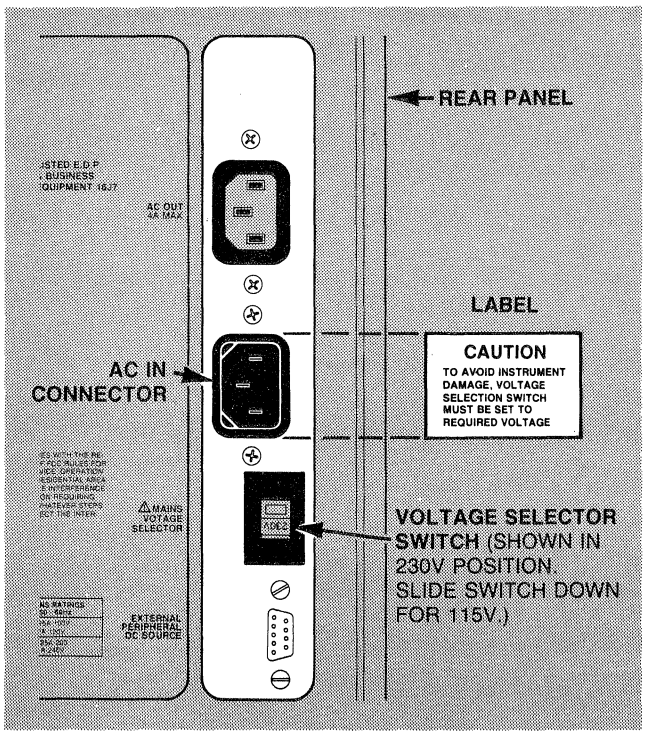
A. ☐ Set the Voltage Selection switch on the back of the 4170 to correspond to your facility line voltage (115V or 230V). If you're not sure, ask someone.

B. ☐ Remove the CAUTION label from the AC IN CONNECTOR. Remove the ac power cord from the accessory box and connect it to the AC IN connector on the rear panel.

C. ☐ Connect the power cord to the nearest ac outlet.

CAUTION

Verify the switch (step A) is set to correct voltage.



D. ☐ Peel "A" and "B" adhesive disk drive labels from the backing paper and stick them next to the disk drives A and B busy indicators as shown. The "C" and "D" labels are extra and can be used externally in the future.

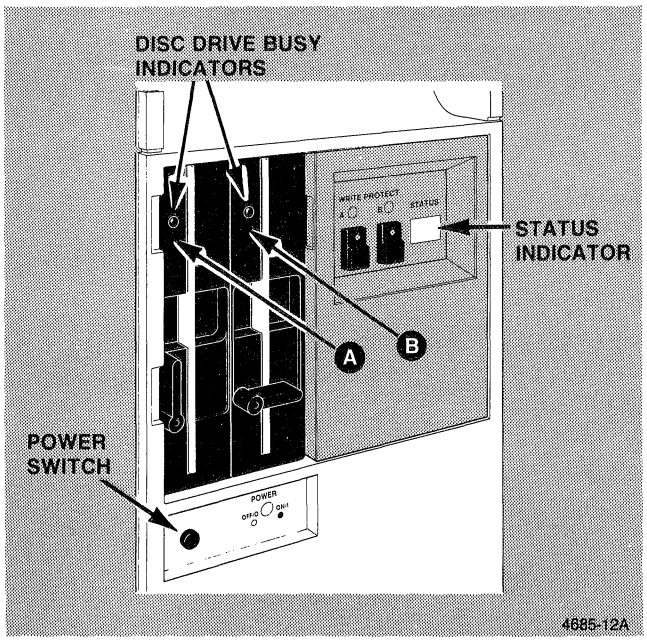
E. ☐ Apply power by pressing the POWER switch.

F. ☐ The 4170 runs an automatic internal test. Satisfactory completion is signaled by a single bell, the code 00 displayed on the status indicator, and both A and B disk drive busy lights are not lit.

NOTE

Any other code indicates the power-up test failed. Contact your local Tektronix service center, DO NOT GO ANY FURTHER.

G. ☐ You are done with Procedure 2. Check it off Table 4-1 and go to Procedure 3.



3. REMOVING THE FRONT COVER

CAUTION

There are no exposed hazardous voltages in the 4170. However, components may be damaged if boards are installed or removed while power is on. To avoid this potential hazard ALWAYS turn off power and disconnect the power cord before removing covers from the unit.

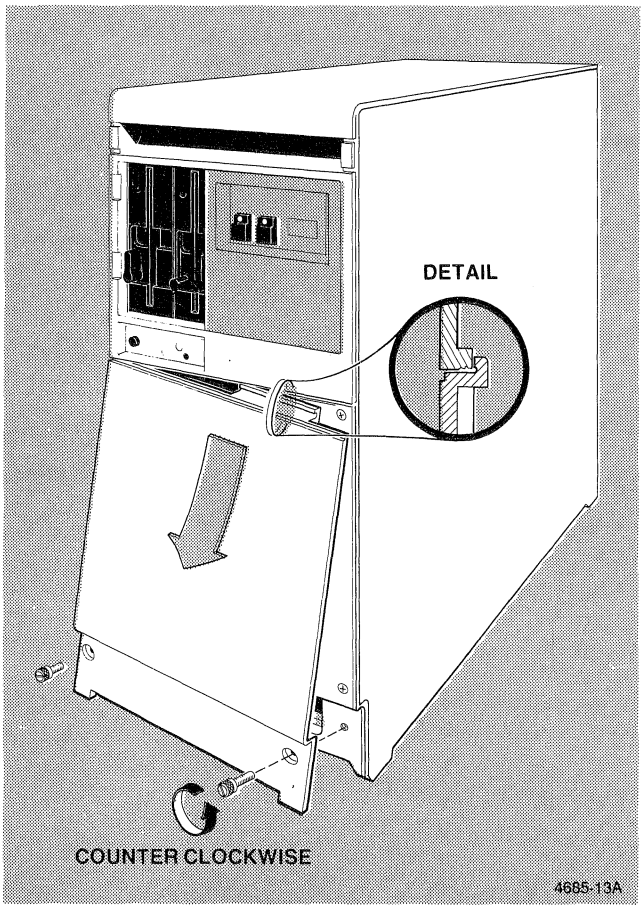
A. ☐ Turn off power.

B. ☐ Unplug the power cord.

C. ☐ Remove the two slotted screws from the bottom of the front cover. You should be able to remove these screws by hand. If not, use a small coin to loosen them.

D. ☐ Remove the front cover by pulling out at the bottom and sliding it down to free it from the top panel.

E. ☐ You are done with Procedure 3. Check it off Table 4-1 and find out what you need to do next.



4. REMOVING THE SIDE COVER

NOTE

When laying the unit on its side, it is advisable to place the plastic bag your 4170 was packed in under it to prevent scratching the paint.

A. ☐ Lay the 4170 down on its left side.

NOTE

The screwdriver is in the accessory carton with this manual, the ac power cord, and the interface cable.

B. ☐ Using the screwdriver that came with the 4170, remove the two large screws from the back of the unit. (These two large screws secure the side cover in place; they are not the smaller screws that secure the individual connector panels in place.)

C. ☐ Remove the three front screws (earlier units had only two front screws).

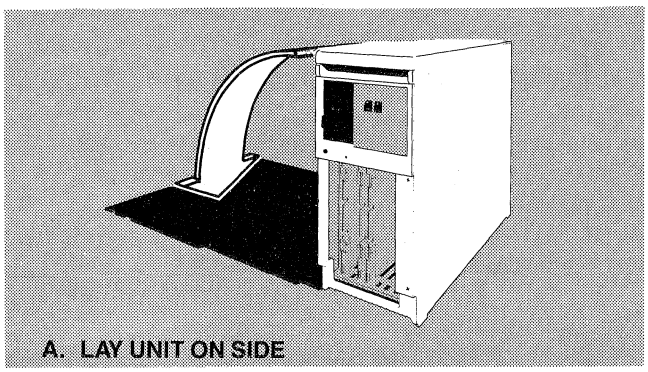
NOTE

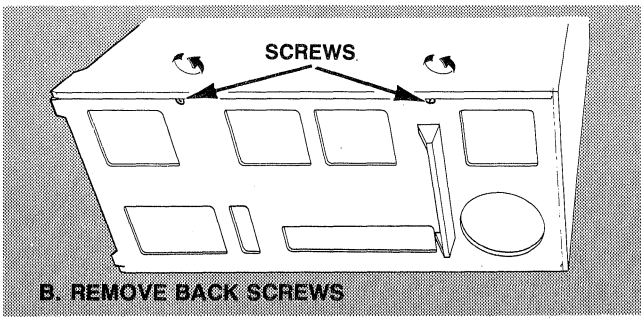
In the next step you may have to gently lift each corner of the cover to free it from the guides before removing the cover.

D. ☐ Grasp the cover at the upper corners as shown and lift it off.

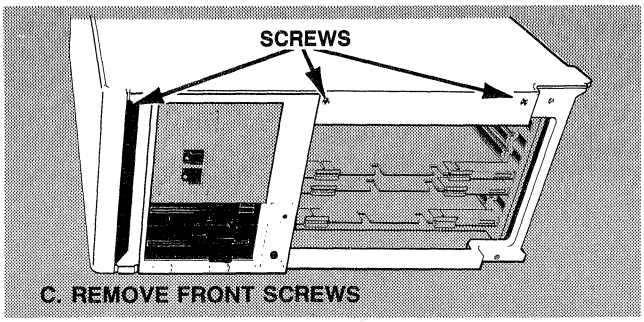
E. ☐ Set the cover aside.

F. ☐ You are done with Procedure 4. Check it off Table 4-1. Before you do the next procedure, review Figure 4-3, which follows, to learn the location of major components.

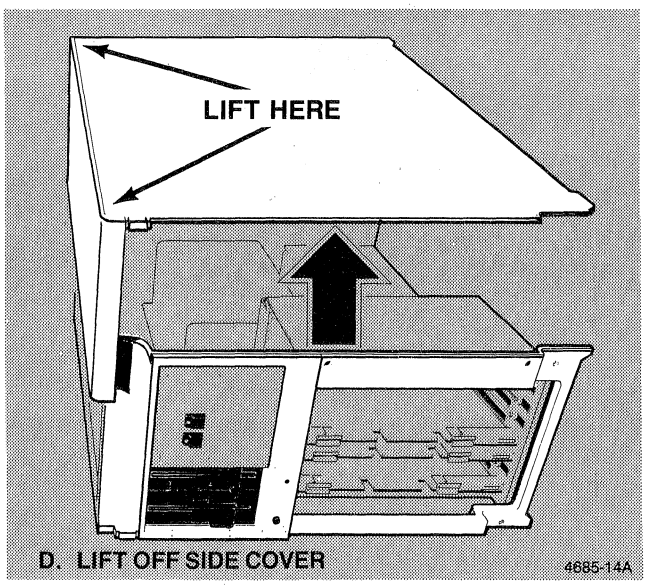

A. LAY UNIT ON SIDE



B. REMOVE BACK SCREWS



C. REMOVE FRONT SCREWS



D. LIFT OFF SIDE COVER 4685-14A

LOCATION OF MAJOR COMPONENTS

The major component locations are shown in Figure 4-3. You should familiarize yourself with these components since they are referred to in the following procedures. The name following the board slot name is the name found on the board installed in that slot.

CAUTION

When you are installing and removing boards from the unit, try to handle the boards only by the ejectors. Some of the boards have static sensitive components on them and there is a risk of damage. Exercise care when handling them.

NOTE

The card cage is held in place by the side cover. With the side cover removed the card cage is loose. This is normal.

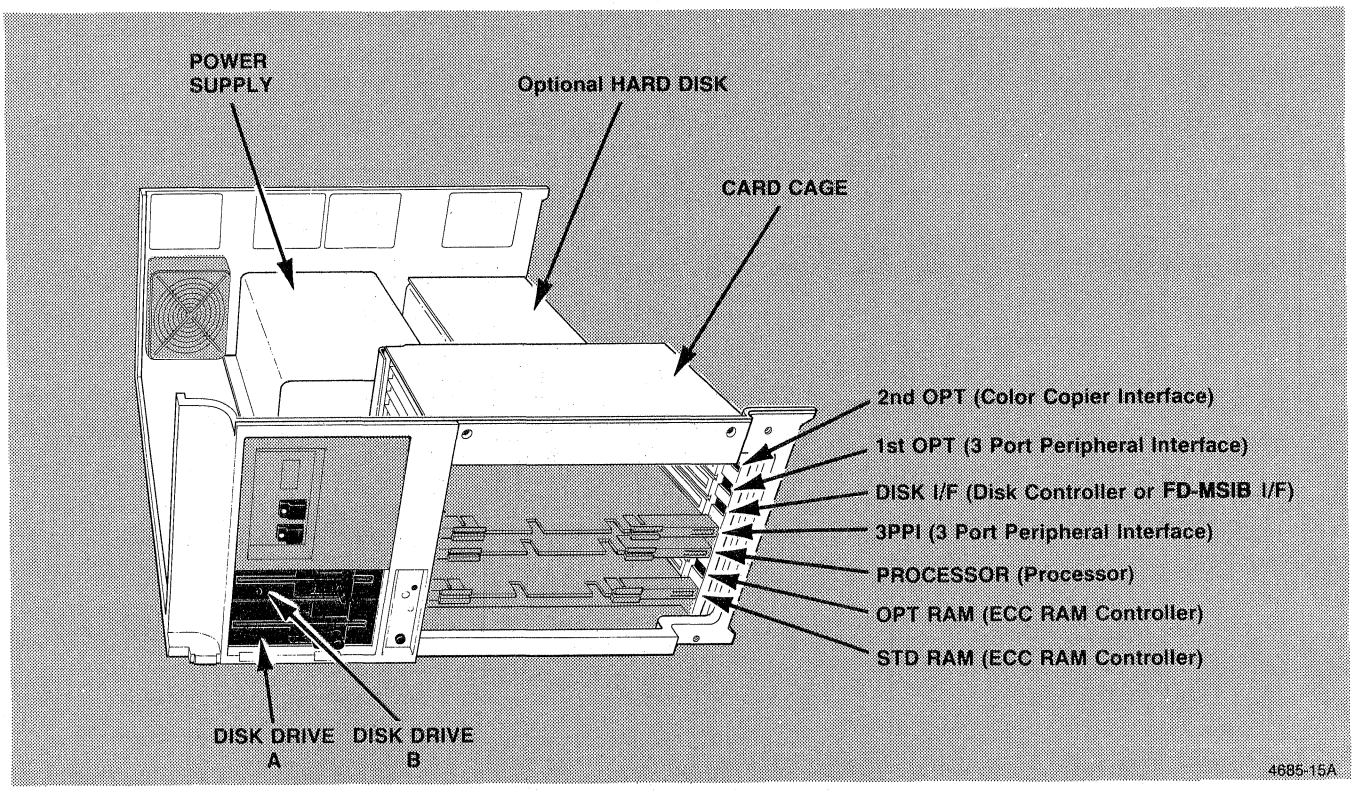


Figure 4-3. Major Component Locations.

5. INSTALLING MEMORY OPTIONS (4170F30, 31, 32, & 51)

BEFORE STARTING

In Steps A, B, and C (which follow), you will remove the Memory boards from the STD RAM and OPT RAM slots of the card cage, and from the carton containing the optional Memory boards. (In the procedures here, the boards are called "Memory" boards although the name "ECC RAM Controller" is printed on the boards.) After you remove the boards, the procedures instruct you to move some *clips* on each board; these *clips* are small rectangular blocks that connect two pins (see the illustration below). After you remove the boards, you will be able to see the clips clearly.

Figure 4-4 shows the clip positions as the boards are shipped from the factory. Follow the procedures to determine exactly what clips you must move on each board. (Some clips are preset and should not be moved.)

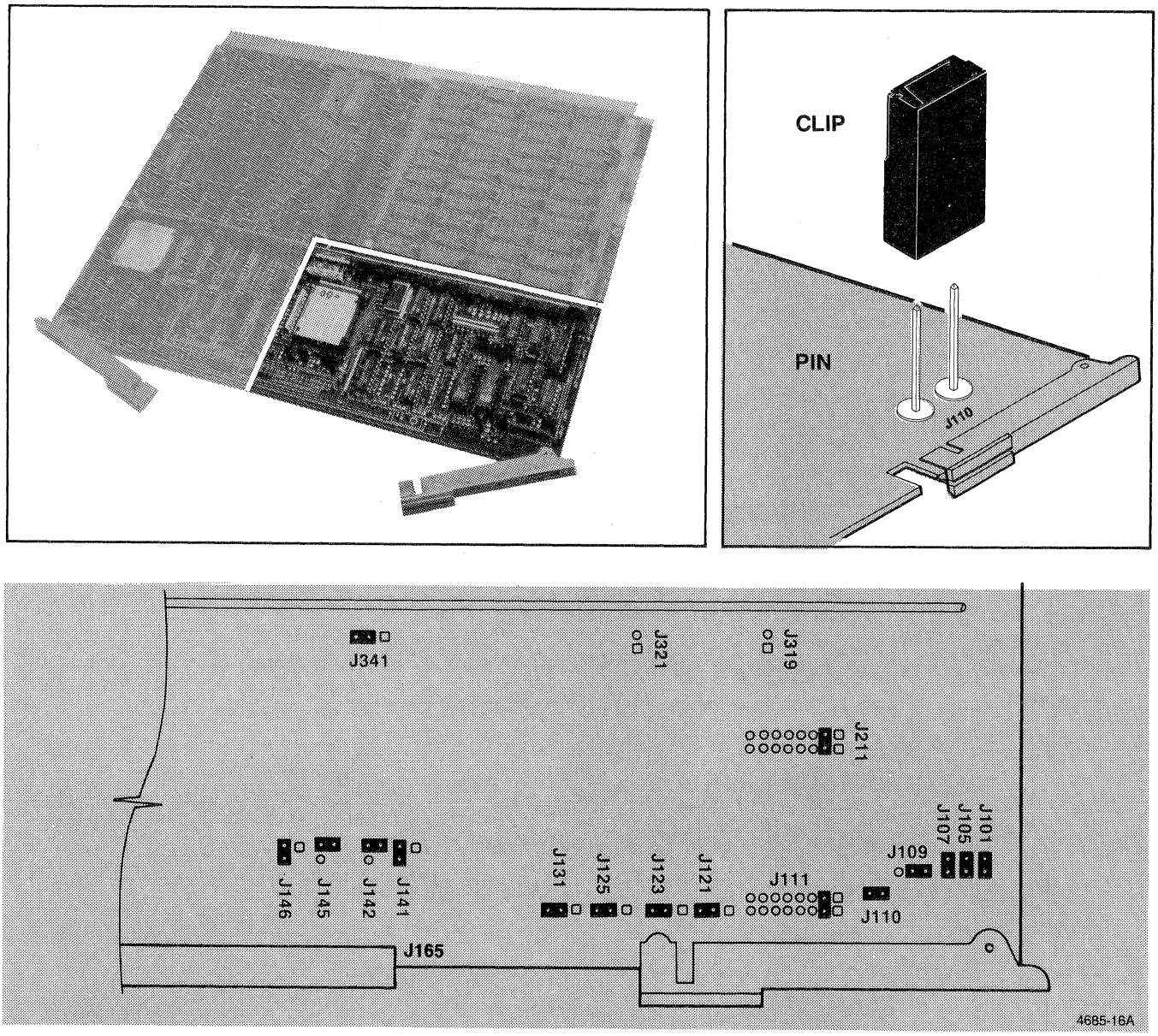


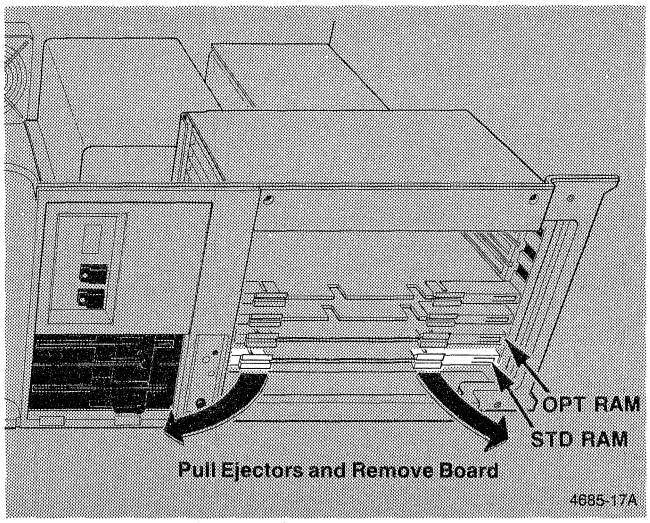
Figure 4-4. Memory Board Clip Locations (as shipped from factory).

A. ☐ Remove the Memory board from the STD RAM (bottom) slot in the card cage by pulling out on the board ejectors to release the board. Slide it out and set it aside for the moment.

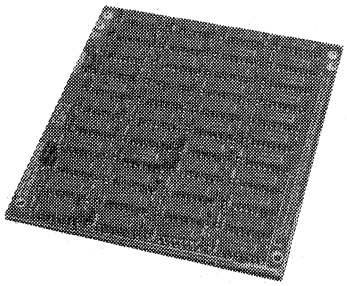
B. ☐ If there is a board in the OPT RAM slot of the card cage remove it as well.

NOTE

The Memory board(s) removed in Step A or Step B may not necessarily be returned to the slot it was removed from. Do not worry; just follow the instructions exactly for a successful installation.

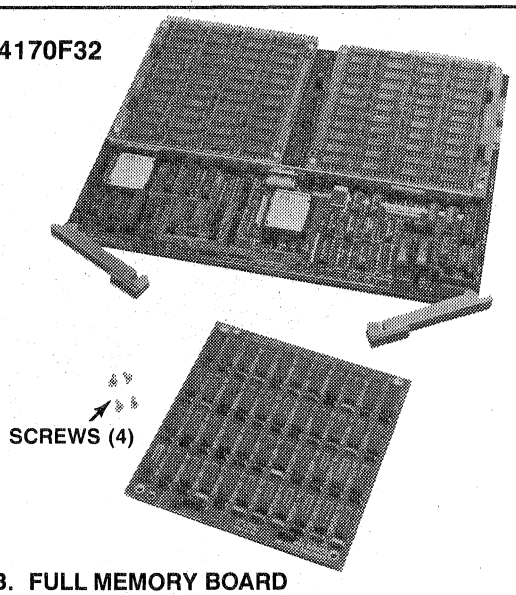


4170F30



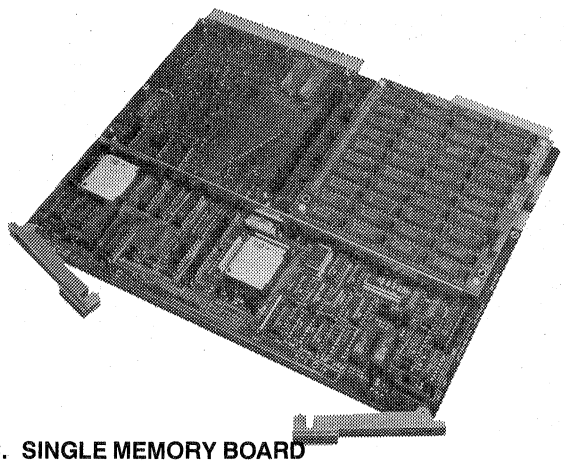
A. SINGLE RAM ARRAY BOARD

4170F32



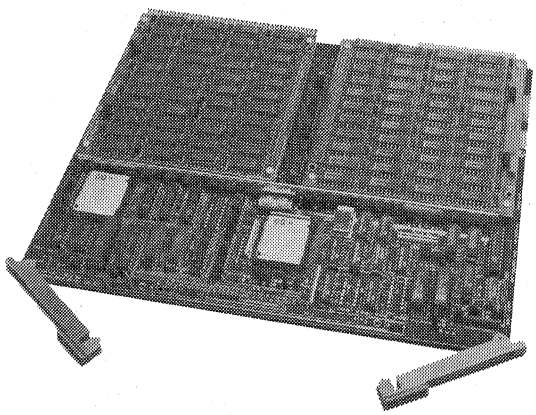
B. FULL MEMORY BOARD + RAM ARRAY BOARD

4170F51



C. SINGLE MEMORY BOARD

4170F31



D. FULL MEMORY BOARD

4685-18A

C. ☐ Open the carton containing the Memory F-kit (4170F30, F31, F32 or F51) and remove the parts from the shipping envelopes.

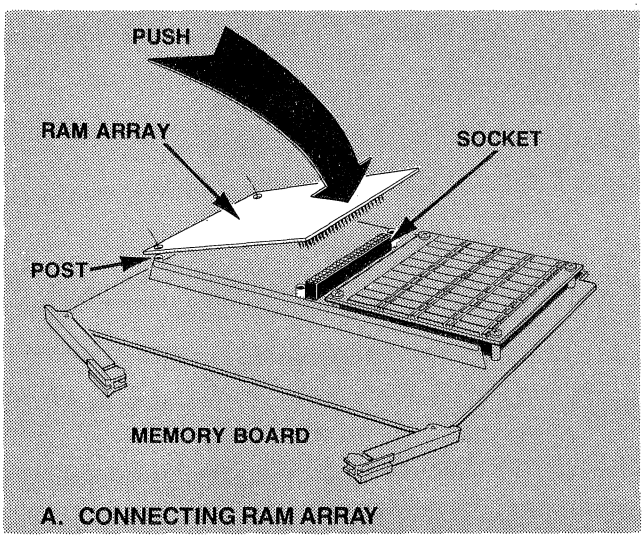
If your kit has a RAM Array board (4170F30 or 32) continue with step D. Otherwise skip to step G.

D. ☐ Install the small RAM Array board on the Memory board removed from the 4170 by carefully aligning the connector pins on the bottom of the RAM Array board with the socket on the top of the Memory board as shown.

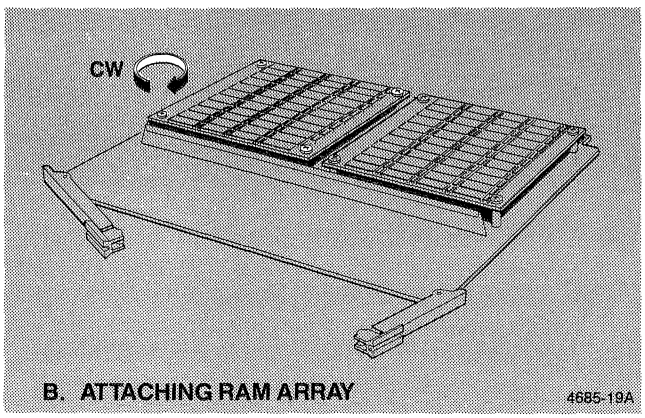
> HINT: Seat RAM Array board by pressing at the center of the pins. Be careful - pins are sharp.

E. ☐ Push the RAM Array board down until the pins are seated in the socket (the bottom side of the RAM Array board is flush with the top of the socket and the board rests on the four posts on the Memory board).

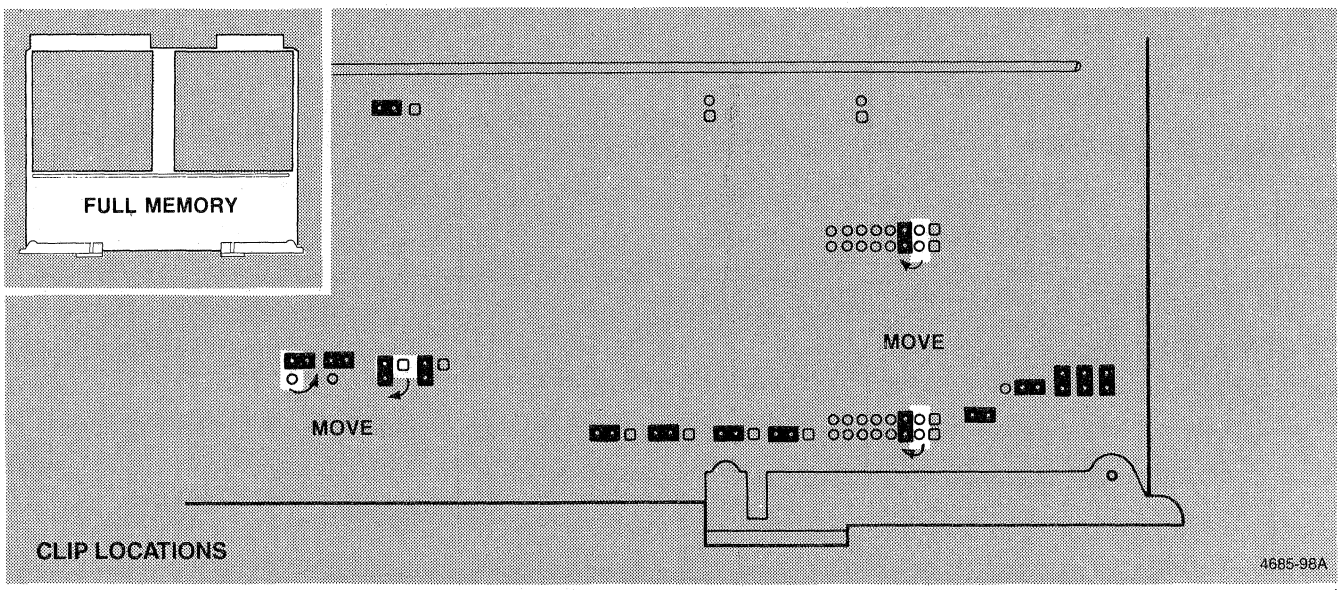
F. ☐ Use the four screws from the kit to secure the RAM Array board to the posts on the Memory board.



A. CONNECTING RAM ARRAY



B. ATTACHING RAM ARRAY 4685-19A



G. ☐ Pick up a full Memory board. (A full Memory board has two small RAM Array boards installed, as shown in the upper left corner of the illustration above.)

H. ☐ Move the four clips as indicated. (The illustration above uses small squares and circles to represent the pins on the board.) Because the clips fit tightly, to remove them you may need to wiggle them while pulling gently on them, or you may need to use your fingernail or a thin object to loosen them.

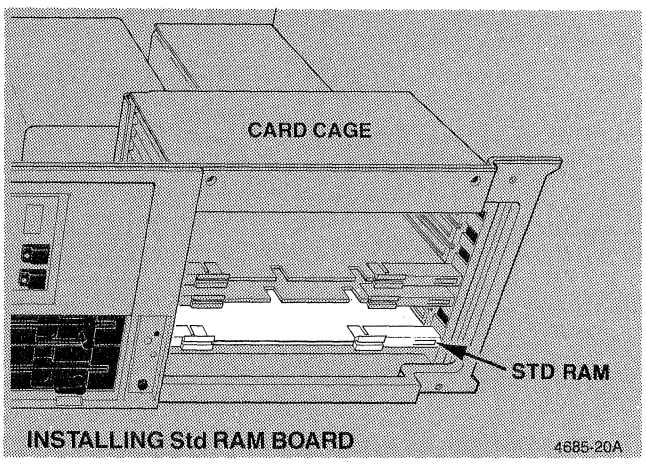
I. ☐ Check the other clips and make sure they are properly installed, if not move them to the proper pins.

NOTE

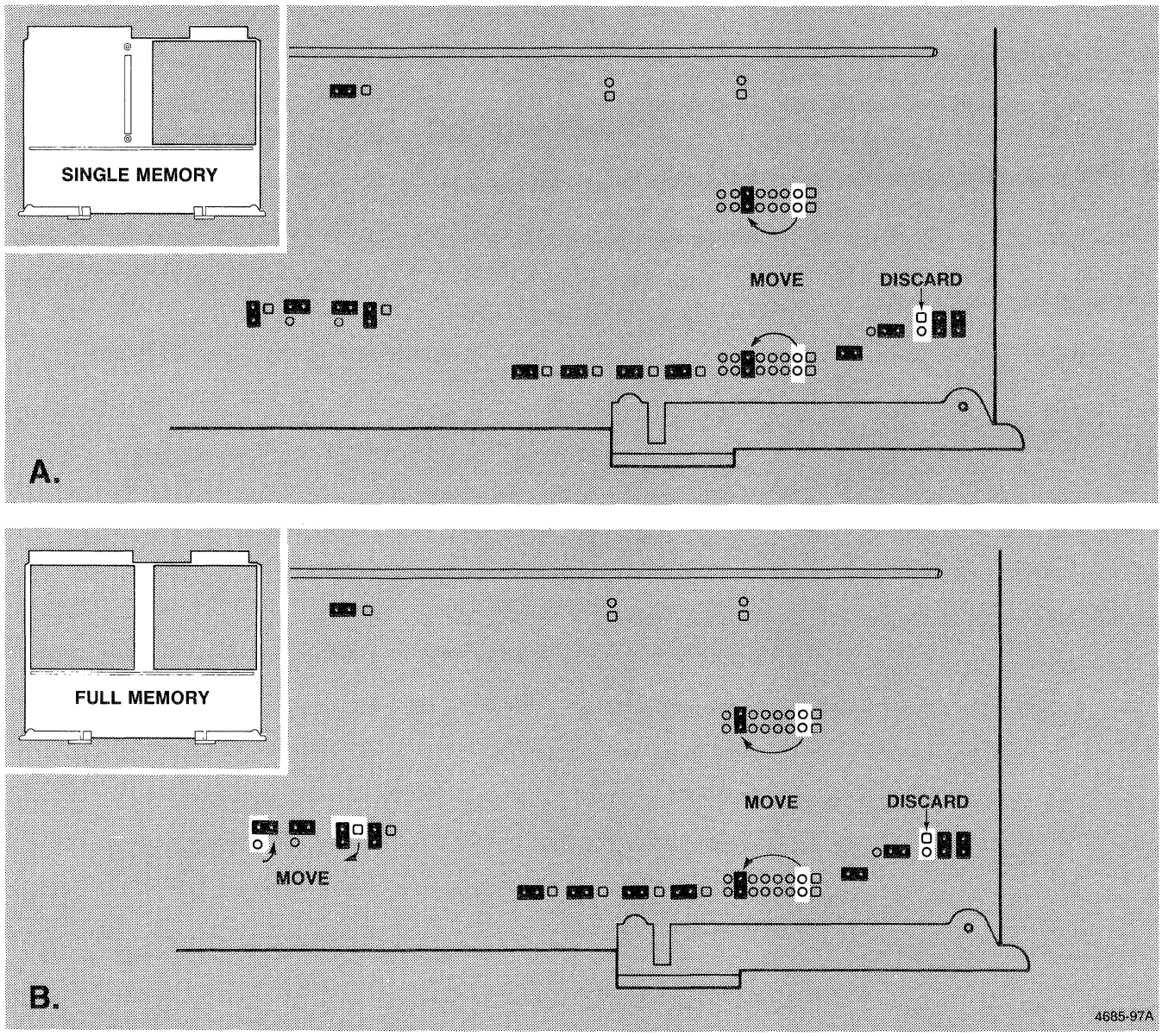
Remember when installing boards that the component side should be down, and the board should be handled by the ejectors to prevent possible damage to components.

J. ☐ Install the board in the STD RAM (bottom) slot of the card cage. Lock the board in place by pressing in the board ejectors until they lock over the edge of the board.

K. ☐ If you only have one Memory board you are done. Check Procedure 5 off Table 4-1 and go to the next procedure. Otherwise continue with step L.



L. ☐ If the remaining board looks like the board in A, set the clips as indicated under A. If it looks like B, use the settings under B.

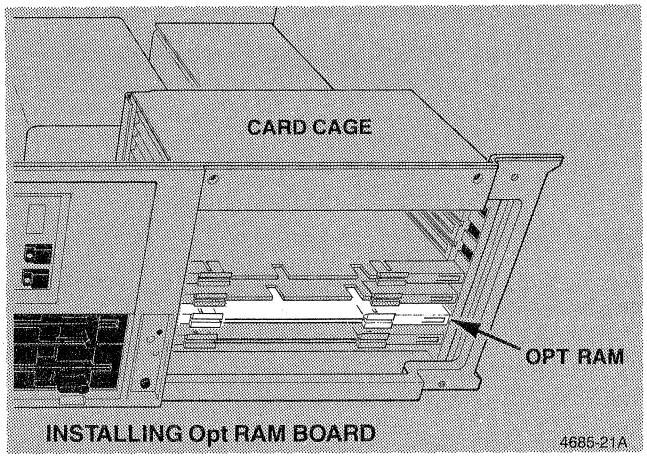


NOTE

Remember when installing boards that the component side should be down, and the board should be handled by the ejectors to prevent possible damage to components.

M. ☐ Install the board in the OPT RAM slot of the card cage. Lock the board in place by pressing in the board ejectors until they lock over the edge of the board.

N. ☐ You are done with Procedure 5. Check it off Table 4-1 and find out what you need to do next.



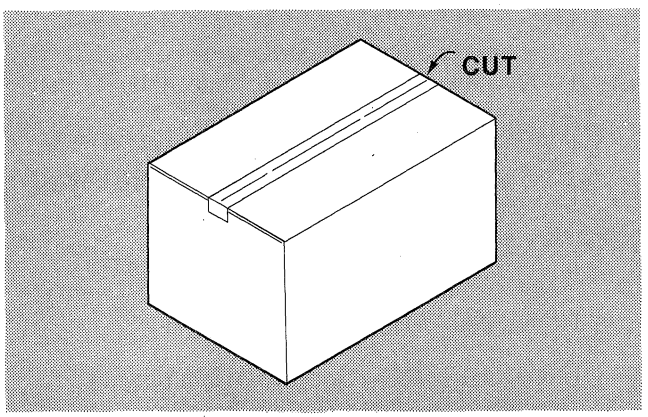
6. INSTALLING THE HARD DISK UNIT (4170F03)

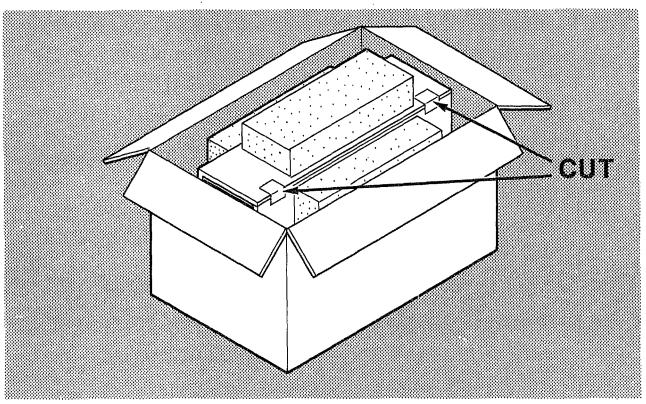
CAUTION

The Disk Unit can be damaged by dropping or by a blow. Handle it with care.

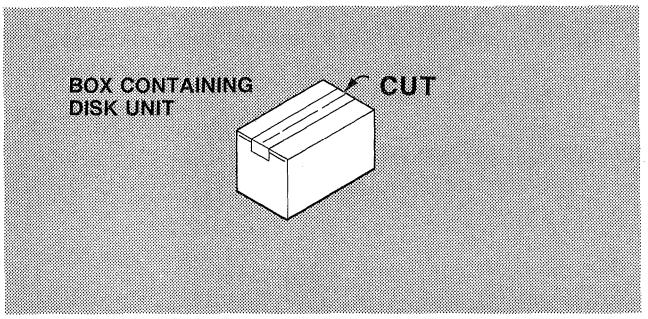
A. ☐ Open the carton containing the Hard Disk by cutting the tape on the top of the carton.

B. ☐ Cut the tape on the inside packing and lift top cover.



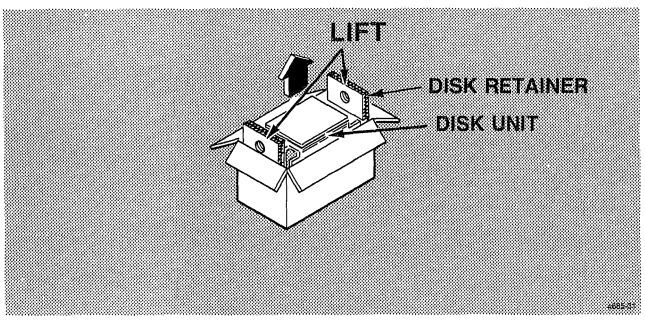


C. ☐ Remove box containing Disk Unit and cut tape to open box.



D. ☐ Insert a finger in the holes on each end of the DISK RETAINER and carefully lift the retainer and Disk Unit out of the box.

E. ☐ **Save all packing materials for use later, if you need to repackage the Hard Disk for shipment to another location.**



NOTE

Look for and save data sheets(s) shipped with the hard disk. This information will be used later.

F. ☐ Loosen (BUT DO NOT REMOVE) the four disk mounting stud screws (located on the bottom of the 4170 cabinet behind the card cage) about ¼ inch.

G. ☐ Look at the way the power supply is held in place. The hard disk will be held in place in the same manner.

H. ☐ Orient the hard disk as shown in the illustration. Note that the plug labeled P9 should be toward the card cage, **not** the back of the 4170.

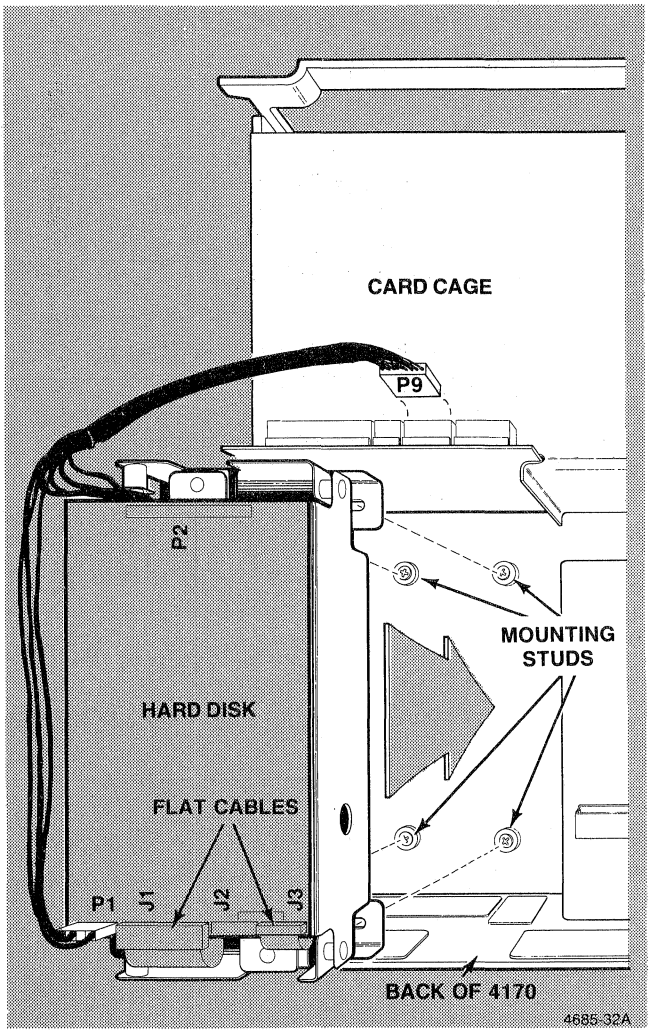
I. ☐ The hard disk may be awkward to position. It requires careful handling, since dropping it or bumping it sharply may result in damage. Hold the hard disk in one hand and move the cable bundle aside with your other hand. Slide the hard disk into position. The mounting studs go through the large holes in the disk mounting brackets.

J. ☐ Move the hard disk so the mounting studs are in the narrow slots in the hard disk mounting brackets.

K. ☐ Tighten the four mounting screws securely. If the screws are loose, the Disk Unit could vibrate loose during operation.

L. ☐ Connect P9 to the connector on the back edge of the card cage as shown. Plug P9 is keyed to aid in proper orientation.

M. ☐ You are done with Procedure 6. Check it off Table 4-1 and go to Procedure 7.

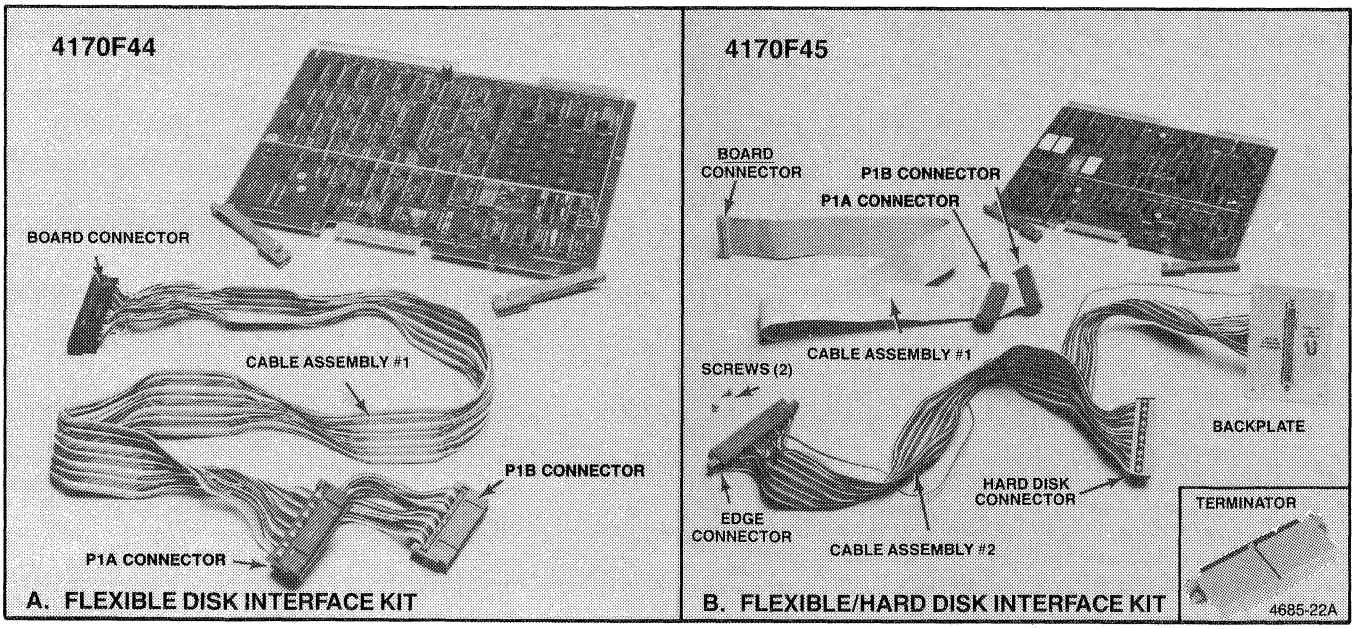


7. INSTALLING THE DISK INTERFACE (4170F44 & 45)

Two different Disk Interfaces are available for the 4170. Only one may be installed. One is used to control the two Flexible Disk drives that are standard in the unit. This disk interface is the Flexible Disk Interface (4170F44). If you have a hard disk (4170F03), this disk interface is the optional Flexible/ Hard Disk Interface kit (4170F45). This procedure installs either disk interface.

A. ☐ Open the carton containing the Disk Interface kit.

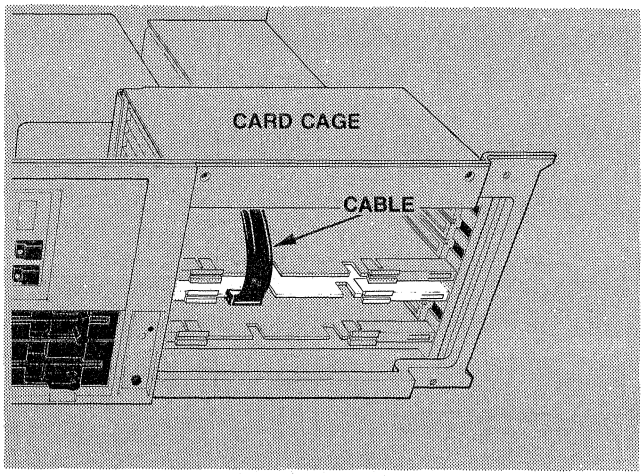
B. ☐ Remove all of the parts from their individual packaging and verify that the kit parts correspond to one of these pictures.



CAUTION

To avoid breaking wires when disconnecting cables, grip the connector by the plug and pull. DO NOT PULL THE CABLE OFF BY PULLING ON THE WIRES.

C. ☐ Disconnect the cable from the front edge of the Processor board.

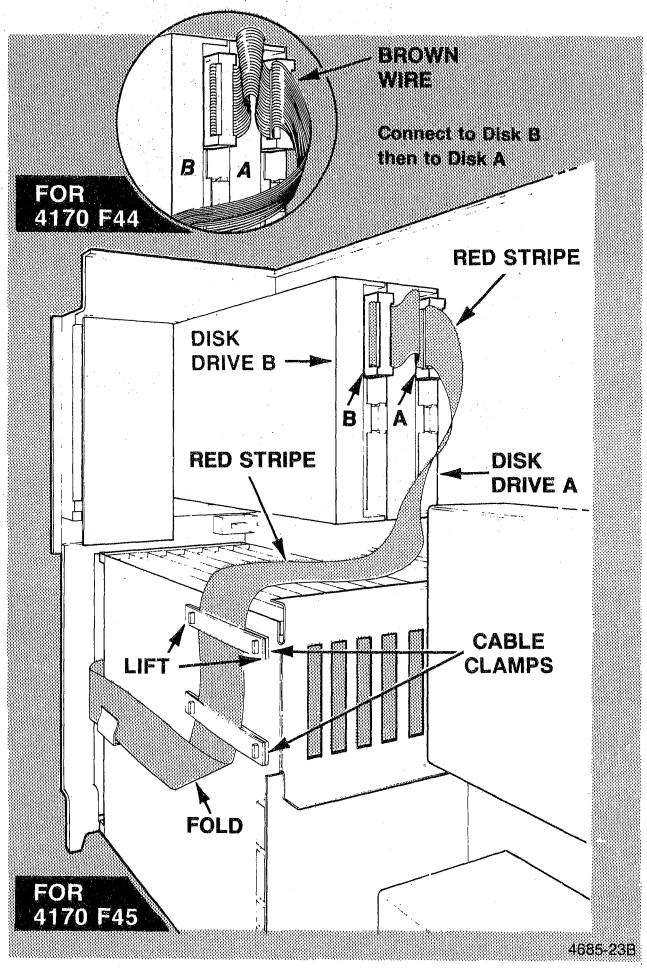


D. ☐ Take cable assembly 1 from the kit. You will note that there are two connectors near one end of the cable, marked P1A and P1B in the picture. Plug the P1B connector (on the end of the cable) into the socket on the back of DISK DRIVE B, orienting the locating stripe as shown. Now plug the other (P1A) connector into the socket on DISK DRIVE A. See inset figure for 4170F44.

> HINT: Compare the disk units to the picture to make sure you understand the cable connections and orientation.

E. ☐ Route the other end of the cable between the edge of the card cage and the cabinet frame as shown. It will be connected in Procedure 10.

> HINT: Remove the cable clamp by simultaneously lifting both ends of the upper portion of the clamp.



NOTE

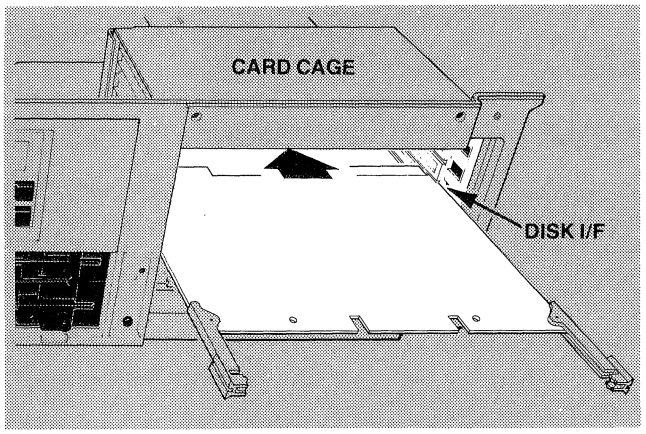
Remember when installing boards that the component side of the board must be down, and the board should be handled by the ejectors to prevent possible damage to the components.

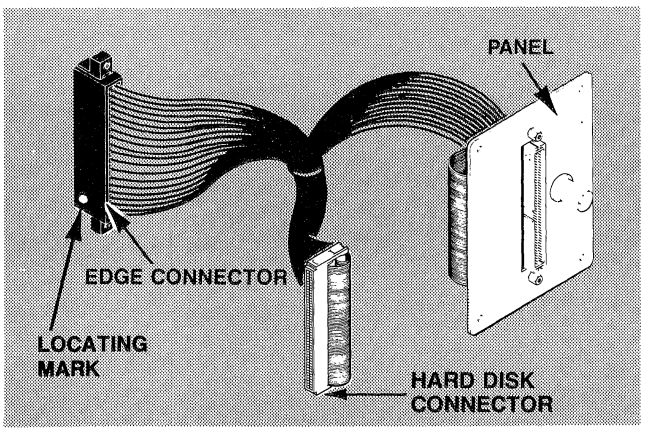
F. ☐ Slide the Disk Interface board into the DISK I/F slot in the card cage. Do not attempt to seat the board. This will be done later.

G. ☐ If you have installed all of the parts in your kit, skip to step P. Otherwise continue with step H.

> HINT: You may need to slide the Disk Interface board part way out of the card cage so it does not interfere with the edge connector.

H. ☐ Remove cable assembly 2 from the kit. Hold the edge connector so that you can see the locating mark as you look down on the edge connector.

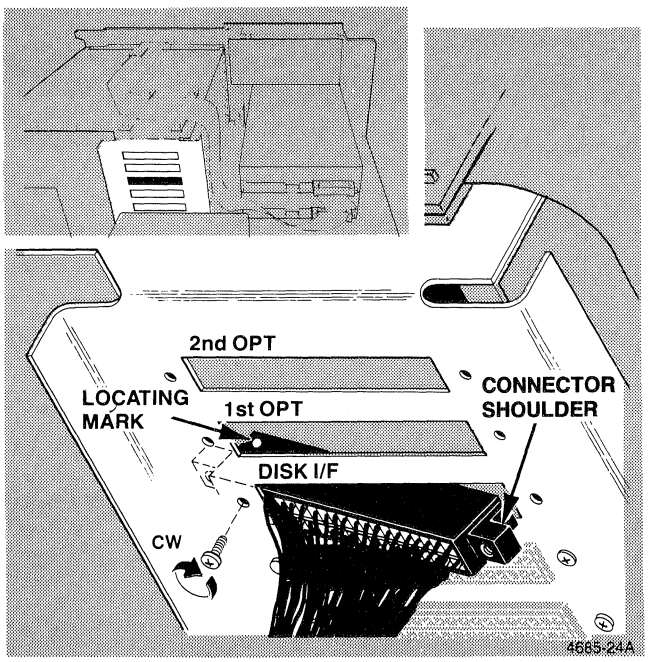




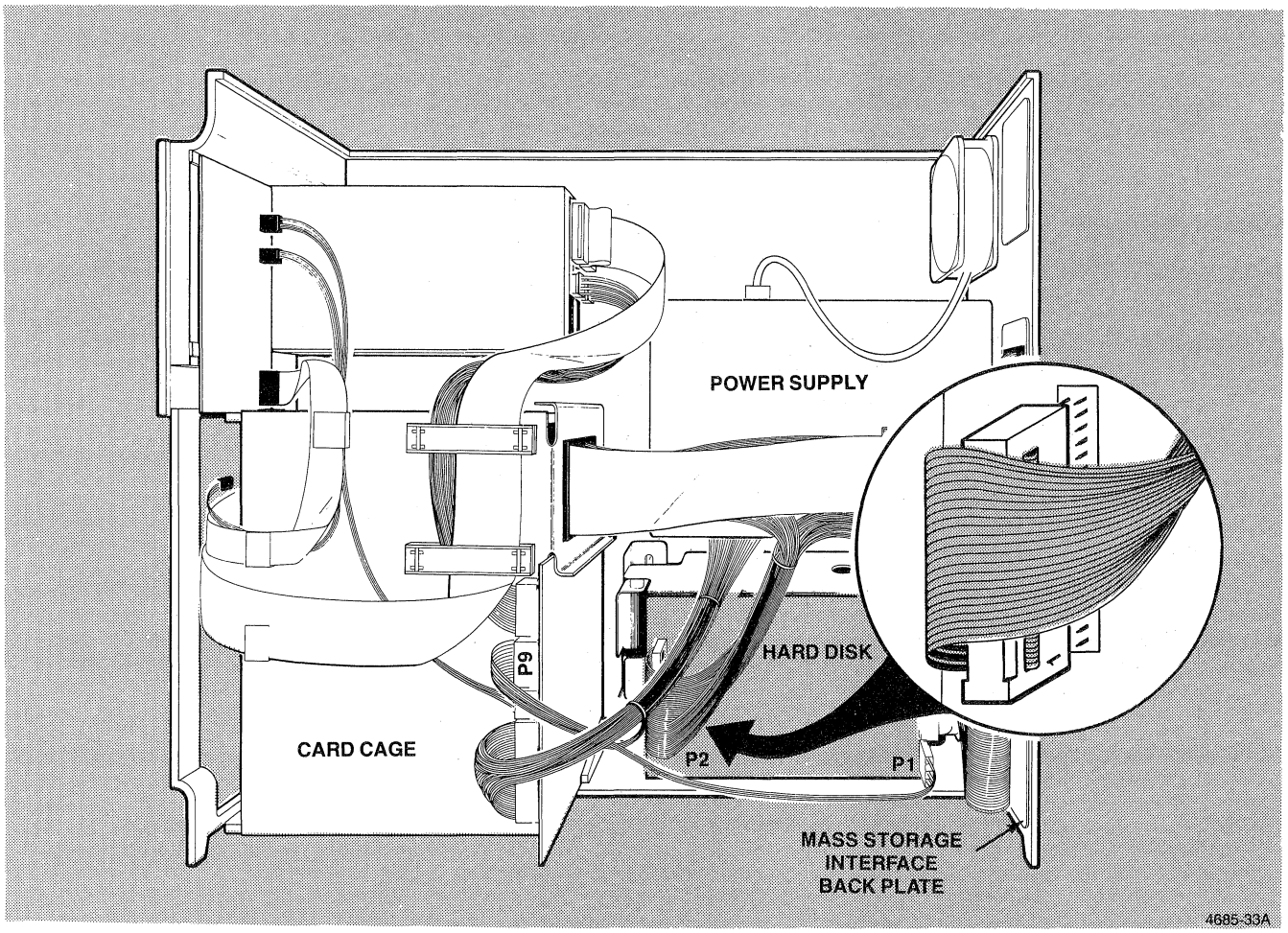
> HINT: Stand with the back of the 4170 (the part with the voltage selector switch on it) in front of you. Now look down and you will see the 4170 as shown in the small inset figure.

I. ☐ Slide one end of the edge connector through the DISK I/F opening as shown. Push the remainder of the connector through the opening then pull it back out far enough so the shoulders on the connector are against the inside of the card cage.

J. ☐ Secure the connector to the back of the card cage using the two screws from the kit.



K. ☐ Plug the Hard Disk connector to P2 on the Hard disk as shown. Make sure orientation is correct (Pin 1 is down and cable exits connector to the front of the 4170).



L. ☐ Keeping the screws for use, remove and discard Blank Panel 1 from the back of the 4170.

M. ☐ Position the panel (connected to the cable installed in H) as shown and secure it using the four screws removed in J. Tighten the screws.

> HINT: Do not connect the rest of the cables at this time. They will be connected later.

NOTE

Do not do steps N and O if you are going to connect Mass Storage Interface Bus (MSIB) compatible devices, such as TEKTRONIX 4926 10M Byte Hard Disk Drive, to the 4170.

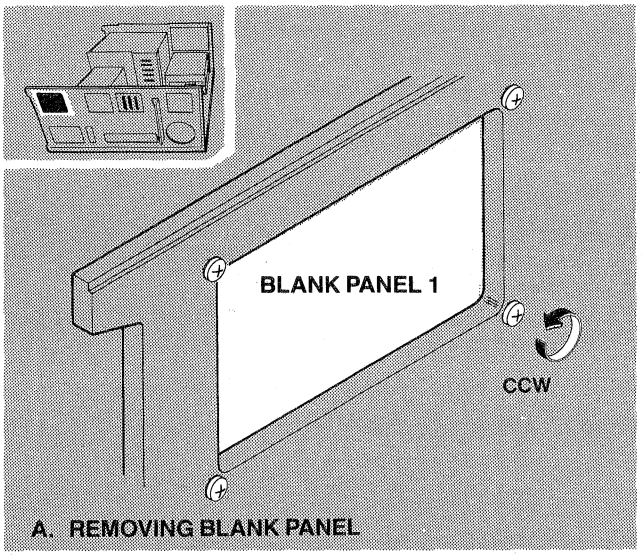
The following is a list of permitted external CP/M-86 disk drive names (C:, D:, E:, etc.) that correspond to user selected MSIB addresses. MSIB address 0 is assigned to internal 4170 hard disks C: and D: (D: is not presently used). Use this list when you are assigning MSIB addresses to the connected MSIB devices. MSIB address assigning is done at the external MSIB device. Refer to the device manual to find out how to set the MSIB address for the device.

CP/M-86 Drive Names	MISB Address to Assign
C: and D:	0
E: and F:	1
G: and H:	2
I: and J:	3
K: and L:	4
M: and N:	5
O: and P:	6
Reserved	7

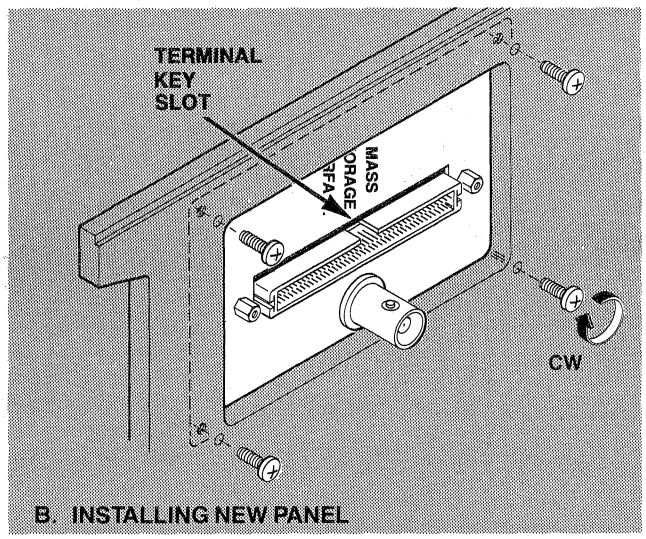
N. ☐ Remove the TERMINATOR from the kit and plug it into the socket on the panel as shown. TERMINATOR is keyed.

O. ☐ Lock the TERMINATOR in place by tightening the two holding screws on the TERMINATOR assembly.

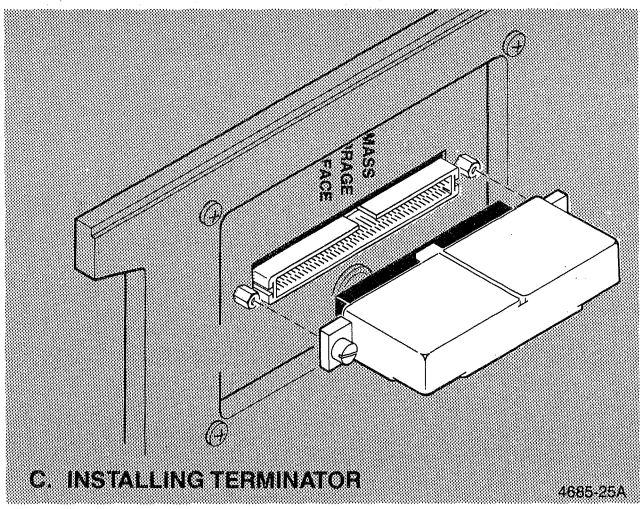
P. ☐ You are done with Procedure 7. Check it off Table 4-1 and find out what you need to do next.



A. REMOVING BLANK PANEL



B. INSTALLING NEW PANEL


C. INSTALLING TERMINATOR

4685-25A

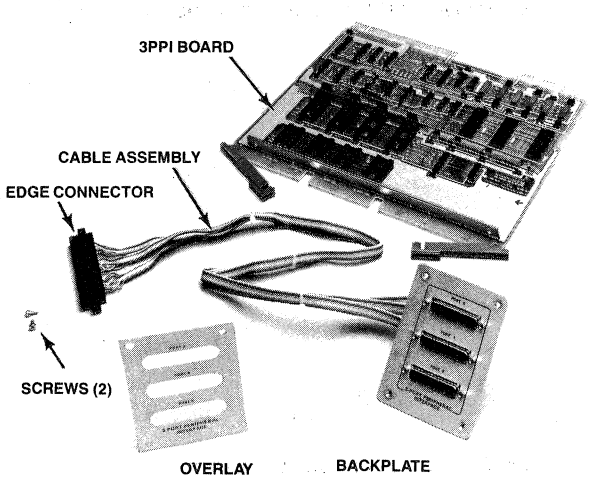
8. INSTALLING THE OPTIONAL PERIPHERAL INTERFACE (4170F10)

A. ☐ Open the carton containing the Peripheral Interface F-kit.

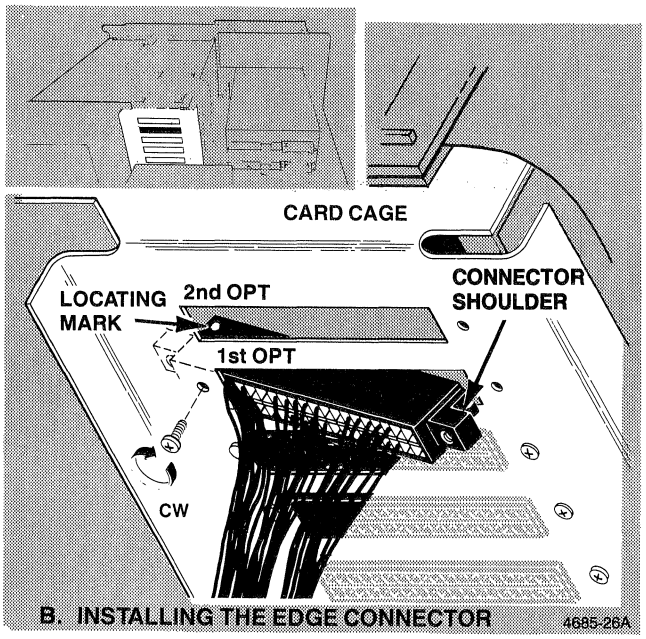
B. ☐ Remove all of the parts from their individual packaging and verify that the kit contains the parts shown in the picture.

> HINT: Stand with the back of the 4170 (the part with the voltage selector switch on it) in front of you. Now look down and you will see the 4170 as shown in the small inset figure.

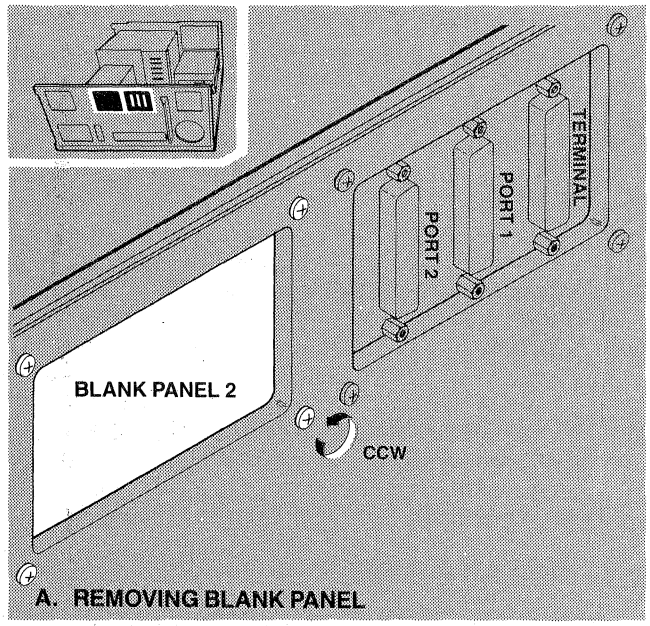
C. ☐ Hold the edge connector with the locating mark so you can see the locating mark as you look down on the edge connector. Slide the end of the edge connector through the 1st OPT opening as shown. Push the rest of the connector through the opening then pull it back out far enough so the shoulders of the connector are against the inside of the card cage. Secure with two screws from kit.



A. OPTIONAL PERIPHERAL INTERFACE KIT



D. ☐ Keeping the screws for later use, remove and discard Blank Panel 2 from the back of the 4170.

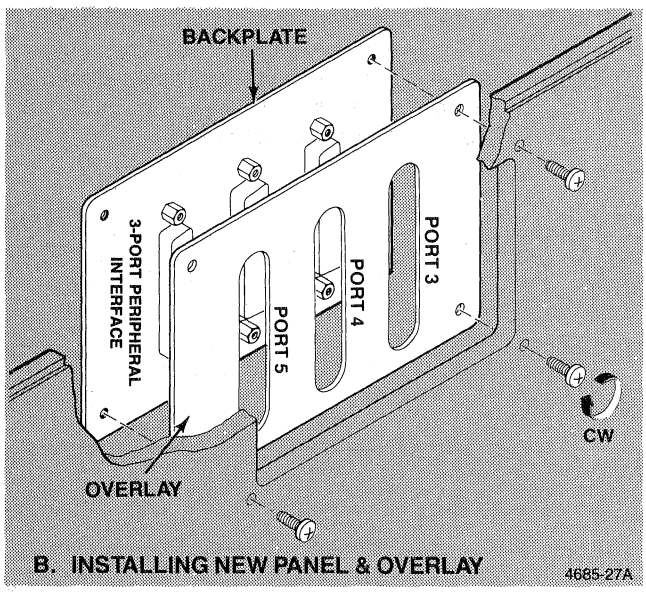


NOTE

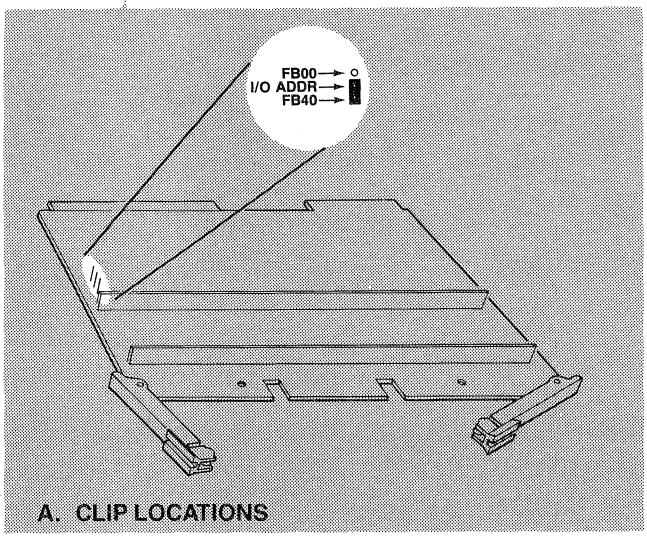
There is printing on both sides of overlay. Ensure that correct label (PORT 3, PORT 4, and PORT 5) faces out.

E. ☐ Place the overlay on the new backplate as shown.

F. ☐ Align the new backplate (with the overlay) behind the opening in place of Blank Panel 2. Secure the backplate with the screws removed in step D.



G. ☐ Verify that the I/O ADDR clip is in position FB40 as shown.

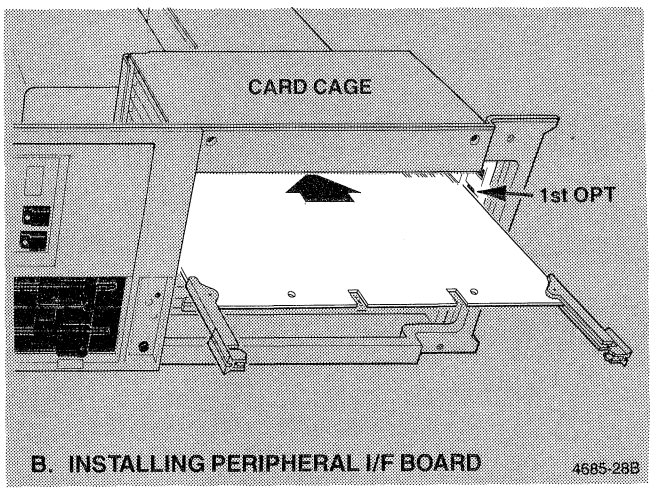


A. CLIP LOCATIONS

NOTE

Remember when installing boards the component side of the board must be down and the board should be handled by the ejectors to prevent possible damage to the components.

H. ☐ Slide the Peripheral Interface board into the 1st OPT slot in the card cage. Lock the board in place by pressing in on the board ejectors until they lock over the edge of the board.



B. INSTALLING PERIPHERAL I/F BOARD 4685-28B

> HINT: If the board does not slide in and lock easily you may have installed the edge connector (step C) upside down.

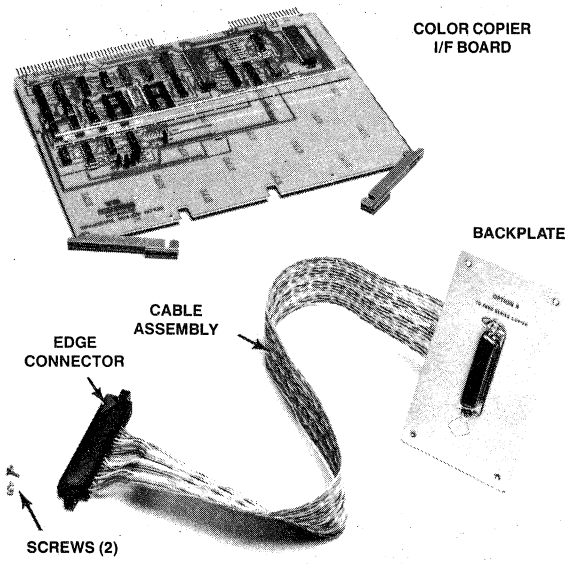
I. ☐ You are done with Procedure 8. Check it off Table 4-1 and find out what you need to do next.

9. INSTALLING THE PARALLEL INTERFACE (4170F09)

A. ☐ Open the carton containing the Parallel Interface F-kit.

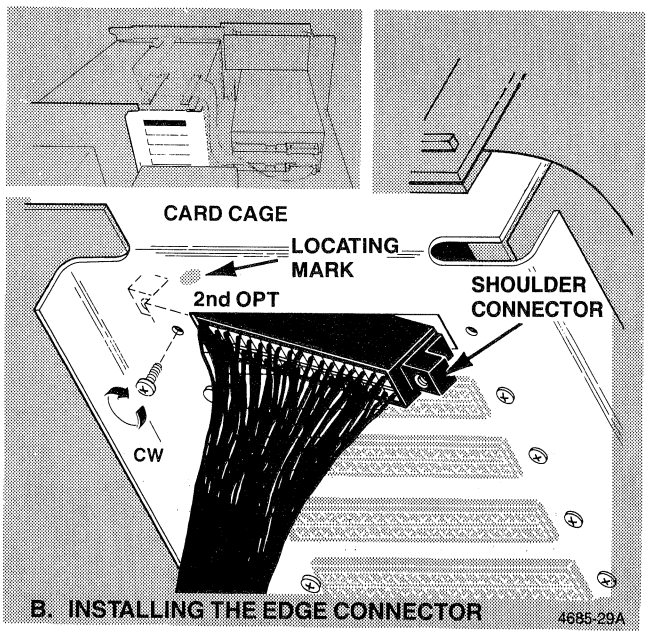
B. ☐ Remove all of the parts from their individual packaging and verify that the kit contains the parts shown in the picture.

> HINT: Stand with the back of the 4170 (the part with the voltage selector switch on it) in front of you. Now look down and you will see the 4170 as shown in the small inset figure.



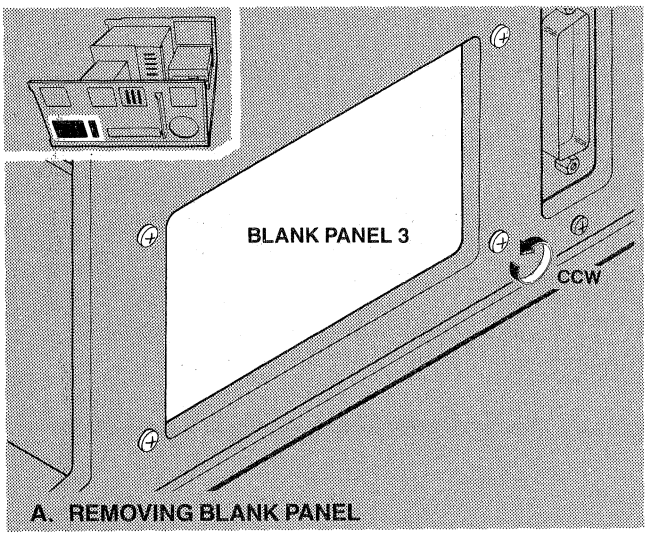
A. PARALLEL INTERFACE KIT

C. ☐ Hold cable assembly edge connector so you can see the locating mark as you look down on the edge connector. Slide end of it through the 2nd OPT opening in the back of the card cage as shown. Push the rest of the connector through the opening, then pull it back far enough so the shoulders of the connector are against the inside of the card cage. Secure the connector using the screws provided in the F-kit.



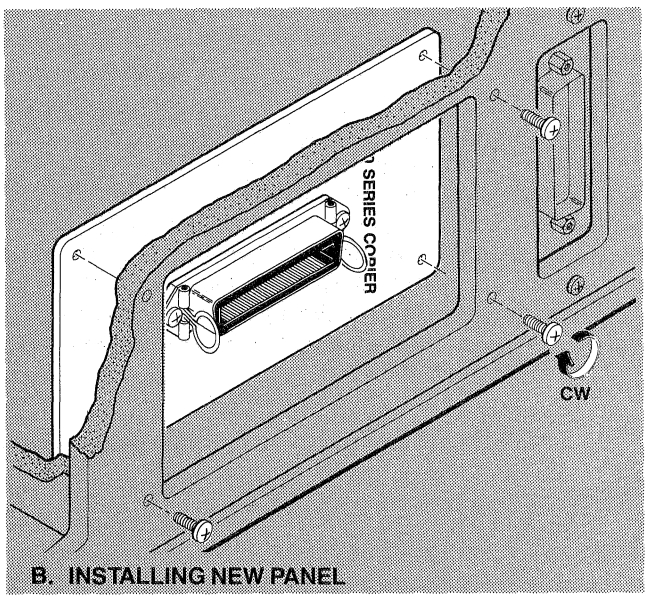
B. INSTALLING THE EDGE CONNECTOR

D. ☐ Keeping the screws for later use, remove and discard Blank Panel 2 from the back of the 4170.



A. REMOVING BLANK PANEL

E. ☐ Align the new backplate behind the opening as shown. Secure the backplate with the screws removed in D.

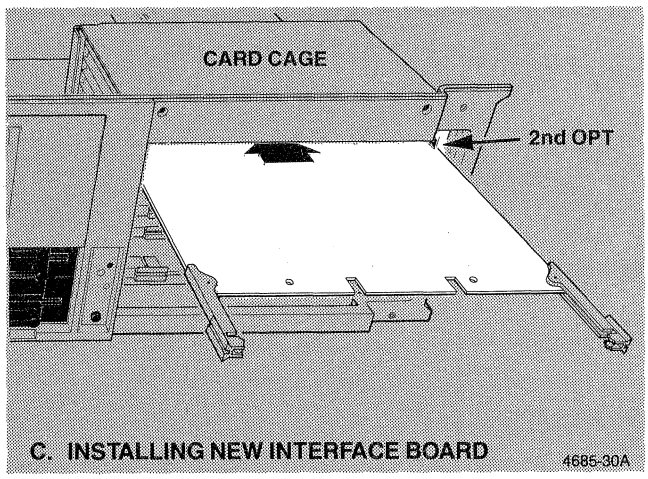


B. INSTALLING NEW PANEL

NOTE

Remember when installing boards the component side of the board must be down and the board should be handled by the ejectors to prevent possible damage to the components.

F. ☐ Slide the Interface board into the 2nd OPT slot in the card cage. Lock the board into place by pressing in the board ejectors until they lock over the edge of the board.



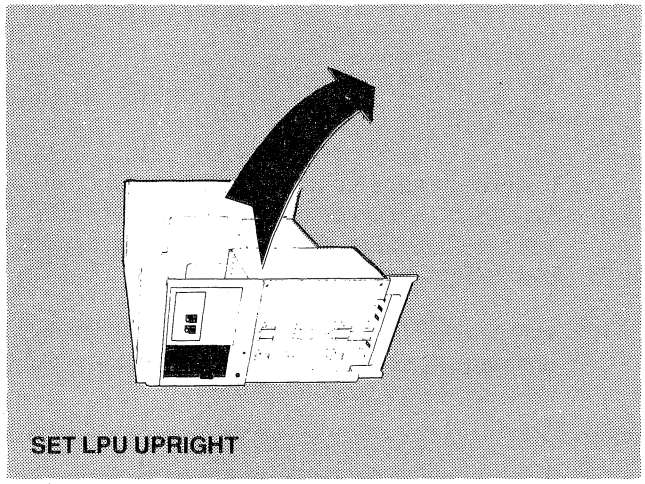
C. INSTALLING NEW INTERFACE BOARD 4685-30A

G. ☐ You are done with Procedure 9. Check it off Table 4-1 and find out what you need to do next.

10. REPLACING THE SIDE COVER

A. ☐ Check to make sure that all internal screws and connections you have made are tight.

B. ☐ Set the 4170 back upright.



NOTE

Look on the side of the card cage for the three-wire cable.

C. ☐ Connect the three-wire cable to J206 (on the side of the DISK I/F card) as shown.

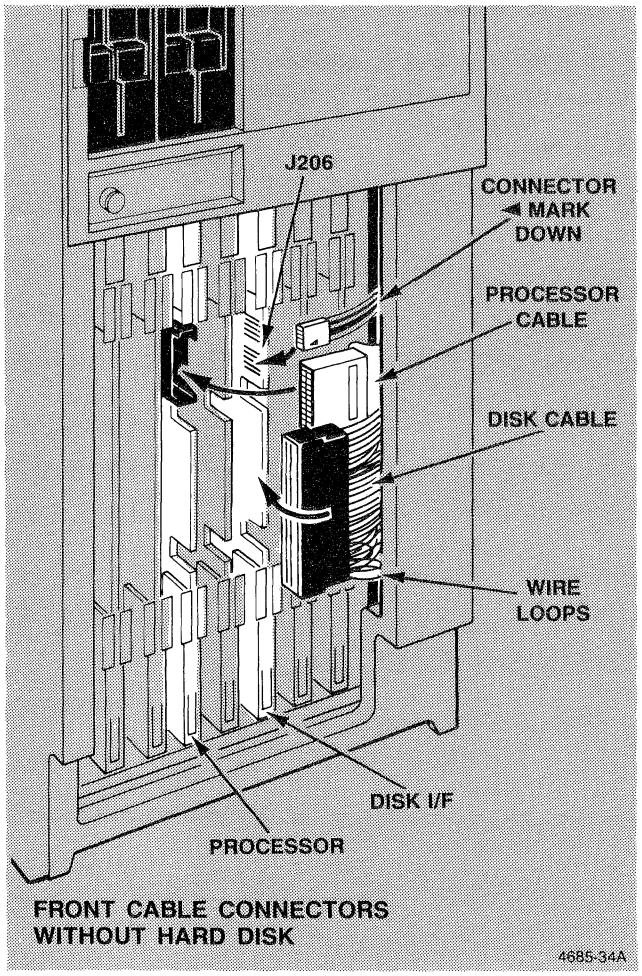
D. ☐ If you have the hard disk interface (4170F45) in your 4170, skip to Step I.

E. ☐ Connect the large disk cable to the connector on the edge of the Disk I/F board, making sure the four white-wire loops are down.

F. ☐ Lock the DISK I/F board in place by sliding it in and pressing in the board ejectors until they lock over the edge of the board.

G. ☐ Connect the processor cable to the connector on the edge of the Processor board as shown. The red stripe on the cable should be down.

H. ☐ Skip to step L.



NOTE

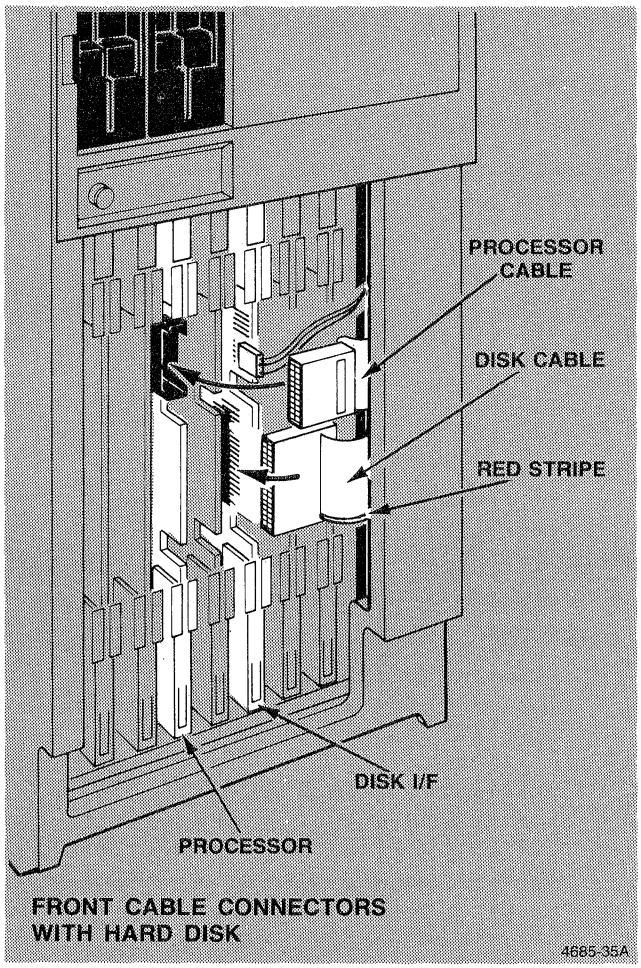
You may find it easier to remove the DISK I/F board to attach the disk cable to it.

I. ☐ Connect the large flat cable to the connector under the edge of the Disk I/F card as shown. Make sure that the red stripe on the cable is down and that the pins and connector are properly aligned.

J. ☐ Lock the Disk I/F board in place by sliding it in and pressing in the board ejectors until they lock over the edge of the board.

K. ☐ Connect the processor cable to the connector along the edge of the Processor board as shown. The red stripe on the cable should be down.

L. ☐ Route cables through cable clamps as shown in Figure 4-5 if you do not have a Hard Disk. Route cables as shown in Figure 4-6 if you have a Hard Disk.



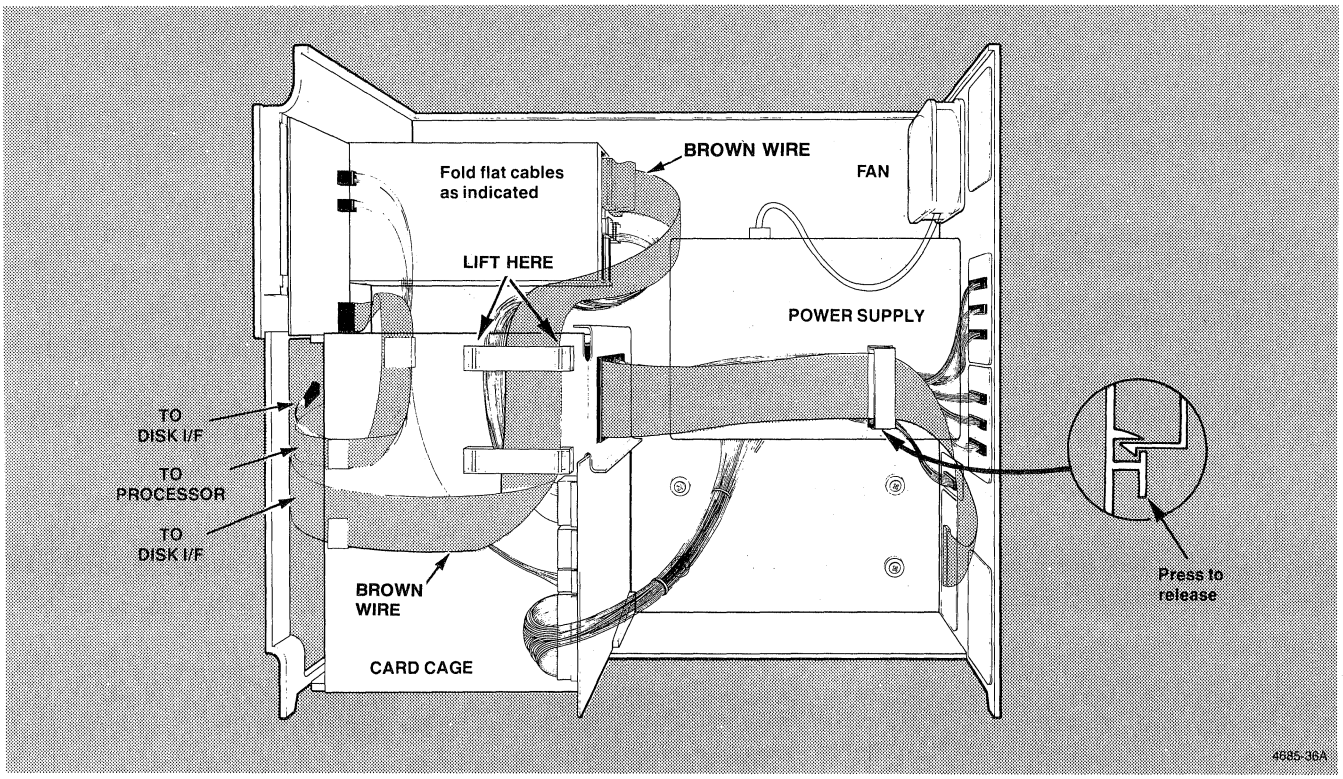


Figure 4-5. Cable Routing (Without Hard Disk).

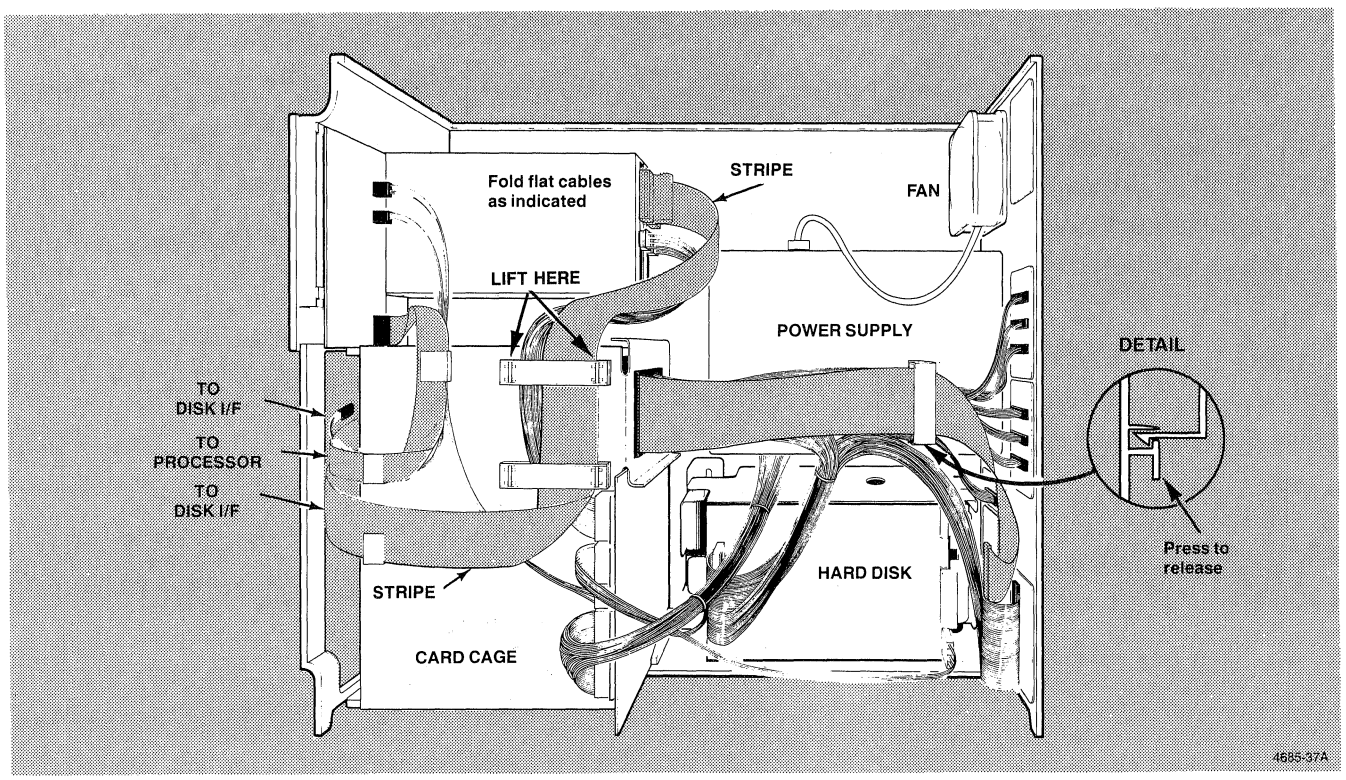


Figure 4-6. Cable Routing (With Hard Disk).

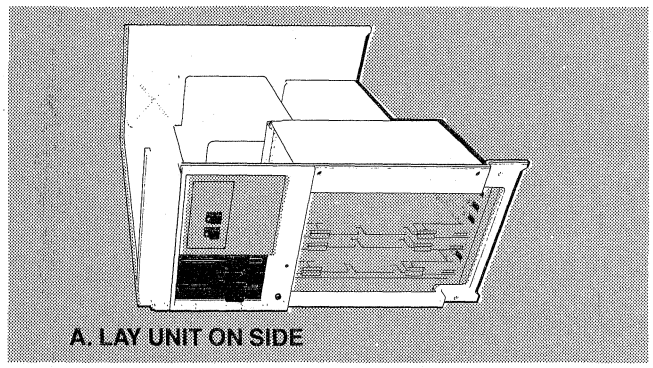
M. ☐ Lay 4170 down on its left side.

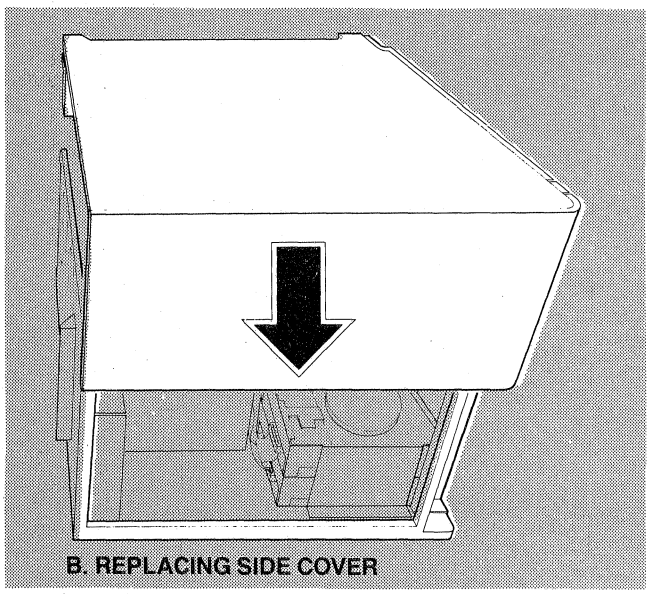
N. ☐ Position the card cage so that it is not resting on the two guides located between the disk drives and the card cage. The guides are part of the cabinet frame.

O. ☐ Carefully slide the side cover on. Pay particular attention to aligning the guides on the inside top of the side cover with the top of the cabinet frame. Be sure the cabinet frame screen clears the side cover.

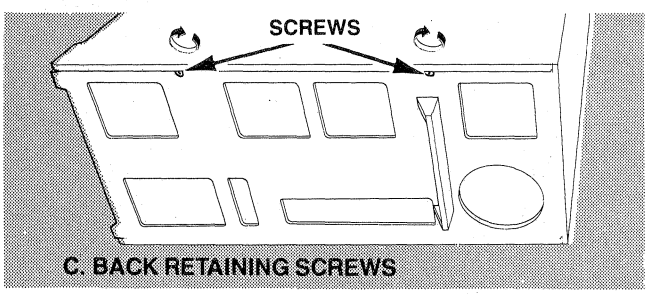
> HINT: The card cage is held in place by the side cover. You may need to slightly reposition it so the side cover will slide into place.

P. ☐ Check around the sides of the side cover to make sure it fits tight on the frame.


A. LAY UNIT ON SIDE

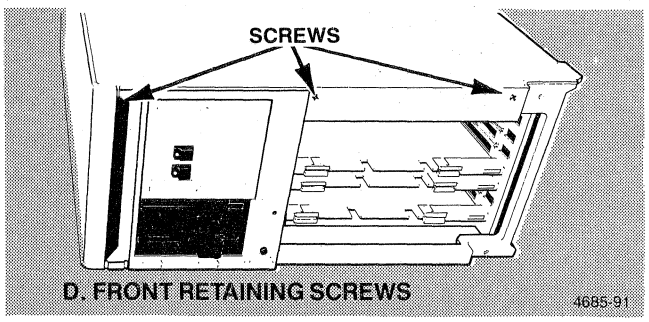

B. REPLACING SIDE COVER

Q. ☐ Replace the two screws on the back of the unit as shown.



C. BACK RETAINING SCREWS

R. ☐ Replace the three screws on the front of the unit as shown.



D. FRONT RETAINING SCREWS

4685-91

S. ☐ You are done with Procedure 10. Check it off Table 4-1 and go to Procedure 11.

11. REPLACING THE FRONT COVER

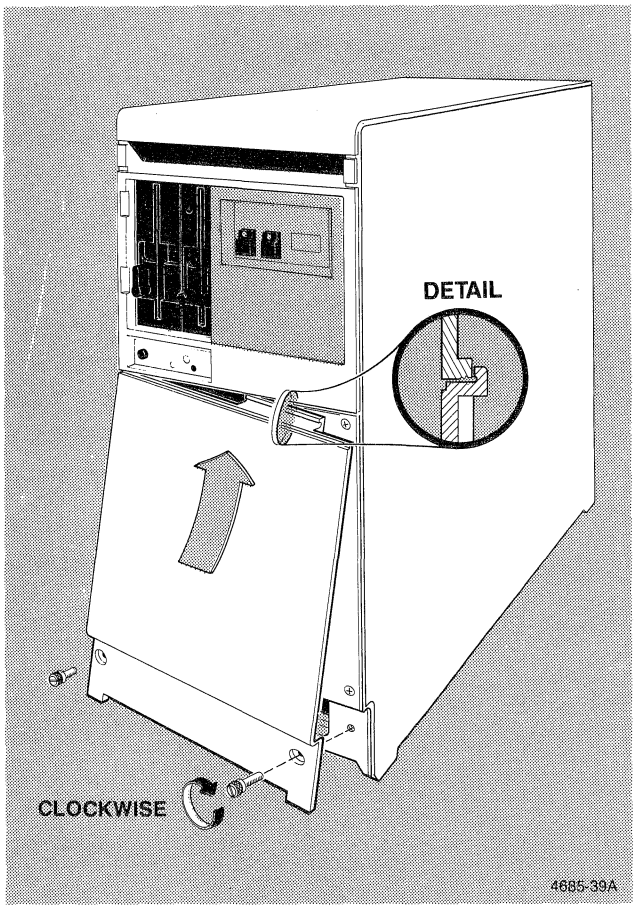
A. ☐ Set unit upright.

> HINT: Look at the circuit board installed in either the 3PPI or PROCESSOR slot to see a locked ejector. Note how ejector contacts the card cage.

B. ☐ Verify that all the boards are properly installed and the board ejectors are locked.

C. ☐ Slide the front cover into place as shown. Make sure the upper right edge of the cover slips behind the back of the control panel as shown.

D. ☐ Replace the two knurled screws as shown to secure the front cover.



E. ☐ You are done with Procedure 11. Check it off Table 4-1 and find out what you need to do next.

12. CONNECTING A TERMINAL

CAUTION

When connecting cables make sure they are out of the way and do not present a hazard to people walking by.

A. ☐ Connect one end of the terminal interface cable to the TERMINAL connector on the back of the 4170 as shown.

B. ☐ Connect the other end of the cable to the connector on the back of the terminal. On a Tektronix 4105, 4107, or 4109 terminal, the connector is labeled COMPUTER.

C. ☐ Connect the power cord from the terminal to the AC OUT connector on the back of the 4170 as shown.

> HINT: Some terminals are designed to only be connected to a wall outlet. They cannot be connected to the 4170. In this case, connect the terminal to facility power and turn it on. You can, however, connect the Tektronix 4105, 4107, and 4109 terminals directly to the 4170 using the interface cable supplied.

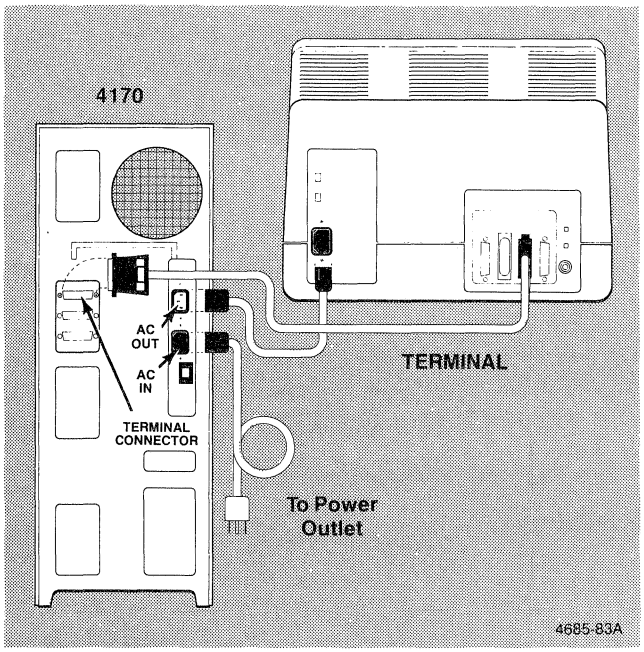
D. ☐ Connect the 4170 power cord to the AC IN connector on the back of the 4170 as shown.

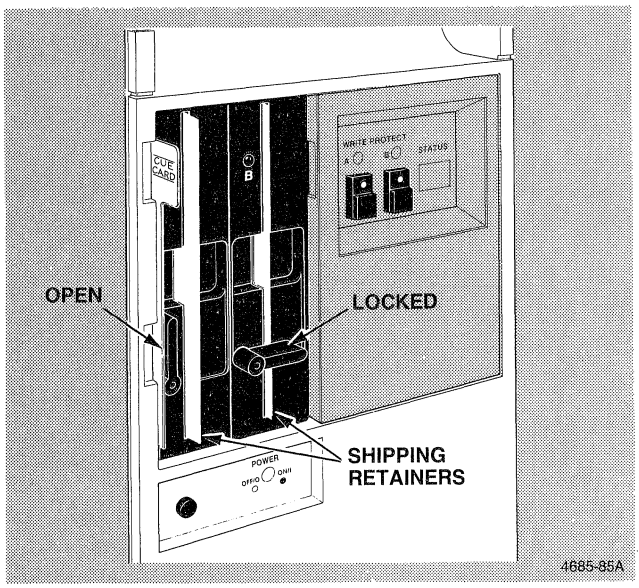
E. ☐ Connect the other end of the 4170 power cord to the nearest power outlet and turn on power to the terminal.

F. ☐ Turn disk drive A and B diskette retainers to open position as shown. Pull out shipping retainers. Save retainers with the rest of the packing materials.

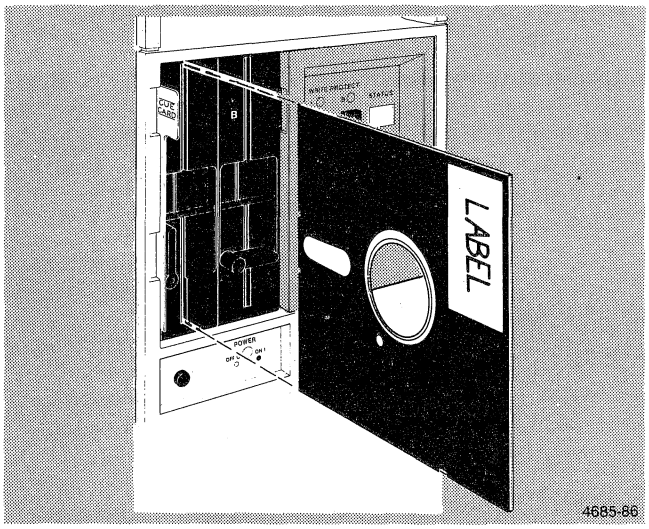
G. ☐ Insert Cue Card in slot to left of Drive A.

H. ☐ You are done with Procedure 12. Check it off Table 4-1 and go to Procedure 13 to verify operation.





13. VERIFYING OPERATION



A. ☐ Press the power switch to apply power to the 4170.

NOTE

Failure of the following steps indicates the internal diagnostics of the 4170 detected an error. Verify that you completed the installation procedures correctly. If the problem is still present, contact your local Tektronix Service Center for assistance.

B. ☐ Verify the bell rings once, 00 status is displayed, and the red disk drive A busy indicator is on.

C. ☐ Save the test connector that came in the accessory carton (with the screwdriver, the power cords, and the terminal interface cable). This connector is used during self-test procedures that may be performed at a later time.

D. ☐ You have completed the installation and checkout procedures and your 4170 is ready for use. Go to the Getting Started section in this manual.

NOTE

Remove and open remaining box in 4170 shipping box. This box contains your diskettes and additional manuals.

REPACKAGING INSTRUCTIONS

If you need to repackage the 4170 for shipment to another site you should use the original materials that were used to ship the 4170 to you. If these materials are no longer available contact your local Tektronix representative to obtain the materials you need. Once you have the materials, proceed as follows:

1. Turn off power to the 4170 and disconnect all external cables and power cords.

2. If your 4170 has a Hard Disk installed you will need to remove it. The Hard Disk removal and packing is described in steps 3 through 6. If you do not have a Hard Disk go to step 7.

3. Remove the front and side covers from the 4170. (Refer to Installation Procedures 3 and 4 for details.)

NOTE

In step 4 the letters in parentheses refer to the corresponding steps of 9, INSTALLING THE HARD DISK UNIT.

4. (L) Unplug the cable from P9 on the card cage.
   (K) Unplug the cable from P2 on top of the Hard Disk unit.
   (F) Loosen the four disk mounting screws.
   (H) Remove the disk by sliding it toward the power supply module (to clear the mounting screws) and lifting it out of the mounting cabinet. Tighten the four mounting screws.
   (D) Set the disk in the Disk Retainer and place it in the box.
   (C) Seal the box with tape and place the box in the large shipping carton (B).
   (A) Seal the shipping carton with tape.

5. Replace the side cover (see steps M, N, O, and P of Installation Procedure 10).

6. Replace the front cover (see 11, *REPLACING THE FRONT COVER*).

7. Install the shipping retainers in the disk drives (Installation Procedure 12, step G).

8. Place the 4170 in the shipping cartons as shown in Figure 4-7.

9. Place the diskettes and software manuals in the large box, seal the box with tape and place it in the bottom of the shipping carton.

10. Place this manual, the power cable, POZIDRIVE® screwdriver, and other accessories in the small accessory box and place the box in the shipping carton.

11. Seal the shipping carton with reinforced tape, apply the necessary shipping labels to the 4170 and Hard Disk cartons.

The 4170 is now ready to ship to another site.

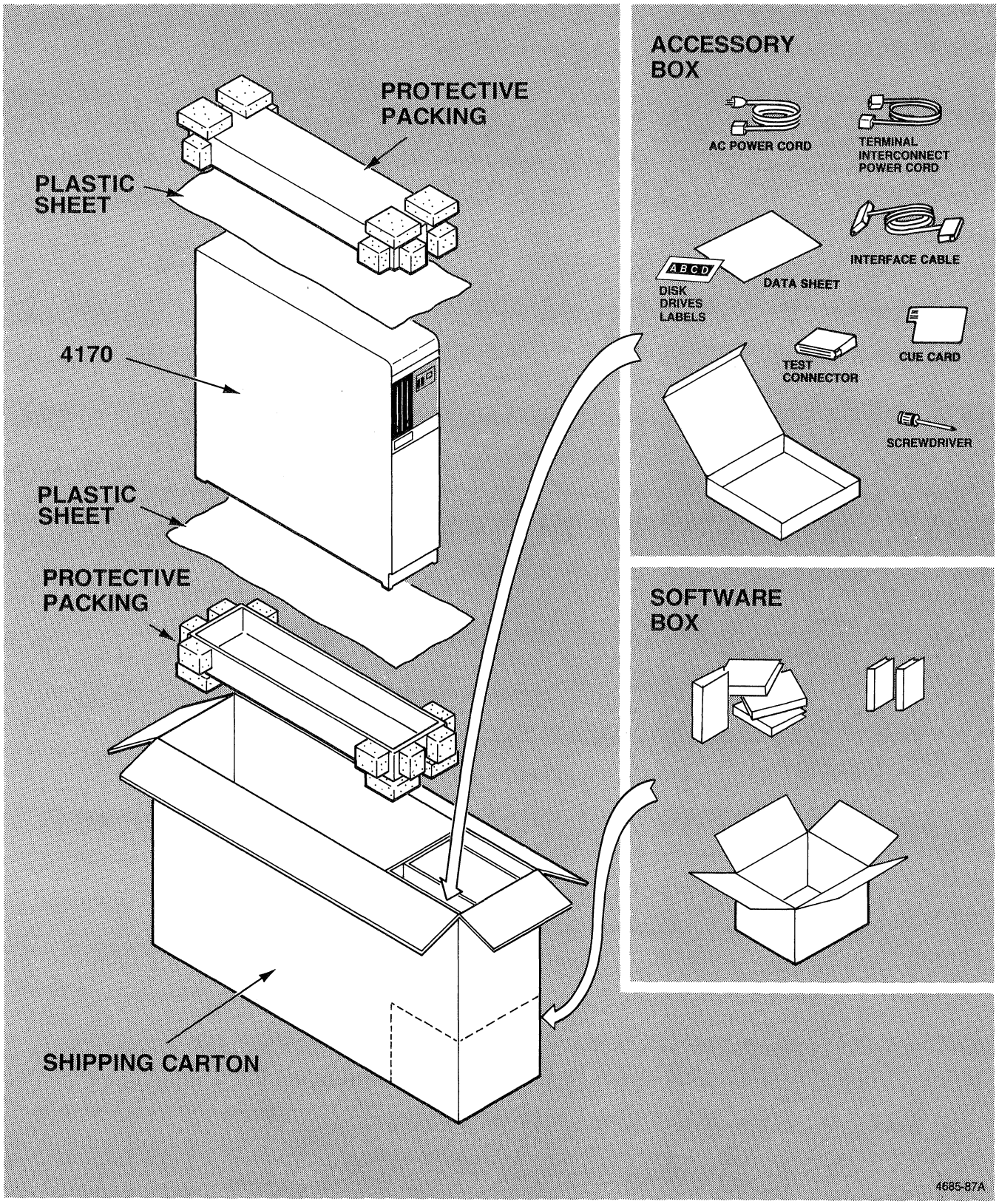


Figure 4-7. Repacking.

Section 5

GETTING STARTED

INTRODUCTION

This section introduces you to the operation of the 4170 and tells you how to get the 4170 running for the first time. Before reading this section, follow the instructions in Section 4, Installation, for installing your 4170 and for connecting a terminal to your 4170.

This section covers:

- Setting up communications with your terminal.
- Loading the CP/M-86 operating system.
- Making backup copies of the operating system diskette and other software.
- Using the optional hard disk (including how to format the hard disk and how to set up the hard disk so that the operating system automatically loads from the hard disk).
- Connecting the 4170 to a host.
- Connecting the 4170 to a printer.

HOW TO USE THIS SECTION

If you just installed the 4170 and this is the first time that you are using it, follow the first three procedures in this section (to establish communications with your terminal, to load the operating system, and to make a backup copy of the operating system diskette).

After you have made a back-up copy of the operating system diskette, you may want to read other procedures in this section if you are going to use a hard disk, host communications, or a printer. If you are not going to use a hard disk, host communications, or a printer, you may decide to skip those procedures. Instead, you may want to load an applications program, using the instructions provided with it, or you may want to go directly to Section 6, Operating Information, to do the exercises there and learn some other CP/M-86 operating system commands.

PROCEDURES FOR FIRST-TIME OPERATION

SETTING UP COMMUNICATIONS WITH YOUR TERMINAL

Use these procedures to establish communications between your terminal and the 4170. Because some applications packages require you to change communications parameters, even after using the 4170 for some time, you may lose communications on occasion; if so, refer back to the procedures here to reestablish communications between the terminal and the 4170. Before you begin, check that the cable between your terminal and the 4170's terminal port (Port 0) is securely connected.

1. If the 4170 and the terminal are not already on, press the power button on the 4170 (Figure 5-1) and on the terminal. (When you turn on the 4170, it tests certain parts of its circuitry. When this self-test is complete, the 4170's bell rings once and the STATUS display on the front panel changes from "FF" to "00".)

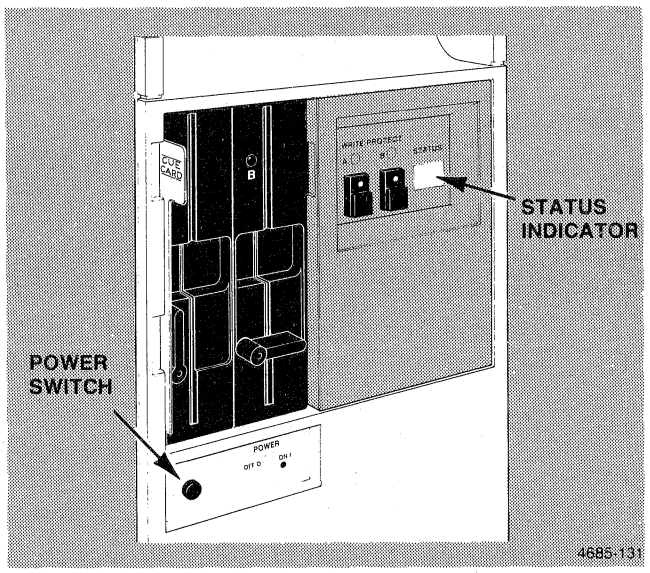


Figure 5-1. Turning on the 4170.

2. Set the terminal's communication parameters. How you do this depends on the kind of terminal you have. Here are the parameters that you must set:

   - Baud rate - 2400
   - Flagging - in/out ("in/out" flagging is DC1/DC3 flagging on a 4100 Series terminal)
   - Parity - none
   - Stop bits - 1
   - Data bits - 8

NOTE

Flagging **for the terminal** should be set to in/out (which is DC1/DC3 flagging). Flagging **for the 4170** is set to "none" (by the CMOS-reset procedure in the next step). If you are going to use applications programs that use CTRL-S or CTRL-Q (such as Wordstar, SuperCalc or Emacs), flagging for the 4170 must be set to none or these programs will not work properly; the terminal's flagging may be set to in/out (DC1/DC3) as described here.

Tektronix 4100 Series Terminals. If you are connecting the 4170 to a Tektronix 4100 Series terminal, refer to Table 5-1.

Table 5-1

CHANGING A 4100 SERIES TERMINAL'S COMMUNICATION PARAMETERS

Terminal Type	To Change Parameters Using Setup Commands	Alternate Method
Tektronix 4105, 4107, or 4109 terminals	*baudrate 2400 <CR> *flagging in/out <CR> *parity none <CR> *stopbits 1 <CR>	*factory <CR> *flagging in/out <CR>
Tektronix 4112, 4113, 4114, 4115, or 4116 terminals	*baudrate 2400 <CR> *flagging in/out <CR> *parity none <CR> *stopbits 1 <CR>	Reset the terminal's Setup memory and set the terminal flagging to in/out. Refer to the operators manual for the specific terminal.

The middle column of Table 5-1 lists the Setup commands that you must enter to set the terminal's communication parameters. To set the parameters on a 4105, for example, put the terminal in Setup mode (by pressing the Setup key) and enter these Setup commands (the <CR> means that you should press the RETURN key; the * is the Setup mode prompt):

```
*baud 2400<CR>
*flagging in/out<CR>
*parity none<CR>
*stopbits 1<CR>
```

NOTE

Press the Setup key to take the terminal out of Setup mode after you are finished entering Setup commands. When Setup mode is in effect, the asterisk (*) appears in front of the blinking cursor.

Alternate Method for 4100 Series Terminals. Because the Setup values are similar to the factory default parameters for the 4100 Series terminals, there is an alternate method -- resetting the terminal to its factory defaults and setting flagging to in/out. The right column of Table 5-1 shows how to do this. On a 4105, for example, you can enter these commands:

```
*factory <CR>
*flagging in/out <CR>
```

Resetting the terminal to its factory defaults, however, resets other parameters besides the communication parameters. If you do not want to reset other parameters, enter the Setup commands (in the middle column of Table 5-1) instead of resetting all parameters to the factory defaults. After you enter Setup commands, be sure to take the terminal out of Setup mode by pressing the Setup key again.

Other Terminals. If you are using a terminal that is not a Tektronix 4100 Series terminal, refer to the terminal's operating manual for information on setting the communication parameters.

3. Reset the communications parameters on the 4170 to these factory default settings:

   - o Baud rate - 2400
   - o Flagging - none
   - o Parity - odd and disabled
   - o Stop bits - 1
   - o Data bits - 8

   To do this, use the CMOS-reset procedure that follows.

The 4170 CMOS-Reset Procedure. To set the communications parameters on the 4170 to the factory defaults, you'll do what is called a "CMOS-reset" (so-called because you'll be resetting the 4170's parameters to the factory default values stored in the constant or CMOS memory).

NOTE

When you power up the 4170 for the first time, the factory default communication parameters are in effect, and it is not necessary to do a 4170 CMOS-reset. If you suspect that the 4170's communications parameters have been changed, then go ahead and do the CMOS-reset procedure given here; otherwise, skip to the next procedure, Loading CP/M-86 from the Diskette.

a. Remove the pop-off cover to expose the control panel shown in Figure 5-2. Press and hold the TEST and RESET buttons.

b. Release the RESET button. When you see the numbers in the STATUS display start to step from 00 to 01, 02, 04, etc., release the TEST button. The STATUS display continues to change, and the T1, T2, and T3 lights blink on and off until finally the bell rings. (Figure 5-2 also shows the locations of the STATUS display and the T1, T2, and T3 lights.)

c. **On the 4170's front panel** (NOT on the terminal's keyboard), hold the CTL button down while you press the C button, and then release both buttons. You should see FF in the 4170's STATUS display (if you don't see the FF, go back to the beginning of the CMOS-reset procedure and start again with step "a"). When you release the buttons, you may see a menu (on the terminal screen) similar to the one in Figure 5-3; if you don't, continue with the next step. (The menu will be visible only if the 4170 was using a baud rate of 2400 bits/second; the menu entries will vary according to the options installed.)

d. **On the 4170's front panel** (NOT on the terminal's keyboard), press the button labeled F2. Again, you may see a menu like the one in Figure 5-4, but even if you don't, proceed to the next step.

e. **On the 4170's front panel** (NOT on the terminal's keyboard), press the button labeled F1; at this point, you may see what is displayed in Figure 5-5, but if you don't, proceed to the next step.

f. **On the 4170's front panel** (NOT on the terminal's keyboard), press and hold the CTL button while you press the E button. Finally, release both buttons.

g. Put the pop-off cover back on the 4170's control panel.

This completes the 4170 CMOS-reset procedure.

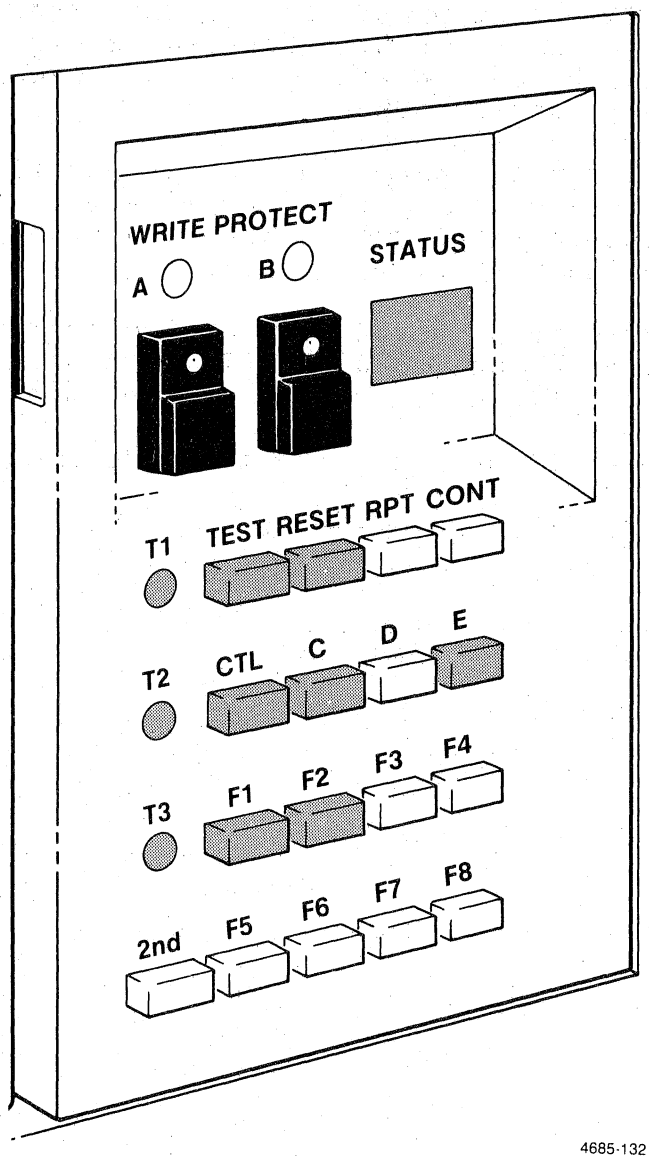


Figure 5-2. The 4170 Control Panel.

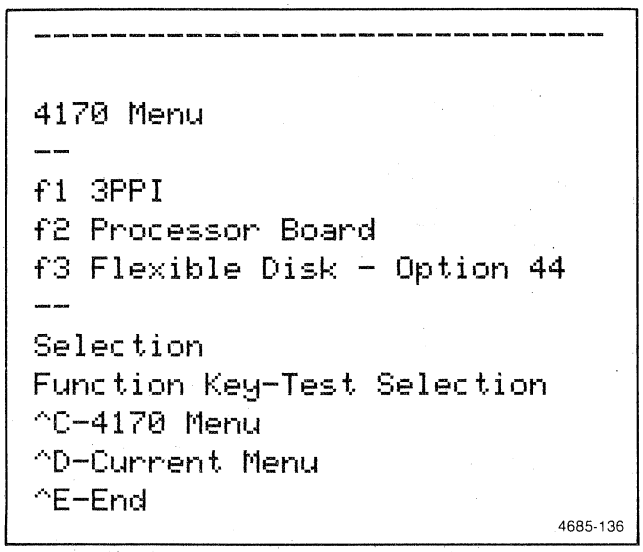


Figure 5-3. 4170 Menu.

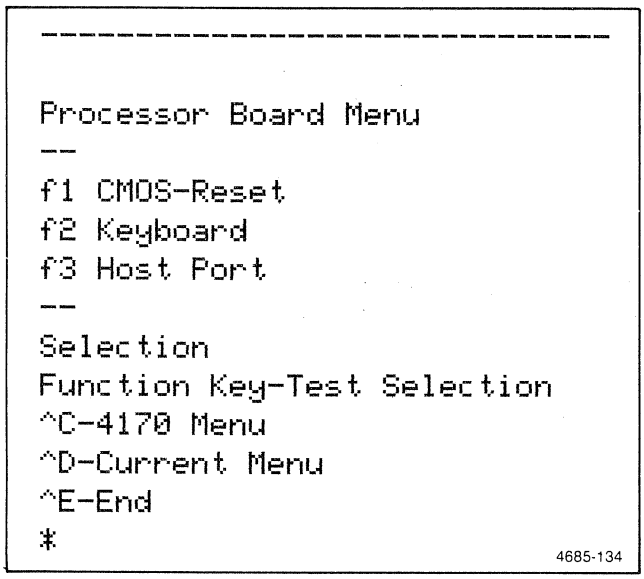


Figure 5-4. Processor Board Menu.

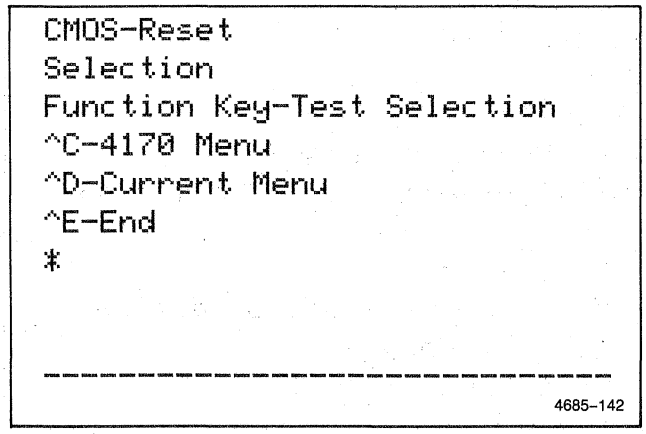


Figure 5-5. CMOS Reset Selection.

LOADING CP/M-86 FROM THE DISKETTE

Insert the CP/M-86 operating system diskette in Drive A and close the drive door latch. (Figure 5-6 shows how to insert the disk. Be sure that your fingers do not touch the surface of the disk where it is exposed.) While the 4170 loads the operating system, the drive busy light for drive A goes on, and the STATUS display steps from 00 through 0F.

When the operating system has been loaded, the light indicator in the disk drive goes off, the STATUS display changes to 00 and then 01, and this message appears on the terminal screen:

```
Tektronix 4170 CP/M-86 V1.1    Release 1.2
A>
```

This message indicates that the CP/M-86 operating system is loaded in the 4170's memory, and that the terminal is communicating with the 4170.

If you do not see this message on your terminal screen or if you see some random characters on the screen, it may be because your terminal's communication parameters do not match those of the 4170. Go back to the first procedure in this section, Setting Up Communications with your Terminal, and reset the communications parameters for **both** the terminal and the 4170. If this message still does not appear, then contact your local Tektronix field office for assistance.

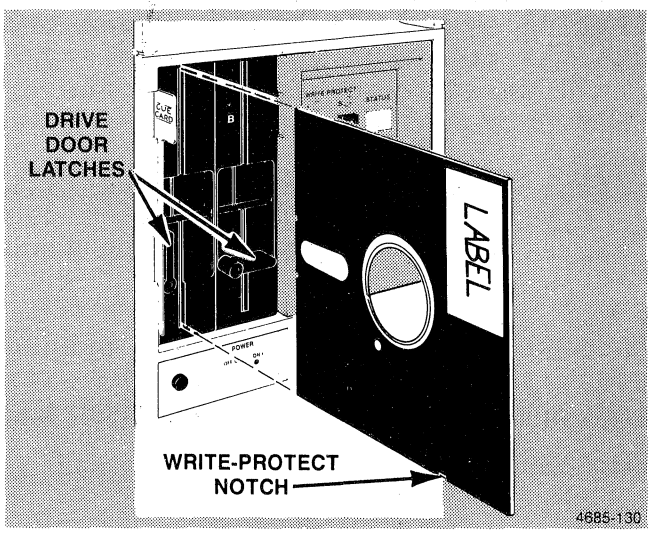


Figure 5-6. Inserting a Disk.

Setting a Faster Baud Rate

After you have successfully loaded the operating system, you may want to increase the rate of communications between the 4170 and the terminal. To do this, change the baud rate first on the 4170 and then on the terminal. (The baud rate settings on the 4170 and the terminal must be identical.)

For example, you can increase the baud rate to 19200 bits/second. (This is eight times faster than 2400 bits/second.) First, enter this command to change the baud rate on the 4170:

A>SETDEV con, baud=19200 <CR>

Then change the baud rate for your terminal. If you have a Tektronix 4100 Series terminal, press the Setup key and enter this Setup command:

***baud 19200 <CR>**

Press the Setup key again to take the terminal out of Setup mode, and press <CR> (the carriage return key) to get the system prompt **A>**.

MAKING A BACKUP OF THE OPERATING SYSTEM DISKETTE

This procedure describes how to make a copy of the CP/M-86 operating system diskette. After you have successfully loaded the operating system for the first time, you should immediately make a copy of the operating system diskette. Store the original in a safe place, and use the copy for your day-to-day operations. This way, if the copy is damaged, you still have the original diskette and can make another copy.

The procedure consists of two steps: (1) formatting a new diskette, and (2) copying the operating system diskette. You will need a new blank diskette before you can copy the operating system diskette.

Tektronix recommends that you use this same procedure to copy all diskettes supplied to you. You may use the 10 extra blank diskettes (supplied with the 4170) for this purpose.

Formatting a New Diskette

This procedure formats a disk in drive B. If you make a typing mistake at any time, use the Backspace or Rubout keys to correct the error.

1. Enter the command **FORMAT B:** after the system prompt **A>**:

```
A>FORMAT B:<CR>
```

CAUTION

Formatting destroys any data that may be on the diskette. Be sure that you do not need any data on the diskette you're going to format, or that it is a new blank diskette, before you confirm the FORMAT operation in the next step.

2. The system responds with this prompt:

```
Formatting drive B:
    5 1/4" Double-Sided Floppy (512 bytes/sector)
    with interleave of 0.
Insert a disk in drive B.
Format the disk? (Y or N; default N):
```

Insert a new diskette into Drive B and close the drive door latch. Enter **Y** and press <CR> to format the diskette in drive B. (Entering any other key besides **Y** cancels the format operation.) As soon as you press <CR>, the 4170 begins formatting the diskette. The drive busy light on the drive goes on, the drive makes a rhythmic clicking sound, and this message is displayed:

```
Formatting (this takes approximately 30 seconds)...
```

If you press <CR> and nothing happens, check to see that the drive door latch is closed (refer back to Figure 5-6).

An error message indicates if the disk or drive is write-protected, and instructs you how to proceed:

Check that the disk and drive are not write-protected. Remove the write-protect tab from the disk, or turn off the drive's write-protect switch. If the disk was not write-protected, formatting a second time may correct the error.

Check to see that the diskette is correctly inserted in the drive, and that the diskette and drive are not write-protected. (On 5-1/4" diskettes, when the notch on the diskette is covered, the diskette cannot be written to or formatted. Figure 5-6 shows the write-protect notch.)

After you've corrected the problem, enter **Y** and press <CR> when the prompt reappears.

3. When the format is completed, the system prompts for another diskette:

Formatting completed.

Insert a disk in drive B.
Format the disk? (Y or N; default N):

If you want to format other diskettes at this time, enter **Y<CR>**; otherwise, press any key except **Y** and then <CR>.

Copying the Operating System Diskette

Next, you will copy the CP/M-86 operating system diskette to the diskette that you just formatted. Make sure the CP/M-86 operating system diskette is in drive A and that the formatted diskette is in drive B.

1. Enter the command **COPYDISK** after the system prompt **A>**:

   A>COPYDISK<CR>

   When the COPYDISK utility is loaded into memory from the operating system diskette, this prompt appears:

   CP/M-86 Full Disk COPY Utility
   Version 2.0
   Enter Source Disk Drive (A-D)?

2. Enter **A<CR>** to indicate that you want to copy the operating system diskette in drive A.

   Enter Source Disk Drive (A-D)? A<CR>

   If the nothing happens after you press <CR> and the light in the source drive door remains on, check that the diskette is correctly inserted in the source drive and that the drive door latch is closed.

3. The system then prompts you for the destination drive:

   Enter Destination Disk Drive (A-D)?

   Enter **B<CR>** to indicate that you are going to write on the formatted diskette in drive B.

   Again, if nothing happens after you press <CR> and the light in the destination drive remains on, check that the diskette is correctly inserted in the destination drive and that the drive door latch is closed.

   This error message is displayed if the diskette in the destination drive is not properly formatted:

   Source and destination disks must be the same type.

   Refer back to Formatting a New Diskette for the procedure.

4. This prompt appears, asking you to verify the copy operation:

Copying Disk A: to Disk B:
Is this what you want to do (Y/N)?

CAUTION

COPYDISK destroys all data on the diskette in the destination drive. Be sure that you have the CP/M-86 operating system diskette in the source drive and the formatted diskette in the destination drive. Check the diskettes before you confirm the copy operation. You may press the write-protect switch for drive A to write-protect the operating system diskette.

If you are sure that the diskettes are in the correct drives, enter **Y<CR>** to confirm the copy.

The system first reports that the copy is started:

Copy started

This is followed by a message that tells you what stage of the copy is taking place. There are three stages during the copy. First, the system reads each track on the diskette in the source drive. Then, it writes each track to the diskette in the destination drive. Finally, the system verifies that each track was correctly written. During the three stages, the system updates a message on your screen that looks like this:

Reading track ##

If the destination drive is write-protected (either by the switch or the notch in the diskette), an error like this is displayed:

Permanent Error Writing Track 79, Sector 31
Ignore error (Y/N)?

If this error occurs, unprotect the diskette or the drive, enter **N** to exit from the COPYDISK program, and try again. This kind of error may also occur during the verification stage; if so, enter **N** and try the copy again.

5. If the copy is successful, the system reports when the copy is completed and prompts you for another copy:

```
Copy completed.
Copy another disk (Y/N)?
```

   Enter **N<CR>** to exit the copy program.

6. Remove the operating system diskette from Drive A. Store it in a safe place.

CAUTION

> The next step instructs you to create a label for the copy of the operating system diskette. To prevent damage to the diskette, write out the label **before** you place it on the disk.

7. Remove the duplicate operating system diskette from drive B. Attach a label to the diskette so you can identify it as a copy of the operating system diskette.

8. Install the duplicate operating system diskette in drive A.

9. Press CTRL-C (the CTRL and the C keys at the same time) on the terminal keyboard to "log-in" the new disk in drive A. After you press CTRL-C, the system responds with the system prompt **A>**. (When you insert a different diskette or when you switch between drives A and B, a CTRL-C at the terminal informs the system of the change. If you don't press the CTRL-C and you try to write to the diskette, you may get an error.)

Now you should duplicate your other diskettes, repeating the procedures here for formatting and copying diskettes. Then, if you are going to use a hard disk, host communications, or a printer, you may continue with the following procedures. Or, after making copies of your other diskettes, you may want to skip to Section 6, to do the exercises and learn more of the CP/M-86 commands.

USING THE HARD DISK

If you have the optional hard disk (Option 4170F03), read the procedures here to get started using the hard disk. There are two hard disk procedures given here: formatting the hard disk and setting up the hard disk as the default/boot disk.

You have the option of setting up the hard disk as the default/boot disk. (The default disk is the disk the operating system defaults to after booting; the boot disk is the disk that the system automatically boots from when the 4170 is powered up.) If you have a hard disk and you configure the system as the default/boot disk, the system will boot from the hard disk and default to it after booting. Even when the hard disk is used as the boot drive, you must still have a formatted diskette in drive A: before the system will boot. (You do not have to use the operating system diskette; any formatted diskette will work.)

Formatting the Hard Disk

NOTE

Tektronix formats the hard disk before it is shipped. You do not need to use this procedure to format the disk before using it.

However, if you intend to configure the hard disk as the default/boot drive and there are files already on the hard disk, you may want to format the disk first to delete those files. (The file **CPM.SYS** must be the first physical file on the hard disk, so the hard disk must be empty before it can be made the default/boot disk.) If you have not yet used the hard disk, you may skip to the next step, Setting Up the Hard Disk as the Default/Boot Drive.

1. Insert the CP/M-86 operating system diskette in drive A. Press CTRL-C to log in drive A:

```
A> (CTRL-C)
A>
```

2. Enter the FORMAT command after the system prompt **A>**. Specify drive C:, the hard disk, as the disk to be formatted:

```
A>FORMAT C:<CR>
```

The FORMAT utility is loaded into memory from the operating system diskette, and this message appears on your terminal screen:

```
Formatting drive C:
    Winchester drive (512 bytes/sector) with interleave of 5.

 Please enter the numbers of any tracks known to be bad.
 More than one number may be entered on a line but they
 must be separated by commas.  Any spaces are ignored.
 Consecutive tracks may be specified by entering the
 starting track, a dash ('-'), and the ending track (e.g.,
 '14-18').  To end the list, press RETURN without
 entering any characters.  If you don't know of any bad
 tracks, just press RETURN.

 Enter bad tracks:
```

Occasionally certain parts of a hard disk are known to be unreliable; these are called "bad tracks." If your disk has bad tracks, it does not mean that the disk is unusable, but that small parts of it cannot reliably record information. When you format the disk, the system identifies up to 160 bad tracks and formats alternate tracks. Even when FORMAT detects bad tracks, you still have the full 8 megabytes of hard disk storage that CP/M-86 supports.

When you installed the hard disk option in your 4170, there should have been a sheet of paper (like the one shown in Figure 5-7) in the box with your disk drive. This sheet contains information from the disk drive manufacturer about bad tracks on your disk. To figure out the bad track numbers from the information on this sheet, use this formula:

(4 * Cylinder#) + Head# = Badtrack#

Using Figure 5-7 as an example, 79 is the only bad track for that disk (the Cylinder# is 19 and the Head# is 3). Thus, you would specify:

Enter bad tracks: 79<CR>

If there was more than one bad track, separate the tracks with commas, like this:

Enter bad tracks: 79,87<CR>

If you do not have a sheet (like that in Figure 5-7) to calculate the bad track numbers, simply press <CR> when you see the prompt for bad track numbers. FORMAT can detect bad tracks on its own. (Because the manufacturer's testing is so extensive, it is preferable to specify known bad tracks if you have the manufacturer's list; if you have the manufacturer's list, you should use it.) If FORMAT detects more than 160 bad tracks, it reports an error; if this occurs, contact your Tektronix field office.

When FORMAT finds bad tracks, the 4170 displays this message:

Formatting alternate tracks.

FORMAT identifies the bad tracks and formats alternate tracks (which are otherwise unused). Thus, you still have the full amount of storage on the disk available.

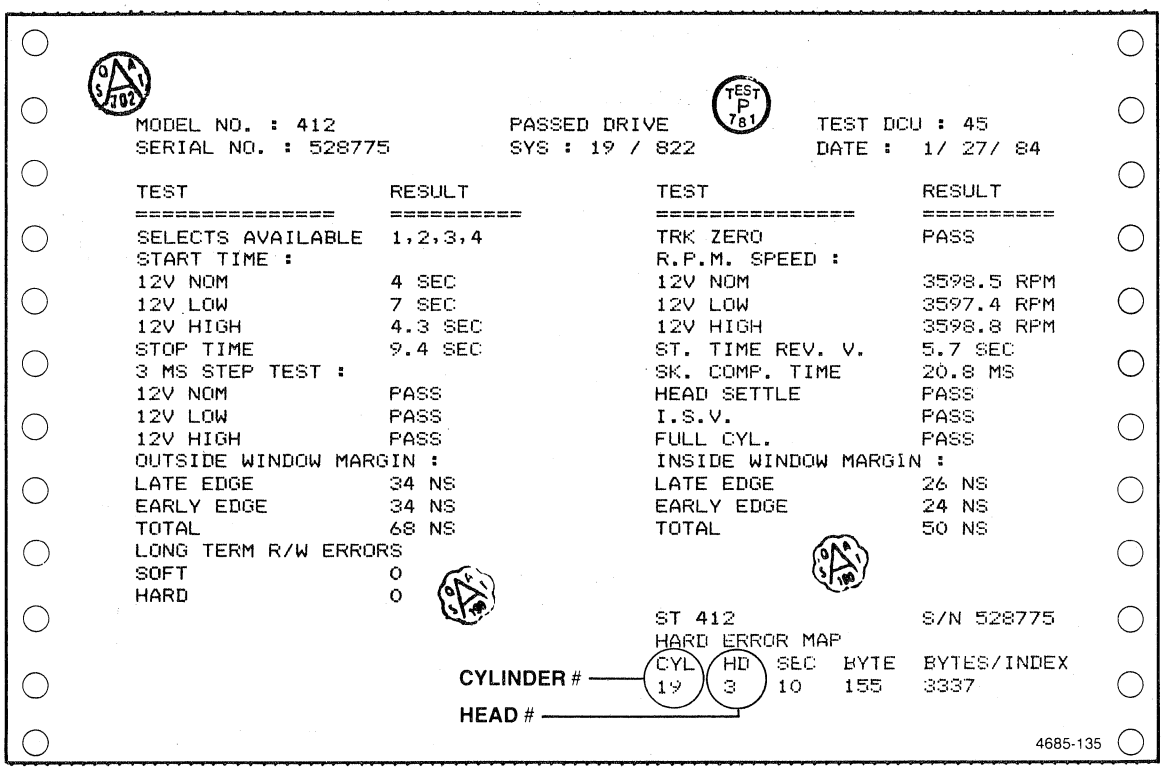


Figure 5-7. Calculating Bad Tracks from the Manufacturer's Test Report.

FORMAT asks you to confirm the FORMAT operation before it begins:

```
Format the disk?  (Y or N; default N): Y
Are you SURE?  (Y or N; default N): Y
This will take about 5 minutes
Formatting Winchester
Checking for bad tracks
Writing loader
Format complete.
```

After the hard disk is formatted, if you are going to configure it as the default/boot drive, complete the following procedure.

Setting Up the Hard Disk as the Default/Boot Drive

To set up the hard disk as the default/boot drive, you must:

- Copy the file CPM.SYS as the first physical file on the hard disk.
- Configure the system (using SETDEV) so that the hard disk is the default/boot drive.

Copying CPM.SYS to the Hard Disk. To do this, you will use the Peripheral Interchange Program (PIP) to copy the file from the CP/M-86 operating system diskette to User 15 on the hard disk. Enter the command:

```
PIP C:[G15]=A:CPM.SYS[ROVG15]<CR>
```

This copies CPM.SYS as the first physical file on the hard disk.

Making the Hard Disk the Default/Boot Drive. Next, you use the SETDEV utility to set up the hard disk as the default/boot disk. Enter this command:

```
A>SETDEV bootdrive=C:, defaultdrive=C:<CR>
```

The 4170 now boots from the hard disk, and after booting the system defaults to the hard disk drive (C:). To verify this, press RESET on the 4170 (located on the 4170's control panel, beneath the pop-off door).

NOTE

Make sure that there is a formatted disk in drive A.

The terminal screen should display a message like this:

```
Tektronix 4170 CP/M-86 V1.1 Release 1.2
C>
```

CONNECTING THE 4170 TO A HOST

This discussion covers how to connect your 4170 to a host computer and how to establish communications between the host and the 4170 using the MODEM86 program. Tektronix provides several preconfigured versions of the MODEM86 program; this section introduces you to those versions and discusses which version you should use. The MODEM86 program is described in more detail in Section 7, Programming Information.

The discussion here of host/4170 communications includes:

- Connecting to a host computer.
- Getting started with the MODEM86 program.
- Transferring files between a host and the 4170.

Connecting to a Host Computer

Figure 5-8 shows the 4170 in a typical configuration with a host. As the figure illustrates, the 4170 is often connected to the host via a telephone modem, which is, in turn, usually connected to the 4170's HOST port. If you want to connect the 4170 to more than one host, you can use any one of the 3PPI (Three-Port Peripheral Interface) ports in addition to the HOST port. Figure 5-9 shows the locations of the HOST port, the standard 3PPI ports, and the optional 3PPI ports. (The optional 3PPI ports are available as Option 4170F10.)

If your 4170 is not already connected to the host, connect the cable (from the host or modem) to the HOST port or to a 3PPI port; be sure to secure the cable with two screws.

To transfer files between two 4170s, you may need a special cable connector, depending on which ports you intend to use. A standard RS-232-C cable is required to connect the HOST port on one 4170 to a 3PPI port on the other. If you intend to use a 3PPI port on each unit, or the HOST port on each unit, a "null modem" cable with the following pin connections is required:

Connector A Pins	Connector B Pins	Connector A Pins	Connector B Pins	Connector A Pins	Connector B Pins
1	1	5	4	12	12
2	3	6	20	19	19
3	2	7	7	20	6
4	5	11	11		

To connect two 3PPI ports, Tektronix cable 012-0689-01 provides these pin connections on a cable with two male ends.

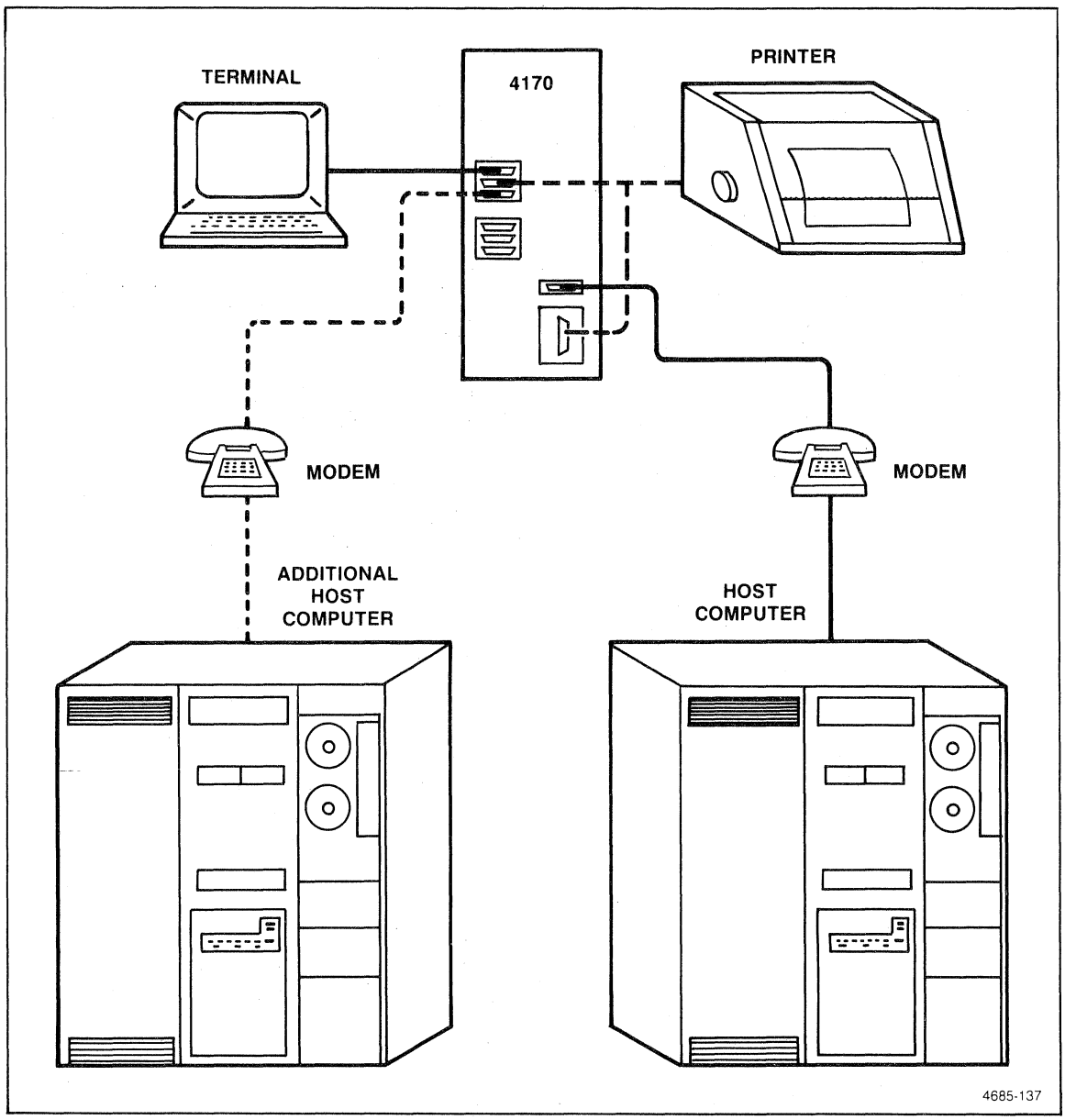


Figure 5-8. Typical 4170 System Configuration.

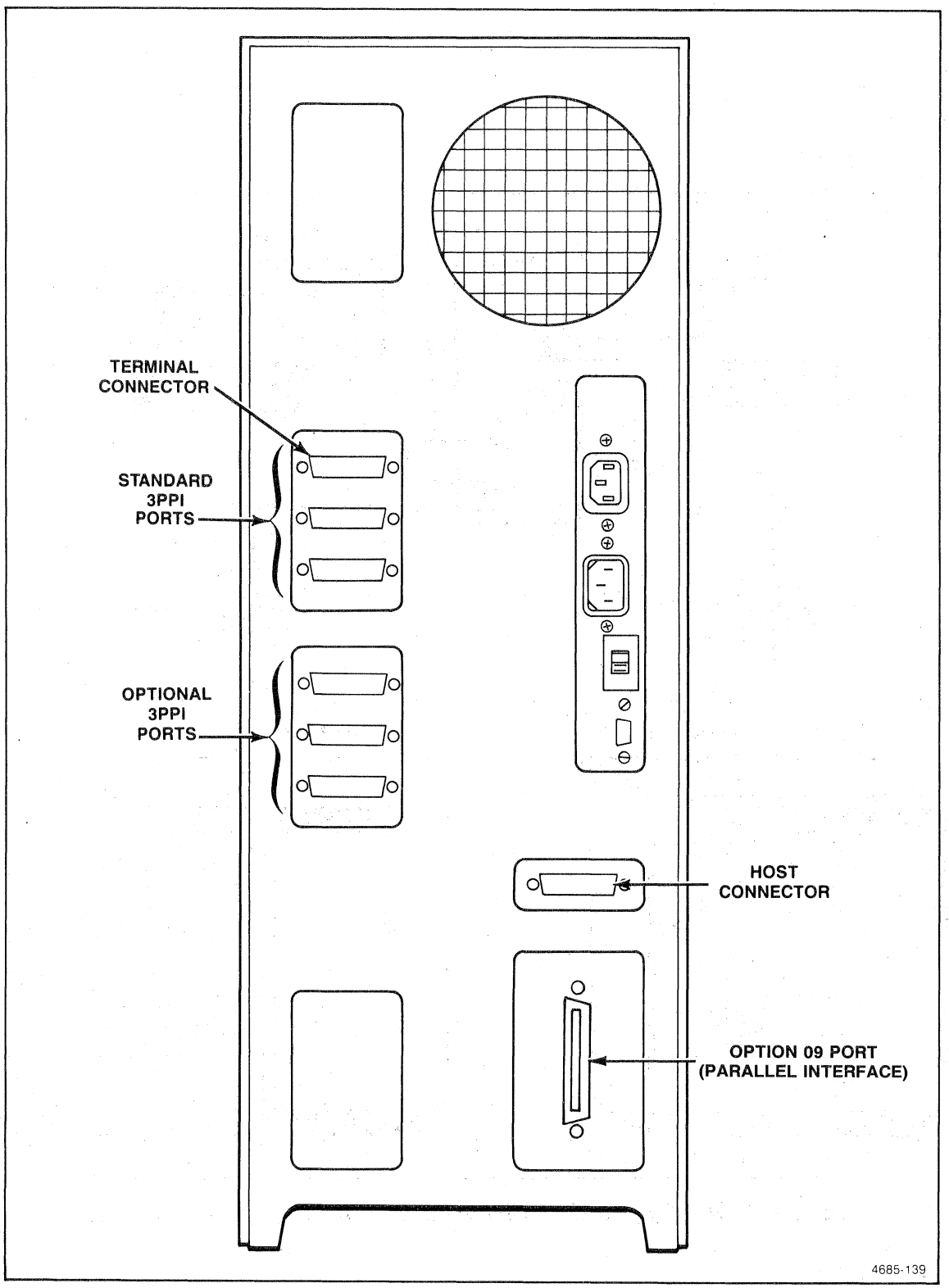


Figure 5-9. 4170 Port Locations.

Getting Started with the MODEM86 Program

Use the procedures here to run the MODEM86 program on your system and to establish communications between the host and the 4170. To follow these instructions, you'll need the MODEM86 and CP/M-86 operating system diskettes.

1. Turn on the 4170 and terminal. Load the operating system.

2. Insert the MODEM86 diskette in drive B, and close the drive door latch.

3. Enter the directory command to view what's on the MODEM86 diskette in drive B:

   A>DIR B:<CR>

   Figure 5-10 shows a directory of the MODEM86 diskette. Tektronix provides several configurations of MODEM86, the program for communicating between a 4170 and a host. The different configurations are MODM86-H.CMD, MODM86-1.CMD, MODM86-2.CMD, MODM86-3.CMD, MODM86-4.CMD, and MODM86-5.CMD. (Other files on the disk are described in Section 7.)

4. Change the default drive to drive B:

   A>B:<CR>

   When the system prompt **B>** appears, type a CTRL-C to log in drive B.

```
A>dir b:
B: COM2CMD  COM : MODEM86  COM : COM2CMD  CMD : DISTMDM  SUB
B: EXPAND   CMD : MODM86-3 CMD : MODEMSET COM : PHONES   DAT
B: REMOVECC CMD : SHRINK   CMD : SIOBIOS  A86 : SIOBIOS  MAC
B: UNSQZ    CMD : MODM86-4 CMD : MODM86-H CMD : CMD2COM  COM
B: CMD2COM  CMD : MODEMSET CMD : MODM86-5 CMD : BIN2HEX  COM
B: MODEM86  SET : EXPAND   COM : UNSQZ    COM : SHRINK   COM
B: REMOVECC COM : MODEM86  CMD : MODM86-1 CMD : MODM86-2 CMD
B: README   1ST :

SYSTEM FILE(S) EXIST
A>
```

4685-143

Figure 5-10. Directory of the MODEM86 Diskette.

5. You are now ready to use one of the MODEM86 programs. The program you will use depends on the 4170 port to which your host is connected. For example, if the host is connected to the HOST port on the 4170, you will use the program MODM86-H. Refer to the list below to determine which MODEM86 program you will use.

Port	Program
HOST Port	MODM86-H
Port 1	MODM86-1
Port 2	MODM86-2
Port 3	MODM86-3
Port 4	MODM86-4
Port 5	MODM86-5

6. Enter the program name after the system prompt. If your host is connected to the host port, for example, enter:

   B>MODM86-H<CR>

   (Note that you must type **MODM86-H**, not MODEM86-H.)

   The program displays the MODEM86 main menu (Figure 5-11). (Section 7 describes the menu functions in more detail.)

```
B>modm86-h

MODEM86 as of 03/29/84
(C) Copyright 1982-1984 Mark Hersey

  T - Terminal mode in full-duplex        H - Terminal mode in half-duplex
  E - Terminal mode with echo
  R - Receive a file                      S - Send a file
  L - List file directory entries         F - Display file data
  X - Toggle expert mode (menu on/off)    M - Menu
  ? - Help, command syntax description    V - Current values display
  D - Disconnect (hang up phone)          Q - Quit, exit to operating system

Default drive: B
Enter command:
                                                              4685-144A
```

Figure 5-11. MODEM86 Main Menu.

7. View the communications parameters that the MODEM86 program has set, and verify that they are acceptable for your particular host. To view the parameters, enter a **V** (Current Values Display) after the prompt, like this:

   Enter command: V<CR>

   The MODEM86 program displays the current values of the communications parameters it expects. For example, if you are using MODM86-H, you see this message:

   ```
   Communication port #: 1
   Baud rate: 1200
   Parity: None
   Stop bits: 1
   Host handshaking signals: On
   Conversation save file: None
   Press any key to continue:
   ```

   Notice that "1" is the communication port number for the MODM86-H program. This is the MODEMSET communication port number (and not the number of a 4170 3PPI port); MODEMSET communication port 1 corresponds to the 4170's HOST port. (See Section 7 for a cross-reference of MODEMSET numbers and 4170 physical ports.)

   Next, determine if the baud rate of 1200 is what your host expects. For example, if the host transmits at a rate of 4800, then you must change the baud rate from 1200 to 4800. The next step instructs you how to do this. If you want to change other program defaults (such as parity or stopbits), you can do so using the MODEMSET program supplied on the MODEM86 diskette; see Changing Preconfigured MODEM86 Programs in Section 7. In most cases, changing the baud rate is sufficient to establish communications.

   Press any key to redisplay the MODEM86 main menu.

8. To change the baud rate, enter **T** after the prompt on the MODEM86 main menu:

Enter command: T<CR>

The MODEM86 program responds with:

Conversation save file: None
Terminal Mode Control Characters:
B)Break D)Disconnect E)Exit L)Literal R)Rate T)Transfer

The first line of this message reports that there was no conversation save file specified when you gave the T command. (Conversation save files are discussed under the heading Transferring Files from the Host to the 4170, later in this section.) The last two lines of the message summarize the functions of the control characters that you can enter.

To change the baud rate, press CTRL-R (hold down the CTRL key and press the R key). The CTRL-R (the control character for Rate) does not print, but this prompt appears:

Enter new baud rate:

Enter the baud rate that your host expects for communications between it and the 4170. A baud rate of 4800 bits/second or less is recommended for literal data transfers. (Higher rates could possibly result in lost data during literal transfers. Rates of up to 19200 baud, however, may be used when transferring files with the S and R menu options because these options use the Ward Christiansen error-checking protocol; see Section 7 for more information.)

If you have a phone modem, dial up a telephone line and connect to the host. The terminal now acts as if it is directly connected to the host.

Enter a CTRL-E to return to the MODEM86 main menu. If you enter Q from the MODEM86 main menu (to quit and exit to the CP/M-86 operating system), you can use CP/M-86 without disconnecting from the host. To return to using the host, simply reenter the program name (for example, MODM86-H) and change the baud rate again, if necessary. Thus, you can bounce back and forth between using the host and using the 4170 and CP/M-86.

Transferring Files Between the Host and 4170

When the MODEM86 main menu (Figure 5-11) is visible, you can transfer files between the host and the 4170. To transfer files, you use the T (Terminal Mode) menu function. The following discussion covers:

- How to transfer a file from the host to the 4170.
- How to transfer a file to the host from the 4170.

It is recommended that you select small files when you experiment with the MODEM86 examples here. This will allow you to run through the steps quickly to speed up your understanding of how to get started using MODEM86.

See Section 7, Programming Information, for additional details on transferring files with MODEM86.

Transferring Files from the Host. To transfer a file from the host to the 4170, it is recommended that you have room on the diskette that is the file's destination. Because the MODEM86 diskette is full, the following procedure first instructs you to copy the MODEM86 program you'll need to the operating system diskette; make sure that there is space on the operating system diskette for the copy. (After you've used MODEM86 for a while, you'll realize that the program allows you to insert a different diskette when a transferred file is too long to fit on the first diskette.)

1. Insert the operating system disk in drive A and the MODEM86 diskette in drive B. Log in both drives (make the drive active and enter a CTRL-C). Make sure the write-protect switches on the drives are off; otherwise, you may get a BDOS error later.

2. Copy the appropriate MODEM86 program for your system to the operating system diskette. If your host is connected to the host port, for example, then copy the MODM86-H.CMD program; in this case, enter:

   A>PIP A:=B:modm86-h.cmd[OV]<CR>

   Remove the MODEM86 diskette and put it away.

3. Establish communications with the host. To do this, type the name of the appropriate MODEM86 program. For example:

   A>MODM86-H<CR>

   The MODEM86 main menu appears (refer back to Figure 5-11). From that menu, select **T** (Terminal Mode), and if necessary, do a CTRL-R to set the baud rate. Dial a phone line or otherwise connect to the host. When you have established host communications, type a CTRL-E to exit back to the MODEM86 main menu.

4. Select the **T** (Terminal Mode) function again, only this time, specify a conversation save file. For example:

   Enter command: T save.txt<CR>

   A conversation save file (save.txt, in this example) saves your dialog between the host and the terminal; it provides a medium in which you can save whatever the host transmits to the terminal -- including host commands as well as transferred files.

   After you press the carriage return, this message appears:

   Conversation save file: A:SAVE.TXT
   Start saving conversation (Y/N) [Y]?

   (The terminal displays a BDOS error instead of this prompt if the drive is write-protected. Unprotect the drive and press <CR> to get the prompt.)

5. Enter **N** because you do not yet want to begin saving conversation in the file save.txt. This list of control characters appears:

   Terminal Mode Control Characters:
   B)Break D)Disconnect E)Exit L)Literal P)Purge R)Rate
   T)Transfer Y)Yank

   When you are ready to turn on the conversation-save feature to capture the file from the host, you will enter a CTRL-Y (Yank); CTRL-Y acts as a toggle to turn conversation saving off and on. At this point, what you type on the terminal is sent to the host but not saved.

6. Enter the host command that begins transmitting text, but before you press the carriage return key, enter a CTRL-Y. After the CTRL-Y, press <CR>. (Consult your host system's manual or your host programmer for the specific command that initiates output from the host.) On some host systems, this command may be simply a "type hostfile.txt <CR>" command, so that what you enter after the host prompt (**@** in this example) looks like this:

   @type hostfile.txt (CTRL-Y) <CR>

   The CTRL-Y toggles the conversation save feature on, so that when the host begins transmitting, the MODEM86 program saves the data in a file save.txt. (When the conversation save feature is on, a colon (:) appears on the left side of your screen. The colon is not saved as part of the file.) As soon as you press <CR>, the host starts transmitting the file to the 4170; you should see it scroll across your terminal screen.

   When the transfer is complete, if your system has a host prompt, it reappears. (The prompt, unfortunately, is also saved in the file, because the conversation-save feature is still on. You may use ED, the CP/M-86 line editor, or another text editor to delete the prompt from the file, if necessary.)

7. Press CTRL-Y to turn off conversation save and CTRL-E to exit Terminal Mode. The MODEM86 main menu appears, preceded by this message:

   ++Remember to write or purge conversation save file.++

   At this point, the conversation save file has been saved in a buffer, but it has not been written to a diskette. The prompt reminds you to write the file (**W** from the MODEM86 main menu) or to purge or delete it from the buffer (**P** from the main menu).

8. Enter the appropriate command after the prompt to write or to purge the conversation save file. For example, entering W writes the file save.txt to the default drive (drive A in this example):

   Default drive: A
   Enter command: W <CR>

   When the file is written to the diskette in drive A, the MODEM86 main menu appears and indicates that the file was written correctly.

9. Enter **L** (List file directory entries) to check that the file transfer was successful:

   Enter command: L<CR>

   You should see that the file save.txt now exists on the diskette in drive A. To view the contents of that file, quit MODEM86 (Q from the MODEM86 main menu) and enter this CP/M-86 command:

   A>type save.txt<CR>

If MODEM86 does not work properly, refer to the heading If You Have Problems with MODEM86 in Section 7.

Transferring Files from the 4170. To transfer files from the 4170 to the host, you must know how to open a file on your particular host. This usually involves using some kind of text editor to create a new file in which data can be captured. As an example, this discussion uses a text editor called TECO, but how you open a file depends on your particular host.

This example uses the operating system diskette and the MODM86-H program (in drive A) as in the previous example.

1. Establish communications with the host. As before, enter the program name to get the MODEM86 main menu:

   A>modm86-h<CR>

   Then enter **T** after the prompt:

   Enter command: T<CR>

   This list of control characters is displayed:

   Terminal Mode Control Characters:
   B)Break D)Disconnect E)Exit L)Literal R)Rate T)Transfer

   Change the baud rate, if required, with CTRL-R, and dial a phone line, if required.

2. Open a file on the host that will capture what is transmitted by the 4170 and MODEM86. What you enter depends on your particular host; consult your system manual or system programmer for assistance.

   Here is an example for a particular host system. The following commands (after the host prompt **@**) invoke a text editor called TECO. After the TECO prompt *****, the commands open a file called "from4170.txt" in which the transmitted data will be inserted:

   @r teco<CR>
   ***ew from4170.txt$$**
   ***i <CR>**

3. After you have opened a file on your host (using whatever particulars your host requires), press CTRL-T. This is the control character for a Transfer (refer back to the list of control characters under Step 1).

   The MODEM86 program prompts you for a filename:

   Enter file name to be transferred - CR to quit:

4. Type in the name of the file you want to transfer from the 4170. For example, to transfer the file "intro.doc" from the operating system diskette in drive A:

   Enter file name to be transferred - CR to quit: intro.doc

   You can specify a drive (such as **B:**) before the filename; otherwise, the default drive is assumed.

   If the MODEM86 program finds the file on the specified drive, it responds with a message like this:

   A:INTRO.DOC open, size: 25 (00000019H) sectors
   Minimum duration of transfer: 0 minutes and 3 seconds
   Should LF's be sent (Y/N) [N]?

   The MODEM86 program does some internal calculations, using the current baud rate and the size of the file (in sectors), to estimate how long the file transfer will take.

   If the file does not exist on the specified drive (or if you made an error in typing the filename), you'll see a message like this instead:

   Can't open filename.ext ++ File does not exist ++ Enter file name to be transferred - CR to quit:

   You can then reenter the filename or specify a different file.

5. Enter **N** after the prompt for sending linefeed characters (LF's):

   Should LF's be sent (Y/N) [N]? N

   (Answering **Y** attaches an extra linefeed character to every carriage return-linefeed sequence. Usually you will want to answer **N**.)

   NOTE

   No carriage return is required; the transfer begins immediately after you respond to the prompt for sending LF's.

   This message concludes the transfer:

   ++ File transfer completed ++

6. Close the file on the host that was previously opened for the transferred data. Again, this depends on your particular host system.

   Using the earlier example of a host file opened with TECO, these commands close the file "from4170.txt":

   ```
   $$
   *ex$$
   @
   ```

   After the host prompt appears (**@** in the example), you may want to do a directory to verify that the file "from4170.txt" exists on the host.

This completes the examples of getting started with host communications. Section 7, Programming Information, discusses the causes of some typical problems that you may encounter. See that section for more information about using MODEM86.

CONNECTING A PRINTER

If you have a printer, this section instructs you how to connect the printer, how to configure the 4170, and how to use the printer.

This section talks specifically about a printer, but the procedure is similar for most peripherals. Generally, to connect a peripheral, you must:

- Attach the peripheral to one of the 3PPI ports or to the Option 09 port.
- Assign the peripheral to a logical device (using SETDEV).
- Configure the communications for that logical device (using SETDEV) to match the peripheral's requirements.

Finally, to use the peripheral, you can specify the logical device in several CP/M-86 commands (PIP, TYPE, etc.). As discussed in the following, you may also use CTRL-P to toggle output to the logical device LST:.

Connecting a Printer

If your printer uses an RS-232 interface, connect the printer cable to one of the 3PPI ports (see Figure 5-12). If your printer uses a Centronics-style parallel interface, you must have Option 09 for the 4170; in this case, connect the printer cable to the Option 09 port (Figure 5-12).

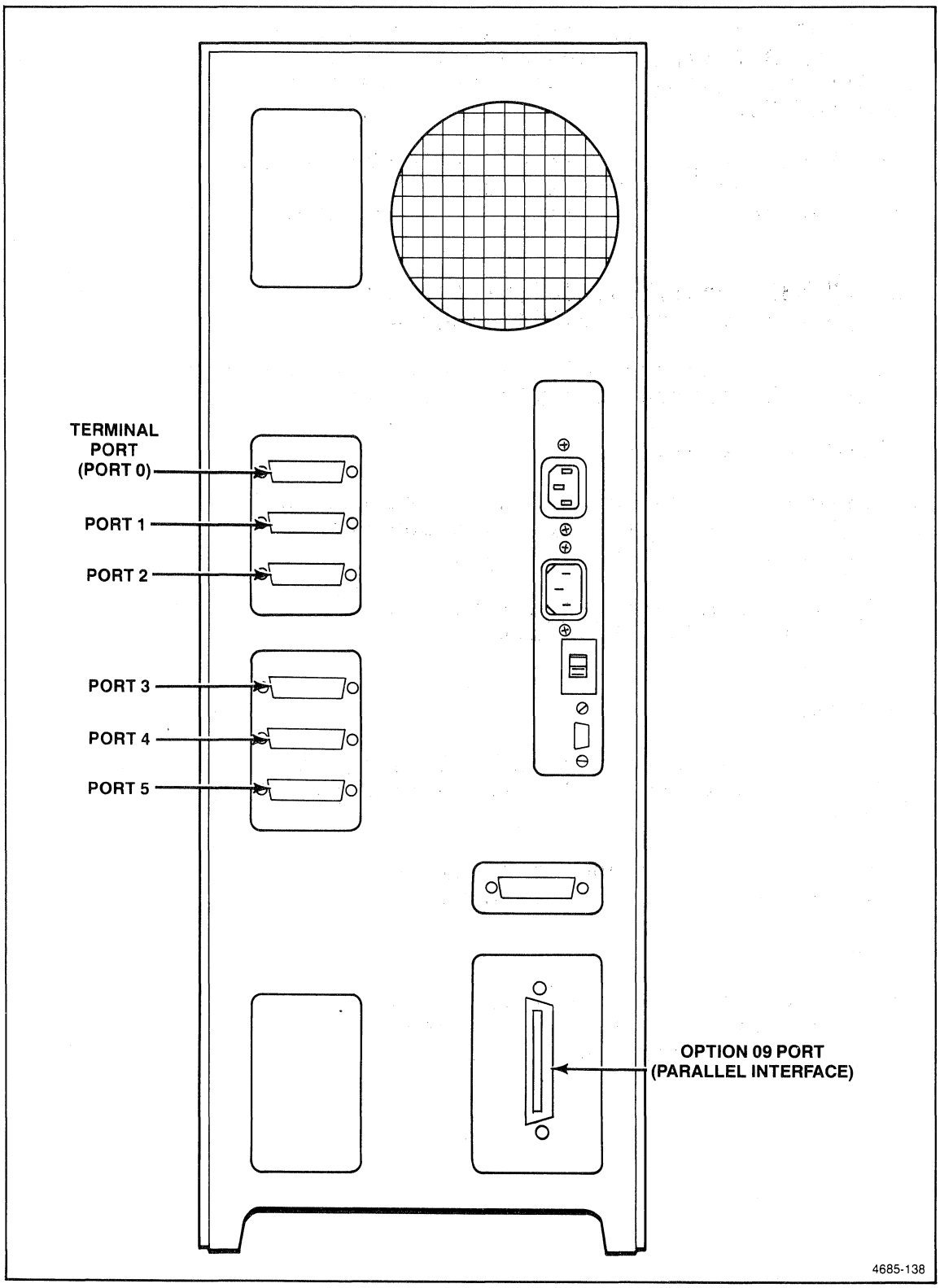


Figure 5-12. Attaching a Printer.

Configuring the 4170

Use the SETDEV utility to configure the 4170 to talk to the printer. With SETDEV, you assign the printer port to the logical device LST: and set the communications parameters that the printer requires.

This example assigns the printer (which is connected to port 1) as the logical device LST:, and sets the communications parameters as shown ("xon" flagging means that DC1/DC3 flagging is used):

```
SETDEV LST=port1,baud=300,parity=none,
 stopbits=1,databits=8,flagging=xon
```

This example assigns the LST: device to the Option 09 port, which uses a parallel Centronics-style interface:

```
SETDEV LST=parallel
```

The parameters that you should use with the SETDEV utility depend on the type of printer that you are connecting. Refer to the printer's operating manual.

Section 7, Programming Information, also contains a detailed description of the SETDEV utility.

Using a Printer

By assigning a printer to the LST: device, you can use CTRL-P (the CTRL and P keys pressed simultaneously) to toggle output to the printer.

For example, insert the operating system diskette in drive A. Press CTRL-P and then type this command:

```
A>DIR<CR>
```

The DIR command and the directory appear on the terminal and are output to the printer. After the directory of the operating system diskette has printed, enter another CTRL-P to turn off printing.

The CTRL-P can also be used in conjunction with the CP/M-86 command TYPE. Try entering this command, but before you press the RETURN key, enter a CTRL-P:

A>TYPE intro.doc (CTRL-P) <CR>

This prints the entire file intro.doc. Another way to print the same file is with the PIP command:

A>PIP lst:=a:intro.doc <CR>

This concludes the exercises for getting started with the 4170. Section 6, Operating Information, gives examples of some other CP/M-86 commands.

Section 6

OPERATING INFORMATION

INTRODUCTION

This section describes how to run and use the CP/M-86 Operating System with your 4170. This introduction gives you some needed background information about mass memory names and terminal operating modes. Then, the Familiarization Exercises start you using the 4170 and its operating system.

If you are a first-time user, it is recommended that you go through Section 5 first and then through the exercises in this section. The exercises here direct you to perform some frequently used CP/M-86 procedures, and it is a starting point for learning more about the operating system; the Digital Research manual, CP/M-86 Operating System User's Guide, provides additional information. Finally, you should read Section 7 for detailed information about specific programming applications.

If you are an experienced user of the CP/M-86 operating system, most of this section is already familiar to you. However, you should read the summary of the differences between standard CP/M-86 and the 4170's version of CP/M-86; see the heading Differences from Standard CP/M-86 at the end of this section.

MASS STORAGE AND LOCAL MEMORY NAMING CONVENTIONS

The flexible-disk drives and the hard disk drive are collectively referred to as mass storage. This is in contrast to local memory, which consists of RAM (random access memory) on a memory board in the 4170's card cage. (Also, there is fixed memory, ROM (read-only memory) on the Processor board, where firmware codes are stored; these contain the power-up and self-test routines.)

Many CP/M-86 routines allow you to specify a device name for one of the flexible drives or for the optional hard disk drive. Your 4170 names its two flexible disk drives as devices A: and B:; the optional hard disk drive is device C:. (Even if you do not have the optional hard disk, the device name C: is reserved. The device name D:, which is presently unused, is also reserved.)

External mass storage devices (such as the Tektronix 4926, the 4925, and the 4926 Option 25 drives) use device names E: through P:. The exact device name depends on how the device is strapped at installation. (For more information, refer to the instructions in Section 4 for installing the 4170F45 disk interface and to the appropriate device manual.)

SECTION 6
OPERATING INFORMATION

KNOW YOUR TERMINAL

Before using the 4170, read the operator's manual for your terminal. You need to be familiar with the terminal's various operating modes.

FAMILIARIZATION EXERCISES

These exercises focus on the general aspects of system use and give you an overview for operating the 4170 with CP/M-86.

To do these exercises, the operating system must have been properly loaded into the 4170 as explained in Section 5. When directed to enter commands, enter them exactly as shown (you can use either uppercase or lowercase characters). Each character entered at the terminal's keyboard is displayed on the screen. Your input appears on the same line as (and immediately following) the system prompt, A>. The character <CR> means that you should press the RETURN key.

1. Enter the following command:

   STAT USR:<CR>

   This command is one form of the STAT (status) command. It displays a message like the one below, which shows the current user number and all the user numbers for which there are files stored on the disk in Drive A. Notice that on the operating system disk (in Drive A) there are files stored under User Numbers 0 and 15, and that the current User Number is 0.

   ```
   A:Active User:0
   A:Active Files:0 15
   ```

2. Enter the DIR command:

   DIR<CR>

   This command displays the names of the user files on the disk in Drive A.

3. Now enter the DIRS command to see if there are any system files stored under User Number 0:

   DIRS<CR>

   If there are no system files present, DIRS simply displays a message saying "NON-SYSTEM FILES EXIST."

The previous two commands produced a complete list of the files contained under User Number 0 on the operating system diskette. There are two forms of the DIR command because some of the files have the system attribute and some files have the directory attribute. (These attributes are explained in the CP/M-86 Operating System User's Guide.)

4. Recall that the STAT Command in Step 1 indicated that there were one or more files under User Numbers 0 and 15. To obtain a directory of the files under User Number 15, you must first change your current user number to 15; do so by entering the USER command:

   USER 15<CR>

5. Again enter the DIR command. The file that appears (CPM.SYS) is the operating system itself.

6. Return to User Number 0 by entering:

   USER 0<CR>

This is the end of the familiarization exercises. At this point, you may want to refer to the CP/M-86 Operating System User's Guide to begin getting acquainted with more of the operating system's features and commands.

DIFFERENCES FROM STANDARD CP/M-86

If you have used a CP/M-86 operating system before, you'll notice that there are some differences in the 4170's CP/M-86 from the versions you've used. Here is a summary of the differences:

- In standard versions of CP/M-86, the TOD command lets you examine and set the time of day. The TOD command is not included, however, in the 4170 version of CP/M-86.

- Tektronix has added several utilities to the standard set of CP/M-86 commands. They are AR, CONFIG, DEL, FORMAT, RUN, SD, and SETDEV. In the first part of Section 7, these Tektronix-supplied utilities are described for your reference.

- Tektronix also has provided a communications program, called MODEM86, for transferring files between a host and the 4170. Section 5 describes how to get started with host communications, and Section 7 explains MODEM86 in more detail.

Error Messages

Error messages for standard CP/M-86 commands are listed in the CP/M-86 Operating System User's Guide. Error messages for Tektronix-supplied utilities are listed in Section 7, under the descriptions of those utilities.

Section 7

PROGRAMMING INFORMATION

INTRODUCTION

Before reading Section 7, see Sections 5 and 6, which provide an overview and introduction to using the 4170 and the CP/M-86 Operating System. Section 7 provides reference information that programmers may need to write and install software packages for the 4170.

This section includes:

- General information about 4170 device names.
- Descriptions of the utilities (AR, CONFIG, DEL, FORMAT, SD, SETDEV, and RUN) that Tektronix has added to the standard CP/M-86 utilities. The descriptions here include reference information and supplement the standard CP/M-86 command descriptions in the Digital Research manual, the CP/M-86 Operating System User's Guide.
- A description of the MODEM86 utility, which enables communications between the host and the 4170. (Section 5 provides an introduction to using MODEM86; this section provides more detailed information.)
- Information for FORTRAN programmers who want to use FORTRAN-86 on the 4170.
- Information for programmers who want to use IGL (Interactive Graphics Library) on the 4170.
- Information for programmers who want to use GSX-86 (Graphics System Extension).
- Information for programmers who want to use the DTI (Direct Terminal Interface) on the 4170.

GENERAL INFORMATION

MASS STORAGE DEVICE NAMES

The device names for the flexible disk drives are A: and B:; the optional hard disk drive is C:. Device names for external mass storage devices (such as the Tektronix 4926 Hard Disk Drive) are E: through P:, depending on how the device is strapped at installation. (See the instructions for installing the 4170F45 disk interface in Section 4 and the appropriate device manual.) The device name D: is not presently used and is reserved for future use.

If you transport a program written for a 4110 Series terminal to the 4170, 4110 external devices (of the form S0, S1, etc.) map into 4170 external devices as follows:

4110 Series Device	4170 Device
S0	D: (reserved)
S1	C: (optional hard disk)
T0	E:
T1	F:
U0	G:
U1	H:
V0	I:
V1	J:
W0	K:
W1	L:
X0	M:
X1	N:
Y0	O:
Y1	P:
Z0	No assignment
Z1	No assignment

In some cases, you may need to modify the device names used by the program.

LOGICAL-TO-PHYSICAL DEVICE MAPPING

The logical devices for the 4170 are Console (CON:), Auxiliary (AUX:), and List (LST:). Each logical device is assigned or "mapped" to a corresponding physical port on the 4170. The factory default logical-to-physical assignments are CON: mapped to Port 0, AUX: mapped to Port 2, and LST: mapped to Port 1.

There are three ways to change logical-to-physical device mapping: SETDEV, CONFIG, and STAT. Most users will find that SETDEV is the easiest way to assign logical devices to a physical port. (When "show" is specified with SETDEV, you can also view the current assignments.)

A few experienced users or programmers may choose to use STAT in conjunction with the IOBYTE to change logical-to-physical device mapping; those users should refer to the discussion of the IOBYTE in the CP/M-86 Operating System, System Guide, and they should read the following discussion of how the IOBYTE is implemented for the 4170. If, like most users, you intend to use SETDEV and not STAT (in conjunction with the IOBYTE), you may skip the following discussion of the IOBYTE.

One advantage of using SETDEV or CONFIG is that the 4170's CMOS memory stores the logical-to-physical device assignments (along with other parameters) that are set with these utilities; thus, the device assignments are retained even when the 4170 is powered down. If STAT is used to change device assignments, the assignments are lost when the 4170 is turned off.

If you have the 3PPI option, you can assign logical devices to 4170 physical ports 3, 4, and 5. To do this, use SETDEV or CONFIG; do not use STAT with the IOBYTE. (The IOBYTE will not detect the change correctly for the optional 3PPI ports.)

The 4170 logical devices CON:, AUX:, and LST: are different than the standard CP/M-86 logical I/O devices CONSOLE (CON:), READER (AXI:), PUNCH (AXO:), and LIST (LST:), documented in the CP/M-86 Operating System, System Guide. (Instead of the input device READER (AXI:) and the output device PUNCH (AXO:), the 4170 device AUX: is both an input and output device.)

The IOBYTE

To provide some additional flexibility in logical-to-physical device mapping, the IOBYTE has been implemented in the 4170 BIOS (Basic Input/Output System). (The IOBYTE is a location in BIOS that defines how logical devices are mapped to physical devices at any given time.) Each logical device uses a bit field in the IOBYTE, and the value in that bit field (0-3) represents a pseudo-physical device that corresponds to a physical port.

Because AUX: is both an input and output device, Bits 4 and 5 in the IOBYTE bit field are ignored. If you use the CP/M-86 function call "Set IOBYTE," described in the CP/M-86 Operating System, System Guide, only Bits 2 and 3 in the bit field are used for the AUX: device; if you use "Get IOBYTE," Bits 4 and 5 will return the same values as Bits 2 and 3.

These are the default IOBYTE assignments for the logical devices CON:, AUX:, and LST:.

Logical Device	Bit Field	Bit Value	Pseudo-Physical Device	Corresponding Port
CON:	Bits 0, 1	00	TTY:	Port 0 (Terminal)
		01	CRT:	Port 1
		10	BAT:	Port 2
		11	UC1:	HOST port
AUX:	Bits 2, 3 (AXI:)	00	TTY:	Port 0 (Terminal)
		01	PTR:	Port 1
		10	UR1:	Port 2
		11	UR2:	HOST port
	Bits 4, 5 (AXO:)	00	TTY:	Port 0 (Terminal)
		01	PTP:	Port 1
		10	UP1:	Port 2
		11	UP2:	HOST port
LST:	Bits 6, 7	00	TTY:	Port 0 (Terminal)
		01	CRT:	Port 1
		10	LPT:	Port 2
		11	UL1:	Option 09 port

Using STAT to View and Change Device Assignments

You can use STAT DEV: to view the current IOBYTE assignments for the logical devices. For example, when the factory default assignments are in effect, STAT DEV: returns:

```
CON: is TTY:
AXI: is UR1:
AXO: is UP1:
LST: is CRT:
```

Since Bits 0 and 1 of the IOBYTE have a value of 0, the logical device CON: is assigned to the pseudo-physical device TTY:, which corresponds to the 4170 physical port 0 (the terminal port). Because the Auxiliary (AUX:) device is both input and output, the value 2 in Bits 2 and 3 returns UR1: and UP1: for STAT DEV:, which indicates that the corresponding physical port is Port 2.

To use STAT to change logical-to-physical device mapping, assign one of the three 4170 logical devices to a pseudo-physical device. For example, this command assigns the LST: device to the pseudo-physical device UL1:, which corresponds to the Option 09 port:

```
STAT LST:=UL1:
```

In this first example, the value in Bits 6 and 7 of the IOBYTE is set to 3.

To change the assignment of the Auxiliary (AUX:) device, use the CP/M-86 logical device name AXI:. In the next example, the AUX: device is assigned to the Host port:

```
STAT AXI:=UR2:
```

The device connected to the Host port can then be used for both input and output. (To assign the logical device AUX: to a physical port, specify AXI:, not AXO:, because AXO: is ignored.)

STAT VAL: lists the pseudo-physical device names that you can use; refer to the CP/M-86 Operating System Users Guide for a description of STAT VAL:.

TEKTRONIX-SUPPLIED UTILITIES

The following utilities are products of Tektronix: AR, CONFIG, DEL, FORMAT, SD, SETDEV, and RUN. Similar utilities with the same names may exist on other systems, but Tektronix has designed the utilities described here specifically for the 4170.

This section contains reference material only for the Tektronix-supplied utilities; refer to the CP/M-86 Operating System User's Guide for descriptions of the other CP/M-86 utilities. Remember that there is one standard CP/M-86 command -- TOD (Time of Day) -- that is not included with the 4170's version of CP/M-86.

SYNTAX CONVENTIONS

Each description of the Tektronix-supplied utilities includes a syntax line. A box outlines the syntax line and separates it from other information on the utility. For example:

```
AR outdev:[Gnn]=filespec {[options]}
```

Generally the syntax conventions are the same as those used in the CP/M-86 Operating System User's Guide. The conventions are:

- The utility name appears in uppercase letters, although it may be typed in either uppercase or lowercase.
- Parameters (such as "outdev" and "filespec" in the example) appear in lowercase and describe the information to be entered. The colon (:) after "outdev" is a delimiter for a device name.
- Curly braces ({}) are used to indicate that a parameter is optional. The braces are a syntax convention only; **do not enter them.** Square brackets ([]), on the other hand, are entered when you choose to enter an option (they are not entered unless the option is entered).

AR (Archive)

The AR utility makes archive copies of files. Unless it is requested to do otherwise, AR examines a bit of information in the directory of each file and copies the file only if the information reveals that the file has been altered since it was last archived. This allows you to save copies of your work periodically without having to remember which files you may have changed. AR is distributed in the file AR.CMD.

```
AR outdev:{[Gnn]}=filespec{[options]}
```

outdev: is the disk drive on which the files are to be archived. You may include a user number of the standard form [Gnn] (by default, files are archived under the current user area).

filespec names the file or files to be archived. AR accepts ambiguous filespecs (wild cards) represented by asterisks or question marks.

option is one of the following instructions:

A	Archive previously Archived files
D	Archive Directory (non-System) files
Gnn	Archive files only from user area nn
G*	Archive from all user numbers those files for which the filespec is valid, starting with user 0 and continuing through user 15. The destination file retains the same user number as the source file. G* is valid only in the source filespec. Gnn may not be used in the destination filespec when using G*.
H	Help: display help text and exit
N	Archive New (non-Archived) files
Q	Query: tells AR to ask for permission to archive each candidate file
R	Archive Read-only files
S	Archive System files
W	Archive Writable(R/W) files

If you specify no options, AR assumes Directory and New as the defaults, and uses the current drive and user number.

NOTE

If invoked without arguments, AR displays a message describing its syntax and returns control to the operating system.

Examples

Suppose that you had just created four files: TRACE.FOR, FILL.FOR and OUTLIN.FOR under user number 0, and TEST.DAT under user number 1 on diskette A; furthermore, you wish to archive all but FILL.FOR. You could use the G* and Q options like this:

```
AR B:=A:*.*[QG*]
```

AR responds with a message echoing your command and identifying its version number. Before a file is archived, AR asks if you want the file to be archived. You can respond in one of three ways:

- Press the "y" key if you want the file to be archived.
- Press the "n" key if you do not want the file to be archived.
- Press the "g" key if you want AR to go on archiving without querying.

NOTE

If you press any key during file archiving, AR automatically goes into query mode. AR remains in query mode until the operation is done or you press the "g" (go) key.

Discussion

To stop AR before archiving is complete, press CTRL-C.

NOTE

Only AR sets the bit that marks a file as having been archived. REN resets the bit; so do PIP, ED, and other programs that make new copies of the file.

Before copying a file, AR checks the destination device to make sure that there is room for the new file. If it needs more disk space, AR asks you to insert a new disk, continuing only after it verifies that it can copy the entire file successfully.

Error Messages

^Command line syntax error: for help type AR<CR>
A device, filename, or filetype is wrong or absent; or, a G option was given and was not followed by a decimal number 0 through 15 (or by *).

^Device must be A:,B:,...P:
The device name is missing or is not the name of a disk drive. For example, files cannot be archived with LST: as a device name.

^Cannot archive file to itself.
The drive and user number for the destination are the same as the drive and user number for the source. Archive to a different disk or user number.

^User numbers must be 0 through 15.
The characters immediately following the G option specifier must be 0 through 15 or G*. G* cannot be used in a destination filespec.

^Not valid when destination user number specified.
G* was used in the source filespec after Gnn was used in the destination filespec.

^Not Enough Memory.
There is not enough memory for the AR program and the directory of candidate files. Archive a smaller set of files.

^No Files Found.
The filespec given in the command did not match files.

CONFIG (Configure)

CONFIG allows the user to define peripheral devices to be used with the 4170, extending the physical-to-logical device mapping facility provided by the CP/M-86 command STAT. CONFIG is distributed in the file CONFIG.CMD. (The SETDEV utility performs a similar function to CONFIG and may be easier to use; see the description of SETDEV in this section.)

```
CONFIG {<filespec}
```

filespec — Names the file containing the parameters for CONFIG; the filename must be preceded by the left angle bracket (<). You may include a drive specifier; the default drive is the current drive.

CONFIG without a filespec prompts the user for input. Specifying FACTORY.TXT as the filespec configures the 4170 to its factory default values.

Examples

CONFIG<CR>

When CONFIG is invoked without a filespec, it goes through a complete session of prompting for input from the user via the terminal. Figure 7-1 shows the prompts. (Responding to the CONFIG prompts can be rather time-consuming, especially if you want to reconfigure only one device. Use the SETDEV utility, also supplied by Tektronix, instead.)

In this example, CONFIG begins execution and uses the values in the file FACTORY.TXT (when you specify a filename, the left angle bracket (**<**) **must** precede the filename):

CONFIG <FACTORY.TXT <CR>

Because there is no prompting when you specify a file with CONFIG, it executes rapidly.

The responses to the prompts in Figure 7-1 are the same as the values in the file FACTORY.TXT; thus, responding to the prompts as in Figure 7-1 has the same effect as specifying CONFIG with FACTORY.TXT.

Here is a summary of the default values contained in FACTORY.TXT:

CON: - Port 0 (Terminal Port)
LST: - Port 1
AUX: - Port 2

All ports are set as follows:
Baud Rate(Transmit and Receive) - 2400
DC1/DC3(CTRL-S/CTRL-Q) Flagging - No
RTS/CTS (Request to Send/Clear to Send) Flagging - No
DTR (Data Terminal Ready) Flagging - No
ETX/ACK (End of Text/Acknowledge) Flagging - No
Data bits - 8
Stop bits - 1
Parity - Odd (Transmitted from 4170) and Disabled (Received by the 4170)
Boot Drive - A
Default Drive - A

When FACTORY.TXT is used with CONFIG, flagging for the terminal (the logical device CON:) is set to none. You may specify only one kind of flagging at a time with CONFIG. ETX/ACK flagging is a no-op.

NOTE

If you use CONFIG to change the flagging for the terminal, do not select DC1/DC3 flagging if you are going to use applications programs (such as Wordstar, SuperCalc, or Emacs) that use CTRL-S or CTRL-Q; these programs will not work properly if DC1/DC3 flagging is enabled.

NOTE

Although the CONFIG menu does not list it as a valid selection for the baud rate, you may specify "19" to CONFIG's request for a baud rate and thus select 38400 baud.

You can edit the file FACTORY.TXT to create a file for your specific configuration requirements. If you edit FACTORY.TXT to supply your own configuration parameters, first copy the file and rename it. Be sure that you edit the file carefully and that you specify the exact number of characters that CONFIG expects; otherwise CONFIG will not work and you may lose communications between the terminal and the 4170.

Error Messages

If CONFIG prompts for input and it expects a response of Y or N, it checks the character entered and requests reentry if any other character is entered. When CONFIG expects numeric responses, it does not check the values entered; therefore, you must enter numeric values carefully. If you specify a filename to CONFIG and the file contains extraneous or incorrect parameters, the system will not work properly; if this occurs, use the procedure described in Section 5 to restore communications between the terminal and the 4170.

```
A>config

Tektronix CP/M-86 System Configuration Utility
Version 1.0

Enter the Physical Port assigned to the Console.
0 - Terminal Port
1 - Port 1
2 - Port 2
3 - Port 3
4 - Port 4
5 - Port 5
6 - Host
7 - Printer Port
Enter the physical port number. (0-7) ? 0

Baud Rate:
0 - No baud rate for this device
1 - 50 Baud
2 - 62.5 Baud
3 - 75 Baud
4 - 110 Baud
5 - 134.5 Baud
6 - 150 Baud
7 - 200 Baud
8 - 300 Baud
9 - 600 Baud
10 - 1200 Baud
11 - 1800 Baud
12 - 2000 Baud
13 - 2400 Baud
14 - 3600 Baud
15 - 4800 Baud
16 - 7200 Baud
17 - 9600 Baud
18 - 19.2K Baud
Enter the baud rate number. (0-18) ? 13


Flagging:
Do you want DC1/DC3 (control-S/control-Q) flagging. (Y/N) ? n
Do you want RTS/CTS (hardware) flagging. (Y/N) ? n
Do you want DTR (Data Terminal Ready) flagging. (Y/N) ? n
Do you want ETX/ACK flagging. (Y/N) ? n

Number of Data Bits per character:
0 - 5 Data Bits
1 - 6 Data Bits
2 - 7 Data Bits
3 - 8 Data Bits
Enter the number of data bits per character. (0-3) ? 3

Number of Stop Bits per character:
0 - 1 Stop Bit
1 - 2 Stop Bits
Enter the number of stop bits per character. (0-1) ? 0


Parity:
0 - Odd Parity
1 - Even Parity
Enter the number for data parity. (0-1) ? 0

Do you want parity enabled. (Y/N) ? n
```

(continued)

4685-141

Figure 7-1. CONFIG with the Same Values as Factory.Txt.

```
Enter the Physical Port assigned to the List Device.
0 - Terminal Port
1 - Port 1
2 - Port 2
3 - Port 3
4 - Port 4
5 - Port 5
6 - Host
7 - Printer Port
Enter the physical port number. (0-7) ? 1

Baud Rate:
0 - No baud rate for this device
1 - 50 Baud
2 - 62.5 Baud
3 - 75 Baud
4 - 110 Baud
5 - 134.5 Baud
6 - 150 Baud
7 - 200 Baud
8 - 300 Baud
9 - 600 Baud
10 - 1200 Baud
11 - 1800 Baud
12 - 2000 Baud
13 - 2400 Baud
14 - 3600 Baud
15 - 4800 Baud
16 - 7200 Baud
17 - 9600 Baud
18 - 19.2K Baud
Enter the baud rate number. (0-18) ? 13


Flagging:
Do you want DC1/DC3 (control-S/control-Q) flagging. (Y/N) ? n
Do you want RTS/CTS (hardware) flagging. (Y/N) ? n
Do you want DTR (Data Terminal Ready) flagging. (Y/N) ? n
Do you want ETX/ACK flagging. (Y/N) ? n


Number of Data Bits per character:
0 - 5 Data Bits
1 - 6 Data Bits
2 - 7 Data Bits
3 - 8 Data Bits
Enter the number of data bits per character. (0-3) ? 3


Number of Stop Bits per character:
0 - 1 Stop Bit
1 - 2 Stop Bits
Enter the number of stop bits per character. (0-1) ? 0


Parity:
0 - Odd Parity
1 - Even Parity
Enter the number for data parity. (0-1) ? 0

Do you want parity enabled. (Y/N) ? n
```

(continued)

4685-141

Figure 7-1. CONFIG (cont).

```
Enter the Physical Port assigned to the Auxiliary Device.
0 - Terminal Port
1 - Port 1
2 - Port 2
3 - Port 3
4 - Port 4
5 - Port 5
6 - Host
7 - Printer Port
Enter the physical port number. (0-7) ? 2

Baud Rate:
0 - No baud rate for this device
1 - 50 Baud
2 - 62.5 Baud
3 - 75 Baud
4 - 110 Baud
5 - 134.5 Baud
6 - 150 Baud
7 - 200 Baud
8 - 300 Baud
9 - 600 Baud
10 - 1200 Baud
11 - 1800 Baud
12 - 2000 Baud
13 - 2400 Baud
14 - 3600 Baud
15 - 4800 Baud
16 - 7200 Baud
17 - 9600 Baud
18 - 19.2K Baud
Enter the baud rate number. (0-18) ? 13

Flagging:
Do you want DC1/DC3 (control-S/control-Q) flagging. (Y/N) ? n
Do you want RTS/CTS (hardware) flagging. (Y/N) ? n
Do you want DTR (Data Terminal Ready) flagging. (Y/N) ? n
Do you want ETX/ACK flagging. (Y/N) ? n

Number of Data Bits per character:
0 - 5 Data Bits
1 - 6 Data Bits
2 - 7 Data Bits
3 - 8 Data Bits
Enter the number of data bits per character. (0-3) ? 3


Number of Stop Bits per character:
0 - 1 Stop Bit
1 - 2 Stop Bits
Enter the number of stop bits per character. (0-1) ? 0


Parity:
0 - Odd Parity
1 - Even Parity
Enter the number for data parity. (0-1) ? 0

Do you want parity enabled. (Y/N) ? n


Enter the Boot Drive. (A-P) ? a
Enter the Default Drive. (A-P) ? a

Setting System Configuration.
Is that what you want to do (Y/N) ? y
Configuration Started.
Tektronix 4170 CP/M-86 V1.2    Release 1.2
A>
```

4685-141

Figure 7-1. CONFIG (cont).

DEL (Delete)

The DEL utility, like the standard CP/M-86 ERA command, erases files. Unlike ERA, however, DEL can be made to ask your permission before deleting each file in a list. DEL always requires confirmation of blanket file deletion requests (using the ambiguous filespec *.* or ??????.???). DEL is distributed in the file DEL.CMD.

```
DEL {filespec{[GnnQ]},...}
```

filespec — is either a literal name or an ambiguous file name (constructed with the wildcards "*" or "?"). You may include a drive specifier; the default drive is the current drive.

[GnnQ] — is an optional modifier. Gnn specifies the user number from which the file or files are to be deleted (the default is the current user number). Q(Query) tells DEL to ask before deleting each file.

NOTE

If you invoke DEL without arguments, it displays a brief message describing its syntax and returns control to the operating system.

When you use the Query option (Q), DEL finds each file that matches the filespec, displays its name, and asks you with a question mark whether it should be deleted. DEL recognizes these responses:

y — Yes, delete the file and continue.

n — No, keep the file and continue.

g — Go ahead and delete this file or all files matching the filespec without asking.

CTRL-C — Stop and return control to the operating system.

NOTE

If you press any key during file deletion, DEL automatically goes into query mode. DEL remains in query mode until the operation is done or you press the "g" (go) key.

Even without the Query option, DEL asks before deleting Read-Only files and requests confirmation before obeying commands to delete all files (filespecs such as *.* and ??????.???).

Examples

The sequence shown below deletes all but one of the files with the filetype ".bak" under user number zero:

```
A>del *.bak[gOq]
Deleting files:

A:CATALOG.BAK[GO]?y
A:TEACH.BAK[GO]?y
A:COMMANDS.BAK[GO]?n

A>
```

Error Messages

Illegal file name given.
The command included an incorrect filespec or option. Check the filespec and command syntax.

****Insufficient memory to continue.**
Not enough memory is available to the operation system to store the filespecs during processing. If adequate memory is available and the problem persists, reload the operating system.

Filename does not exist.
The file specified in the command does not exist. This message is issued only if you have included an unambiguous but incorrect file name in the command; it is never issued in response to a command to delete an ambiguous filespec.

y/Y	**yes**
n/N	**no**
g/G	**go (and stop asking for confirmation)**
CTRL C	**abort**

DEL prints this message in response to an invalid confirmation. Note that "G" only prevents requests for confirmation for the filespec being processed; if a subsequent filespec on the command line is followed by the "Q" option, requests for confirmation resume.

FORMAT (Format a Disk)

Before you can use either a hard disk or a flexible diskette for the first time, you must format it. To do so, use the FORMAT utility. This utility erases any data present on the disk being formatted, and writes track and sector information on the disk surface for the operating system to use. When FORMAT is followed by a device name, prompts explain how you should continue. FORMAT is distributed in the file FORMAT.CMD.

```
----------------------
| FORMAT {device:}   |
----------------------
```

device: is the drive (A: through P:) containing the disk you want to format. Drives A: and B: are the flexible disk drives; the optional hard disk is drive C:. (External disk drives -- such as a Tektronix 4926, 4925, or 4926 Option 25 -- use device names E: through P:. The exact device name is determined by strap settings at installation.)

NOTE

If you invoke FORMAT without a device name, it displays a brief message describing its function and returns control to the operating system. FORMAT issued with the device name Q: causes a quick-reference syntax message to be displayed.

CAUTION

Formatting a disk destroys any data on the disk. Be sure that the disk does not contain valuable data before you format it.

Formatting Flexible Diskettes

Before you format a diskette in drive A or B, be sure that the diskette in the drive does not contain data you want to save. Type one of these commands:

FORMAT A: (to format the disk in drive A)
FORMAT B: (to format the disk in drive B)

FORMAT asks you to confirm that you do indeed want to format the diskette and thus destroy any data on it. If you confirm that it should go ahead, FORMAT erases and formats the diskette.

Formatting a Hard Disk

As with a flexible disk, FORMAT erases any data on the hard disk and writes the track and sector information that the operating system requires on the disk surface.

In addition, when a hard disk is formatted, FORMAT detects "bad tracks." (Hard disks usually contain a number of tracks that cannot be reliably written to and read from.) FORMAT marks these tracks, and then assigns alternate (normally unused) tracks so that you have the full amount of storage available.

NOTE

The optional hard disk is formatted at the factory and does not need to be reformatted before you use it.

If you format the optional hard disk, FORMAT lets you input the list of bad tracks supplied by the disk manufacturer. If you do not have this list, or suspect that it is in error, FORMAT proceeds and marks the tracks that it cannot verify. (Section 5 describes how to specify bad tracks from the manufacturer's list.)

FORMAT can handle up to 160 bad tracks. If the disk contains more than 160 bad tracks, FORMAT returns an error message. If your hard disk contains more than 160 bad tracks, it should be replaced; contact your local Tektronix field office.

Example

```
A>FORMAT B:<CR>
Formatting drive B:
   5 1/4" Double-Sided Floppy (512 bytes/sector)
          with interleave of 0

Insert a disk in drive B
Format the disk? (Y or N; default N): Y<CR>

Formatting (this takes approximately 30 seconds)...
Formatting completed.

Insert a disk in drive B
Format the disk? (Y or N; default N): N<CR>
A>
```

See Section 5, Getting Started, for an example of formatting a hard disk.

Error Messages

Error messages are standard CP/M-86 error messages and the following:

Check that the disk and drive are not write-protected.
Remove the write-protect tab from the disk, or turn off the drive's write-protect switch. If the disk was not write-protected, formatting a second time may correct the error.

If formatting a second time does not correct the error, the diskette may be bad; try another diskette.

Only drives A: through P: are allowed.
The drive you specified is not in the valid range A: through P:.

Drive not present. Aborting.
The specified drive (A: through P:) is not connected.

Number of bad tracks exceeds 160
The hard disk contains more than 160 bad tracks and should be replaced. Contact your local Tektronix field office.

If either of these two errors occurs, try formatting the hard disk again:

Error - in requesting alternate track.
Error - in writing alternate track.

If the error recurs, there may be a problem with the disk or the disk controller; contact your local Tektronix field office.

RUN (Run .LTL Program)

RUN reads load-time-locatable programs (extension .LTL) from disk files, loads them into memory, and begins executing them. RUN loads ONLY the load-time-locatable object programs created by LINK86 from the output of Intel language compilers such as ASM86 and FORTRAN-86. The names of such files have the extension ".LTL", meaning "load time locatable." RUN is contained in the file RUN.CMD.

```
RUN progname {arguments}{&}{;comment}
```

progname	is the name of the load-time-locatable program file to be loaded and run. The file specifier may include a drive specifier and a user number of the usual form ([Gnn]). The file extension .LTL is assumed.
arguments	is list of any arguments required by the program to be loaded.
&	is an optional continuation character. The ampersand (&) indicates that the command or arguments continue on the next line after a carriage return; this enables you to break long command sequences into as many lines as necessary.
;comment	optional remarks which have no effect on program loading or execution.

Examples

To load and run the load-time-locatable file FORTST.LTL contained under user number 2 on drive B, type:

```
A>RUN B:FORTST[G2]<CR>
```

Long command lines that have too many arguments to fit on a single line may be continued on subsequent lines. To demonstrate how to continue a command line, the following command is arbitrarily divided into several lines. To show how explanatory comments can be added, one is included in this example. Notice how the operating system prompts for continuation lines with double angle brackets (>>).

```
A>RUN FORT86 &
A>>TEST.FOR  ;program listing will appear in TEST.LST
```

Discussion

When used to load FORTRAN-86 load-time-locatable programs, the RUN command may optionally include a "preconnection" statement assigning a logical device or a filename to a FORTRAN logical unit number. See the following discussion of FORTRAN-86 for details.

SD (Sorted Directory)

The SD utility provides an alphabetically sorted directory of files on a specified disk drive. In addition to filenames (which the standard CP/M-86 command DIR also lists), SD displays file sizes, the number of files, and the amount of available disk space. SD displays the directory on the terminal screen, writes it in a file, or sends it to another specified device. SD is distributed in the file SD.CMD.

```
SD {outdev=}{filespec}{[options]}
```

outdev — is an output device (CON, LST, AUX) or disk file (optional drive, filename, optional filetype, optional user number) that receives the directory listing. The terminal is the default output device.

filespec — same as {d:}filename{.filetype}. Ambiguous filenames and filetypes are allowed (using "*" and "?"). If a filename and/or filetype is omitted, it is assumed to be "*".

options — must be preceded by "[" and may be followed by "]". The available options are:

A	list Archived files (see the previous description of AR)
D	list Directory (non-System) files
F	show flags (attributes) of files as they are listed
Gnn	list from user number nn
G*	list from all user numbers starting with 0 and going through 15
H	print a help text
N	list New (non-Archived) files
R	list Read-only files
S	list System files
W	list Writable (R/W) files

Discussion

To be listed, a file must meet all of these requirements:

- Be on the indicated (or default) drive;
- Be in the specified (or current) user area;
- Match the filename and extension pattern;
- Match the attribute flags (ADNRSW) -- match every specified attribute, or one member of each pair of mutually exclusive attributes.

A/N, S/D, R/W are mutually exclusive; the default filename and extension are "*"; if NO attributes are specified, "D" (the Directory attribute) is assumed.

Examples

The following example lists all System and Directory files having the currently active user number (in this case, 1) on the current drive and displays file attributes.

```
A>SD [SDF]
```

SD responds with a message like this:

```
DiskA:,User 01        "SD [SDF]"                    V1.20

prog1.for -SA[4k]   prog2.for RS-[6k]   prog3.for -A[4k]

Total of 14k in 3 files with 2998k space remaining.
```

Suppose that the current user number is 0, and that you wish to send to the listing device the sorted directory of user number 1, including only archived FORTRAN source files (System and Directory). To do this, type the following command (notice that the filename may be omitted):

```
A>sd lst:=.for[ag1]
```

The listing device displays a message like this:

```
Disk A:,User 01     "SD LST:=.FOR[AG1]"          V1.20

prog1.for[4k]    prog2.for[4k]

Total of 14k in 3 files with 2998k space remaining.
```

SETDEV (Set Device)

SETDEV allows the user to check and selectively change the logical-to-physical device assignments and to reconfigure device parameters. Unlike CONFIG, which requires you to specify all parameters for all devices, SETDEV allows you to specify as few parameters as you want for a specific device. Also unlike CONFIG, SETDEV allows you to view the current configuration settings. SETDEV is distributed in the file SETDEV.CMD.

SETDEV has two syntax forms, which follow. Most SETDEV arguments can be abbreviated to as few as one or two characters; in the discussion of SETDEV here, the characters in bold **UPPERCASE** show how you may abbreviate each argument.

```
SETDEV logdev{=physdev}, {parameter1=setting},
          {parameter2=setting},...{SHow}

or

 SETDEV {Query}
```

logdev — is a logical device name. The valid logical device names are **Con**sole, **Aux**iliary, and **Lst** (or **Lst**). The logical device name must precede any parameter specifications; the only time it may be omitted is when SETDEV is issued with only "show" or "query". A single SETDEV command allows only one logical-to-physical device assignment.

physdev — is a physical device port. The ports can be the standard 3PPI ports (Ports 0-2), the optional 3PPI ports (Ports 3-5), the Host port, and the Option 09 port. An error is reported if you specify a port that does not exist on your 4170. The valid port names are:

PORT0 or P0
PORT1 or P1
PORT2 or P2
PORT3 or P3
PORT4 or P4
PORT5 or P5
Host
Option9 (or Opt9), or Parallel

parameter — is a communications parameter or a drive specification being changed. Table 7-1 lists the parameters and shows how they may be abbreviated. An equals sign (=) separates the parameter from its setting; commas (and optionally, spaces) separate one parameter specification from another.

setting — is a valid value for the communications parameter being changed. Table 7-1 lists valid settings for each parameter, and shows how they may be abbreviated.

SHow — displays a list of parameter settings for all devices. If "show" is included with parameters to be changed, SETDEV first shows how the parameters will look once they are changed. "Show" can be specified without specifying a logical device.

Query — causes SETDEV to prompt for parameter changes.

NOTE

If you invoke SETDEV with no arguments, it displays a brief help message and returns control to the operating system.

Table 7-1

PARAMETERS AND SETTINGS FOR SETDEV

Parameter	Valid Settings
Baud	50, 62.5, 75, 110, 134.5, 200, 300, 600, 1200, 1800, 2000, 2400, 3600, 4800, 7200, 9600, 19200, or 38400
Parity	Odd, Even, or None
Databits	5, 6, 7, or 8
Stopbits	1 or 2
Flagging	None, Xon, Dtr, or Rts/cts
DEfaultdrive	A:, B:, ... P:
BOotdrive	A:, B:, ... P:

NOTE

"Xon" flagging is DC1/DC3 flagging. If you are going to use applications programs that use CTRL-S or CTRL-Q (such as Wordstar, SuperCalc, or Emacs), do not select Xon flagging for communications between the 4170 and the logical device Console (the terminal); these applications programs may not work properly if Xon (DC1/DC3) flagging is enabled.

Only one kind of flagging may be specified with SETDEV at a time.

Discussion

SETDEV allows you to selectively change communication parameters, as opposed to CONFIG, which requires you to reenter all parameters. SETDEV also lets you display the current parameters with the optional argument "show".

When the second syntax form is used:

```
A>SETDEV query <CR>
```

SETDEV displays a menu and prompts you for changes. Figure 7-2 shows the menu that **SETDEV query** issues and illustrates how to change device parameters by responding to the prompts. Note that you can press <CR> to leave a parameter unchanged in response to the prompting.

Examples

Suppose that you are adding a printer, that you are connecting it to Port 1, and that you want it to be the logical device LST:. After connecting the printer, use SETDEV to assign the printer to the list device and to change the required communication parameters (with the first syntax form, parameters not specified retain their values). Note that the logical device must precede any parameter specifications.

```
A>SETDEV lst=port1, ba=300, pa=none <CR>
```

After you press the <CR>, SETDEV responds with this message:

```
SETDEV CP/M-86 Device Configuration Utility V1.0
Setting new configuration....
```

SETDEV can also be used to change the default drive (the drive selected after the operating system is booted) and the boot drive. To change the default drive, enter:

```
A>SETDEV def=C: <CR>
```

Valid boot and default device names are A: through P:. Device names A: and B: refer to the flexible disk drives, while device name C: refers to the optional hard disk drive. (The device name D: is presently unused and is reserved for future use.) Device names E: through P: refer to external devices, such as the Tektronix 4926 Hard Disk Drive; the exact device name depends on strap settings that are made during installation. SETDEV reports an error if the device name specified is not in the valid range A: through P:. If the device specified (from A: to P:) is not connected, however, SETDEV does not report an error and uses drive A: instead.

```
A>setdev query
SETDEV CP/M-86 Device Configuration Utility V1.0

SETDEV Query Mode.


Enter one of the following (may be abbreviated to a single letter):
  - SHOW to display the current configuration
  - CONSOLE, AUXILIARY, or LIST to change parameters for that device
  - DEFAULTDRIVE to change the drive selected after CP/M-86 is booted
  - BOOTDRIVE to change the drive used to load CP/M-86
  - KEEP to quit SETDEV and keep the changes
Press CTRL-C (^C) to quit without saving any changes.

Enter choice and press RETURN -> aux


The AUXILIARY device is currently assigned to PORT2.

Choose:
   PORT0, PORT1, PORT2, ...PORT5, HOST, or OPTION9
and press RETURN, or just press RETURN for PORT2 -> p3


The current baud rate for the AUXILIARY device is 4800.

Choose:
   50, 62.5, 75, 110, 134.5, 150, 200, 300, 600, 1200, 1800, 2000,
   2400, 3600, 4800, 7200, 9600, 19200, 38400
and press RETURN, or just press RETURN for 4800 -> 9600


The current parity for the AUXILIARY device is NONE.

Choose:
   NONE, ODD, or EVEN
and press RETURN, or just press RETURN for NONE ->


The number of databits for the AUXILIARY device is 8.

Choose:
   5, 6, 7, or 8
and press RETURN, or just press RETURN for 8 ->


The number of stopbits for the AUXILIARY device is 1.

Choose:
  1 or 2
and press RETURN, or just press RETURN for 1 ->


The flagging mode for the AUXILIARY device is XON/XOFF.

Choose:
   NONE, XON/XOFF, RTS/CTS or DTR
and press RETURN, or just press RETURN for XON/XOFF ->


Enter one of the following (may be abbreviated to a single letter):
  - SHOW to display the current configuration
  - CONSOLE, AUXILIARY, or LIST to change parameters for that device
  - DEFAULTDRIVE to change the drive selected after CP/M-86 is booted
  - BOOTDRIVE to change the drive used to load CP/M-86
  - KEEP to quit SETDEV and keep the changes
Press CTRL-C (^C) to quit without saving any changes.

Enter choice and press RETURN -> keep

Setting new configuration...                                    4685-133
```

Figure 7-2. An Example of SETDEV Query.

SETDEV specified only with "show" displays the current settings for all devices (Figure 7-3 shows a typical display):

A>SETDEV show <CR>

In the next example, a parameter to be changed is specified along with "show":

A>SETDEV lst, ba=300, pa=none, show <CR>

SETDEV displays the new configuration, with the baud rate and parity revised to the specified values.

```
C>
C>setdev show
SETDEV CP/M-86 Device Configuration Utility V1.0

Current CP/M-86 Device Configuration:

CONSOLE device is PORT0
        baudrate....2400
        parity......NONE
        databits....8
        stopbits....1
        flagging....XON/XOFF

AUXILIARY device is PORT2
        baudrate....2400
        parity......NONE
        databits....8
        stopbits....1
        flagging....XON/XOFF

LIST device is PORT1
        baudrate....300
        parity......NONE
        databits....8
        stopbits....1
        flagging....XON/XOFF

Defaultdrive....C:
Bootdrive.......A:

C>
```

4685-145

Figure 7-3. SETDEV Show.

Error Messages

setdev: can't understand 'xxxxx'.
The SETDEV command did not understand the string 'xxxxx'. Either you mistyped the string or it is an invalid parameter.

setdev: a CP/M logical device must be specified first.
A logical device name must be specified before a physical port.

setdev: physical device not available: 'xxxx'.
The specified physical device does not exist in the 4170. See the list of valid physical device ports above.

setdev: '=' and argument required after 'xxxx'.
The parameter 'xxxx' must be separated from its argument by an equals sign (=).

setdev: invalid baud rate.
Refer to Table 7-1 for valid baud rates.

setdev: invalid parity type.
The valid parity types are ODD, EVEN, and NONE.

setdev: invalid number of stopbits.
The valid values are 1 or 2.

setdev: invalid number of databits.
The valid values are 5, 6, 7, or 8.

setdev: invalid flagging data.
The valid values for flagging are NONE, XON, DTR, or RTS/CTS.

setdev: invalid drive specified.
Valid drive specifications for the boot and default drives are A: through P:.

HOST COMMUNICATIONS AND THE MODEM86 UTILITY

The MODEM86 utility is used to establish communications between the 4170 and a host computer. Section 5 explains how to get started using MODEM86. If have not used MODEM86 before, follow the instructions in Section 5 under the heading Connecting the 4170 to the Host.

Tektronix provides several preconfigured versions of the MODEM86 program on the MODEM86 diskette. The different versions are MODM86-H.CMD, MODM86-1.CMD, MODM86-2.CMD, MODM86-3.CMD, MODM86-4.CMD, and MODM86-5.CMD.

WHICH PROGRAM TO USE

The program to run depends on the 4170 port to which your host is connected. For example, if the host is connected to the HOST port on the 4170, you run the program MODM86-H. Refer to the list below to determine which MODEM86 program to run.

Port	Program
HOST Port	MODM86-H
Port 1	MODM86-1
Port 2	MODM86-2
Port 3	MODM86-3
Port 4	MODM86-4
Port 5	MODM86-5

(The file README.1ST on your MODEM86 diskette also describes the MODEM86 programs and their differences.)

The simplest way to run the MODEM86 program is to just type the program's file name. If the host is connected to the host port, for example, enter:

A>MODM86-H<CR>

(Note that you must type **MODM86-H**, not MODEM86-H.)

MODEM86 starts by displaying the following menu options:

```
P - Purge (delete) present file (from terminal mode)
W - Write file to disk (from terminal mode)

C - Call number (intelligent modem only)
D - Disconnect (hang up phone)

T - Terminal mode in full-duplex
H - Terminal mode in half-duplex
E - Terminal mode with echo

R - Receive a file
S - Send a file

M - Menu
X - Toggle expert mode (menu on/off)
? - Help, command syntax description

F - File data display
L - List directory
V - Current value display

Q - Quit, exit to operating system
```

The menu options are described later in this document. Some menu options may be missing from the menu at any given time, because certain options only make sense at certain times.

After displaying the menu, MODEM86 then indicates the current default disk drive and prompts for a command as follows:

```
Default drive: A
Enter command: _
```

ENTERING MODEM86 COMMANDS

MODEM86 expects a command from the main menu, followed by a <CR>, whenever this prompt appears:

Enter command: _

Generally, the command contains a menu option that may be preceded by a drive specification and may be followed by suboptions and a file specification. Here is the syntax:

{d:}{m{sssssss}{.baud}{filespec}}

where:

d:	if specified, d: becomes the default disk drive
m	m is a menu option from the MODEM86 main menu; the menu option may be omitted only when you are changing the default drive
s	up to seven suboptions may be included with an option from the main menu (See Overview of Command Suboptions, which follows.)
baud	baud rate in bits per second
filespec	indicates the name of a conversation save file or the name of a file to be transferred. When you use the M (Multifile) suboption, you may include more than one filespec to indicate which files will be sent.

Lowercase letters used in a command are translated into uppercase.

The "?" main menu option may be used to display a command syntax description and a list of suboptions (suboptions are discussed later in this section):

Enter command: ?<CR>

At the conclusion of each command, MODEM86 redisplays the menu and default drive and prompts for another command. You can change the default drive as follows:

Enter command: B:<CR>

You can change the default drive and display the help text with one command:

Enter command: B:?<CR>

The following example shows how suboptions and a baud rate may be included with a command:

Enter command: VN1.9600

The suboptions are N (no parity) and 1 (1 stopbit) and the baud rate is changed to 9600 bits/second. The V option (from the main menu) displays the changed communication parameters.

The baud rates that may be specified are 110, 300, 600, 1200, 2400, 4800, 9600, and 19200. The baud rate used must be the same as the modem baud rate.

The default communication parameters (such as the baud rate) are set by the MODEM86 program that you invoke (MODM86-H, MODM86-1, etc.) For example, the MODM86-H program defaults to a baud rate of 1200 bits/second. To change the defaults (for baud rate, parity, stopbits, etc.), you must use the MODEMSET program; see Changing Preconfigured MODEM86 Programs, later in this section.

NOTE

The MODEM86 program overrides (but does not change) the baud rate and other communication parameters set with the CONFIG or SETDEV utilities. When you exit MODEM86, the parameters set with CONFIG or SETDEV are still in effect.

Subsequent commands use the same communications settings unless the settings are otherwise specified. Answer or Originate mode (the 'A' suboption or the 'G' suboption, respectively) also stays in effect until the mode is specified again. If the baud rate needs to be changed, it may be done without changing the mode. For instance, if MODEM86 is in Originate mode at 300 bits/second, then the command **R.600 NAME.TYP** will allow MODEM86 to receive a file at 600 bits/second in Originate mode. If the Answer/Originate mode needs to be changed, then that may be done without selecting the baud rate again.

A main menu option may be specified with the MODEM86 program name, in a manner similar to specifying file names for CP/M-86 programs. For example, the following program call displays the help text before displaying the menu and prompting for a second command:

```
A>MODM86-H ?<CR>
```

The following command puts MODEM86 in expert mode immediately, and thus avoids displaying a menu even before the first command prompt:

```
A>MODM86-H X<CR>
```

In this manner, MODEM86 does not start by displaying the main menu. Instead, it goes directly into the mode specified by the program command. When that task is completed, MODEM86 displays the main menu.

Overview of the MODEM86 Main Menu Options

Here is a summary of the options (in alphabetical order) from the MODEM86 main menu. Several options require an "intelligent" modem, such as a Hayes modem, for the menu option to work.

C used to call the host using an autodial modem connected to the serial port. (Requires intelligent modem.)

D used to disconnect from the host. (Requires intelligent modem.)

E used to enter terminal mode with local echoing.

F used to display a file or set of files on the console. There may be more than one filespec and each filespec may be ambiguous. The contents of all files with matching names are displayed on the console along with the file size and name. If expert mode is off, then each file is separated by a pause. CTRL-S may be used to temporarily stop the display at any time; CTRL-Q starts it again. MODEM86 then waits for another character to be typed before it continues. CTRL-C may be used to terminate the display at any time.

H used to enter terminal mode in half-duplex. This is useful for talking to a smart modem or for hosts that require helf-duplex mode operation.

L lists the directory of the current default disk drive if no filespecs are added. There may be more than one filespec and each filespec may be ambiguous. If any filespecs are included, matching file names are displayed.

M used to get another menu. This is useful when expert mode is on.

P used to purge (delete) the conversation save file.

R used to receive a data file from the host computer using an error-checking protocol. (See Error-Free File Transfer, later in this section.)

S used to send a data file to the remote computer using an error-checking protocol. (See Error-Free File Transfer, later in this section.)

T used to enter terminal mode in full-duplex, without local echoing.

Q used to quit (that is, exit) the program. If there is a conversation save file open, MODEM86 does not allow the use of Q to return to the operating system until either the W menu option or the P menu option is used to close the conversation save file. (This prevents data from being accidentally lost.)

V used to display the current values of various settings which are remembered from command to command. This includes the bit rate, modem mode, parity, number of stop bits, and the conversation save file name.

W used to write out the conversation buffer to the conversation save file, and to close the file.

X toggles expert mode on and off. In expert mode the menu is not displayed prior to asking for the next command.

? prints help text on the terminal.

Some of these main menu options are described in more detail in a following discussion, Transferring Files Using Terminal Mode.

Overview of Command Suboptions

You may specify up to seven suboptions from this list after an option from the main menu:

A The A (answer mode) suboption is used to put an intelligent modem into answer mode. Once answer mode has been selected, any subsequent commands use the same mode. (Requires intelligent modem.)

B The B (batch) suboption is used to prevent the program from going into menu option mode after the command. Instead, MODEM86 exits to the operating system.

C The C (clear space parity) suboption is used to select 7-bit characters with an additional parity bit, which is set to zero. The parity is not checked with this option.

D The D (disconnect) suboption is used to disconnect from the host by hanging up the phone, or turning off the handshaking signals after the menu option is completed. This is done after the T suboption if both are chosen. This is similar in function to the D menu option. (Requires intelligent modem.)

E The E (even parity) suboption is used to select 7-bit characters with an additional even parity bit for serial I/O.

F The F (file data viewing) suboption is used to view file data as it is received.

G The G (originate mode) suboption is used to put the modem board into originate mode. Once originate mode has been selected, any subsequent commands use the same mode. (Requires intelligent modem.)

H The H (half-duplex) suboption is used to return to half-duplex terminal mode after another menu option is completed.

M The M (multifile) suboption is used only with the R and S menu options to indicate a multifile transfer.

N The N (no parity) suboption is used to select 8-bit characters with no parity for serial I/O.

O The O (odd parity) suboption is used to select 7-bit characters with an additional odd parity bit for serial I/O.

Q The Q (quiet) suboption is used to keep any MODEM86 messages from being displayed. This can speed up file copies at higher bit rates, or sometimes make file copies at higher bit rates work. This is because writing the error messages can take long enough to miss characters on the serial port. When characters are missed, the data must be re-sent, but usually the results of resending are the same. When this happens, quiet mode can keep MODEM86 from missing the data, since it doesn't print the message. This option also suppresses many prompts for verification.

R The R (redial) suboption, when you are using an intelligent modem, continuously redials the phone number until there is an answer. The user may abort the redialing by typing a CTRL-X. This is only used with the C menu suboption. (Requires intelligent modem.)

S The S (set mark parity) suboption is used to select 7-bit characters with an additional parity bit, which is always set to one. The parity is not checked with this suboption.

T The T (terminal mode) suboption is used to return to full-duplex terminal mode automatically after another menu option is completed.

V The V (view all data) suboption is used to view all file data as it is transferred as well as any extra characters used for headers and checksums.

X The X (extended checksum) suboption makes the file copy checksum a 16-bit CRC-16 checksum. This is more reliable than the parity checksum otherwise used.

1 The 1 suboption is used to select 1 stop-bit for each serial character to and from the serial port. This is the default.

2 The 2 suboption is used to select 2 stop-bits for each serial character to and from the serial port.

TRANSFERRING FILES USING TERMINAL MODE

To transfer files between the 4170 and the host (without error-checking protocol), you can use the T (Terminal mode) option from the MODEM86 main menu.

In Terminal mode, characters typed at the keyboard are sent to the host computer, which is expected to echo them back to the MODEM86 program. Characters received from the host, including echoed characters, are displayed on the screen.

T Menu Option -- Terminal Mode in Full-Duplex

The T menu option may be used with or without a file name. If a file is specified (it should be a new file), then this enables the conversation save feature. Conversation save files are used for capturing files transmitted from the host to the 4170.

A list of all special control characters is given at the start of Terminal mode. The list may include these characters (what actually appears varies according to what was specified with T):

CTRL-B sends a break on the serial line. This is usually used to get the attention of the host computer or to abort something.

CTRL-D causes the program to turn off all the modem handshaking signals it can, and do as much as possible to disconnect from the host.

CTRL-E exits Terminal mode.

CTRL-L is used as a literal-next indication. The next character typed is sent to the host computer without being interpreted as a special MODEM86 command. This is used to send the host control characters (such as CTRL-B).

CTRL-P purges the conversation save buffer. (Note that it purges the buffer, not the conversation save file.)

CTRL-Q causes host output to continue after it has been halted with CTRL-S.

CTRL-R allows the baud rate to be changed without changing other parameters. The program asks you for the new baud rate.

CTRL-S halts host output.

CTRL-T is used to initiate a file transfer from the 4170 to the host. (See the discussion Transferring Files to the Host, later in this section.)

CTRL-Y toggles conversation saving on and off.

If these particular control characters are inconvenient for some reason, their functions can be reassigned to other control characters with MODEMSET. (See Changing Preconfigured MODEM86 Programs, later in this section.)

Transferring Files From a Host Computer

Section 5 describes the procedure for using the T (Terminal mode) menu option and the conversation save feature to capture files transmitted from the host. (Before reading the discussion here, refer to the the heading Transferring Files from the Host in Section 5.)

When the conversation save feature is active, anything received by the modem is saved in memory and may later be written to the conversation save file. The conversation save feature is toggled on and off by typing a CTRL-Y. A colon (:) is printed at the beginning of each new line when the conversation save feature is active. The colon is not transmitted over the modem nor will it be saved in memory. If you have not entered a filename with the T (Terminal mode) since the last W (Write) or P (Purge), then the conversation save feature cannot be activated.

If the conversation save memory buffer is full, the contents are automatically written to the file specified in the T command, but the file is not closed. Communications will then continue with the buffer reinitialized. Your host computer should accept CTRL-S and CTRL-Q (for the Host Output Pause and Host Output Continue functions, respectively) or data may be lost during the file write. If your host uses different characters for these functions, you can use MODEMSET to change the characters to be compatible (see Changing Preconfigured MODEM86 Programs, later in this section). On small files that do not fill up the memory buffer, this will not be a problem, even if your host computer does not use CTRL-S and CTRL-Q.

If errors are encountered while writing out the contents of the conversation save buffer to the disk, MODEM86 closes the conversation save file after writing as much as possible. Then it indicates that the disk may be removed, if desired, and a new disk inserted. Finally, it asks for the name of a new conversation save file. Thus, even if the conversation file is too big for one disk, it can be saved. (This may also happen after you use the W main menu option to write and close the conversation save file.)

While in Terminal mode, a CTRL-P can be used to purge the conversation save buffer. This empties just the buffer, not the entire conversation save file. MODEM86 asks if this is really intended, since CTRL-P may have been entered by mistake and it destroys data. You may decide to purge the conversation save buffer if, after saving something to the buffer, you discover that it should not be saved.

Use CTRL-E to exit from the Terminal mode and enter the Menu Option mode. This is normally done when communications are over, but there are other times when exiting Terminal mode is desirable.

The conversation save file must be closed after exiting Terminal mode using the W menu option. If this is not done before exiting MODEM86, all saved conversation data is lost. The file is not closed automatically because there are times you may want to leave Terminal mode using CTRL-E, do another menu option, and then reenter Terminal mode and continue saving to the same file. To reenter Terminal mode in this fashion, use the T menu option again with no file name. You may reenter Terminal mode (with the conversation saved in the same file) as many times as required, as long as you do not close the save file with the W menu option.

Transferring Files to a Host Computer

Section 5 explains how to transfer files to the host from the 4170. (See the heading Transferring Files from the 4170 in Section 5 before reading the following discussion.)

Briefly, transferring files to the host involves using Terminal mode and CTRL-T; while in Terminal mode, CTRL-T allows MODEM86 to send an ASCII (as opposed to binary) file over the serial connection to the host computer. You can also use CTRL-T to send common sequences of commands to the host.

This type of transfer does no error-checking; there is no protocol specified between the MODEM86 program and the host other than that the host should be ready to receive data via the serial connection. If the host computer sends a CTRL-S (X-OFF), then MODEM86 stops sending until it receives a CTRL-Q (X-ON). (If your host uses different characters for these functions, you can use MODEMSET to change the characters to be compatible -- see Changing Preconfigured MODEM86 Programs, later in this section). Since host computers normally echo data being input and binary files can have the X-ON and X-OFF bytes in them, transfers using CTRL-T in Terminal mode are really only safe for ASCII (character) files. (However, the MODEM86 utility program BIN2HEX can be used to convert a binary file into a character file prior to transfer; see Conversion of Binary Files, later in this section.)

In most cases CTRL-T is preceded by a host command that captures the data that will be sent into a host file. MODEM86 asks for the name of the file to transfer and whether line feeds should be sent after carriage returns at the end of each line. Typing a CTRL-X during a simple file transfer cancels the transfer.

MODEM86 estimates how long it will take to transfer the file to the host. The estimate is a minimum, based on the file size and the baud rate. This minimum will probably be exceeded by at least a little bit, because of time taken to do disk reads and the delays caused by the host sending a CTRL-S (X-OFF).

If you specify the F or V suboptions with the T option, the terminal displays the host's responses to file data (including host echo of the file data, if any). If you do not specify either F or V, then no host characters are displayed during the file copy. In general, it is suggested that you choose either the F or V suboption, since error messages from the host are then displayed; otherwise, you will not see these messages. Displaying the host's responses also slows down the file data transfer at line endings and allows the host more time to digest each line.

You may also use the CTRL-T simple file transfer for things like autodialing and autologon. If you use an autodial modem, then you may create a file with appropriate commands in it, and later specify that file in a CTRL-T simple file transfer command. This results in the file being sent to the modem, which takes the appropriate actions according to the commands. Similarly, you can create a file with what is needed to log-on to a host. You can use the CTRL-T simple file transfer, then, to send the commands to the host, as though you had typed them in.

H Menu Option -- Terminal Mode in Half-Duplex

The H (Terminal Mode in Half-Duplex) option enables terminal mode in half-duplex. This mode is useful to talk to an intelligent modem; when the automatic dialing commands is used, for example, the modem's response echoes on the screen. Also, some host systems may require half-duplex terminal mode.

E Menu Option -- Terminal Mode With Echoing

The E (Terminal Mode with Echoing) option is useful when the 4170 is required to communicate with another terminal or computer running MODEM86. One terminal should be in Terminal mode with Echoing, and the other terminal should be in Terminal mode.

Terminal mode with echoing is similar to regular Terminal mode. The difference is that when you type characters at your terminal, you will see the characters on your terminal's screen and they will "echo" or appear on the screen of the other terminal. Characters typed at the other terminal will also appear on your screen.

W Menu Option -- Write and Close Conversation File

The W option must be used after leaving Terminal mode. This writes the last conversation save memory buffer to the disk and closes the conversation save file. If there is a conversation save file open, you cannot return to the operating system (the Q option) until either the W option or the P option is used to close the conversation save file. This prevents data from being accidentally lost.

If errors are encountered while writing out the contents of the conversation save buffer to the disk, MODEM86 closes the conversation save file after writing as much as possible. Then it indicates that the disk may be removed and a new disk inserted. Finally, it asks for the name of a new conversation save file in which it saves the remainder of the conversation save buffer. Thus, even if the file conversation is too big for one disk, it can be saved.

P Menu Option -- Purge Conversation File

The P (Purge) option erases (deletes) the most recent file accessed in Terminal mode. This is useful when, after communications, you decide that the conversation did not need to be saved in the file. If there is a conversation save file open, MODEM86 does not allow you to use the Q option to return to the operating system until either the W option or the P option closes the conversation save file.

Examples

To enter Terminal mode, to save communication in the file save.txt, to use the default baud rate, and to specify Originate mode:

Enter command: **TG save.txt**

Once in Terminal mode, type CTRL-Y to start saving, CTRL-E to exit Terminal mode, and W to write and close the file save.txt.

To act as the host computer to another computer that is using the above command, with no saving:

Enter command: **EA**

To write out the rest of the save buffer to the file save.txt (specified in the first example):

Enter command: **W**

To purge or delete the file save.txt rather than saving it:

Enter command: **P**

The next example is the same as the first example, but there is no intelligent modem. The modem is set to originate by hand, and the baud rate is 1200 bits/second:

Enter command: **T.1200 save.txt**

ERROR-FREE FILE TRANSFER

S Menu Option -- Send File Mode
R Menu Option -- Receive File Mode

The S (send file) and the R (receive file) menu options are used to do error-free file transfers between two 4170s, between a 4170 and another system running MODEM86, or between a 4170 and a system running a MODEM86-compatible program such as XMODEM. (Many free-access bulletin board systems use MODEM86 or a MODEM86-compatible program, enabling you to transfer data between those systems and the 4170.)

To do error-free file transfers between two systems, one unit runs MODEM86 and uses the S option to send the file data, while the other unit runs MODEM86 and uses the R option to receive the same file data.

A special protocol (a procedure to follow for sending and receiving the data) is used to ensure that the file is received intact without the possibility of lost or changed data, which often happens in file transfers without a protocol.

It is not necessary to understand the protocol to use the S and R options; it is necessary only to understand how to give the commands that start the transfer to the program. The S and R menu options normally require one filename to be specified -- the name of the file to be sent or received.

Using the M (Multifile) suboption, you can specify more than one file and/or ambiguous filenames to designate the files to be transferred. To send files, use the S option and the suboption M (along with any other suboptions and a baud rate, if desired). To receive the files being sent, use the R menu option and the M suboption. For multifile transfers, you must not name the received files since filenames are sent by the sending program, but you may specify a disk drive (otherwise, the files are written to the default drive). Existing files with the same name as a received file are deleted.

For the S menu option, MODEM86 estimates how long it will take to transfer the file. The estimate is a minimum, based on the file size and the baud rate. This minimum will probably be exceeded by at least a small amount because of the time taken to do disk reads, delays caused by the other unit, and possible resending of blocks in the event of errors. For the R menu option, MODEM86 cannot estimate the duration of the file transfer because the size of the file is not known until the file is received.

Examples

To send the file stuff.txt with the modem in the Originate mode at the current baud rate:

Enter command: **SG stuff.txt**

To send another file (more.txt) with the same modem mode and the same baud rate:

Enter command: **S more.txt**

To receive the file junk.txt on drive B (at the new baud rate of 600 with the modem in Answer mode), to view just the file data being received, and to return to Terminal mode when done:

Enter command: **RAFT.600 B:junk.txt**

To send all the ".com" files, with no messages to be displayed, at the current baud rate and with the current modem mode:

Enter command: **SMQ *.com**

To send two files (afile.txt and bfile.txt) from two different drives with one command:

Enter command: **SM A:afile.txt B:bfile.txt**

To receive on drive A the two files sent by the previous example (afile.txt and bfile.txt):

Enter command: **RM A:**

Note that for the R command to work the baud rates must be the same and the modem modes must be the opposite. Because the above example uses R with the multifile suboption (M), no file names are specified.

To receive the file safe.com on drive B using the Extended (CRC-16) Checksum mode at a new baud rate of 1200 bits/seconds:

Enter command: **RX.1200 B:safe.com**

To send all the ".com" files from disk B:, use the Originate mode (intelligent modem only) at 600 bits/second, and return to Terminal mode when done:

Enter command: **SMGT.600 B:*.com**

To receive all the files being sent from the previous example on the default drive using Answer mode (intelligent modems only) at 600 bits/second:

Enter command: **RMA.600**

To send asm.com from the default drive, all the ".hex" files from the default drive, and all the files starting with "m" from the drive B:

Enter command: **SM.600 asm.com *.hex B:m*.***

CHANGING PRECONFIGURED MODEM86 PROGRAMS

The original MODEM86 program was designed to enable host communications on several types of computers, including the Tektronix 4170. The MODEMSET program was used to preconfigure to create the preconfigured 4170 versions of MODEM86 (MODM86-H, MODM86-1, etc.). To create these different versions, these parameters were specified to MODEMSET:

(1) Computer - Tektronix 4170

The MODEM86 communications port depends on the particular MODEM86 program. Here is a cross-reference of the preconfigured MODEM86 programs, the MODEMSET communications port number, and the corresponding 4170 physical port. (Note that MODEMSET communications port 2, which is reserved for OEM use, corresponds to Port 0, the terminal port, on the 4170.)

Preconfigured Program	MODEMSET Communications Port	4170 Physical Port
MODM86-H	1	HOST Port
MODM86-1	3	Port 1
MODM86-2	4	Port 2
MODM86-3	5	Port 3
MODM86-4	6	Port 4
MODM86-5	7	Port 5

(2) CPU and Memory
CPU Clock - 5MHz
Data Path Width - 16 bits
Wait States - 1

(3) Serial I/O Port Hardware
Serial I/O Device - For MODM86-H, the device is the Tektronix Host Port; for MODM86-1 thru MODM86-5, it is the INS 8250 ACE.
Serial I/O Device Memory Mapped - No
Serial I/O Device Base Address (in hex) - E0
Serial I/O Register Separation - 02H
Bit Rate Generator Clock Input - 2458 kHz
Serial Interrupt Enable Modified - Yes
Serial Receive Character Interrupt Enabled - Yes
Interrupt Controller Base Address (in hex) - E8
Interrupt Controller Register Separation - 02H
Interrupt Controller Base Vector Number - 80
Serial Receive Character Interrupts
(IRQ number in hex) - 1
Interrupt Routine Checked - Yes

(4) Terminal Mode Control Characters

CTRL-B	Break
CTRL-D	Disconnect
CTRL-E	End Terminal Mode
CTRL-L	Literal
CTRL-P	Purge conversation buffer
CTRL-R	Bit Rate Change
CTRL-T	File Transfer
CTRL-Y	Toggle Conversation Saving
CTRL-S	Host Output Pause
CTRL-Q	Host Output Continue

(5) Default Settings Group

Baud Rate - 1200 bits/second
Minimum Timeout Period - 0 seconds
Parity - None
Stop Bits - 1
Expert Mode - No
Use of Bells - Yes
Host Requires Line Feeds - No
XOFF/XON's Sent Automatically - Yes

NOTE

It is recommended that you only change terminal mode control characters or default settings listed in (4) and (5).

You can make changes to any of the preconfigured 4170 versions using the following procedure. (It is suggested that you copy the MODEM86 diskette first and only make changes to the copy.) The diagram in Figure 7-4 outlines the procedure. Notice how you may repeat the procedure (in an almost cyclical fashion) until you get the precise configuration that you want.

You may want to reconfigure a version of MODEM86, for example, to specify a different baud rate as the default, or to change the control characters that MODEM86 uses.

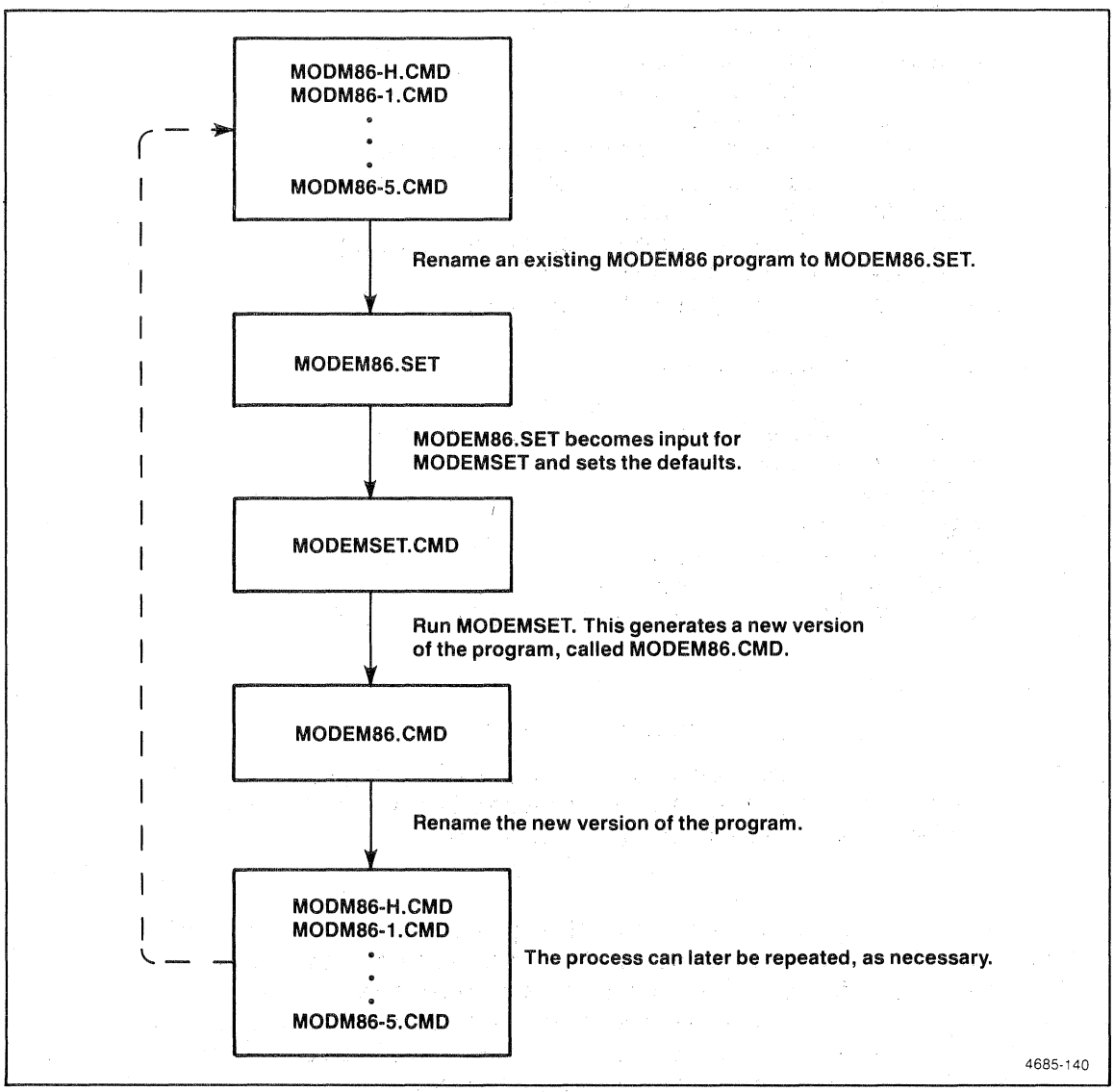


Figure 7-4. Changing a Preconfigured MODEM86 Program.

Procedure for Reconfiguring MODEM86. First, rename one of the 4170 versions of MODEM86 (MODM86-H.CMD, MODM86-1.CMD, ... MODM86-5.CMD) to the filename MODEM86.SET. MODEM86.SET is input to MODEMSET.CMD and sets the default values for the MODEMSET program.

Then enter the command:

MODEMSET<CR>

and follow the prompts. Press <CR> to use the default value. (Note that the default values in MODEMSET are the values from the renamed version. For example, if you renamed MODM86-H.CMD to MODEM86.SET, when MODEMSET prompts for the baud rate, the default within the prompt is 1200.)

Running MODEMSET creates a new configuration -- called MODEM86.CMD. To check how this new configuration works, you can now run MODEM86 by simply typing:

MODEM86 <CR>

If the new configuration is okay, rename it, using the name of the appropriate version. Thus, if you started by renaming the file MODM86-H.CMD, rename MODEM86.CMD (which is the new configuration) to MODM86-H.CMD.

UTILITIES

There are several additional utilities provided on your MODEM86 diskette. This discussion covers the REMOVECC, UNSQZ, SHRINK, EXPAND, and BIN2HEX utilities.

Printing Host Files

Many host files have FORTRAN style "logical" carriage controls in the first column of every line. These carriage controls are not standard for all computers. These carriage controls can be removed and the file converted to the more standard ASCII physical carriage controls using the REMOVECC utility provided with MODEM86. The REMOVECC program can be used with most documents copied from a host that have logical carriage controls.

REMOVECC takes up to two parameters. The first parameter is the source file, the file to be read. The second parameter is the object file, the file to be created. If the first parameter is left blank, then the source file is assumed to be "HOST.DOC". If the second parameter is left blank, then the object file is assumed to have the same name as the source, with an extension of ".LST". The wild card characters "?" and "*" may be used in either file name to use the corresponding part of the default file name and to override the unambiguously specified part.

Squeezed Files

Most remote CP/M bulletin board systems have a program called SQ for encoding files in a squeezed format, and a corresponding program called USQ for decoding the squeezed format files back into a copy of the original file. The purpose of this process is to save space and to reduce the amount of time necessary to transfer the file. In the squeezed form, a typical ASCII file takes up about 30% less space. The UNSQZ program can be used with squeezed files that have been copied from a remote CP/M host to unsqueeze the file.

UNSQZ takes up to two parameters. The first parameter is the source file, the file to be read. The second parameter is the object file, the file to be created. If the first parameter is left blank, the source file is assumed to be "HOST.SQZ". If the second parameter is left blank, the object file is assumed to have the same name as the source, with the extension of the original (unsqueezed) file as indicated in the squeezed source. (The squeezed file contains the name of the original source file.) The wild card characters "?" and "*" can be used in either file name to use the corresponding part of the default file name and to override the unambiguously specified part.

Tab Character Processing

CP/M-86 processes tab characters by expanding them with blanks when displaying them. The tabs are replaced with enough blanks to go to the next tab column. The tab columns are regularly spaced at 8 columns apart. MODEM86 comes with two utilities that do tab character processing, mostly for the purposes of saving space in text files on the disk and in saving time for text file transfers.

Using SHRINK, ASCII files can be made to take up less space than they might normally take by reducing sequences of blanks with tab characters wherever possible. This averages to about a 30% saving. Using EXPAND, ASCII files with tab characters can be expanded by removing all tab characters and replacing them with an appropriate number of blanks. This mechanism is different than squeezing the files, because the resulting file is still readable.

SHRINK takes up to two parameters. The first parameter is the source file, the file to be read. The second parameter is the object file, the file to be created. If the first parameter is left blank, the source file is assumed to be "SHRINK.LST". If the second parameter is left blank, the object file is assumed to have the same name as the source, with the extension ".SHR". The wild card characters "?" and "*" can be used in either file name to use the corresponding part of the default file name and to override the unambiguously specified part. Appropriate files to SHRINK include listings from compilers and assemblers, most assembly language source programs, and most documents.

EXPAND takes up to two parameters. The first parameter is the source file, the file to be read. The second parameter is the object file, the file to be created. If the first parameter is left blank, the source file is assumed to be "EXPAND.SHR". If the second parameter is left blank, the object file is assumed to have the same name as the source, with the extension ".LST". The wild card characters "?" and "*" can be used in either file name and to override the unambiguously specified part.

Conversion of Binary Files

When it is desirable to send a binary file to a host that only accepts ASCII (character) files, the MODEM86 utility program BIN2HEX can be used.

BIN2HEX takes up to two parameters. The first parameter is the source file, the file to be read. The second parameter is the object file, the file to be created. If the first parameter is left blank, the source file is assumed to be "MODEM86.CMD". If the second parameter is left blank, the object file is assumed to have the same name as the source, with the extension ".H86". The wild card characters "?" and "*" can be used in either file name and to override the unambiguously specified part. Appropriate files to use BIN2HEX on include program run files, program object files, and binary data files.

On a receiving CP/M-86 system, the resulting ".H86" file can be converted back to binary by issuing commands similar to the following:

```
A>GENCMD MODEM86 8080

BYTES READ    0000
RECORDS WRITTEN 00

A>DDT86
DDT86 1.1
-RMODEM86.CMD
  START      END
6DC0:0000 6DC0:4CFF
-WMODEM86.CMD,0180,4CFF
-<STACKED PLUS SIGNS>C
```

Notice that the starting address used in the W command should always be 0180, and the ending address should always be the same as the one displayed by DDT86.

IF YOU HAVE A PROBLEM WITH MODEM86

If you encounter problems with MODEM86, refer to the following headings, which describe some typical problems with MODEM86. If your problem does not match one of these descriptions, contact your local Tektronix field office.

MODEMSET Gives Strange Default Answers

MODEMSET gets the default answers from the MODEM86.SET file. The MODEM86.SET file is just a version of MODEM86 that has been renamed for MODEMSET. If you renamed the wrong file, the values will not look right. Press return until the MODEMSET menu is printed again, choose the first selection again, and respond to the prompts without using the defaults, unless they are the correct answers.

This problem may also indicate that the version of MODEMSET is not compatible with the version of MODEM86 in MODEM86.SET. This means that the MODEM86 diskette may not have the right files; contact your local Tektronix field office for a new diskette.

SIO Parameter Error Message

MODEM86 only prints this error when it detects something wrong with the serial I/O port description provided to it by MODEMSET. Usually this is because when you reconfigured a MODEM86 program you answered a prompt incorrectly for the system that MODEM86 is currently running on. The solution is to run MODEMSET again, and to carefully answer the questions.

Sometimes MODEM86 cannot detect a configuration error. Usually when this happens, it does something wrong immediately and the system often requires resetting. Again, the solution is to reconfigure MODEM86 using MODEMSET, carefully answering the questions. The default configuration answers can be used, but there is a good chance that they are wrong. It is a good idea to verify the answers by carefully; refer to the earlier discussion in this section, Changing Preconfigured MODEM86 Programs.

Incompatible Versions Error Message

Occasionally an incompatible MODEMSET program is placed on a disk with MODEM86. Both MODEMSET and MODEM86 have internal version numbers to prevent this from happening without a warning. MODEMSET leaves a copy of its version number in MODEM86 during the configuration process. The version numbers are later compared when MODEM86 runs. If the versions are not the same, this error message occurs. Contact your local Tektronix field office.

Data is Lost From Each Line

Sometimes, serial data is missed when a line feed is followed too closely by characters from the next line. In this case, it is wise to tell the host to delay after a line feed prior to sending more data, if it can be done. Otherwise, MODEM86 may miss a few characters at the beginning of each line at the higher data rates (such as those greater than 1200 bits/seconds).

If a host delay after a line feed does not solve this problem, the baud rate is probably higher than the 4170 or its terminal can handle. Try lowering the baud rate to 300; this should be slow enough to be handled by most terminals. If this works, the baud rate was probably too high originally. Try higher baud rates until you find the highest baud rate that the system can handle.

Checksummed File Copying Aborts

This is often a baud rate problem. There are several possible solutions.

The most common problem is that the baud rates used by the 4170 and host are not the same. The solution is to match the baud rates and try again.

The quiet suboption Q can be used to keep any MODEM86 messages from being displayed. This can speed up file copies at higher baud rates, or sometimes make file copies at higher baud rates work. Displaying characters may take long enough so that characters on the serial port are missed. When the characters are missed, the data is resent, but usually the result of resending is the same. When this happens, Quiet mode can keep MODEM86 from missing the data, since the message is not printed.

If the Q suboption fails to work, another possible solution is to use the F suboption on both the 4170 and the host. This could slow both down at critical points.

If neither the Q or F suboptions allow the file copy to proceed, you may have to reduce the baud rate for the file copy. Set the baud to 300 bits/second, and then increment it until you find the highest baud rate that works. Sometimes it is possible to do file copying with the 'Q' or 'F' suboptions at higher baud rates than normal terminal operations.

File Data and Host Messages not Displayed

During a simple non-checksummed file transfer (started by a CTRL-T while in Terminal mode), file data and host messages are not normally displayed. This speeds up the file transfer because there is no delay caused by the scrolling of the terminal. However, if something goes wrong, any error message generated by the host is not displayed. This may make it difficult to determine what went wrong. The solution is to use the F or V suboptions on the T option to enter Terminal mode. When at least one of these options is used, all echoing and messages from the host are displayed during the simple file transfer. Depending on the kind of terminal, this often has the effect of slowing down the transfer by delaying after each line feed. This can help the file copy if the host needs time to digest each line as it comes in.

MODEM86 Runs but Nothing is Received From the Host

There are a number of situations that have this symptom. Check out all of the following possibilities:

- Check the handshaking signals line using the V option. If MODEM86 indicates that the handshaking signals are not all on, then one of the signals (DCD -- Data Carrier Detect -- Pin 8, CTS -- Clear To Send -- Pin 5, or DSR -- Data Set Ready -- Pin 6) is not connected to the modem and may be required before the serial IC will send data to the host. It may also be the case that one of the signals RTS (Request To Send -- Pin 4) or DTR (Data Terminal Ready -- Pin 20) was not connected and the host did not see any data from the computer running MODEM86.

- For some computers, a null modem cable is necessary for connection to a modem. For most computers, a null modem is not necessary to connect two computers together. A null modem is a cable with the wires "crossed" so that: Pin 2 on each end is connected to Pin 3 on the other end; Pin 8 on each end is connected to Pin 20 on the other end and to Pins 5 and 6 on the same end; and Pins 1 and 7 are connected directly to the same numbered pins on the other end.

- Check that the the 3-Port Peripheral Interface is installed correctly.

- If the modem being used has a half/full duplex switch, is it set to full duplex? If it is set to half duplex, change it to full.

- If the modem being used has an originate/answer switch, is it set to originate? If it is set to answer, change it to originate. Most hosts use Answer mode, and expect terminals to use Originate mode.

- If the modem is an acoustic modem, is the handset placed in the modem correctly? One end of the modem acoustic coupler must always have the cord end of the handset, and the modem is usually marked to indicate which end. Also, is the handset being placed in the modem acoustic coupler quickly enough? If not, the host will probably hang up the phone before it is in. If the handset never seems to be placed in the acoustic coupler quickly enough, try putting the handset into the coupler after the first ring, instead of when the carrier signal comes on as the host answers the phone.

- Is the modem a "smart" modem, capable of handling autodialing? The modem must be told to dial up the host. Refer to the modem documentation for help with this.

- Was the baud rate specified correctly using MODEMSET, or in the MODEM86 command? If specified incorrectly, MODEM86 will not receive any characters correctly, although it will be able to detect the carrier signal and will think the host is connected. The solution is to specify the correct baud rate.

- Is the network or host phone number being dialed while MODEM86 is running? If not, the network or host may hang up because the DTR (Data Terminal Ready) signal may not be provided when the host answers the phone. The solution is to hang up, and then dial the host again while MODEM86 is running.

- Is the phone number dialed the correct number for the host being used? Is it the correct phone number for the baud rate being used? If not, the solution is to use the correct phone number.

PROTOCOLS USED

The protocols used are the same as the CP/M MODEM 7.412 program, with a modification for the possibility of a CRC-16 checksum. (The CP/M MODEM 7.412 program is used on many public bulletin board systems and RCPM systems. Because the same protocols are used, you can transfer data between those systems and a 4170 with MODEM86.)

Whenever MODEM86 expects to receive something, there is a timeout period; when the timeout expires, MODEM86 tries to recover, up to a limit of ten times. When MODEM86 reaches the limit, it asks whether it should retry another ten times or quit. When it succeeds after recovery, the error count is reset to zero.

Single File Transfer

1. Receiver sends an NAK.
2. Transmitter (sets checksum to zero and) sends:
    - o SOH for more data (included in the checksum), or
    - o EOT for end of transfer (go to 6), or
    - o CAN for cancel (go to 7).
3. Transmitter sends:
    - o Block number mod 256 (included in the checksum).
    - o 255 - (Block number mod 256) (included in the checksum).
    - o 128 bytes of data.
    - o Checksum bytes (1 or 2 depending on type used).
4. Receiver sends ACK if checksum is OK, sends NAK if not (or a timeout), or sends a CAN (go to 7).
5. Transmitter starts at Step 2 with next block ACKed, or same block NAKed. If no response at all (timeout), assumes NAK.
6. Receiver sends ACK.
7. End of transfer.

Multifile Transfer

1. Receiver sends a NAK.
2. Transmitter sends an ACK (then sets checksum to zero).
3. Transmitter sends:
   - o EOT for no more files (go to 9), or
   - o 11 characters of file name (padded with blanks).
4. Receiver sends back an ACK after each file name character.
5. Transmitter sends a CTRL-Z (included in the checksum).
6. Receiver sends back the checksum.
7. Transmitter sends an ACK if the checksum is good (go to 8) or sends a 075H if the checksum is bad (go to 1).
8. Use single file protocol to send file data (go to 1).
9. End of transfer.

CRC-16 CHECKSUM

The CRC-16 checksum is a standard block checksum generated by the binary generator polynomial X**16+X**15+...X**2+1. For a reference, see Technical Aspects of Data Communications by John E. McNamara. (CRC stands for Cyclical Redundancy Check.)

FORTRAN-86

FORTRAN-86 is an extended version of the FORTRAN 77 subset defined by the American National Standards Institute (ANSI). The FORTRAN-86 compiler translates your FORTRAN source programs into relocatable object code. This code can then be linked with other object code modules (generated by FORTRAN-86, ASM86, or other compatible languages) and executed.

THE FORTRAN-86 FILES

The FORTRAN files are distributed on two flexible disks -- FORTRAN-86, Volumes I and II. The FORTRAN files include FORT86.LTL -- the FORTRAN compiler; LINK86.LTL -- the linkage editor; LIB86.LTL -- the librarian utility; and the run-time libraries. Also included on the FORTRAN-86 diskettes is FDTI.LIB, an object library of the FORTRAN Direct Terminal Interface source code.

NOTE

The Direct Terminal Interface (DTI) is recommended for application programs that run on 4100 terminals with Local Programmability and that use specific features of these terminals. Because it directly controls terminal features, DTI offers high performance and small code size. However, using DTI limits transportability specifically to the 4100 terminals. (See Using the DTI on the 4170, at the end of this section.)

When writing applications requiring transportability to non-Tektronix graphics devices, you should use the GSX-86 drivers. The Interactive Graphics Library is appropriate for applications that will be uploaded or downloaded from a host mainframe and that follow the SIGGRAPH CORE proposed standard.

SECTION 7
Programming Information

The FORTRAN-86 diskettes contain several libraries of object code subroutines that perform I/O and other important functions for FORTRAN programs during execution. These files are also known as "run-time" libraries and are called:

CEL87.LIB
F86RN0.LIB
F86RN1.LIB
F86RN2.LIB
F86RN3.LIB
F86RN4.LIB
EH87.LIB
8087.LIB
LARGE.LIB

Although these files are not needed for FORTRAN compilation, you will need them to link and execute your compiled FORTRAN object code.

COMPILING A FORTRAN-86 PROGRAM

To compile a FORTRAN source program, type:

RUN FORT86 myprog.for {controls}<CR>

myprog.for — is the file containing your source program. The file specification should conform to the operating system's convention for naming files.

controls — is an optional list of compiler controls separated by spaces. (See Additional Details later in this section.)

For example, to compile the source code file MYFORT.FOR on drive B: with the compiler and loader on the current default drive (A:), type:

RUN FORT86 B:MYFORT.FOR<CR>

FORT86 generates an object code file with the same name as your source file and an extension OBJ. It may also produce various other files containing listings, symbol tables, etc.; this depends on the controls you specify during invocation.

During compilation, FORTRAN-86 creates and uses several temporary work files, deleting them when compilation is complete. Always leave plenty of space for these work files on the disk (typically 60 to 80 K bytes). The FORTRAN compiler also creates a file with a .LST extension. TYPE this file to read any errors produced during compilation.

LINKING AND EXECUTING A FORTRAN-86 PROGRAM

To link the object file generated by FORT86 with the run-time libraries and other object files of your own, type the invocation as follows (filespecs of files that are not on the default drive must include a drive specifier):

```
run link86 myprog.obj,{file_list,}cel87.lib,f86rn0.lib, &<CR>
f86rn1.lib,f86rn2.lib,f86rn3.lib,f86rn4.lib,eh87.lib, &<CR>
8087.lib,large.lib to myprog.ltl bind purge<CR>
```

myprog.obj — is the file containing the compiled source program

file_list — is an optional list of files containing compiled subprograms that are called by the main program. For example, IGL, GSX-86, and FDTI libraries may be included here. Separate the file names with commas (the braces are not actually entered; they are used here to indicate that file_list is optional).

myprog.ltl — is the name you wish to give the file that contains the linked modules. (The name is up to you, but the extension should be LTL.)

Notice that the command is broken twice with &<CR>. These continuation characters enable the linker to accept a command longer than one line. The linker creates a file with a .MP1 extension. This file contains results of the linking operation. TYPE myprog.MP1 to review results of the link operation.

NOTE

Use the STREAM or QS programs (supplied on the Program Exchange diskette) to simplify link operations. Type out the files STREAM.DOC or QS.DOC (also on the Program Exchange diskette) for more information. (STREAM.DOC gives an example of using STREAM to compile and link an object file.)

To execute your linked program, type:

RUN myprog<CR>

The RUN utility loads and runs your program. Control returns to the operating system after your program is finished.

Compiling and Linking with Flexible Disks

To compile and link FORTRAN programs using only the two flexible disk drives, it is recommended that you put the compiler (FORT86.LTL) on one diskette, the linker (LINK86.LTL) and run-time libraries on another, and the FORTRAN source program on a third diskette.

Then, when you compile and link the source program, specify the disk drive before file names, as required. For example, this command compiles the source "myprog.for" on the diskette in drive A (using the compiler on the diskette in drive B):

A>RUN B:FORT86 myprog.for <CR>

Because the compiling operation creates several temporary work files (and then deletes them upon completion), you should allow plenty of space on the source diskette for the work files.

To link the object file, remove the diskette that has the compiler, and insert the diskette with the linker and run-time libraries in its place. Then, specify the link command, as in this example (the example assumes that the diskette in drive A has the object file "myprog.obj" and that the diskette in drive B has the linker and run-time libraries):

```
A>RUN B:LINK86 myprog.obj,B:ce187.lib,B:f86rn0.lib, &<CR>
A>>B:f86rn1.lib,B:f86rn2.lib,B:f86rn3.lib,B:f86rn4.lib, &<CR>
A>>B:eh87.lib,B:8087.lib,B:large.lib to myprog.ltl bind purge<CR>
```

If you are using only flexible disk drives to compile and link an IGL program, refer to the procedure IGL With Flexible Diskettes, later in this section.

FORTRAN-86 FEATURES UNIQUE TO 4170 CP/M-86

As explained in the FORTRAN-86 User's Manual, FORTRAN-86 allows you to associate logical unit numbers with specific files and physical devices. This is done using OPEN statements (connection) and with UNIT controls appended to RUN commands (preconnection). Only two unit numbers are connected by default:

- Unit 5 - keyboard of connected terminal
- Unit 6 - display screen of connected terminal

Other logical units in the range 0 - 255 are undefined. To associate a unit number with a file on a specific disk or drive or under a particular user number, include the appropriate drive and user number specifiers in the name you give in the FORTRAN OPEN statement. The following OPEN statement, for example, opens the file MYFILE.DAT on drive B: under user number 4, connecting it to unit 99:

```
OPEN(99,FILE='B:MYFILE.DAT[g4]',STATUS='NEW')
```

Similarly, you can talk to the 4170 CP/M-86 logical devices Console (CON:), Auxiliary (AUX:), and List (LST:). Use the following names in place of disk file names:

```
CP/M Console device:
   CON:      Console input and output
   CONOUT:   Console output
   CONIN:    Console input

CP/M Auxiliary device:
   AUX:      Auxiliary input and output
   AXO:      Auxiliary output
   AXI:      Auxiliary input

CP/M List device:
   LST:      List output
```

For example, to set FORTRAN logical unit number 7 for Auxiliary input and unit 8 for Auxiliary output, you can use the following OPEN statements:

```
OPEN(UNIT=7,FILE='AXI:')
OPEN(UNIT=8,FILE='AXO:')
```

The CP/M logical devices can be used with file pre-connection also. The following command assigns FORTRAN logical unit numebr 15 to the CP/M List device:

```
RUN MYPROG(UNIT15=LST:)
```

FORTRAN OVERLAYS

When a FORTRAN program is too large to load into memory in its entirety, you may need to "overlay" the program to make it usable. Overlaying is a practice of loading program modules into a common region of memory as they are needed. Essentially, this reduces the needed memory to an amount sufficient to hold both the main or "root" program and the largest subprogram.

The operating system has a special subroutine, DQOVER, to load overlays. A call to DQOVER has this form:

```
CALL DQOVER (MODNAM,IERROR)
```

MODNAM is the character string identifying the overlay module to be loaded. An overlay module can include several subprograms.

IERROR returns an integer indicating whether the overlay was loaded successfully: 0 indicates success; less than 0 indicates a failure, probably because the overlay was not found.

Once DQOVER has loaded an overlay module, you can invoke subprograms from that module as usual. Overlay modules remain in memory until overwritten by subsequent overlays.

The following is an example of a simple overlayed program:

```
File ROOT.FOR:
      PROGRAM ROOT
      print *,'This is the root module"
C Load the first overlay and call it
      call dqover('OV1',ierr)
      if (ierr  .ne. 0) goto 100
      call ov1
C Load the second overlay and call it
      call dqover('OV2',ierr)
      if (ierr  .ne. 0) goto 100
      call ov2
      goto 900

100   print 'Error on overlay load: ',ierr

900   end

File OV1.FOR:
      subroutine ov1
      print *,'This is overlay 1'
      return
      end

File OV2.FOR:
      subroutine ov2
      print *,'This is overlay 2'
      return
      end
```

Linking an overlayed program is a three-step process. First. link the root and any required libraries. Second, link each overlay against the root and any required libraries. Third, link the files produced by all the previous links into the final executable file.

Here are the commands to link the root. These commands also force some routines required explicitly for proper overlay execution into the root.

```
Run link86 Root.obj, eh87.lib(tqinstruction_retry), cel87.lib,&
f86rn0.lib,f86rn2.lib(input_edit_table,output_edit_table),&
f86rn1.lib,f86rn2.lib,&
f86rn4.lib(format_seq_device_driver,unformat_seq_device_driver),&
f86rn3.lib,f86rn4.lib,&
eh87.lib,8087.lib,large.lib &
to root.lnk overlay(root)
```

Then link the overlays against the root:

```
Run link86 ov1.obj,cel87.lib,&
f86rn0.lib,f86rn1.lib,f86rn2.lib,f86rn3.lib, f86rn4.lib,&
eh87.lib, 8087.lib, large.lib &
to ov1.lnk overlay(ov1) assumeroot (root.lnk)

Run link86 ov2.obj, cel87.lib,
f86rn0.lib,f86rn1.lib,f86rn2.lib,f86rn3.lib, f86rn4.lib,&
eh87.lib, 8087.lib, large.lib &
to ov2.lnk overlay(ov2) assumeroot(root.lnk)
```

Finally, link everything together:

```
Run link86 root.lnk,ov1.lnk,ov2.lnk to root.ltl purge bind nomap
```

It is normal to receive warnings for UNRESOLVED EXTERNALS from the linker in all the links except the last. However, if the final link also reports an UNRESOLVED EXTERNAL, then the missing symbol must be located and the problem corrected.

The PURGE and BIND directives are used only on the final link. Also note the use of the ASSUMEROOT directive. This directive tells the linker that a subroutine call in an overlay should point to the subroutine in the root, if the file ROOT.LNK contains that subroutine. This avoids loading the same subroutine into both the root and the overlay.

The above examples were designed for the sample program. If additional object files need to be added to the root or overlay files, add them following the main program or overlay. For example:

```
Run link86 root.obj,sub1.obj,sub2.obj,eh87.lib(tqins...etc.),

Run link86 ov1.obj,sub3.obj,sub4.obj,cel87.lib,& etc..
```

If additional library files are used, add them to the root file as well as to the overlays:

Run link86 root.obj,**fdti.lib**,eh87.lib, etc...

Run link86 ov1.obj,**fdti.lib**,cel87.lib,etc...

ADDITIONAL DETAILS ABOUT FORTRAN-86

The Intel manual,FORTRAN-86 Users Manual describes FORTRAN-86 in detail. The FORTRAN-86 Pocket Reference Guide provides a concise list of FORTRAN-86 statements and compiler controls for quick reference.

CAUTION

Do not use the invocation commands and device descriptors as given in the two Intel manuals; FORTRAN-86 under the 4170 CP/M-86 recognizes only the invocation sequence described in this section.

IGL

INTRODUCTION

The 4170 software includes an object library version of the Interactive Graphics Library (IGL). This version contains the Primary Command Set, Panel Support, Panel Emulation, and selected device drivers. For expanded IGL capabilities (Line Smoothing, Segments, 3-D, and additional terminal support), you may order the Tektronix 4170P73, Options 23 and 24.

Refer to the IGL User's Manual (that supports Level 5 and up) and the IGL Reference Guide for more detailed information on using IGL. (These manuals are included with the 4170.)

This section contains information for getting IGL programs running on your 4170. It includes:

- General information for using the subset of IGL provided.
- Information for setting up IGL on the hard disk and for compiling, linking, and running programs on the hard disk.
- Information for using IGL if you don't have a hard disk (including how to redistribute files on diskettes so that you can compile, link, and run your programs).

Requirements for Running IGL

The minimum requirements for running IGL are a graphics terminal attached to the 4170 and at least 256K bytes of memory. It is strongly recommended that the optional hard disk be a part of the 4170, and that IGL, FORTRAN, and your application program be resident on the hard disk. A system with a hard disk (Options 03 and 45) can handle the largest IGL programs, and affords the added benefits of increased development speed and greater convenience. If you do not have a hard disk, you may need some blank disks to redistribute the IGL files.

Diskette Contents

Your two IGL diskettes contain these files:

- IGLA.LIB--the first part of the IGL library
- IGLB.LIB--the second part of the IGL library
- ERRFIL.DAT--the random access error message file
- DEMO.FOR--a sample program

Terminals Supported

IGL is configured with the following device drivers:

4110 Series terminal driver (supports the 4110 Series terminals -- the 4112, 4113, 4114, 4115, and 4116 terminals).

4014 terminal driver (supports the 4014, 4015 and 4016 graphics terminals with the Enhanced Graphics Module).

4105 terminal driver (supports the 4105 graphics terminal; for the 4107 and 4109 terminals, use the 4113 device driver).

4662 plotter driver (supports the standard 4662 plotter and the eight-pen 4662 Option 31).

Integer Size

IGL has been installed using 32-bit integers. All IGL routines using integer arguments must be passed 32-bit integers. You may compile your application with the STORAGE(INTEGER*4) compiler control or explicitly declare all integers passed to IGL routines as INTEGER*4.

Logical Unit Numbers

IGL may use FORTRAN logical unit numbers 10 through 17 for file processing. You should not attempt to use these unit numbers in your programs.

File Names

IGL expects file names used by HFOPEN (Host File Open) to be exactly six characters long, starting with an alphabetic character. Lower case letters are acceptable. The file name supplied to HFOPEN must not have an extension - HFOPEN appends an extension of ".DAT" to the file name.

File Types

IGL for the 4170 does not support file format 6 - random read-only file. You must use file format 5 - random read/write file. If you port programs from an IGL host I/O package that supports random read-only files, you must change the calls to HFOPEN to use format 5.

Error Message File

IGL is shipped with an error message file. You may call the IGL routines REPORT and CHECK to get the latest error reported by IGL. These routines open a disk file with the CP/M-86 name ERRFIL.DAT. If you use REPORT or CHECK, be sure the file is in your user number or in user number 0 with a SYS attribute.

USING IGL WITH A HARD DISK

Setup for Hard Disk Operation

When you receive IGL, FORTRAN, and the 4170 Operating System, you get these programs on flexible diskettes. To make working with IGL more convenient, put the following files (using PIP) in User 0 of the hard disk:

ED.CMD
RUN.CMD
FORT86.LTL
LINK86.LTL
IGLA.LIB
IGLB.LIB
CEL87.LIB
F86RN0.LIB
F86RN1.LIB
F86RN2.LIB
F86RN3.LIB
F86RN4.LIB
EH87.LIB
8087.LIB
LARGE.LIB

Use the STAT utility to change all of these file attributes to SYS (so they may be referenced from any other user number).

Put the file DEMO.FOR in a user number of your choice.

Compiling an IGL FORTRAN Application Source File

After you have created an IGL FORTRAN 77 source file (or would prefer to use DEMO.FOR as a source file) you will need to compile it with the FORTRAN-86 compiler. The compiler accepts your application source file as input and produces a relocatable object file suitable for linking with the IGL library files, IGLA.LIB and IGLB.LIB.

To compile your source program, type the following:

RUN FORT86 myprog.FOR NOTYPE NOLIST STORAGE(INTEGER*4)<CR>

<CR> means "press the RETURN key". MYPROG is the name of your source file (it could be DEMO).

When the compiler is finished, it outputs a message to the terminal specifying how many errors and warnings it found in your source file. If the compiler message says you have any errors or warnings you can use the TYPE command to examine the myprog.LST file that the compiler creates. You can locate your error or warning messages in this file. If your source file produces no errors, you are ready to link myprog.OBJ, the object module file that the compiler produced.

NOTE

During compilation, FORTRAN-86 creates and uses several temporary "work files", deleting them when compilation is complete. Because of this, you should make certain that you have from 60K to 80K free file space for a modest application program.

Linking Your IGL FORTRAN Object File

Once you have successfully compiled your source file, you are ready to link myprog.OBJ with the IGL and FORTRAN-86 library files. Enter the following command:

```
RUN LINK86 myprog.OBJ,IGLA.LIB,IGLB.LIB,CEL87.LIB,&<CR>
F86RN0.LIB,F86RN1.LIB,F86RN2.LIB,F86RN3.LIB,F86RN4.LIB,&<CR>
EH87.LIB,8087.LIB,LARGE.LIB TO myprog.LTL &<CR>
BIND PURGE NOMAP<CR>
```

Note the use of the ampersand character (&) to specify to the linker that you have a continuation line.

NOTE

Use the STREAM or QS programs (supplied on the Program Exchange diskette) to simplify link operations with IGL. Type out the STREAM.DOC or the QS.DOC files for more information on how to use these programs.

Running Your IGL FORTRAN Program

When you have successfully created myprog.LTL, you can execute the .LTL file with the RUN.CMD file, a load-time locatable program loader.

To load and run the linked FORTRAN object file myprog.LTL (or DEMO.LTL if you have compiled and linked DEMO.FOR), enter the command:

RUN myprog<CR>

or

RUN DEMO<CR>

This completes the set-up instructions for those 4170's with a hard disk.

IGL WITH FLEXIBLE DISKETTES

Setup for Flexible Diskette Operation

When you receive IGL, FORTRAN and the 4170 Operating System, you get each of these programs on flexible diskettes. To make working with IGL convenient, you should use the PIP utility (with the [OV] options) to redistribute the files on working diskettes as follows:

NOTE

Whenever you change disks, you must log in the new disk by typing CTRL-C.

1. Put these files on Work Disk 1.

   FORT86.LTL
   RUN.CMD

2. Put these files on Work Disk 2.

   CEL87.LIB
   IGLA.LIB
   IGLB.LIB

   Copy the DEMO.FOR program from the distribution medium to this disk (or place your own IGL application program on this disk).

3. Put these files on Work Disk 3.

   LINK86.LTL
   RUN.CMD
   F86RN3.LIB
   F86RN4.LIB
   8087.LIB
   LARGE.LIB

4. Put these files on Work Disk 4.

   F86RN0.LIB
   F86RN1.LIB
   F86RN2.LIB
   EH87.LIB

Compiling an IGL FORTRAN Application Source File

After you have created an IGL FORTRAN 77 source file (such as DEMO.FOR) you will need to compile it with the FORTRAN-86 compiler. The compiler accepts your application source file as input and produces a locatable object file suitable for linking with the IGL library files, IGLA.LIB and IGLB.LIB.

Use this sequence of steps to compile your FORTRAN source file:

1. First make sure that the Work Disk 1 is in Drive A, Work Disk 2 is in Drive B and that both have been logged in. Assign drive A to be the current drive (the prompt on the screen will be A>). Make certain that the disk drive latches are closed.

2. To compile your source file, type the following:

   RUN FORT86 B:myprog.FOR NOTYPE NOLIST STORAGE(INTEGER*4)<CR>

   After you press RETURN, the compiler runs for a length of time that depends on the size of your source file. When the compiler is finished, it outputs a message to the terminal specifying how many errors and warnings it found in your source file. If the compiler message indicates you have any errors or warnings, you can use the TYPE command to examine the myprog.LST file (on work Disk 2) that the compiler creates. You can locate your error or warning messages in this file. If your source file produced no errors or warnings, you are ready to link myprog.OBJ, the object module file that the compiler created.

   NOTE

   During compilation, FORTRAN-86 creates and uses several temporary "work files", deleting them when compilation is complete. Because of this, you should make certain that the working disk (in this example - Work Disk 1) has from 60K to 80K free file space.

Linking Your IGL FORTRAN Object File

Once you have successfully compiled your source file, you are ready to link myprog.OBJ with the IGL and FORTRAN-86 library files. Due to the diskette space limitations, the linking must be done in two steps.

The following is a sequence of steps you can use to link myprog.OBJ:

1. Be certain that Work Disk 2 is still in Disk Drive B.

2. Put Work Disk 3 in Drive A. Assign to drive A (the prompt on the screen will be A>). Log in the drive by typing CTRL-C.

3. To run the first link step, type in:

```
RUN LINK86 b:myprog.OBJ,B:IGLA.LIB,B:IGLB.LIB,B:CEL87.LIB &<CR>
to myprog.LNK PURGE NOMAP<CR>
```

NOTE

When the linker is finished, you will see the following message:

WARNING 12: UNRESOLVED SYMBOLS

The linker outputs this message whenever you do a two or more step linking job. The unresolved symbols should be resolved upon completion of the final stage in the link job.

4. After LINK86 creates the myprog.LNK file, remove Work Disk 2 from Disk Drive B and replace it with Work Disk 4. Log Work Disk 4 in by making the current drive B, then type in CTRL-C. You will see the prompt B>. Set Drive A to be the current drive. You will see the prompt A>.

5. To run the second link step, type in:

```
RUN LINK86 myprog.LNK,B:F86RN0.LIB,B:F86RN1.LIB,B:F86RN2.LIB,&<CR>
F86RN3.LIB,F86RN4.LIB,B:EH87.LIB,8087.LIB,LARGE.LIB TO &<CR>
B:myprog.LTL BIND PURGE NOMAP<CR>
```

Note the use of the ampersand character (&) to specify to the linker that you have a continuation line. Be sure to specify Disk Drive B (B:) as shown above. This causes the linker to search the correct disk for the files.

```
ERROR 3: I/O ERROR
?Exception 0029h too many entries in directory
```

This is an error message from CP/M-86. If you see it while you are attempting to carry out one of the steps detailed above, a likely cause is that your application program is too large to be compiled and/or linked on a system with flexible diskettes as the only storage medium. In this case the message should not be taken literally but rather as an indication that diskette storage capacity has been exceeded.

Running Your IGL FORTRAN Program

When you have successfully created myprog.ltl, you can execute the .LTL file with the RUN.CMD file, a load-time locatable program loader.

To load and run the linked FORTRAN object file myprog.ltl (or demo.ltl if you have compiled and linked demo.for), enter the command:

RUN B:myprog<CR>

or

RUN B:demo<CR>

GSX-86

INTRODUCTION

GSX-86 is a device-independent graphics system for CP/M-86. It is provided to you on two diskettes. In addition to the diskettes, you should have the GSX-86 Graphics Extension Programmer's Guide that describes GSX-86 in detail.

You should use the programmer's guide as the primary documentation for GSX-86. This section describes the contents of the GSX-86 diskettes, the FORTRAN GSX-86 interface, how to run the demonstration program, the extensions for advanced alpha text-editing features, and device-specific information for the Tektronix device drivers.

Diskette Contents

Your GSX-86 diskettes contain these files:

- GRAPHICS.CMD--this file enables GSX-86 and loads the default device driver.
- ASSIGN.SYS--this is a sample device driver assignment table file.
- DD41XX.SYS--device driver for Tektronix 4100 Series terminals (4105, 4107, 4109, 4112, 4113, 4114, 4115, and 4116); the 4112, 4113, and 4115 must have firmware version 6 or higher.
- DD466X.SYS--device driver for Tektronix 4662/4663 plotters (including the 4662 with the 8-pen Option 31).
- GSX86.A86--assembly language source of the FORTRAN-to-GSX-86 interface subroutine.
- GSX86.OBJ--assembled object code of GSX86.A86.
- GSXLIB.DOC--a documentation file for GSXLIB.FOR and GSXLIB.LIB.
- GSXLIB.FOR--the FORTRAN source code.
- GSXLIB.LIB--an object library of the FORTRAN source code.
- GSXDEMO.FOR--a short GSX-86 demonstration program (this program is used as an example later in this section).

NOTE

On one of the GSX-86 diskettes are the Tektronix-written device drivers DD41XX.SYS and DD466X.SYS. This diskette also includes some device drivers not written by Tektronix that are included for your convenience. These device drivers are provided "as is," without warranty of any kind, either expressed or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose.

GSX-86 FORTRAN INTERFACE

The files GSX86.A86 (source) and GSX86.OBJ (assembled object code) contain a FORTRAN callable subroutine that enables a FORTRAN program to communicate directly with GSX-86. The calling sequence for this subroutine is:

Call GSX86 (contrl, intin, ptsin, intout, ptsout)

where contrl, intin, ptsin, intout, ptsout are all integer*2 arrays.

NOTE

The arrays contrl, ptsin, and ptsout must be explicitly declared as integer in FORTRAN. These names are used for compatibility with the GSX-86 Graphics Extension Programmer's Guide.

The virtual device specification contained in the GSX-86 Graphics Extension Programmer's Guide describes the use of these integer arrays. The file GSXLIB.FOR provides examples using this subroutine to access most of the features of GSX-86. The file GSXLIB.LIB is a library of the subroutines in GSXLIB.FOR plus the interface subroutine GSX86.A86. A brief description of each subroutine is in the file GSXLIB.DOC.

AN EXAMPLE: COMPILING, LINKING, AND RUNNING THE DEMONSTRATION PROGRAM

The file GSXDEMO.FOR is a short, FORTRAN demonstration program that uses the GSXLIB subroutines. It is first necessary to compile the source program by typing the command:

```
RUN FORT86 GSXDEMO.FOR
```

Next, the object code produced by the compiler must be linked with the GSXLIB subroutines and the FORTRAN run-time libraries. Type the command:

```
RUN LINK86 GSXDEMO.OBJ, GSXLIB.LIB, CEL87.LIB, F86RN0.LIB, & <cr>
F86RN1.LIB, F86RN2.LIB, F86RN3.LIB, F86RN4.LIB, EH87.LIB, & <cr>
8087.LIB, LARGE.LIB  TO  GSXDEMO.LTL PURGE BIND NOMAP <cr>
```

This produces the file GSXDEMO.LTL.

NOTE

> Use the STREAM or QS programs (supplied on the Program Exchange diskette) to simplify link operations with GSX-86. Type out the STREAM.DOC or the QS.DOC files for more information on how to use these programs.

Before the demonstration can be run, it is necessary to enable GSX-86 by typing the command:

```
GRAPHICS <cr>
```

The GRAPHICS command reads in the default device driver (the first one listed in the table in ASSIGN.SYS) and loads the Graphics Device Operating System (GDOS). (See the Digital Research manual, the GSX-86 Graphic Extension Programmer's Guide.) The ASSIGN.SYS file furnished on the GSX-86 diskettes uses the Tektronix 4100 series driver DD41XX.SYS as the default. (These files must be on the current disk.)

Now, run the demonstration program by typing:

```
RUN GSXDEMO
```

The program displays menus and is self-explanatory.

TEKTRONIX EXTENSIONS TO GSX-86

GSX-86 provides the ESCAPE function to support device-dependent features. Some ESCAPE functions are predefined to support features found only in CRT terminals such as alpha cursor addressing. ESCAPE functions 51-99 are available for extensions.

Tektronix has added 17 extensions (functions 51-68) that improve performance of applications which involve lots of text-editing and support the color features available in Tektronix terminals. These functions are described in the following paragraphs. Alpha cursor positioning and these extensions are not available when a 4114 or 4116 is used. The extensions are presented by function number. The following list is an alphabetical listing of the Tektronix extensions.

Extension Name	Escape Function Id
BOLD RENDITION ON	67
CLEAR HORIZONTAL TAB	59
DELETE CHARACTERS	57
DELETE LINES	53
INQUIRE ADVANCED EDITING FEATURES	51
INQUIRE ALPHA TEXT COLOR REPRESENTATION	66
INSERT CHARACTERS	54
INSERT LINES	52
INSERT MODE OFF	56
INSERT MODE ON	55
SCROLL DOWN	61
SCROLL UP	60
SET ALPHA TEXT COLOR INDICES	64
SET ALPHA TEXT COLOR REPRESENTATION	65
SET DEFAULT GRAPHICS RENDITION	63
SET HORIZONTAL TAB	58
SLOW BLINK ON	68
UNDERLINING ON	62

Inquire Advanced Editing Features

This extension returns availability of advanced features to application program.

Input

```
contrl(1)  -- 5          Indicates ESCAPE function
contrl(2)  -- 0
contrl(6)  -- 51         Escape function id
```

Output

```
contrl(3)  -- 0
contrl(5)  -- 12 Number of data items returned in Intout
intout(1)  -- INSERT LINE capability,0=no,1=yes
intout(2)  -- DELETE LINE capability,0=no,1=yes
intout(3)  -- INSERT CHARACTERS capability,0=no,1=yes
intout(4)  -- INSERT MODE, =0 (no insert mode),1=yes
intout(5)  -- DELETE CHARACTERS capability,0=none,1=yes
intout(6)  -- HORIZONTAL TABS, = 0 (none),1=yes,setable,
              n>1 (yes, but fixed at multiples of n columns)
intout(7)  -- SCROLL UP capability,0=no,1=yes
intout(8)  -- SCROLL DOWN capability,0=no,1=yes
intout(9)  -- GRAPHIC RENDITION FEATURES (e.g. reverse video,
              underlining, bold, and blink)0=no,1=yes
intout(10) -- ALPHA TEXT COLOR CAPABILITY
                = 0  no colors available (monochrome device)
                = 1  colored text available, but all
                     characters on the screen must be the
                     same color, can't be changed on a
                     character by character basis (such
                     as Tektronix 4112, 4113 and 4115)
                = 2  colored text available, and can be
                     changed on a character by character
                     basis (such as Tektronix 4105,4107
                     and 4109)

intout(11) -- NUMBER OF COLORS AVAILABLE FOR ALPHA TEXT
                = 2  monochrome (black and white)
                > 2  color (note: number of colors for
                     alpha text may be less than number of
                     colors available for graphics (such as
                     the 4115)

intout(12) -- SEPARATE ALPHA TEXT COLOR MAP
                = 0  color map for alpha text is the
                     same as for graphics (Tektronix
                     4112,4113 and 4115)
                = 1  separate alpha text color map
                     (4105,4107,4109)
```

Insert Lines

This routine inserts blank lines at the current cursor position.

Input

```
contrl(1) -- 5            ESCAPE function opcode
contrl(2) -- 0
contrl(6) -- 52           function id
intin(1)  -- number of lines to insert
```

Output

```
contrl(3) -- 0
```

Delete Lines

This routine deletes the line containing the cursor and possibly following lines if specified.

Input

```
contrl(1) -- 5            ESCAPE function
contrl(2) -- 0
contrl(6) -- 53           function id
intin(1)  -- number of lines to delete
```

Output

```
contrl(3) -- 0
```

Insert Characters

This routine inserts the requested number of blank character cells at the current cursor position. Text previously at the cursor position is moved to the right and any characters shifted beyond the right margin are lost. The cursor position remains the same.

Input

```
contrl(1) -- 5            ESCAPE function
contrl(2) -- 0
contrl(6) -- 54           function id
intin(1)  -- number of blank characters to insert
```

Output

```
contrl(3) -- 0
```

Insert Mode On

This routine causes the text at and following the current cursor position to be moved to the right when a character is received. Any text shifted beyond the right margin is lost.

Input

```
contrl(1) -- 5          ESCAPE function
contrl(2) -- 0
contrl(6) -- 55         function id
```

Output

```
contrl(3) -- 0
```

Insert Mode Off

This routine causes the received character to replace the character at the current cursor position. It resets the effects of INSERT MODE ON.

Input

```
contrl(1) -- 5          ESCAPE function
contrl(2) -- 0
contrl(6) -- 56         function id
```

Output

```
contrl(3) -- 0
```

Delete Characters

This routine deletes the character at the current cursor position and, if specified, characters following the cursor. Text following the deleted characters is shifted to the left to fill in the gap. Only characters on the current line are affected; the cursor position is left unchanged.

Input

```
contrl(1) -- 5          ESCAPE function
contrl(2) -- 0
contrl(6) -- 57         function id
intin(1)  -- number of characters to delete
```

Output

```
contrl(3) -- 0
```

Set Horizontal Tab

This routine sets a tab stop at the current cursor position.

Input

```
contrl(1) -- 5          ESCAPE function
contrl(2) -- 0
contrl(6) -- 58         function id
```

Output

```
contrl(3) -- 0
```

Clear Horizontal Tab

This routine clears the tab stop at the current cursor position or all tab stops.

Input

```
contrl(1) -- 5          ESCAPE function
contrl(2) -- 0
contrl(6) -- 59         function id
intin(1)  -- 1 to clear stop at current cursor position
          -- 2 to clear all tab stops
```

Output

```
contrl(3) -- 0
```

Scroll Up

This routine scrolls the alpha text region up. Blank lines are inserted from the bottom and lines scrolled off the top are lost. The alpha cursor maintains its position relative to the scroll, but does not scroll up past row 1 (it never leaves the screen).

Input

```
contrl(1) -- 5          ESCAPE function
contrl(2) -- 0
contrl(6) -- 60         function id
intin(1)  -- number of lines to scroll
```

Output

```
contrl(3) -- 0
```

Scroll Down

This routine scrolls the alpha text region down. Blank lines are inserted from the top and lines scrolled off the bottom are lost. The alpha cursor retains its position relative to the scroll, but does not scroll off the bottom of the screen.

Input

```
contrl(1) -- 5            ESCAPE function
contrl(2) -- 0
contrl(6) -- 61           function id
intin(1)  -- number of lines to scroll
```

Output

```
contrl(3) -- 0
```

Underlining On

This routine causes an underline to be drawn under all received characters.

Input

```
contrl(1) -- 5            ESCAPE function
contrl(2) -- 0
contrl(6) -- 62           function id
```

Output

```
contrl(3) -- 0
```

Set Default Graphics Rendition

This routine turns off graphics renditions such as reverse video, underlining, bold, and blink (giving no blink, no underscore, no bold, positive video).

Input

```
contrl(1) -- 5          ESCAPE function
contrl(2) -- 0
contrl(6) -- 63         function id
```

Output

```
contrl(3) -- 0
```

Set Alpha Text Color Indices

This routine sets the color indices to be used for subsequent characters. The wipe index value is the color of the alpha screen when it is erased.

Input

```
contrl(1) -- 5          ESCAPE function
contrl(2) -- 0
contrl(6) -- 64         function id
intin (1) -- character foreground index or -1 to not change
intin (2) -- character background index or -1 to not change
intin (3) -- dialog area wipe index or -1 to not change
```

Output

```
contrl(3) -- 0
```

NOTE

Some terminals cannot change the alpha text character colors character by character (for example, on the Tektronix 4110 series terminals). On these terminals, this escape function causes all text on the screen to be changed to the specified indices.

Set Alpha Text Color Representation

This routine maps the specified color index to the color specified in RGB. If the alpha text color representation and the graphics color representations are the same, this command changes both alpha and graphics representations.

Input

```
contrl(1) -- 5            ESCAPE function
contrl(2) -- 0
contrl(6) -- 65           function id
intin(1)  -- color index
intin(2)  -- red color intensity (in tenths of a percent,
             0-1000)
intin(3)  -- green color intensity (0-1000)
intin(4)  -- blue color intensity (0-1000)
```

Output

```
contrl(3) -- 0
```

Inquire Alpha Text Color Representation

This routine returns the representation of the specified index in RGB units.

Input

```
contrl(1) -- 5            ESCAPE function
contrl(2) -- 0
contrl(6) -- 66           function id
intin(1)  -- requested color index
```

Output

```
contrl(3) -- 0
intout(1) -- color index
intout(2) -- red intensity (in tenths of a percent, 0-1000)
intout(3) -- green intensity (0-1000)
intout(4) -- blue intensity (0-1000)
```

Bold Rendition On

This routine causes subsequent alpha characters to be displayed in color index 1.

Input

contrl(1) -- 5	ESCAPE function
contrl(2) -- 0	
contrl(6) -- 67	function id

Output

contrl(3) -- 0

NOTE

On the Tektronix 4112 terminal, color index 5 is used instead of 1. Bold is turned off by the Select Default Rendition escape function (63).

Slow Blink On

This routine causes subsequent alpha characters to blinked slowly on and off.

Input

contrl(1) -- 5	ESCAPE function
contrl(2) -- 0	
contrl(6) -- 68	

Output

contrl(3) -- 0

DEVICE-SPECIFIC INFORMATION (FOR TEKTRONIX DEVICE DRIVERS)

Table 7-2 describes the device driver DD41XX.SYS and Table 7-3 describes the device driver DD466X.SYS.

Table 7-2

DEVICE DRIVER DD41XX.SYS INFORMATION

Characteristic	Description
Filename	DD41XX.SYS
Terminals Supported	The TEKTRONIX 4105, 4107, 4109, 4112, 4113, 4114, 4115, and 4116. Firmware version number 6 and higher is required for the 4112, 4113, 4114, 4115, and 4116 terminals.
Device Index	In the ASSIGN.SYS file included on the GSX-86 diskette, this driver is assigned index (workstation id) 1.
Escapes	Cursor positioning, editing, and color. Escapes are not supported for the 4114 and 4116.
Color	The 4114 and 4116 support only two indices, 0=background and 1=green; indices greater than 1 are taken as index 1. These may not be redefined. The 4105 has 8 colors for graphics and a separate 8 colors for dialog characters. The 4107 and 4109 have 16 colors for graphics and 8 for dialog characters. The number of colors available on the 4113 and 4115 depends on the number of bit planes of display memory installed. Color indices 0-7 are mapped according to the specification given in Appendix B of the GSX-86 Programmers Guide; indices greater than 7 map to the terminal index. Color representation can only be set and inquired upon for indices from 0-15. On the 4112, the indices 0-7 map to the terminal's gray scale values. It is possible to change the default gray scales using the set color representation function.

Table 7-2 (cont)

DEVICE DRIVER DD41XX.SYS INFORMATION

Characteristic	Description
Graphic Input (GIN)	Two input devices are available: the thumbwheels (or joydisk) and an optional graphics tablet (except the tablet is not available on the 4105). For the 4107 and 4109, the 4957 graphics tablet must be connected to the terminal's port p1.
Generalized Drawing Primitives (GDPS)	One GDP is available: 1=bar
Text	Continuous text scaling and rotation is available, except for the 4105 which has a limited number of character sizes and rotation in 90 degree increments only.
Linestyles	Eight linestyles are available (these are the terminal linestyles). Linestyle 1 is solid, and 2-8 are combinations of dots and dashes.
Fill Patterns	Only "hollow" is available on the 4114 and 4116. On the other terminals, 16 patterns are available (these are the terminal fill patterns).
Markers	Ten marker types are available.

Table 7-3

DEVICE DRIVER DD466X.SYS INFORMATION

Characteristic	Description
Filename	DD466X.SYS
Device Index	The actual device index is determined by the value used in the file ASSIGN.SYS. For plotters, values between 11 and 20 are recommended. In the ASSIGN.SYS furnished, index 11 is used.
Communications	The CP/M-86 logical devices AXI: (auxiliary input) and AXO: (auxiliary output) are used. The CONFIG or SETDEV utility should be used to set one of the peripheral ports to the CP/M-86 AXI and AXO logical device with the proper baud rate. The maximum baud rate is 1200 for the 4662 and 9600 for the 4663. The 4662 rear switches should be set so that the GIN terminator is CR (carriage return), the device address is A, and DEL IGNORE is NO; the baud rate, parity and stop bits should be the same as the corresponding 3PPI terminal settings. For example: 0223 for 1200 baud, no parity, and 2 stop bits. The 4663 parameter entry card should be set for attention character = ESC, serial address = A, command/response format = 1, output terminator = CR, and DEL IGNORE off. The other serial communications settings should be set according to the terminal's values. If Option 36 Media Advance is installed, a Clear Workstation function advances the paper when roll mode is selected.

Table 7-3 (cont)

DEVICE DRIVER DD466X.SYS INFORMATION

Characteristic	Description
Graphic Input	The only available device is the joystick (default). On the 4662, the prompt light is turned on when an Input Locator function is called. A point inserted by pressing the CALL button momentarily (not held down until it beeps). On the 4663, the DRAW POINT light blinks on and off during GIN. Either the MOVE POINT or DRAW POINT button may be used to enter a GIN point. The Input Locator function returns a locator terminator which has the value 32 if the plotter pen is up during a GIN (or the MOVE POINT button was hit on a 4663), or 33 if the pen is down (or the 4663 DRAW POINT was hit).
Colors	Color index 0 is the background and is never plotted. Indices greater than 0 map to a pen station: index 1 corresponds to pen station 1, index 2 to station 2, etc. One pen station is available on a 4662, eight on a 4662 equipped with Option 31, and two on a 4663. When an index greater than the highest available pen station is specified, a message is printed on the terminal requesting the operator to load the pen corresponding to the specified index and enter the pen station number used (when more than one station is available).

Table 7-3 (cont)

DEVICE DRIVER DD466X.SYS INFORMATION

Characteristic	Description
Line Styles	On a 4662, only one linestyle (solid) is available. On a 4663, there are four linestyles available in addition to solid.
Generalized Drawing Primitives	One GDP is available: 1 -Bar (hollow only)
Escapes	Only escape 1--Inquire Addressable Character Cells-- is available.
Text	Continuous text scaling and rotation is available.
Markers	Five markers are available: 1:"."; 2:"+", 3:"*", 4:"0", and 5:"X".

DTI

The Direct Terminal Interface (DTI) for the 4170 is a set of FORTRAN-86 subroutines that support the features of 4100 Series terminals. Every 4100 Series terminal command has a corresponding DTI routine.

The DTI is useful for applications that use the special features of the Tektronix 4100 Series terminals (the 4105, 4107, 4109, 4112, 4113, 4114, 4115, and 4116). (GSX-86 is recommended for applications that use other graphics devices, including non-Tektronix devices for which a GSX-86 device driver is available. Plot 10 IGL provides a device-independent set of subroutines that provide higher-level graphics functions compared to the DTI and GSX-86, and that support Tektronix graphics devices. See the previous discussions of IGL and GSX-86.)

The FORTRAN source code is contained in the files FDTIA.FOR, FDTIB.FOR, and FDTIC.FOR; CHARIO.A86, which is in Intel assembly language, provides character input and output for DTI subroutines. All these files are on the DTI/BIOS Source disk. The file FDTI.LIB (on one of the FORTRAN diskettes) is an object library of the FORTRAN source files.

USING THE DTI ON THE 4170

The discussion here summarizes the differences between using DTI on a 4170 and using the DTI supplied with 4100 Series Local Programmability. The 4110 Series DTI Programmers Reference Manual, which is included with the Tektronix 4010C01 PLOT 10 IGL Users Manual, describes the use of the DTI supplied with 4100 Series Local Programmability. This discussion focuses on 4170 DTI and how it differs from 4100 Series DTI.

The principal difference between 4170 DTI and 4100 Series Local Programmability DTI is how graphic input (GIN) reports and terminal status and settings reports are handled. This is discussed in more detail in the remainder of this section.

Another difference is that certain routines may not be present or may cause undesirable results during execution, depending on the type of 4100 Series terminal you are using. (This is because of the differences in terminal features and is not the fault of 4170 DTI.) Table 7-4, which follows, charts the compatibility of certain 4170 DTI routines against the various 4100 Series terminals.

You should call the DTI routine LLINIT (DTI-Initialize) before any other DTI routine. LLINIT initializes terminal communications parameters (such as the EOL string, the report-EOM-frequency, and the transmit delay), and it determines and returns the terminal type. The terminal is then left in Select Code TEK.

NOTE

The GIN decoding routines (LLGTGN, LLGTG4, and LLGT10) will not work properly if the EOL string is set to other than <CR> or if the report-EOM-frequency is set to less frequent. It is also recommended that you set the transmit delay to a non-zero value. (LLINIT initializes the EOL string to <CR>, the report-EOM-frequency to more frequent, and the transmit delay to 50 ms, so that the GIN decoding routines should work properly.)

TERMINAL MODES

The DTI routines described in the 4110 Series DTI Programmers Reference Manual do not set terminal modes. For example, LLMOVE (move) does not cause the terminal to enter Vector mode. This is not the most efficient way to send graphics information to the terminal. For applications that send thousands of vectors, there is a significant speed improvement when optimized 4014-style graphics are used.

Your 4170 DTI contains the routines LLMV14 (ix,iy) and LLDR14 (ix,iy), which do optimized graphics. LLMV14 causes the terminal to enter Vector mode and must be called before LLDR14. To exit Vector mode, use the DTI routine LLAMOD (DTI-Set-Alpha-Mode). Markers may be drawn by first calling LLMMOD (DTI-Set-Marker-Mode), followed by a sequence of LLDR14 for the remainder of the markers. LLAMOD should be used to exit Marker mode.

Most 4100 Series terminals are capable of understanding ANSI X3.64 alpha editing escape sequences. (These include alpha functions, such as cursor positioning, and deleting and inserting characters, as well as visual attributes, such as underlining and reverse video.) These escape sequences are not compatible with 4100 Series escape sequence commands. The command LLCODE (Select-Code) instructs the terminal to interpret escape sequences according to either ANSI X3.64 standards or 4100 (TEK) format. Because the DTI does not support the ANSI X3.64 escape sequences, these can be sent using FORTRAN WRITE statements after LLCODE places the terminal in ANSI X3.64 mode; the terminal must then be put back into Select Code TEK before DTI routines can be used. (LLINIT also sets Select-Code TEK.)

Graphic Input

Only 4010-style Graphic Input (GIN) is available on the 4105 terminal and only one point can be enabled at one time. Here is a simple GIN input example using the DTI:

```
C * Example GIN for the 4105
        Character*1 Keychr
        Call Llinit (iterm)
C * enable 4010 style GIN using the joydisk
        Call Llgn10
C * read the report from the terminal
        Call Llgt10 (Keychr,Jx,Jy)
C * print out the results
        Call Llmove (0,2850)
        Write (6,10) Keychr,Jx,Jy
10      Format ('The key hit was "',A1,'", at the point:',I5,
     $   2x,I5)
        End
```

Note that no signature characters can be defined for the 4105. Also, LLGT10 should be called once after each LLGN10.

For the other 4100 terminals, more than one point can be input at a time, and segment picking and tablets can be used; because GIN has more options, the GIN commands can be more complex. The DTI routines LLENGN (Enable-GIN) and LLGTGN (DTI-Get-GIN-Report) are used with 4100-style GIN. When GIN is enabled for more than one point, the report signature characters should be set to non-null values. This short program illustrates the process:

```
C * A simple etch a sketch GIN Demo Program
      Character*1 Keychr, Sigchr
      Call Llinit (iterm)
      If (iterm.eq.4105) Stop '***cannot do this on a 4105'
C * Make a little dialog area
      Call Lldaen (1)
      Call Lldavs (0)
      Call Lldaln (4)
      Call Lldacl
      Call Lldavs (1)
C * Set signature characters to "S" and "E"
      Call Llrpsg (-1,83,69)
C * Enable rubberbanding from the gin display start point
      Call Llrbgn (0,2)
C * Move the beam, the graphics cursor, and the gin-display-
C *   start-point to about the middle of the screen
      Call Llmove (2000,1000)
      Call Lltnsg (0,2000,1000)
      Call Llspgn (0,2000,1000)
C * Enable joydisk (or thumbwheel) gin for 100 points
      Call Llengn (0,100)
C * If key "M" or "m", do a move; if "S" or "s", disable GIN
C *  (and wait for the terminating-GIN-report); for any other
C *  character, draw from the last point to the current point.
C *
100   Call Llgtgn (Sigchr,Keychr,Jx,Jy,Jsegno,Jpickid)
C *
C * Exit only when the terminating signature character is received
      If (Sigchr.eq.'E' .or. Sigchr.eq.'e') Go to 999
      If (Keychr.eq.'M' .or. Keychr.eq.'m') Then
            Call Llmove (jx,jy)
      Else if (Keychr.eq.'S' .or. Keychr.eq.'s') Then
            Call Lldsgn (0)
      Else
            Call Lldraw (Jx,Jy)
      Endif
      Go go 100
C * Exit
999   Print *,'GIN Demo Over'
      End
```

The DTI routine LLGTGN is only able to understand the GIN report when GIN is enabled with LLENGN. The Report-EOM-Frequency must be set to "more frequent"; this value is set by LLINIT.

The DTI routine LLCKGN (DTI-Check-for-GIN-Report) cannot tell when a GIN-Report is waiting. It always returns a value of 1.

An application can read and parse GIN reports without using LLGTGN.

NOTE

The GIN decoding routines LLGTGN and LLGTG4 do not correctly interpret terminal GIN reports when the <CR> (Return) key is used to enter a GIN report. <CR> can only be used when known signature characters are used. A user-written GIN decoding routine is recommended if <CR> is used to enter GIN reports.

Consult the programmers reference manual for your terminal for more information on graphic input.

TERMINAL AND ROUTINE COMPATIBILITY

The 4170 DTI does not verify that the terminal used supports the commands that are sent. It is the responsibility of the applications programmer to avoid using 4170 DTI routines that the target terminal cannot support.

Table 7-4 charts the compatibility of Tektronix 4100 Series terminals against 4170 DTI routines. The entries in the columns have these meanings:

blank = The routine is not supported by that terminal.
X = The DTI routine is present and useful for that terminal.
N = The routine is not supported by 4170 DTI, or a no-op results.
E = The routine supports 32-bit extended addressing; on other than a 4115, the coordinates must be 0-4095. When 32-bit addressability is used on a 4115, the DTI routine LLCORD (Set-Coordinate-Mode) should be used to inform the DTI to use extended addressing and to send the Set-Coordinate-Mode escape sequence to the 4115.
N/A = The compatibility question is not applicable; this is used for some internal 4170 DTI routines.

In the "comments" column, the functional names of some 4170 DTI routines are given, followed by an asterisk (*). Routines with the asterisk (*) are **not** documented in the 4110 Series DTI Programmers Reference Manual; instead, you may look up the corresponding terminal command (under the functional name in the "comments" column) in the terminal's programmers reference manual. For example, for a description of the LLDNVM routine, look up the "Define Nonvolatile Macro" command in the programmers reference manual.

Table 7-4

TERMINAL/4170 DTI COMPATIBILITY CHART

DTI Routine	4105	4107	4109	4112	4113	4114 4116	4115	Comments
LLINIT	X	X	X	X	X	X	X	Must be called first.
LLCKGN	N	N	N	N	N	N	N	Always returns 1
LLSYNC	N	N	N	N	N	N	N	No-op
LLGTFW	N	N	N	N	N	N	N	No-op
LLKLFW	N	N	N	N	N	N	N	No-op
LLGTGN	X	X	X	X	X	X	X	Doesn't work if <CR> key is used to enter a GIN report
LLGTG4	E	E	E	E	E	E	E	Doesn't work if <CR> key is used to enter a GIN report
LLGT10	X	X	X	X	X	X	X	Gets a 4010 GIN report
LLXY10	X	X	X	X	X	X	X	Not in 4100 Series DTI; decodes 4010 xy-reports
LLGTRP	X	X	X	X	X	X	X	
LLKLRP	N	N	N	N	N	N	N	Not supported on 4170 DTI
LLGTCH	X	X	X	X	X	X	X	
LLGTIN	X	X	X	X	X	X	X	
LLGTXY	X	X	X	X	X	X	X	

Table 7-4 (cont)

TERMINAL/4170 DTI COMPATIBILITY CHART

DTI Routine	4105	4107	4109	4112	4113	4114 4116	4115	Comments
LLGT14	E	E	E	E	E	E	E	
LLGTRL	X	X	X	X	X	X	X	
LLABLK				X	X	X	X	Requires Option 01 (Half Duplex & Block Mode)
LLBFIL				X	X		X	
LLBGCH		X	X	X	X	X	X	
LLBHSG		X	X	X	X	X	X	
LLTB53				X	X	X	X	Requires Option 13/14 (tablet option)
LLEFIL				X	X		X	
LLEGCH		X	X	X	X	X	X	
LLEPNL	X	X	X	X	X		X	
LLCLSG		X	X	X	X	X	X	
LLAMOD	X	X	X	X	X	X	X	
LLBYPS	X	X	X	X	X	X	X	
LLMMOD	X	X	X	X	X	X	X	
LLVMOD	X	X	X	X	X	X	X	
LLEMAC	X	X	X	X	X	X	X	
LLFRMT				X	X	X	X	
LLTEXT	X	X	X	X	X	X	X	

Table 7-4 (cont)

TERMINAL/4170 DTI COMPATIBILITY CHART

DTI Routine	4105	4107	4109	4112	4113	4114 4116	4115	Comments
LLHCPY	X	X	X	X	X	X	X	
LLIGDL	X	X	X	X	X	X	X	
LLINSG		X	X	X	X	X	X	
LLLFCR		X	X	X	X	X	X	
LLLOAD				X	X	X	X	
LLKBLK	X	X	X	X	X	X	X	
LLVKLK		X	X	X	X		X	
LLINPN		X	X	X	X	X	X	
LLMOVE	X	X	X	X	X	X	X	
LLMOV4	E	E	E	E	E	E	E	
LLMV14	X	X	X	X	X	X	X	
LLPXCP	X	X	X	X	X		X	4105 requires pixel operation ROMs
LLPXRP							X	
LLPLOT		X	X	X	X	X	X	
LLPASG		X	X	X	X	X	X	
LLPCPY				X	X	X	X	
LLAPRM	X	X	X	X	X	X	X	
LLBLSG		X	X	X	X	X	X	
LLBNSG		X	X	X	X	X	X	
LLBPNL	X	X	X	X	X		X	

Table 7-4 (cont)

TERMINAL/4170 DTI COMPATIBILITY CHART

DTI Routine	4105	4107	4109	4112	4113	4114 4116	4115	Comments
LLBPN4	E	E	E	E	E		E	
LLBPX1	X	X	X	X	X		X	4105 requires pixel operation ROMs
LLOPSG		X	X	X	X	X	X	
LLCNCL	X	X	X	X	X	X	X	
LLDACL	X	X	X	X	X	X	X	
LLCOPY	X	X	X	X	X	X	X	Available devices are terminal dependent
LLCRLF	X	X	X	X	X	X	X	
LLDMAC	X	X	X	X	X	X	X	
LLDNVM	X	X	X					Define Nonvolatile Macro *
LLDELF				X	X	X	X	
LLDGCH	X	X	X	X	X	X	X	
LLDLSG		X	X	X	X	X	X	
LLDLVW		X	X	X	X		X	
LLDIR				X	X	X	X	

* This DTI routine is not documented in the 4110 Series DTI Programmers Reference Manual. Refer to the programmers reference manual for your terminal.

Table 7-4 (cont)

TERMINAL/4170 DTI COMPATIBILITY CHART

DTI Routine	4105	4107	4109	4112	4113	4114 4116	4115	Comments
LLDSGN		X	X	X	X	X	X	
LLDSMT				X	X	X	X	
LLDRAW	X	X	X	X	X	X	X	
LLDRA4	E	E	E	E	E	E	E	
LLDR14	X	X	X	X	X	X	X	
LLMRKR	X	X	X	X	X	X	X	
LLMRK4	E	E	E	E	E	E	E	
LLREC4							X	
LLDAEN	X	X	X	X	X	X	X	
LLENGN		X	X	X	X	X	X	
LLENKE	X	X	X					Enable Key Expansion *
LLGN10	X	X	X	X	X	X	X	Only style GIN for 4105
LLPROT				X	X	X	X	
LLRASW	X	X	X	X	X		X	4105 requires pixel operation ROMs
LLRCFL	X	X	X	X	X		X	
LLRENM				X	X	X	X	

* This DTI routine is not documented in the 4110 Series DTI Programmers Reference Manual. Refer to the programmers reference manual for your terminal.

Table 7-4 (cont)

TERMINAL/4170 DTI COMPATIBILITY CHART

DTI Routine	4105	4107	4109	4112	4113	4114 4116	4115	Comments
LLRNSG		X	X	X	X	X	X	
LLRNVW		X	X	X	X	X	X	
LLQQRY					X		X	Requires Option 09
LLJQRY		X	X	X	X	X	X	Available devices terminal dependent
LLKQRY	X	X	X	X	X	X	X	
LLERRP	N	N	N	N	N	N	N	Not supported on 4170 DTI
LLPTGN		X	X	X	X	X	X	
LLPQRY		X	X	X	X	X	X	
LLSQRY		X	X	X	X	X	X	
LLIQRY		X	X	X	X	X	X	
LLST10		X	X	X	X	X	X	
LLHCRE				X	X		X	Requires Option 09
LLREST		X	X	X	X	X	X	
LLRUNW	X	X	X	X	X		X	4105 requires pixel operation ROMs
LLRUN4							E	

Table 7-4 (cont)

TERMINAL/4170 DTI COMPATIBILITY CHART

DTI Routine	4105	4107	4109	4112	4113	4114 4116	4115	Comments
LLSAVE				X	X	X	X	
LLNVSV	X	X	X					Save Nonvolatile Parameters *
LLASZG						X		
LLCODE	X	X	X	X	X		X	
LLSFIL	X	X	X	X	X		X	
LLHCIN	X	X	X	X	X		X	4110 terminals require Option 09
LLSLVW		X	X	X	X		X	
LLACI	X	X	X					Set Alpha Cursor Index *
LLALSZ						X		
LLBGCL		X	X	X	X		X	
LLBGGR				X	X		X	
LLBGIN		X	X	X	X		X	
LLBAUD	X	X	X	X	X	X	X	
LLCBLK				X	X	X	X	Requires Option 01 (Half Duplex & Block mode)

* This DTI routine is not documented in the 4110 Series DTI Programmers Reference Manual. Refer to the programmers reference manual for your terminal.

Table 7-4 (cont)

TERMINAL/4170 DTI COMPATIBILITY CHART

DTI Routine	4105	4107	4109	4112	4113	4114 4116	4115	Comments
LLEBLK				X	X	X	X	Requires Option 01 (Half Duplex & Block mode)
LLHBLK				X	X	X	X	Requires Option 01 (Half Duplex & Block mode)
LLBBLK				X	X	X	X	Requires Option 01 (Half Duplex & Block mode)
LLLBLK				X	X	X	X	Requires Option 01 (Half Duplex & Block mode)
LLMBLK				X	X	X	X	Requires Option 01 (Half Duplex & Block mode)
LLXBLK				X	X	X	X	Requires Option 01 (Half Duplex & Block mode)
LLPBLK				X	X	X	X	Requires Option 01 (Half Duplex & Block mode)
LLTBLK				X	X	X	X	Requires Option 01 (Half Duplex & Block mode)
LLBORD		X	X	X	X		X	

Table 7-4 (cont)

TERMINAL/4170 DTI COMPATIBILITY CHART

DTI Routine	4105	4107	4109	4112	4113	4114 4116	4115	Comments
LLBKTM	X	X	X	X	X	X	X	
LLBYCH	X	X	X	X	X	X	X	
LLCHPA	X	X	X					Set Character Path *
LLHCDA				X	X		X	Requires Option 09
LLCLMD		X	X	X	X		X	
LLHCSZ	X	X	X					Set Copy Size *
LLCORD							X	
LLMTCL		X	X	X	X	X	X	
LLDAAI				X	X		X	
LLDABF	X	X	X	X	X	X	X	
LLDACH				X	X	X	X	
LLDACM	X	X	X					Set Dialog Area Color Map *
LLDAIN	X	X	X	X	X		X	
LLDALN	X	X	X	X	X	X	X	
LLDAXY				X	X	X	X	

* This DTI routine is not documented in the 4110 Series DTI Programmers Reference Manual. Refer to the programmers reference manual for your terminal.

Table 7-4 (cont)

TERMINAL/4170 DTI COMPATIBILITY CHART

DTI Routine	4105	4107	4109	4112	4113	4114 4116	4115	Comments
LLDASF				X	X		X	
LLDAVS	X	X	X	X	X	X	X	
LLDAWM	X	X	X	X	X	X	X	
LLDHCA	X	X	X					Set Dialog Hardcopy Attributes *
LLDMAB							X	Requires Option 03 (DMA)
LLDRBM							X	
LLDUPX				X	X	X	X	Requires Option 01 (Half Duplex & Block mode)
LLECHO	X	X	X	X	X	X	X	
LLEDCH	X	X	X	X	X	X	X	
LLEOF	X	X	X	X	X	X	X	
LLEOL	X	X	X	X	X	X	X	
LLEOM	X	X	X	X	X	X	X	
LLERTH	X	X	X	X	X	X	X	
LLFXUP		X	X	X	X	X	X	

* This DTI routine is not documented in the 4110 Series DTI Programmers Reference Manual. Refer to the programmers reference manual for your terminal.

Table 7-4 (cont)

TERMINAL/4170 DTI COMPATIBILITY CHART

DTI Routine	4105	4107	4109	4112	4113	4114 4116	4115	Comments
LLFLAG	X	X	X	X	X	X	X	
LLARGN		X	X	X	X	X	X	
LLCRGN		X	X	X	X	X	X	
LLCLGN		X	X					
LLGSPD	X	X	X					Set Gin Cursor Speed *
LLSPGN		X	X	X	X	X	X	
LLSPG4		E	E	E	E	E	E	
LLGRGN		X	X	X	X	X	X	
LLGRG4		E	E	E	E	E	E	
LLIKGN		X	X	X	X	X	X	
LLRBGN		X	X	X	X	X	X	
LLTBSF		X	X	X	X	X	X	
LLWIGN		X	X	X	X	X	X	
LLWIG4		E	E	E	E	E	E	
LLGAWM	X	X	X	X	X	X	X	
LLGFNT		X	X	X	X	X	X	
LLGGRD		X	X	X	X	X	X	

* This DTI routine is not documented in the 4110 Series DTI Programmers Reference Manual. Refer to the programmers reference manual for your terminal.

Table 7-4 (cont)

TERMINAL/4170 DTI COMPATIBILITY CHART

DTI Routine	4105	4107	4109	4112	4113	4114 4116	4115	Comments
LLGPRC	X	X	X	X	X	X	X	
LLGROT		X	X	X	X	X	X	Angles limited on 4105 to 0, 90, 180, 270
LLGSIZ	X	X	X	X	X	X	X	
LLGSI4	E	E	E	E	E	E	E	
LLGSLT		X	X	X	X	X	X	
LLHCOR				X	X		X	Requires Option 09
LLKYEX	X	X	X	X	X	X	X	
LLLNIN	X	X	X	X	X	X	X	
LLLNST	X	X	X	X	X	X	X	
LLLNWD						X		
LLMARG						X		
LLMKTY	X	X	X	X	X	X	X	
LLHCPS				X	X		X	Requires Option 09
LLOVWI				X	X		X	
LLOVW4				E	E		E	
LLPGFL				X	X	X	X	
LLFPNL				X	X		X	
LLPRTY	X	X	X	X	X	X	X	

Table 7-4 (cont)

TERMINAL/4170 DTI COMPATIBILITY CHART

DTI Routine	4105	4107	4109	4112	4113	4114 4116	4115	Comments
LLPKAP		X	X	X	X	X	X	
LLPKID		X	X	X	X	X	X	
LLPVSG		X	X	X	X	X	X	
LLPVS4		E	E	E	E	E	E	
LLPXBM	X	X	X	X	X		X	4105 requires pixel operation ROMs
LLPXVW	X	X	X	X	X		X	4105 requires pixel operation ROMs
LLPBAU		X	X	X	X	X	X	4110 terminals require Option 10 (3PPI)
LLPEOF		X	X	X	X	X	X	4110 terminals require Option 10 (3PPI)
LLPEOL		X	X	X	X	X	X	4110 terminals require Option 10 (3PPI)
LLPFLG		X	X	X	X	X	X	4110 terminals require Option 10 (3PPI)

Table 7-4 (cont)

TERMINAL/4170 DTI COMPATIBILITY CHART

DTI Routine	4105	4107	4109	4112	4113	4114 4116	4115	Comments
LLPPRY		X	X	X	X	X	X	4110 terminals require Option 10 (3PPI)
LLPBIT		X	X	X	X	X	X	4110 terminals require Option 10 (3PPI)
LLSPRM	X	X	X	X	X	X	X	
LLQSIZ	X	X	X	X	X	X	X	
LLEOMF		X	X	X	X	X	X	
LLRPMX		X	X	X	X	X	X	
LLRPSG		X	X	X	X	X	X	
LLCSSG		X	X	X	X	X	X	
LLDTSG		X	X	X	X	X	X	
LLPRSG		X	X	X	X	X	X	
LLHISG		X	X	X	X	X	X	
LLIMSG		X	X	X	X	X	X	
LLIMS4		E	E	E	E	E	E	
LLTNSG		X	X	X	X	X	X	
LLTNS4		E	E	E	E	E	E	
LLVISG		X	X	X	X	X	X	

Table 7-4 (cont)

TERMINAL/4170 DTI COMPATIBILITY CHART

DTI Routine	4105	4107	4109	4112	4113	4114 4116	4115	Comments
LLWMSG		X	X	X	X	X	X	
LLSNPY	X	X	X	X	X	X	X	
LLSTBT	X	X	X	X	X	X	X	
LLCLMP	X	X	X	X	X		X	
LLDFSF		X	X	X	X		X	
LLGRSF				X	X		X	
LLPRSF		X	X	X	X		X	
LLVISF		X	X	X	X		X	
LLTBSZ				X	X	X	X	Requires Option 13/14 (tablet option)
LLTBHD				X	X	X	X	Requires Option 13/14 (tablet option)
LLTBSS				X	X	X	X	Requires Option 13/14 (tablet option)
LLTABS	X	X	X					Set Tab Stops *
LLTXIN	X	X	X	X	X	X	X	

* This DTI routine is not documented in the 4110 Series DTI Programmers Reference Manual. Refer to the programmers reference manual for your terminal.

Table 7-4 (cont)

TERMINAL/4170 DTI COMPATIBILITY CHART

DTI Routine	4105	4107	4109	4112	4113	4114 4116	4115	Comments
LLXMTD		X	X	X	X	X	X	
LLXMTL	X	X	X	X	X	X	X	
LLUSER				X	X	X	X	
LLVWAT		X	X	X	X		X	
LLVWDC		X	X	X	X		X	
LLVWPT		X	X	X	X		X	
LLWIND	X	X	X	X	X		X	
LLWIN4	E	E	E	E	E		E	
LLAS14	X	X	X	X	X	X	X	
LLSPOL				X	X	X	X	
LLESPL				X	X	X	X	
LLWAIT	X	X	X	X	X	X	X	In 4170 DTI, sends SYNC characters
INTIX	N/A	N/A	N/A	N/A	N/A	N/A	N/A	Internal 4170 DTI Utilities
INRYIX	N/A	N/A	N/A	N/A	N/A	N/A	N/A	Internal 4170 DTI Utilities
STRIX	N/A	N/A	N/A	N/A	N/A	N/A	N/A	Internal 4170 DTI Utilities

Table 7-4 (cont)

TERMINAL/4170 DTI COMPATIBILITY CHART

DTI Routine	4105	4107	4109	4112	4113	4114 4116	4115	Comments
XY1X	N/A	N/A	N/A	N/A	N/A	N/A	N/A	Internal 4170 DTI Utilities
XY41X	N/A	N/A	N/A	N/A	N/A	N/A	N/A	Internal 4170 DTI Utilities
GETCHR	N/A	N/A	N/A	N/A	N/A	N/A	N/A	Internal 4170 DTI Utilities
PUTCHR	N/A	N/A	N/A	N/A	N/A	N/A	N/A	Internal 4170 DTI Utilities

Section 8

SELF TEST

INTRODUCTION

This section describes the 4170's self test features. It is divided into two subsections according to the intended reader. If you are NOT a service technician, see "WHAT THE USER NEEDS TO KNOW." You need not bother with the rest of this section.

Service technicians should refer to "WHAT THE SERVICE TECHNICIAN NEEDS TO KNOW," later in this section.

WHAT THE USER NEEDS TO KNOW

The 4170 includes a variety of self-test routines. Some of these ("Power-Up Self Test") are performed automatically every time power is applied. Others ("Extended Self Test" and "Adjustment Self Test") are performed only on request. The Adjustment Self Test is only for the service technician; it is described later in this section.

POWER-UP SELF TEST

To perform the Power-Up Self Test, turn on the 4170, or (if it is already turned on), press the RESET switch on the 4170 front-panel keyboard.

If no errors are detected, Power-Up Self Test ends with "00" displayed in the STATUS digits on the 4170 front panel and the 4170 proceeds to load the operating system.

If Power-Up Self Test detects an error it sounds the bell, displays an error code in the STATUS digits, and turns on four decimal points immediately below those STATUS digits. See "Error Code Meanings," later in this section.

SECTION 8
SELF TEST

EXTENDED SELF TEST

To run Extended Self Test, connect a terminal to Terminal Port (also referred to as Port 0), set the terminal communications parameters to the factory default values (see section 4, Connecting Cables) and proceed as follows:

1. On the 4170 front-panel keyboard, simultaneously press RESET and TEST.

2. Release the RESET switch, but continue to press TEST.

3. The STATUS display will display FF, then (very briefly) FE, and then will display the following sequence: 01, 02, 04, 08, 10, 20, 40, 80. When this happens release the TEST switch.

4. The bell will sound, the LEDs will show FC, and the dots at the bottom of the LEDs will blink. When this happens, press any switch (except TEST or RESET) on the 4170 front panel.

5. The Extended Self Test will then continue. As it does, the STATUS lights display a sequence of two-digit hexadecimal codes. (These codes tell a service technician which part of Extended Self Test is being performed.)

6. If an error is detected, the bell will sound, an error code will be displayed in the STATUS digits, and THERE WILL BE FOUR DECIMAL POINTS DISPLAYED IMMEDIATELY BELOW THE STATUS DIGITS. If that happens, see "Error Code Meanings," the next topic in this section. (It is not an error code if the decimal points are not displayed).

7. If no errors are detected, the test will end with "00" displayed in the STATUS lights on the 4170 front panel and the 4170 will proceed to load the operating system.

ERROR CODE MEANINGS

When Power-Up Self Test or Extended Self Test detects an error, the 4170 displays an error code in the STATUS lights on its front panel, together with four decimal points immediately below the STATUS lights.

Sometimes several error codes will be displayed in sequence. The first error code is the primary one, and is listed in the following table. Subsequent error codes are "submessages;" you can invoke them by pressing the CONT key on the 4170 front panel. The 4170 will display them anyway after a wait of about 20 seconds. Pay attention only to the FIRST error code.

NOTE

If the code in the STATUS lights is not accompanied by four decimal points immediately below, then it is NOT an error code.

ERROR CODE ("x" means "any digit")	POSSIBLE DEFECTIVE MODULE ("X" marks the modules which may be defective.)	Opt. 45: Mass Storage Interface Board	Opt. 44: Floppy Disk Controller Board	Opt.10: Second 3PPI Board	Opt. 9: Color Copier Interface Board	3PPI Board	ECC Memory Board	Front Panel Keyboard	Processor Board
00	"No error"								
5x									X
6x								X	
8x									X
Bx								X	
CE									X
EC									X
EE									X
DF								X	X
Ex									X
FC	"Waiting for you to press a key"								
FD								X	X
FE								X	X

SECTION 8
SELF TEST

Table 8-1 lists the (primary) error codes that may be displayed during Power-Up Self Test or Extended Self Test. For each code, the table shows which modules in the 4170 may be at fault. Those are the most likely modules to replace.

Error codes which point to the Disk Controller modules ("8x" codes for the Option 44 Floppy Disk Controller, or "5x" codes for the Option 45 Mass Storage Interface Board) may refer to errors in the cabling to the disk drives or even the disk drives themselves. However, most of these error codes are for tests which exercise only circuitry on the disk controller board itself.

Table 8-1

SELF-TEST SEQUENCE

Hex Digits	Explanation	Executed (a)
FF	Beginning of test.	---
FE	Start of LED checking routine. (During LED checking routine, LEDs T1, T2, and T3 light in sequence, followed by the displays 01, 02, 04, 08, 10, 20, 40, 80; then this is repeated once again, more rapidly.)	SLF
FD	End of LED checking routine.	SLF
FC	Front panel key check.. (the bell sounds, a menu is displayed, and the user is expected to press a key on the front panel keyboard.)	SLF
EF	Timer check	SLF/PUP
EE	Timer set up routine	SLF/PUP
EC	Processor board ROM check	SLF/PUP
DF	Low 32K RAM/Bus address check	SLF/PUP
BF	Low 32K RAM Walking ones check	SLF
BE	Low 32K RAM Walking zero check	SLF
BD	Low 32K RAM all ones check	SLF/PUP
BC	Low 32K RAM all zeros check	SLF/PUP
B5	RAM stack building	SLF/PUP
BA	RAM/ROM memory map tables building	SLF/PUP
DE	Upper RAM bus check	SLF/PUP
B9	Upper RAM walking ones check	SLF

Table 8-1 (cont)

SELF-TEST SEQUENCE

Hex Digits	Explanation	Executed (a)
BB	Upper RAM memory delay check	SLF
B8	Upper RAM walking zero check	SLF
B7	Upper RAM all ones check	SLF/PUP
B6	Upper RAM all zeros check	SLF/PUP
CE	System ROM checksums	SLF/PUP
EB	Interrupt check	SLF/PUP
E9	8087 FPU test	SLF/PUP
DD	Host port register check	SLF/PUP
DC	Host port baud/character check	SLF
6F	3PPI Board Register Check	SLF/PUP
6D	3PPI Board Character Check	SLF
67	Option 10 Board Register Check	SLF/PUP
65	Option 10 Board Character Check	SLF
5F	Option 45 8089 ROM Checksum Test	SLF/PUP
5E	Option 45 8089 RAM Check Test	SLF/PUP
5D	Option 45 MSIB Read/Write Verification	SLF
5C	Option 45 Flexible Disk Status Register Test	SLF/PUP
5B	Option 45 Flexible Disk Controller Initialization	SLF

Table 8-1 (cont)

SELF-TEST SEQUENCE

Hex Digits	Explanation	Executed (a)
5A	Option 45 Flexible Disk Interrupt Check	SLF
59	Option 45 8089 Processor Check	SLF
53	Option 45 MSIB Automatic Acknowledge Test	SLF
8F	Option 44 Board Status Register Check	SLF/PUP
8E	Option 44 Disk Controller Check	SLF/PUP
8D	Option 44 Drive Present Check	SLF/PUP
8C	Option 44 Intersystem Interrupt Check	SLF
8B	Option 44 DMA Operation Check	SLF
8A	Option 44 DMA Addressing Check	SLF

(a) SLF means the test is performed during Self Test, while PUP means the test is performed during the Power Up routine.

WHAT THE SERVICE TECHNICIAN NEEDS TO KNOW

NOTE

This is a preliminary manual. At the time it went to print, not all information about Self Test was available. Additional information will be provided in the final version of this manual.

The 4170 has three self test programs: Power-Up Self Test, Extended Self Test, and the Adjustment Self Test.

POWER-UP SELF TEST

The 4170 performs its Power-Up Self Test automatically whenever it is turned on, or whenever the operator presses the RESET switch on the front panel.

If the test reveals a fault in the 4170's circuitry, the test halts with the following displayed:

(a) A two-digit error code displayed on the 4170's front panel; and

(b) Four decimal points displayed below the two-digit code.

All the tests in the Power-Up Self Test are also performed in the Extended Self Test. For information about particular error codes, see the following Extended Self Test description.

EXTENDED SELF TEST

Starting Extended Self Test

To start the Extended Self Test program, press and hold the TEST switch while pressing and releasing the RESET switch. When the front-panel LEDs begin to light sequentially (cycle), release the TEST switch.

Initial Tests

The Extended Self Test program begins by testing the front-panel indicators. The test starts by sequencing the three LEDs and the two LED numeric displays. First the program turns off the LEDs and displays "00" on the LED displays. Then, each LED (starting with T1) is turned on individually. Next, the LED displays show the following series of numbers: 01, 02, 04, 08, 10, 20, 40, 80. This series is repeated quickly.

Keyboard Test

Next, the 4170 tests its front-panel keyboard. At the same time, you have a chance to select between (a) continuing the Extended Self Test in its entirety, (b) continuing the Extended Self Test but omitting the lengthy memory delay test, or (c) selecting the Adjustment Self Test.

The Keyboard Test works as follows. The 4170 sounds its bell, blinks the T1 LED on the front panel, and waits for a keystroke. At this point, one of the following front panel keys should be pressed:

Key Pressed	Result
CTRL-C	4170 starts the Adjustment Self Test (after PUP tests are finished)
CTRL-D	4170 continues with the Extended Self Test BUT OMITS THE MEMORY DELAY TEST
ANY OTHER KEY	4170 continues with the Extended Self Test (including the Memory Delay Test).

NOTE

If the test proceeds immediately (without pressing a key), spurious keystrokes are being generated. This could be caused by dirt or corrosion in the keys or poor continuity in the connecting cable.

If you wait more than about 20 seconds before pressing a front panel switch, the test will continue as if you had selected the "ANY OTHER KEY" option.

Remainder Of Extended Self Test

Provided the CTRL-D option was not selected, the 4170 continues with the rest of Extended Self Test. As Extended Self Test continues running, the seven-segment displays indicate the hexadecimal code number of the particular test being run. If a fatal error occurs during a test, the identifying error code remains on the displays and the bell sounds twice.

NOTE

Error codes on the seven-segment displays are accompanied by four decimal points. Any lingering codes that are not accompanied by these decimal points are NOT error codes.

The tables that follow list and define the error messages, including submessages, giving an explanation for each error condition and suggesting which part of the circuitry is malfunctioning. The tables are grouped together according to the hardware being tested. (This is not the order in which the tests are performed.)

After the error code is displayed, most problems can be narrowed through the use of submessages. First note the primary error code. Then press CONT key and note the next error code (submessage). If CONT is not pressed, the program times out after 20 seconds and displays the submessage. In some cases, there may be several levels of submessages. The error code tables list the submessages for each code.

After the display port (Port 0) has been checked, all error messages are also shown on the screen of the companion terminal, which must be set to 2400 baud. Such screen-displayed error messages first print the name of the test or the hardware module being tested. A submessage on the next line identifies the part of the test that failed.

KEYBOARD AND LED CHECK (FE, FD)

The following error codes may be displayed during the keyboard tests. For a description of these tests, see "Starting Extended Self Test," earlier in this section.

Table 8-2

FRONT PANEL/PROCESSOR INTERFACE ERRORS

Hex Digits	Explanation
FE	Error at beginning of keyboard LED test (LEDs cycle here)
FD	Error at end of keyboard LED test

CIRCUITS USED: Look for problems on the 4170 keyboard, the keyboard interface on the Processor Board, and the keyboard ribbon cable and connectors.

PROCESSOR CHECK (EX)

Extended Self Test now tests the 4170 Processor Board and Keyboard functions. The processor and its related hardware are systematically exercised.

Table 8-3

PROCESSOR BOARD ERRORS

Hex Digits	Explanation
EF	Error detected during timer test. (NOTE) Submessage Hex Digits: XX Binary 00AB 000C, where bits A, B, and C have the following meanings: A=1 Failed static test: Test timer outputs for high/low values. B=1 Failed dynamic test. Tests timer for correct count versus processor execution. C=0 Failure in timer 0 (I/O address 00E1). C=1 Failure in timer 1 (I/O address 00E3). NOTE -- Timer 2 cannot be tested at this point, but is tested later during the host port check (Dx).
EE	Failure during timer initialization.
EC	Error during Processor board ROM checksum test.

Table 8-3 (cont)

PROCESSOR BOARD ERRORS

Hex Digits	Explanation
EB	Error detected in interrupt handler check (NOTE) Submessage Hex Digits: XX First Submessage: Bits that cannot be set in Interrupt Register (I/O base address 00EA). Second Submessage: Level 5 interrupt performed. Failed to reach interrupt vector address: XX + (4 x 5). (Level 5 interrupt is the easiest level of interrupt to force (hence x5). Each time an increment of a level is made, it moves up four bytes in memory (hence the 4). XX equals the base interrupt address.) Third Submessage: A divide by 0 instruction was performed, but did not vector through location 0. XX equals the base interrupt address. (XX=00 most likely implies an version of an 8086 processor IC that is not acceptable for Processor Board operation. NOTE -- Self-test cannot detect if either T1 or T2 inputs to the PIC are held at a TTL low or if the interrupt outputs from the PIC are held low. If these conitions are suspected, check signals with an oscilloscope.
E9	Error detected during Math floating point Co-processor (FPU) tests

COMMUNICATIONS AND BUS CHECKS (DX)

Self Test now examines the operation of the main 4170 bus and the host I/O port. This it does by writing to low RAM (located between 0000 and 7FFF) using the address of a word as the data to go in that word. It then reads this address back, using the comparison as a bus check. This assumes the RAMs are good.

Table 8-4

RAM BUS AND HOST PORT ERRORS

Hex Digits	Submessage Hex Digits	Explanation
DF		Problems with low bits of main 4170 bus or addressing via this bus.
	XX	Submessage 1. XX indicates the base address of the RAM that caused the bus problem: hexadecimal XX000.
	YY	Submessage 2. YY is the Low data byte, showing which bits are in error. If these bit are all ones, address bit zero is likely to be bad.
	ZZ	Submessage 3. ZZ is the High data byte, showing which bits are in error. If these bits are all ones, BHEN (Byte High Enable) is probably bad.
		NOTE: If YY and ZZ are all ones, this indicates a time-out problem while attempting to access a RAM.

Table 8-4 (cont)

BUS AND HOST PORT ERRORS

Hex Digits	Submessage Hex Digits	Explanation
DD		Error detected during host port I/O address check.
	(On screen)	Submessage one: "Host Port-Registers Expect: XX Receive: YY" This test checks the reset of the USART. after resetting USART, it reads the data at status I/O address (E2). It expects to read XX. If instead it receives YY (where XX is not equal to YY) this means the test failed and the USART cannot be reset.
	(On screen)	Submessage two: "Host Port-Register Expect: XX-YY Receive: ZZ-AA" The test checks USART latch (I/O address EB), and USART Interrupt Request (I/O address E8). XX - Latch status expected; is 00 if no bits are in error. YY - Interrupt request bits in error; is 00 if no bits are in error. ZZ - Latch Status bits in error. AA - Interrupt request bits in error.

Table 8-4 (cont)

BUS AND HOST PORT ERRORS

Hex Digits	Submessage Hex Digits	Explanation
DC		Error detected during host port baud rate/ character check.
	(On screen)	Submessage: "Host Port-Baud/Character Baud: XXXX Expect: YY-AA Receive: ZZ-BB"
		The following table may be used to convert the baud rate from a hexadecimal to a decimal numeral:

Hexadecimal	Decimal	Hexadecimal	Decimal
12C	300	E10	3600
25B	600	12C0	4800
4B0	1200	1C20	7200
708	1800	2580	9600
7D0	2000	4B00	19200
960	2400	9600	38400

Self Test cannot check the following signals if they are tied to a TTL low: MWTC; PFAIL; BCLK; INT1; INT2; INT3; INT4; INT6; INT7; BREQ; BGT.

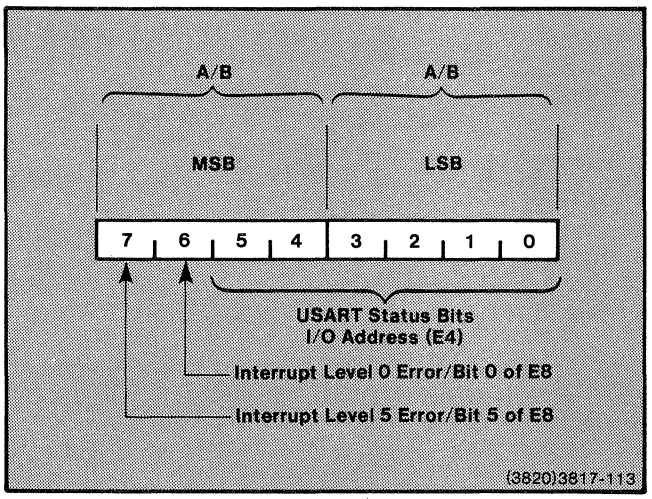


Figure 8-1. Status Byte.

RAM CHECK (BX)

This set of tests perform a systematic check of all system RAMs. Each set of tests is grouped, and each such group uses the same type of error reporting scheme.

In the following tests, the RAM Memory Delay Check (code BB) generates a light code which may remain lit for as much as 14 seconds while the test is running. This does not indicate an error. Errors are indicated by the bell's sounding, together with four decimal points being lit on the two-digit numeric display. The lengthy delay memory check may be bypassed by entering CTRL-D during the earlier keyboard check.

NOTE

In the following table, "Low 32K RAM" refers to the RAM from hexadecimal address 0000 to 7FFF. "Upper RAM" refers to RAM addressed from 8000 through the upper limit of the RAM installed.

Table 8-5

RAM TEST ERRORS

Hex Digits	Sub-message Hex Digits	Explanation
BF		Low 32K RAM, walking ones check. (Here each bit is set and tested, in order from left to right, hence the "walking ones" term.)
	XX	Submessage one. XX indicates the address of the RAM: "Problem at hexadecimal address XX000."
	YY	Submessage two. YY is the Low Data byte, showing which bits are in error.
	ZZ	Submessage three. ZZ is the High Data byte, showing which bits are in error.

Table 8-5 (cont)

RAM TEST ERRORS

Hex Digits	Sub-message Hex Digits	Explanation
BE		Error in "walking zeros" check; low 32K RAM.
	XX	Submessage one. XX indicates the address of the RAM: "Problem at hexadecimal address XX000."
	YY	Submessage two. YY is the Low Data byte, showing which bits are in error.
	ZZ	Submessage three. ZZ is the High Data byte, showing which bits are in error.
BD		Error in "all ones" check; low 32K RAM.
	XX	Submessage one. XX indicates the address of the RAM: "Problem at hexadecimal address XX000."
	YY	Submessage two. YY is the Low Data byte, showing which bits are in error.
	ZZ	Submessage three. ZZ is the High Data byte, showing which bits are in error.
BC		Error in "walking ones" check; low 32K RAM.
	XX	Submessage one. XX indicates the address of the RAM: "Problem at hexadecimal address XX000."
	YY	Submessage two. YY is the Low Data byte, showing which bits are in error.
	ZZ	Submessage three. ZZ is the High Data byte, showing which bits are in error.

Table 8-5 (cont)

RAM TEST ERRORS

Hex Digits	Sub-message Hex Digits	Explanation
BB		Error in RAM memory delay check; low 32K RAM. During this check, data is held for 14 seconds, and then rechecked. The light code will remain lit for this time.
	XX	Submessage one. XX indicates the address of the RAM: "Problem at hexadecimal address XX000."
	YY	Submessage two. YY is the Low Data byte, showing which bits are in error.
	ZZ	Submessage three. ZZ is the High Data byte, showing which bits are in error.
B9		Error in walking ones check; upper RAM.
	XX	Submessage one. XX indicates the address of the RAM: "Problem at hexadecimal address XX000."
	YY	Submessage two. YY is the Low Data byte, showing which bits are in error.
	ZZ	Submessage three. ZZ is the High Data byte, showing which bits are in error.
B8		Error in walking zeros check; upper RAM.
	XX	Submessage one. XX indicates the address of the RAM: "Problem at hexadecimal address XX000."
	YY	Submessage two. YY is the Low Data byte, showing which bits are in error.
	ZZ	Submessage three. ZZ is the High Data byte, showing which bits are in error.

Table 8-5 (cont)

RAM TEST ERRORS

Hex Digits	Sub-message Hex Digits	Explanation
B7		Error in all ones check; upper RAM.
	XX	Submessage one. XX indicates the address of the RAM: "Problem at hexadecimal address XX000."
	YY	Submessage two. YY is the Low Data byte, showing which bits are in error.
	ZZ	Submessage three. ZZ is the High Data byte, showing which bits are in error.
B6		Error in initialization (all zero) check; upper RAM.
	XX	Submessage one. XX indicates the address of the RAM: "Problem at hexadecimal address XX000."
	YY	Submessage two. YY is the Low Data byte, showing which bits are in error.
	ZZ	Submessage three. ZZ is the High Data byte, showing which bits are in error.
B5		Error during RAM stack building.

3PPI CHECK (6X)

This test checks the operation of the 3PPI board. 3PPI error codes are reported only on the 4170 front-panel display.

Table 8-6

3PPI ERRORS

Hex Digits	Sub - message Hex Digits	Explanation
6F		Error detected in 3PPI port register check.
	OP	Submessage one. An error has been been detected in the register check for 3PPI port P. The first digit of this error code is always 0, while the second digit is 0 for port zero, 1 for port 1, or 2 for port 2.
	XX	First two hex digits (MSB) of the 3PPI control register written.
	YY	Second two hex digits(LSB) of the 3PPI control register written.
	ZZ	First two hex digits (MSB) of the expected interrupt data.
	AA	Second two hex digits (LSB) of the expected interrupt data.
	BB	First two hex digits (MSB) indicating the actual interrupt data received.
	CC	Second two hex digits (LSB) indicating the actual interrupt data received.
		Submessages two, three, and four describe the control register pattern sent to the control register for the 3PPI port named in submessage one. The pattern sent to the register was XX, the value expected to be read from the register was YY, and the value actually read was ZZ.

Table 8-6 (cont)

3PPI ERRORS

Hex Digits	Sub - message Hex Digits	Explanation
6F		Error detected during 3PPI character check.
	XX	Submessage two. First two hex digits of baud rate.
	YY	Submessage three. Second two hex digits of baud rate.
	ZZ	Submessage four. The character transmitted during the character test.
	AA	Submessages five. The character received during the character test.
	BB	Submessage six. The interrupt bits expected during the character test.
	CC	Submessage seven. The interrupt bits actually read during the character test.
		EXPLANATION: the baud rate at peripheral port P was set to hexadecimal XXYY. The character transmitted at that port was hexadecimal ZZ, whereas the character received was hexadecimal AA. The interrupt bits expected during the test were hexadecimal BB, while those actually read were hexadecimal CC. (The Character Check test uses internal loopback testing. There is no interaction between this test and external equipment. (Refer to the table earlier in this section to convert the hexadecimal baud rates XXYY to decimal numbers.)

Table 8-6A

3PPI ERRORS (OPTION 10)

Hex Digits	Sub - Message (Hex Digits)	Explanation
67	"Optional 3PPI - Register Register XXXX Expect: YY-ZZ Receive: AA-BB"	Error detected during 3PPI port register check. The 3PPI control register XXXX on optional 3PPI board was written. Bytes yy and zz were expected interrupt data but aa and bb were received.
65	"Optional 3PPI - Charac-ter/Baud Baud: XXXX Expect: YY-ZZ Receive: AA-BB	Error detected during 3PPI port character check. The Baud rate was set to XXXX. During the test the character sent was YY. The character received was ZZ. Interrupt bits expected are AA. What was received was BB.

OPTION 45 DISK CONTROLLER BOARD CHECK (5X)

The Optional Disk Controller Board (Option 45) controls (a) the internal hard disk, (b) the MSIB (Mass Storage Interface Bus), and (c) any Floppy (flexible) disk drives which may be installed. Table 8-7 lists error codes related to this Disk Controller Board.

Table 8-7

OPTIONAL DISK CONTROLLER BOARD ERRORS

Hex Digits	Message On Screen Of Terminal (and Submessages)
5F	"8089 Rom Checksum In Error"
5E	"8089 Ram Check Error"
5D	"Invalid 8089 Write Operation" or "Invalid 8089" Read Operation"
5C	"Floppy Disk Status Register Test - " "Initial HLDS-1 or HDSSSET-1 incorrect" "HLDS-1 not set after 1 second" "HLDS-1 not reset after 3.0 seconds" "HDSSET-1 not reset after 35 milliseconds" "HDSSET-1 not set after 75 milliseconds"
5B	"Floppy Disk Controller Initialization - " "Initial DIO-1 not reset" "DIO-1 not set - write required" "ROM-1 not setable" "Cannot restore 765 protocol"
5A	"Floppy Disk Present Check - " 765 Protocol Error" DS4 through DS1 not in agreement"

Table 8-7 (cont)

OPTIONAL DISK CONTROLLER BOARD ERRORS

Hex Digits	Message On Screen Of Terminal (and Submessages)
59	"Floppy Disk Interrupt Check - " "Initial 765 status incorrect" "765 status incorrect after 1st RECAL byte" "765 status incorrect after 2nd RECAL byte" "Level 7 interrupt not present" "765 status incorrect after SIS" "STO data error" "765 status incorrect after reading STO" "Present track is not track 0" "765 status incorrect after reading present track status" "Level 7 interrupt not reset"
58	"8089 Processor Test failed - " "Invalid Data Transfer"
53	"MSIB Automatic Acknowledge Test - " "AA-1 not reset after 1st write to CSCSIF" "AA-1 not set after write to RCSMWC" "AA-1 not reset after 2nd write to CSCSIF" "AA-1 not set after read from RCSMRC" "AA-1 not reset after 3rd write to CSCSIF"
5n	"Option 45 Not Responding"

OPTION 44 DISK CONTROLLER CHECK (8X)

Hard and Soft Errors

When disk operations fail to execute properly, they are classified as:

- "Soft errors" -- recoverable and random in nature.

OR

- "Hard errors" -- persist after several retries.

Soft Errors. Soft errors, related to the disk, happen when R/W data does not match expected data. The mechanical limitations of the diskette and drive unit allows: temporary misalignments, oxide dropout on the diskette, dirt particles under the R/W head, etc. To keep these transitory problems from hanging up the system, the 4170 processor recognizes the condition and retries the operation.

Hard Errors. Hard errors can occur when the diskette or drive unit is write-protected, or a similar operator oversight.

Disk Media Problems

One of the first and easiest tests to make for disk related problems is to remove the current diskette and replace it with a known good diskette. Try writing to the good diskette and then read it, to see if the problem persists. If the problem remains, it is in the drive unit, the Disk Controller board, the connecting cables, or the power supply.

Table 8-8

OPTION 44 ERRORS

Hex Digits	Submessage (printed on terminal screen)	Explanation
8F		Board Status Register Check
	"Disk -- Board Status Register"	These tests check each bit in the BSR (Board Status Register).
	"Disk -- ADR19-1 not = 0"	Wrote a 0 to ADR19 (D3 of FC01) and read back a 1.
	"Disk -- ADR19-1 not = 1"	Wrote a 1 to ADR19, but read back a 0.
	"Disk -- ADDR19-1 not reset,=1"	Wrote a 0 to reset ADR19 but read a 1.
	"Disk -- INTE-0 not working"	Did not read back what was written at D6.
	"Disk -- BUSW-0 not working"	Did not read back what was written at D5.
	"Disk -- HDL3-0 not working"	Did not read back what was written at D3.
	"Disk -- HDL2-0 not working"	Did not read back what was written at D2.
	"Disk -- HDL1-0 not working"	Did not read back what was written at D1.
	"Disk -- HDL0-0 not working"	Did not read back what was written at D0.
		Circuits Tested: Address Decode. Input Data Buffer, Board Status Register, and the ADR19 Address Counter.

Table 8-8 (cont)

OPTION 44 ERRORS

Hex Digits	Submessage (printed on terminal screen)	Explanation
8E		Initialize and check the Disk Controller The FDC is put into its command phase (regardless of its present state) via a series of command reads and writes. The following bits are tested for correct state: Under FC00 -- D7 (EOC-1) Under FC08 -- D7 (RQM-1) and D6 (DIO-1) A SPECIFY command is then issued by the processor.
	"Disk -- Controller Protocol EOC-0 is 0 after init"	FDC generates an unexpected interrupt, bit is in wrong state after initialization.
	"Disk -- DIO-1 is 0 -- Write required"	FDC required a write to it, while processor expects to read from it.
	"Disk -- DIO-1 is 1 -- Read required"	FDC requires a read from it, but processor expects a write to it.
	"Disk -- DIO-1 is 1 after init"	Bit D6 is in wrong state after initialization.

Table 8-8 (cont)

OPTION 44 ERRORS

Hex Digits	Submessage (printed on terminal screen)	Explanation
	"Disk -- RQM-1 remains 0"	FDC never sends RQM-1 (fails in 1000 tries).
	"Disk -- Cannot restore Protocol"	Runs initialization sequence, but cannot put FDC into a known state. (Processor cannot determine what FDC is doing.) Circuits tested: FDC, Board Status, Control Strobes, Data MUX, and Address Decode.
8D		Drive present check The processor issues a SENSE DRIVE 0 STATUS (SDS) command. Drive 0 should be present. The D7 (RQM-1) and D6 (DIO-1) are tested for correct states before and during the command.
	"Disk -- F0: not present"	Drive 0 should always be present. This message means the FDC or Disk Drive control cannot find drive unit 0. NOTE: If cable is connected to Controller board but not to the drives, this will still detect "Drive ready".

Table 8-8 (cont)

OPTION 44 ERRORS

Hex Digits	Submessage (printed on terminal screen)	Explanation
		Circuits tested: FDC's disk control I/F and Disk Drive Control circuits.
8C		Intersystem Interrupt check The following procedure is used to generate an interrupt: The drive motors are turned off and the processor issues a RECAL DRIVE 0 command. The FDC does not know that the motors were off, so it generates an interrupt. The processor then issues the SENSE INTERRUPT STATUS (SIS) command to read the results of the RECAL command. The results of the SIS command are checked as well as RQM-1 and DIO-1.
	"Disk -- Intersystem Interrupt Check"	
	"Disk -- Bad FC00 init"	Board Status Register contains error (different state than expected).
	"Disk -- Bad FC08 init"	FDC status register contains an error (different state than expected).

Table 8-8 (cont)

OPTION 44 ERRORS

Hex Digits	Submessage (printed on terminal screen)	Explanation
	"Disk -- Bad FC00 RECAL-1 status"	Error in Board Status Register (BSR) after first byte of RECAL command written.
	"Disk -- Bad FC08 RECAL-1 status"	Error in FDC after first RECAL byte written.
	"Disk -- Bad FC00 RECAL-2 status"	Error in BSR after second RECAL byte written.
	"Disk -- Bad FC08 RECAL-2 status"	Error in FDC after second RECAL byte written.
	"Disk -- Level 7 interrupt not present"	Expected interrupt not present.
	"Disk -- Bad FC00 SIS status"	Error in BSR after Sense Interrupt status byte written.
	"Disk -- Bad FC08 SIS status"	Error in FDC after Sense Interrupt status byte written.
	"Disk -- STO data error"	Error in STO, read at FDC.
	"Disk -- Bad FC00 STO status"	Error in BSR after STO status byte read.
	"Disk -- Bad FC08 STO status"	Error in FDC after STO status byte read.

Table 8-8 (cont)

OPTION 44 ERRORS

Hex Digits	Submessage (printed on terminal screen)	Explanation
	"Disk -- Track error"	Error in track number (PCN), read at FDC.
	"Disk -- Bad FC00 Track status"	Error in BSR after track number read.
	"Disk -- Bad FC08 Track status"	Error in FDC after track number read.
	"Disk -- Level 7 interrupt still present"	Interrupt still present. Circuits tested: Address Decode, Board Status, Control Strobes, Data MUX, and FDC.
8B	"Disk -- DMA transfer failed"	DMA Operation check Data ('80') not found at current DMA address (RAM only). Circuits tested: DMA State Machine, Data MUX, Control Strobes, Address Decode. Address Counters, Input Data Buffers, and Board Status blocks.
8A		DMA Addressing check Both this test and the previous test must be completed before the DMA State Machine and the DMA Address Counters are fully tested; the testing of one requires the operation of the other.

Table 8-8 (cont)

OPTION 44 ERRORS

Hex Digits	Submessage (printed on terminal screen)	Explanation
		The DMA Address Counters are write only. The only way to look at these counters is to examine the most significant bit (MSB), which is at D4 of the Board Status Register (FC00). The twenty-bit counters are subdivided into three groups: High Nibble (FC01), Middle Byte (FC02), and Low Byte (FC03). Each group is tested individually in the following manner: HIGH NIBBLE TEST: Test high nibble first because all bits must ripple through the high nibble while testing the middle and low bytes. STEP SET-UP 1. Load High Nibble with 0. 2. Does D4=0? (NOTE) TEST 3. Load Middle Byte and Low Byte with FF. 4. Step DMA. 5. Does D4=0? TEST

Table 8-8 (cont)

OPTION 44 ERRORS

Hex Digits	Submessage (printed on terminal screen)	Explanation
		6. Repeat steps 3-5 eight times. 7. On eighth try, does D4=1? 8. Repeat another 8 times. (D4 should =1 for these 8). 9. On sixteenth try, does D4=0? NOTE -- If D=1 an error message is reported. All "D=N?" steps will cause an error unless the number is as expected. Error messages are in the accompanying tables. MIDDLE BYTE TEST STEP SET-UP 10. Load High Nibble with 7. 11. Does D4=0? 12. Load Middle Byte with FF. TEST 13. Load Low Byte with FF. 14. Step DMA. 15. Does D4=0? 16. Reload Low Byte with FF (leave Middle Byte since it is being tested).

Table 8-8 (cont)

OPTION 44 ERRORS

Hex Digits	Submessage (printed on terminal screen)	Explanation
		17. Repeat steps 13-15; on 256th try does D4=1? (Does High Nibble =8?) SET-UP 18. Reload High Nibble with F. 19. Does D4=1? TEST 20. Reload Low Byte with FF and Step DMA. 21. Repeat step 19 256 times. 22. On 256th try does D4=0? (High Nibble =0) LOW BYTE TEST Same procedure as Middle Byte test except the Low Byte and Middle Byte are reversed.
	"Disk -- DMA Address Counters ADR19-1 Error"	
	"Early 0-1 @ FC01"	Error detected at step 2 or 5 (Counter reached 8 too soon)
	"No 0-1 @ FC01"	Error detected at step 7
	"Early 1-0 @ FC01"	Error detected at step 8
	"No 1-0 @ FC01"	Error detected at step 9

Table 8-8 (cont)

OPTION 44 ERRORS

Hex Digits	Submessage (printed on terminal screen)	Explanation
	"Early 0-1 @ FC02"	Error detected at step 11 or 15
	"No 0-1 @ FC02"	Error detected at step 17
	"Early 1-0 @ FC02"	Error detected at step 19
	"No 1-0 @ FC02"	Error detected at step 22
	"Early 0-1 @ FC03" "No 0-1 @ FC03" "Early 1-0 @ FC03" "No 1-0 @ FC03"	Same explanation as for FC02 messages, except substitute FC03 for FC02 in explanation of Middle Byte tests. These tests check the clocking and carries of one counter to the next. Circuits tested: Same circuit blocks as tested in the DMA Operation Check, except the Data MUX is not tested.

ADJUSTMENT SELF TEST

Adjustment Self Test is the third diagnostic program that the 4170 uses to test itself. Adjustment Self Test is generally used to aid in adjusting or troubleshooting the 4170. This program requires that the technician select what pattern is to be generated or what test is to be run. Some of the things that Adjustment Self Test can do are:

- Give the hexadecimal equivalent of the downstroke and upstroke of each key on the 4170 front-panel keyboard.
- Perform a Host Port check for the 4170. (This requires attaching a loopback connector.)
- Perform head movements of the flexible disk drives. (This can aid in head alignment.)
- Test the hard disk drive, if the 4170 is equipped with that option.
- Test each of the RS-232 peripheral ports (except "Terminal Port").

Starting Adjustment Self Test

To start Adjustment Self Test, connect a terminal to "Terminal Port", set the terminal communication parameters to the factory default values (see Section 4, Connecting Cables) and proceed as follows:

1. Press and hold both the SELF TEST and RESET switches.
2. Release the RESET switch.
3. After the 4170 front-panel LEDs start of cycle, release the SELF TEST switch. The terminal is now in Extended Self Test.
4. When the bell sounds (at the "Keyboard Test" of Extended Self Test), press the CTRL and C keys ON THE 4170 FRONT PANEL at the same time. (Pressing CTRL-C on the companion terminal won't work.)
5. The 4170 will pause a few seconds during the PUP tests and then display a general menu on the attached terminal. The 4170 is now in Adjustment Self Test.
6. The General Menu tells you which key to press in order to test one part of the 4170. All input to Adjustment Self-test must be done through the 4170 front panel keyboard.
7. Once one of the keys designated by the General Menu has been pressed, a second menu will be displayed.

8. Any selection that requires a number to be entered (eg. "17" for track seventeen) is entered with the function keys:

   2nd key F1 = "0"
   F1-F8 = "1-8"
   2nd key F8 = "9"

   Any "numbers" entered in this fashion must be terminated with the CONT key.

The General Menu

The General Menu is the first menu that Adjustment Self Test displays. The exact form of this menu depends on which options have been installed in the 4170. Two examples of this menu are shown below:

```
4170 Adjustment Self Test Menu
--
f1 3PPI
f2 Processor Board
f3 Flexible Disk - Option 44
--
Selection
*
```

```
4170 Adjustment Self Test Menu
--
f1 3PPI
f2 Processor Board
f4 MSIB - Option 45
f5 Optional 3PPI
--
Selection
*
```

During Adjustment Self Test, pressing CTRL-C (on the 4170 front panel) will always cause this General Menu to be displayed on the screen.

Processor Board Menu

Once the General Menu has been displayed, the Processor Board Menu can be selected from the General Menu by pressing function key F2 on the 4170 front panel. Once F2 has been pressed, the Processor Board Menu will be displayed. An example of this menu is as follows:

```
Processor Board Menu
--
f1 CMOS Reset
f2 Keyboard
f3 Host Port
--
Selection
*
```

F1:CMOS Reset. The term CMOS Reset means that the setup memory has been reset and must be restored to factory default values. This is accomplished by completing self test and then booting the operating system and executing the utility CONFIG, using the file FACTORY.TXT.

Example:

A>CONFIG <FACTORY.TXT<CR>

The terminal must also be reset to 2400 baud (or 4105 factory default values).

F2: Keyboard. From the Processor Board Menu, pressing function key F2 will cause every key on the 4170 front-panel keyboard to display two two-digit hexadecimal numerals. These hexadecimal codes represent the 8-bit codes that the processor generates for each upstroke and downstroke of a key. There is a unique code for the upstroke and downstroke of each key. The first number of the upstroke code is always 8 plus the first number of the downstroke code. Figure 8-2 shows the keyboard keycodes.

The only way to return to the Processor Board Menu is by pressing CTRL-D.

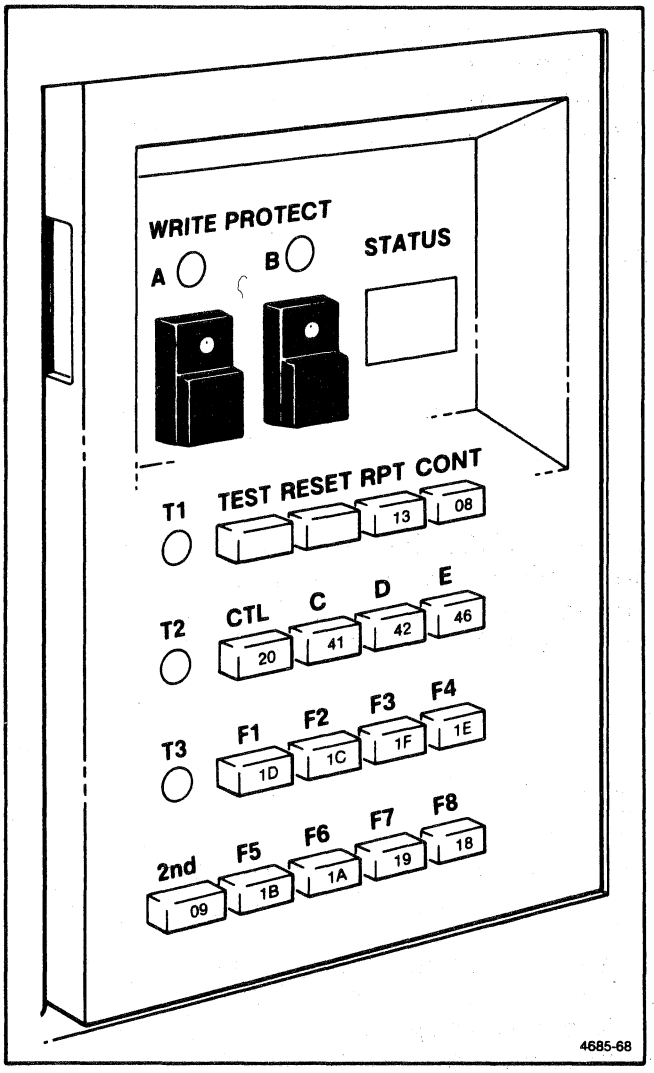


Figure 8-2. Keyboard Keycodes.

F3: Host Port. The Host Port routine checks the validity of the data port by which the 4170 communicates with a remote host computer. Enter this routine from the Processor Board Menu by pressing function key F3. Once this is done, the following is displayed on the companion terminal:

```
Host Port
Attach Loopback
Press CONT key
```

Connect the special loopback connector (available as an optional accessory) to the 4170's host port connector. Once the loopback connector is attached, press the CONT key (on the 4170 front panel keyboard).

The test is repeated at various baud rates, from 19200 down to 300 bits per second. At each data rate, the 4170 sends each ASCII character (hexadecimal 7F down to hexadecimal 00) to the loopback connector and receives that character back from that connector. If no errors are detected, the end of the test is indicated by the following message:

```
Complete
```

This message means the 4170 is ready to go on to the next test.

If there is an error during this test, the following submessage appears on the screen of the companion terminal:

```
Host Port-Baud/Character
Baud: XXXX  Expect: YY-AA  Receive: ZZ-BB
```

Here,

XXXX is the baud rate, expressed as a hexadecimal numeral
YY is the signal sent (for example, 7F)
AA is the expected bits in error, this should always read 00
ZZ is the signal received
BB is the bits in error.

3PPI Menu

The key used to select this 3PPI Menu from the General Menu may depend on which options are installed in the 4170. Once that key has been pressed (on the companion terminal or on the 4170 front panel), the 3PPI Menu is displayed as follows:

```
3PPI Menu
--
f1 3PPI Ports
--
Selection
*
```

This routine is designed to check the RS-232 peripheral ports. (Port P0 cannot be tested here, as it is used for the companion terminal on which the menus and messages are displayed.)

F1: 3PPI Ports. To start the test from the 3PPI Menu, press F2. The following message is displayed on the companion terminal:

```
*f1
Select Port (key f1=Port 1, f2=2)
*
```

Connect an RS-232 cable from the 4170 host port to the peripheral port to the be tested. Then type the number of the port (eg f1 for Port 1) to be tested, followed by CONT. If there are any errors in the test, the following message may be displayed:

```
3PPI-Baud/Character
Sent: XXXX Expect: YY-ZZ Receive: AA-BB
```

Explanation: the baud rate was set to hexadecimal XXXX. During the test, the character sent was hexadecimal YY. The character received was hexadecimal ZZ. Interrupt bits expected are ZZ (00), what was received was BB.

If no errors are found during the test, a prompt (asterisk) is displayed.

Option 10 3PPI Menu. The following menu is present when an additional (Option 10) 3PPI is installed:

```
3PPI Menu
--
f1 Optional 3PPI Ports
--
Selection
*
```

To perform the test on the optional 3PPI ports, press f1 and select the number of the port to be tested.

```
*f1
Select Port (Key f1=Port 3, f2=Port 4, f3=Port 5)
*
```

The test is run in the same manner as the standard 3PPI test and the error messages are the same.

Option 44 Flexible Disk Menu

The Flexible Disk Menu may be obtained from the general menu by pressing f3. This menu will only appear if Option 44 is installed. As Self-test checks the circuitry on the Disk Controller board, this routine checks the drive unit, also. This menu is useful for drive unit head alignment. For this purpose a special alignment diskette (not used in this functional check) is required.

This is an example of the Disk Menu (items are self explanatory):

```
Flexible Disk -- Option 44
--
f1 No Operation
f2 Step Up One Track
f3 Step Down One Track
f4 Seek Track 0
f5 Seek Track 1
f6 Seek Middle Of Disk
f7 Select Head
f8 Seek Last Track
2nd f1 Motor On
2nd f2 Motor Off
2nd f3 Arms Write Mode
2nd f4 Writes Track 39 With a 2F Pattern
2nd f5 Select Your Own Track
2nd f6 Change Device Address
```

With the disk option installed, properly insert a diskette in the drive unit before starting this test. Then make sure the WRITE PROTECT switch is set to the off position (the LED is not lit).

CAUTION

It is best to perform this part of the functional check using a diskette free of data. Performing this test may cause some data to be written on the diskette. With the write-protect switch off, it is possible to write over existing data. Therefore, it is best to use a new diskette or one that contains unwanted data.

Press each of the disk menu keys in succession. No error messages should be received on any of the tests, and the head should move to the selected area on the diskette. After running through the head movements (in the menu), press CTRL C to return to the general menu.

NOTE

This is merely a check of the disk drive unit to ensure that no error messages are received and that the drive unit is operational. For a full alignment procedure of the drive unit, see the 4110 Series F42/43 Disk Options Service Manual.

Option 45 Disk Menu

Some of the operations in the Option 45 Disk Menu do not return a prompt when completed. These tests continue to check the keyboard for further function key requests and will execute the operations requested; however, if the user wants a menu displayed (using control-C or control-D) or to exit adjustment self-test (by pressing control-E), a function that produces a prompt must be requested. Once self-test has generated a prompt ('_') on the attached terminal, a control key can be entered.

The following tests DO NOT respond with a prompt after completion:
f2 Step Up One Track,
f3 Step Down One Track,
f4 Seek Track 0,
f5 Seek Track 1,
f6 Seek Middle of Disk,
f7 Select Head,
f8 Seek Last Track,
Sh F1 Run Motor,
Sh F2 Read Current Track.

The following tests DO respond with a prompt after completion:

f1 MSIB -- ID Verification
f7 Select Head
Sh F3 Arm Write Mode
Sh F4 Write Current Track With a 2F Pattern
Sh F5 Select Your Own Track
Sh F6 Change Device Address
Sh F7 No Operation
Sh F8 MSIB -- External Peripheral Test

For example, if the user presses key F4, the current drive will seek track zero and no prompt will be given to the attached terminal. Even though there is no prompt, key F6 could be pressed and the current drive would seek the middle of the disk. If key Sh F2 were pressed at this point, the current track would be read. Any of the function keys could be pressed. If F7 were pressed, the test would display the new head selected and would display a prompt when completed. It is only at this point with a prompt displayed that the control characters could be entered to request menus or to exit Adjustment Self Test.

Section 10

GLOSSARY

This glossary pertains to this manual and is not intended to be a universal reference. See CP/M 86 manual (included in package) for more lists of terms and accompanying definitions.

3PPI
Three port peripheral interface.

ASCII character
Any one of 128 characters contained in the character set used by American Standard Code for Information Interchange.

ASCII code
Seven-digit binary numbers which express any of the 128 ASCII characters.

Argument
An independent variable.

Back-up file
Any copy of a file located on an alternate storage medium, or under a different user ident code, thus providing back-up protection for the file.

Baud
Signaling units per second; an expression of serial data trasmission bit rate.

Binary
A number system which uses two as its base. Only the digits zero and one appear in binary expressions.

Bit
Binary digIT.

Break
A signal sent from the terminal to the computer to interrupt computer transmission in some installations. Also, the command which intiates the action.

Byte
A group of adjacent binary digits operated on as a unit and usually shorter than a computer word (frequently denotes a group of eight or sixteen bits).

CTS
Clear to send.

SECTION 10
GLOSSARY

Concantenate
To link together in a series or chain.

Control character
A character whose occurence in a particular context, initiates, modifies or stops a control action.

DEL
Delete.

DIR
Directory.

DTI
Direct terminal interface.

DTR
Data terminal ready.

Duplex, full
Method of communication where each end may simultaneously transmit and receive.

Duplex, half
Method of communication where each end may transmit and receive; however, not simultaneously.

ECC
Error correcting code.

EMI
Electromagnetic interference.

Format, disk
A series of characters on a disk that enable it to be used. An address.

FPU
Numeric co-processor.

File
A collection of related records treated as a unit.

GIN
Graphic input.

GSX
Graphics system extensions

IEC
International Electrotechnical Commission.

IGL
Interactive graphics library.

LPOS
Local Programmable Operating System.

Line voltage
90-130 V or 180-260 V, 47 to 400 Hz.

Modem
Modulator/Demodulator. Used primarily for telephone interface.

Operating system
An aid in using the system.

PUT
Programmable Unijunction Transistor.

Simplex
One way transmission. See **Duplex** .

Snubber
Passive circuit waveshaping that allows primary current in a flyback transformer to flow after the main switching transistor is cut off.

Strap
A wire joining two contact points. This is generally used for an option or modification.

Transmission, parallel.
Used to identify a system wherein the information is fed in parallel. Implies two or more transmission channels.

Transmission, serial.
Used to identify a system wherein the bits of a character occur serially in time. Implies a single transmission channel.

Wild card
A character in a program or sub-program that accomplishes what would normally take several keystrokes.

Write-protect
A system that protects information on disks from being written over.

SECTION 11

SERVICE SAFETY SUMMARY

FOR QUALIFIED SERVICE PERSONNEL ONLY

Refer also to the preceding Operators Safety Summary.

DO NOT SERVICE ALONE

Do not perform internal service or adjustment of this product unless another person capable of rendering first aid and resuscitation is present.

USE CARE WHEN SERVICING WITH POWER ON

Dangerous voltages may exist at several points in this product. To avoid personal injury, do not touch exposed connections and components while power is on.

Disconnect power before removing the power supply shield, soldering, or replacing components.

DO NOT WEAR JEWELRY

Remove jewelry prior to servicing. Rings, necklaces, and other metallic objects could come into contact with dangerous voltages and currents.

X-RADIATION

X-ray emission generated within this instrument has been sufficiently shielded. Do not modify or otherwise alter the high voltage circuitry or the CRT enclosure.

POWER SOURCE

This product is intended to operate from a power source that will not apply more than 250 volts rms between the supply conductors or between either supply conductor and ground. A protective ground connection by way of the grounding conductor in the power cord is essential for safe operation.

HANDLING

Due to the wieght of the Display Module, and its component subassemblies, at least two persons are required to perform installatin or service to prevent injury to personnel or damage to the Display Module.

IMPLOSION PROTECTION

Whenever the implosion shield is removed from the CRT, protection against implosion hazard is reduced. Service personnel should wear full face masks and protective clothing at any time the CRT is removed from the CRT module or the implosion shield is not in place.

Section 12

THEORY OF OPERATION

This section contains detailed theory descriptions for some boards, and summary information for other boards/modules. Those boards/modules that are documented in separate manuals are merely referenced here. Any special application information about such boards are included with the summaries.

OVERVIEW

The 4170 Local Graphics Processing Unit provides the needed intelligence and storage capacity for local processing by a connected terminal. This outboard unit acts as a host for the terminal in the simplest applications or it may function with other 4170s in a distributed processing situation. Also the 4170 may act as a host interface, between a terminal and a mainframe computer (acting as the host).

The 4170 contains the following hardware modules/board, arranged as indicated in Figure 12-1:

- Motherboard/card-cage
- Processor board
- ECC RAM board
- Disk Controller board
- Peripheral Interface board (3PPI)
- 5-1/4 inch Flexible Disk Drive Units (2)
- Front-panel (Controls/Indicators) Unit
- Rear-panel connectors
- Power Supply module

Optional components in the 4170 (also shown in Figure 12-1) are:

- Optional Additional 3PPI board
- Optional Additional ECC RAM board
- 5-1/4 inch Winchester Disk Unit (10 M Byte)
- Color Copier Interface board

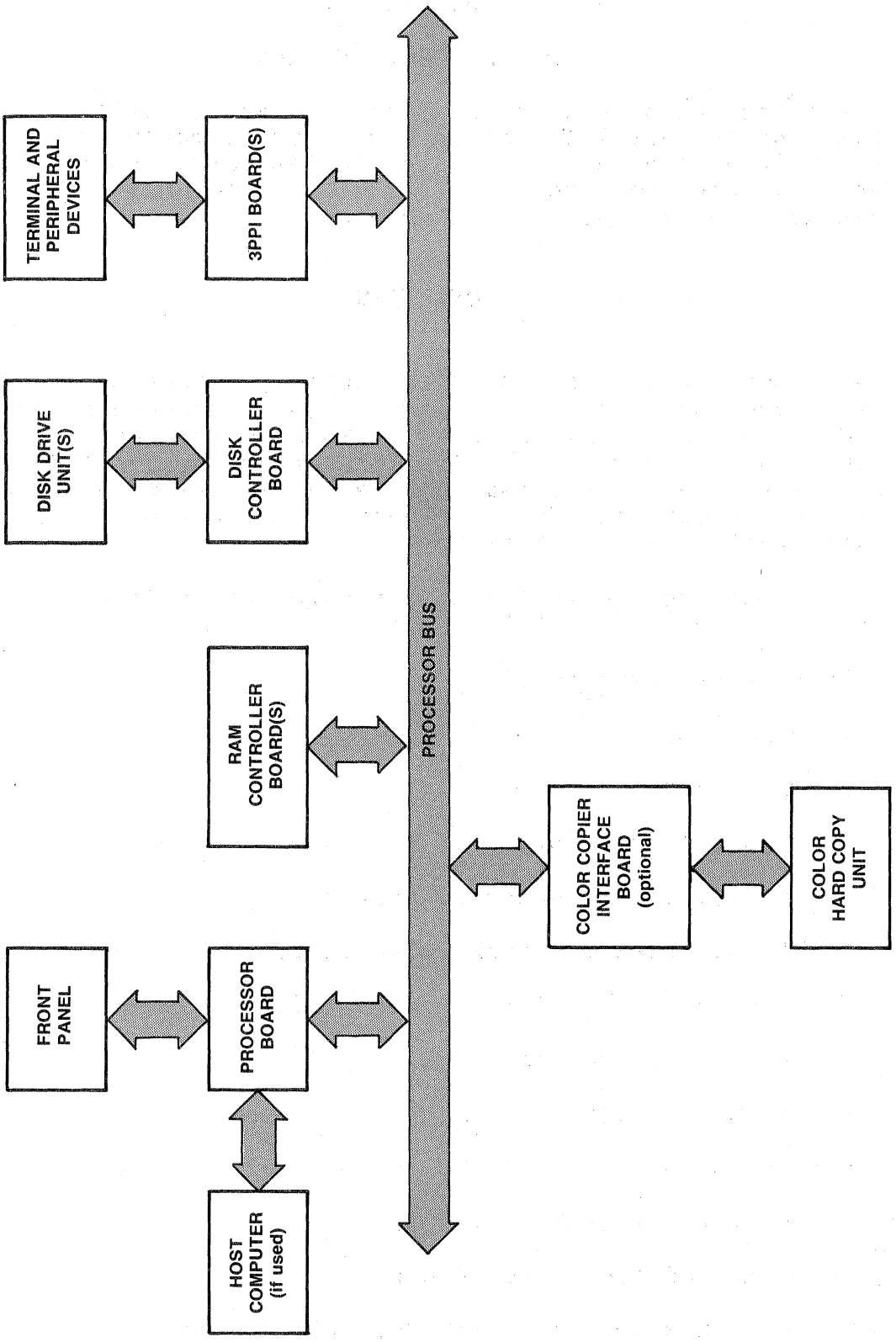


Figure 12-1. 4170 Functional Block Diagram.

This section describes the overall operation of the 4170 and provides summary descriptions of each circuit board or module. Many of the circuit boards are documented in separate manuals, so references are made to such manuals. Those boards/modules that are only used in this instrument receive a full theory description in this section.

CARD-CAGE/MOTHERBOARD

The main circuit boards plug into the Motherboard (Schematic A1-1) in the back of the 4170 card-cage. This motherboard contains seven slots, and each slot is an 80-pin connector that accepts the mating edge-connection of a circuit board. The order in which the circuit boards are inserted is determined by the relative priorities of these boards as bus masters and slaves. Figure 12-2 shows the layout of the Motherboard and the recommended positioning of the inserted boards. Table 12-1 lists the signal names and descriptions for each of the 80-pin connections on the Motherboard.

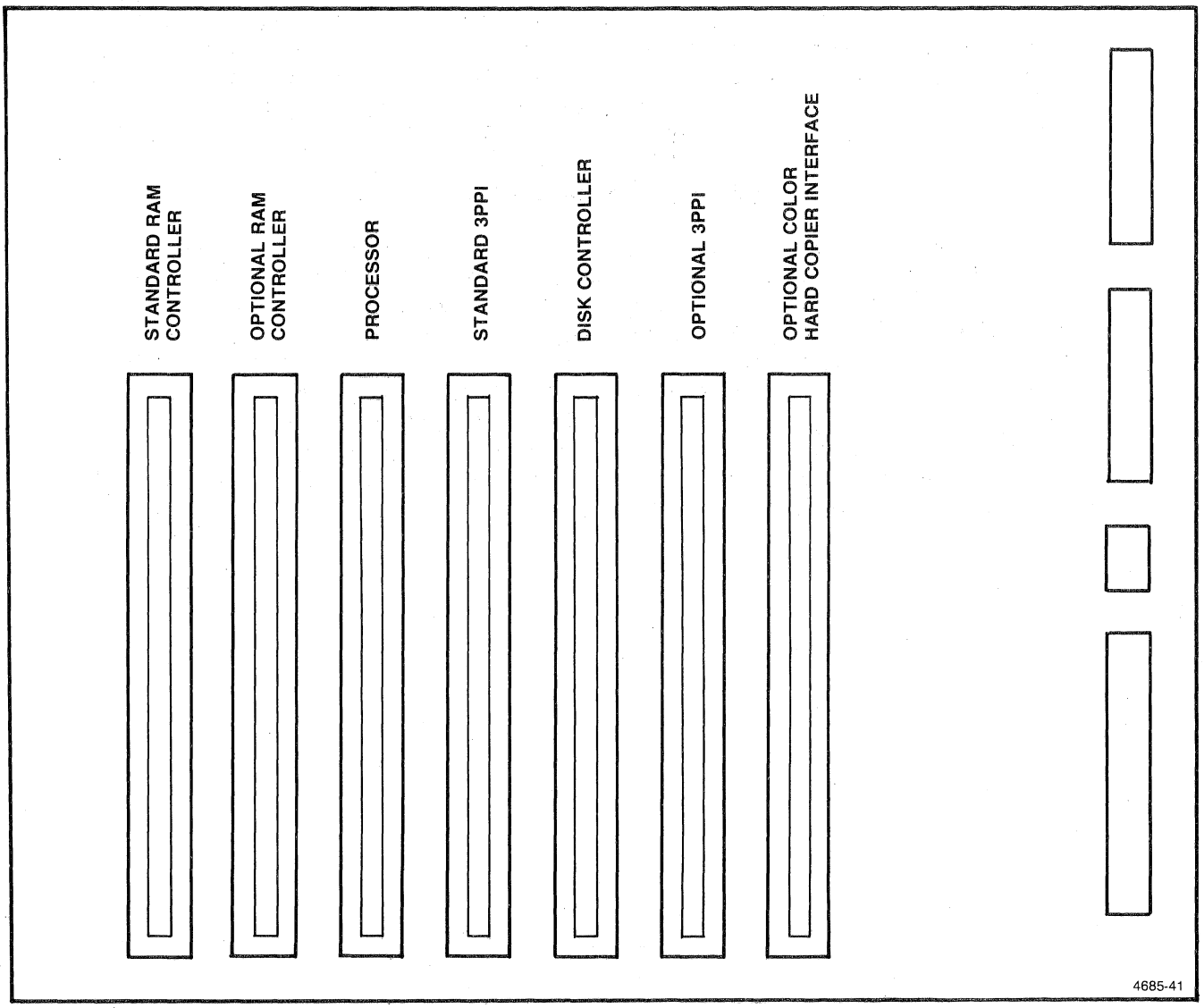


Figure 12-2. Physical Layout of the 7-slot Motherboard.

Table 12-1

80-PIN CONNECTOR SIGNALS

Pin	Signal	Description	Pin	Signal	Description
1	-12V	---	41	BUSY-O	Bus Busy
2	-12V	---	42	CBRQ-O	Common Bus Request
3	GND	Ground	43	STEST-O	Self-Test
4	GND	Ground	44	---	(not used)
5	-5.2V	---	45	BCLK-O	Bus Clock
6	-5.2V	---	46	GND	Ground
7	PFAIL-O	Power Fail Warning Ground	47	GND	Ground
8	AGND	Analog Ground	48	---	(not used)
9	INIT-O	Initialize	49	INTO-O	Interrupt Request
10	BHEN-O	Byte High Enable	50	INT1-O	Interrupt Request
11	ADRO-O	Address Bus	51	INT2-O	Interrupt Request
12	ADR1-1	Address Bus	52	INT3-O	Interrupt Request
13	ADR2-1	Address Bus	53	INT4-O	Interrupt Request
14	ADR3-1	Address Bus	54	INT5-O	Interrupt Request
15	ADR4-1	Address Bus	55	INT6-O	Interrupt Request

Table 12-1 (cont)

80-PIN CONNECTOR SIGNALS

Pin	Signal	Description	Pin	Signal	Description
16	ADR5-1	Address Bus	56	INT7-0	Interrupt Request
17	ADR6-1	Address Bus	57	DAT0-1	Data Bus
18	ADR7-1	Address Bus	58	DAT1-1	Data Bus
19	ADR8-1	Address Bus	59	DAT2-1	Data Bus
20	ADR9-1	Address Bus	60	DAT3-1	Data Bus
21	ADR10-1	Address Bus	61	DAT4-1	Data Bus
22	ADR11-1	Address Bus	62	DAT5-1	Data Bus
23	ADR12-1	Address Bus	63	DAT6-1	Data Bus
24	ADR13-1	Address Bus	64	DAT7-1	Data Bus
25	ADR14-1	Address Bus	65	DAT8-1	Data Bus
26	ADR15-1	Address Bus	66	DAT9-1	Data Bus
27	ADR16-1	Address Bus	67	DAT10-1	Data Bus
28	ADR17-1	Address Bus	68	DAT11-1	Data Bus
29	ADR18-1	Address Bus	69	DAT12-1	Data Bus
30	ADR19-1	Address Bus	70	DAT13-1	Data Bus
31	INTA-0	Interrupt Acknowledge	71	DAT14-1	Data Bus
32	INH-0	Read Inhibit	72	DAT15-1	Data Bus
33	AMWC-0	Adv. Memory Write Cmd.	73	BREQ-0	Bus Request

Table 12-1 (cont)

80-PIN CONNECTOR SIGNALS

Pin	Signal	Description	Pin	Signal	Description
34	MWTX-O	Memory Write Command	74	BPRN-O	Bus Grant
35	MRDC-O	Memory Read	75	+5 V	---
36	IORC-O	I/O Read Command	76	+5 V	---
37	AIOWC-O	Adv. I/O	77	Ground	---
38	IOWC-O	I/O Write	78	Ground	---
39	ACK1-O	Command Acknowledge 1	79	+12 V	---
40	ACK2-O	Command Acknowledge 2	80	+12 V	---

PROCESSOR BUS

The Processor board consists of circuitry that performs the following functions:

- Processes system firmware commands and data along with interrupt data from on and off the board.
- Accepts interrupt signals from the host computer communications port, the front panel, peripheral devices and other circuitry connected to the System bus, and from other circuitry on the Processor board itself.
- Transmits and receives data from the System bus.
- Stores the system firmware and the 4170 initialization information.
- Provides control and status signals primarily for the microprocessor (MPU).
- Communicates with the host computer using the RS-232 communication standard.
- Communicates with the front panel.

Each of these functions is performed by the blocks shown in the simplified block diagram of Figure 12-3.

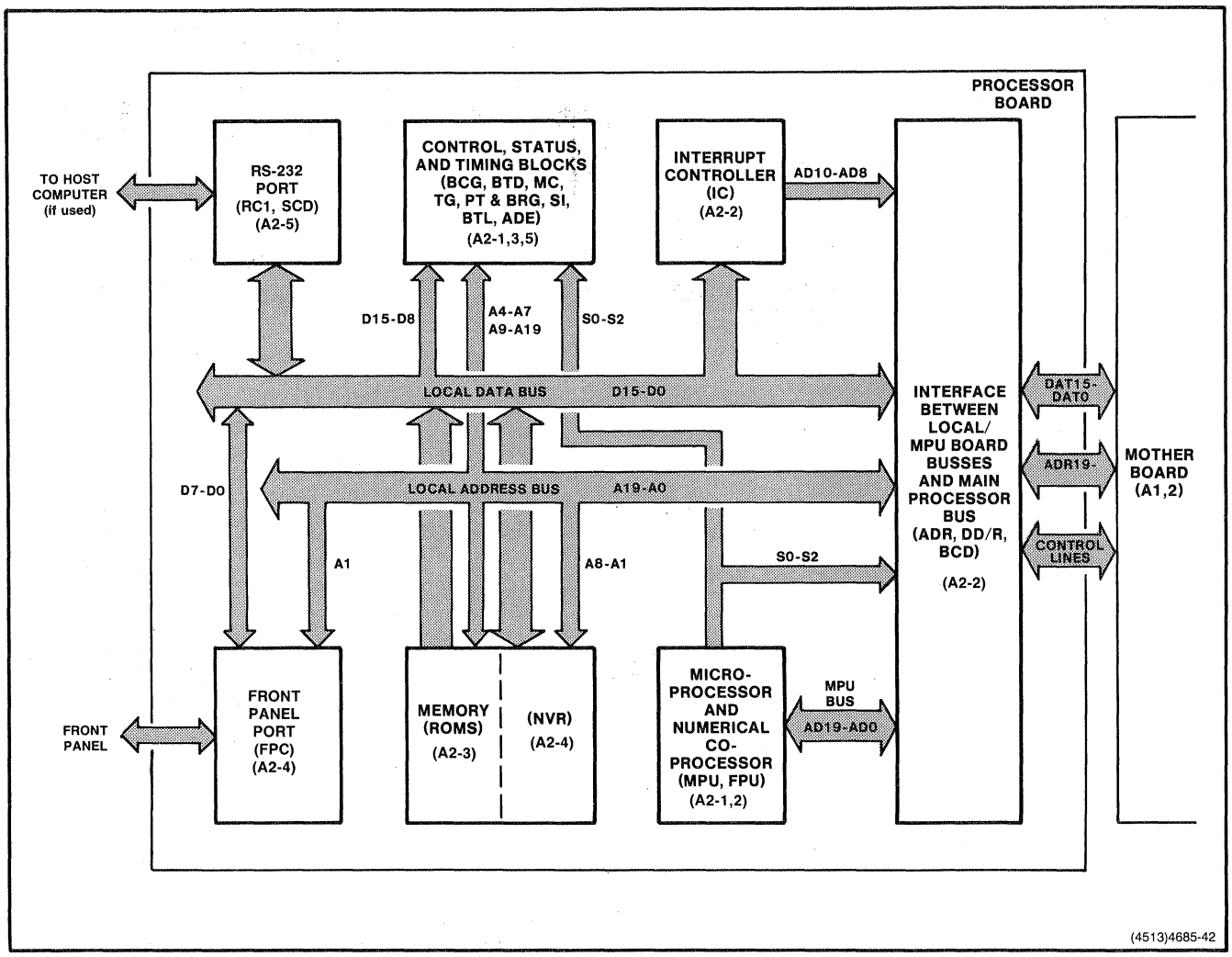


Figure 12-3. Simplified Block Diagram of the Processor Board.

MICROPROCESSOR (MPU) AND NUMERIC CO-PROCESSOR (FPU)

This block consists of the microprocessor (MPU), and the numeric floating point co-processor (FPU). Both 16-bit processors time-multiplex their AD0-1 through AD19-1 lines; allowing the lines to be used for either address or data information. The MPU and FPU output three internal state identifiers (S0-0, S1-0, and S2-0), which identify to the bus the type of processor cycle being run and are used by blocks of circuitry on the Processor board to synchronize their operations with the MPU and FPU. Inputs to the block are primarily a clock signal, a reset signal, a ready signal, and an interrupt signal to the MPU from the Interrupt Controller.

INTERRUPT CONTROLLER

The Interrupt Controller circuitry consists primarily of the Peripheral Interrupt Controller (PIC). This circuit determines which of the eight possible interrupting sources is to be serviced if more than one source requests servicing at one time. Sources can be other boards in the 4170, such as the Three-Port Peripheral Interface (3PPI) board, or internal Processor board circuitry, such as the front panel port, FPU, or host computer port. The MPU communicates with the interrupt controller block through the Process data bus, D0-1 through D7-1. The PIC is programmed by the MPU shortly after the 4170 is powered up. The MPU can also read the current status of the PIC and change its mode of operation by reprogramming its internal registers.

PROCESSOR AND SYSTEM BUS INTERFACE

The primary inputs to the Processor board and System Bus Interface circuitry are the lines on the multiplexed Processor address data bus (AD0-1 through AD19-1), which are connected directly to the MPU and FPU. The AD0-1 through AD19-1 signal lines carry, at different times, data and address information. This block separates the data and address information into the following four distinct buses: D0-1 through D15-1, the Processor and Co-processor data bus; A0-1 through A19-1, the Processor and Co-processor address bus; DAT0-1 through DAT15-1, the System data bus; and ADR0-1 through ADR19-1, the System address bus. This block also outputs control and status signals onto the System bus.

SYSTEM MEMORY

There are approximately 32-thousand bytes of ROM in the System Memory circuitry on the Processor board. The Processor data and address buses connect to this block. Memory is addressed using the Processor address bus, and the selected data is placed on the Processor data bus. The System Memory also includes non-volatile RAM where setup parameters are retained when the 4170 power is off.

CONTROL, STATUS, AND TIMING

This circuitry consists of MPU and FPU control, status detection, and timing functions. Refer to the Processor board schematic diagrams to reference the circuit descriptions that follow.

The MPU Control Logic generates the 4.9152 MHz clock signal for the MPU and FPU. The block also synchronizes the reset signal and ready signal inputs of the MPU and FPU.

The Bus Transfer Logic provides signals that inform bus masters and slaves whether they can use the System bus. Priority-determining logic on the Motherboard works in conjunction with this circuitry to establish bus mastership.

The Address Decoding circuitry creates signals from address bits A12-1 through A7-1 and A9-1 through A19-1 which indicate what area of I/O or memory address space that data is to be sent or received.

The Bus Timeout Detector detects when a slave device fails to respond with an acknowledge signal (ACK1-0) to a command from any bus master device. If the circuitry does detect this failure to respond, it drives its own acknowledge signal (ACK1-0) onto the bus sets an error status bit, and causes an interrupt to be generated. This action prevents the bus from "hanging", that is, remaining in a state that cannot be responded to by any master or slave device.

The Status Input circuitry allows the MPU to read two Processor board status signals, LS0-0 and OBIOD&F-0. This circuitry also outputs STATEN-1 which is input to the Bus Timeout Detector.

The Bus Clock Generator produces a clock signal (BCLK-0) for the System bus.

The Microprocessor Timing Generator receives three signals from the MPU and FPU (S0-0, S1-0, and S2-0) that indicate whether the MPU is: acknowledging an interrupt; reading or writing to I/O or memory space; fetching an instruction; in a halt state; or in a no-bus-cycle state. This information synchronizes operations both on and off the Processor board.

The Programmable Timer and Baud Rate Generator consists of a Programmable Interval Timer that primarily provides variable timing functions. This circuitry provides the transmit baud rate, a firmware interval timer, the bus timeout interval, and the RS-232 intercharacter delay.

HOST COMPUTER PORT

The Host Computer Port circuitry communicates with the host computer by means of RS-232 signals. A programmable integrated circuit accepts RS-232 control and data signals from the host computer and then converts this information to parallel data and interrupts for use by the MPU. In addition to various control and status signals, this block outputs data on the Processor data bus, D0-1 through D15-1.

The RS-232 State Change Detector is related to the Host Computer Port. This circuit detects state changes on the incoming RS-232 status lines and generates interrupts if changes occur.

FRONT PANEL PORT

The Front Panel Port circuitry accepts data from the front panel. The Peripheral Interface Microcomputer (PI MPU), dedicated to servicing the front panel, processes this data and outputs it to part of the Processor data bus, D0-1 through D7-1. The PI MPU also outputs various control signals, including an interrupt (KBINT-0), which reaches the MPU after being processed by the Interrupt Controller circuitry.

DETAILED PROCESSOR BOARD CIRCUIT DESCRIPTIONS

Circuitry that appears on the Processor board schematics is described here. For each block, the description follows the same format: Schematic number, Purpose, Signals (input and output), Description, and Operation. The intent of this format is to give the reader a clear understanding of each block. This formatting also serves the purpose of quick reference once the circuitry is understood.

Refer to Figure 12-4 for an overall view of how the circuit blocks communicate. Note that the Address Drivers and Data Drivers/Receivers blocks perform similar tasks for the address and data information that the MPU emits. They either output this information to the System bus or the Processor bus, depending on the state of control signals from the MPU and FPU.

In the description of some blocks of circuitry, parentheses are used to enclose names that are pin assignments. These names are usually taken from the input or output pin name of a Large Scale Integration (LSI) circuit. Some examples are (CLK-1), (OSC-1), and (RDY-1) in the Microprocessor Control block.

Eight of the nineteen blocks have an LSI circuit as a main component. Many of these are programmable and all are multipurpose. The LSI circuits are used in a definite configuration (and in some cases a definite mode) in the design of the Processor board. The block descriptions do not give complete accounts of the unused modes and/or configurations of these LSI circuits. However, reference to further information, along with the abbreviations used for these LSI circuits, is given in Table 12-2.

Table 12-2

MANUFACTURERS' NOMENCLATURE FOR ICs

NAME	ABBREVIATION	NUMBER
Programmable Interrupt Controller	PIC	8259A (a)
Microprocessor	MPU	8086 (a)
Numerical Co-processor	FPU	8087 (a)
Clock Generator and Driver	CGD	8284A (a)
Bus Controller	BC	8288 (a)
Front Panel Controller MPU	FPC MPU	8041A (a)
Programmable Interval Timer	PIT	8253 (a)
Programmable Communications Interface	PCI	2661 (b)

(a) Description in Intel Component Data Catalog 1982
(b) Description in Signetics Data Manual 1982

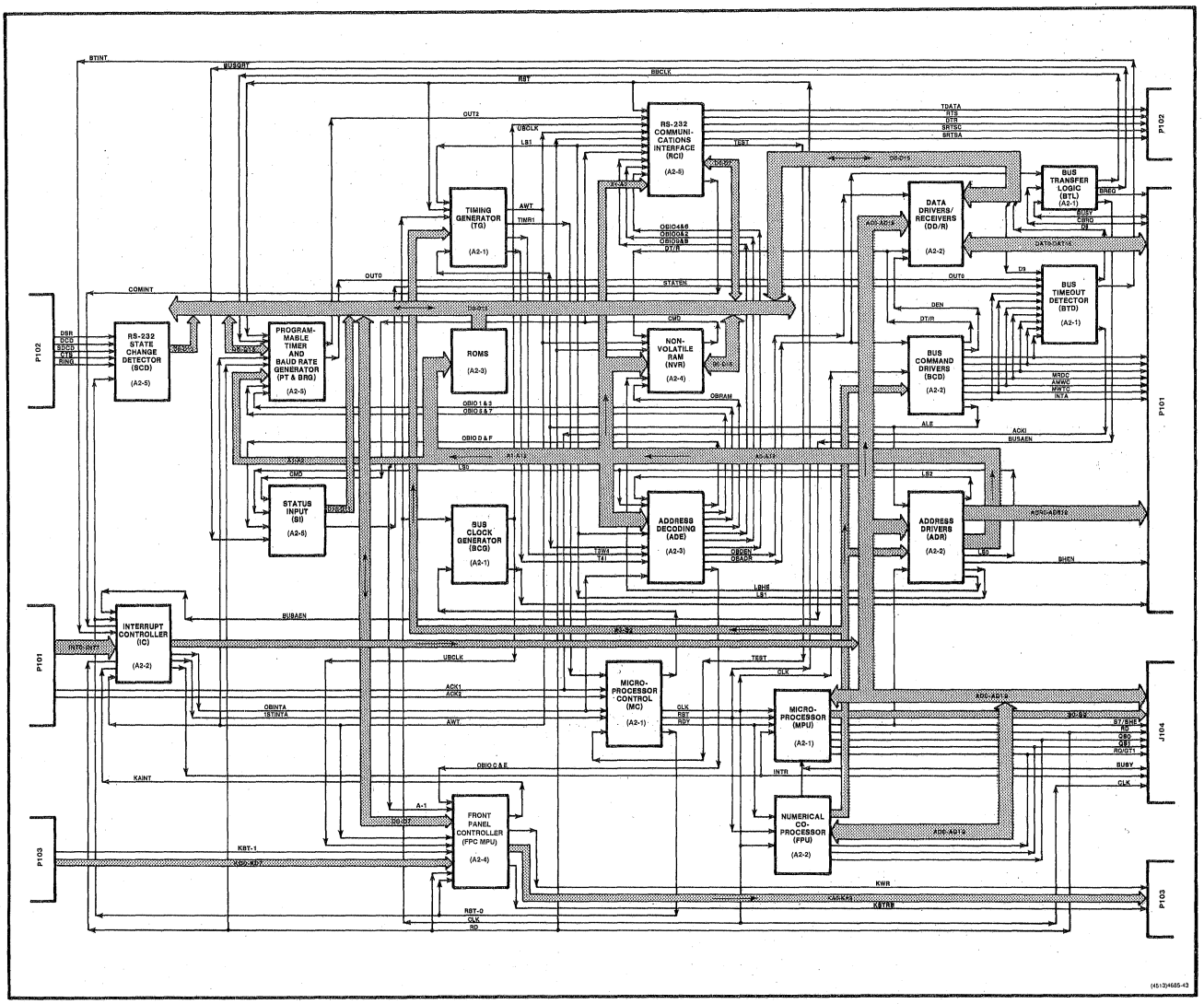


Figure 12-4. Processor Board Block Diagram.

MPU (SCHEMATIC A2-1) AND FPU (SCHEMATIC A2-2)

The MPU controls the general-purpose reading and writing of data and address information for the entire 4170 system in conjunction with the FPU which performs numerical computations for the MPU. The MPU gives up control of the System bus in favor of other bus masters when necessary.

The input signals are:

- CLK-0 (Clock). Direct 4.9152 MHz clock signal from the Clock Generator.(a)

- RST-1 (Reset). Directly input from the Clock Generator. Causes the MPU or FPU to stop current activity, but restarts execution when it goes false. RST-1 must be true for at least four clock cycles to stop the MPU and FPU.(a)

- RDY-1 (Ready). Directly input from the Clock Generator. When true, RDY-1 signals that the addressed memory or I/O device will complete its data transfer.(a)

- TEST-0 (Test). If TEST-0 is true, execution continues. If TEST-0 is false, the MPU inserts idle states between bus cycles until TEST-0 goes true. TEST-0 may be driven by an input from Test Connector J104 or by the BUSY-1 signal from the FPU.(b)

- NMI-0 (Non-Maskable Interrupt). When true, the MPU stops normal execution and jumps to a predetermined subroutine in firmware. Software cannot disable this interrupt. NMI-0 is triggered on the low to high transition. This signal is available only on Test Connector J104.(b)

- INTR-1 (Interrupt). When true, after the last clock cycle of the current instruction, the MPU enters an interrupt acknowledge operation. INTR-1 may be disabled by software.

(a) Input to both the MPU and FPU.
(b) Input to the MPU only.

The output signals are:

- ADO-1 through AD19-1 (Processor Address Data Bus Bits 0 through 19). These bits carry both address and data information at different times. The Address Drivers circuitry outputs this information to the Processor bus or System bus according to its destination.(a)

- LOCK-0 (Lock). Prevents other bus masters from gaining control of the System bus. LOCK-0 is activated when special lock prefixes are added to firmware instructions.(b)

- RD-0 (Read). When true, indicates that the MPU is performing a memory or I/0 read cycle, depending on whether S2-0 is false (memory) or true (I/0).(b)

- QS0-1 and QS1-1. Used to provide FPU with the following status information from the MPU instruction queue: no operation; first byte of op code from queue; empty the queue; or subsequent byte from queue. These signals are also available to the MPU from Test Connector J104.(a)

- RQ-0/GT0-0 (Request/Grant 0). Used by the FPU to gain control of the local bus from the MPU for operand transfers. Also available to the MPU from Test Connector J104.(a)

- RQ0/GT1-0 (Request/Grant 1). Not used in normal operation but is available on J104 to the MPU and on TP40 from FPU.(a)

- S0-0, S1-0, and S2-1 (Status Bits 0 through 2). These bits, when taken together, show which of eight states the MPU is in: interrupt acknowledge; I/0 read; I/0 write; halt; code access; memory read; I/0 read; or passive. In the FPU, these bits are used to show one of three possible states: read memory; write memory; or passive.(a)

- BHE-0 (Byte High Enable). During MPU or FPU State T1, BHE-0 enables the high byte of data onto D8-1 through D15-1 for read, write, and interrupt acknowledge cycles.(a)

- BUSY-1 (Busy). Output from the FPU to the MPU which is tested by the MPU "Wait" instruction and is used to synchronize the operation of the FPU with that of the MPU.(b)

- INT-1 (Interrupt). Used to create a level 7 interrupt through the Interrupt Controller when FPU generates an exception (abnormal or error condition).(c)

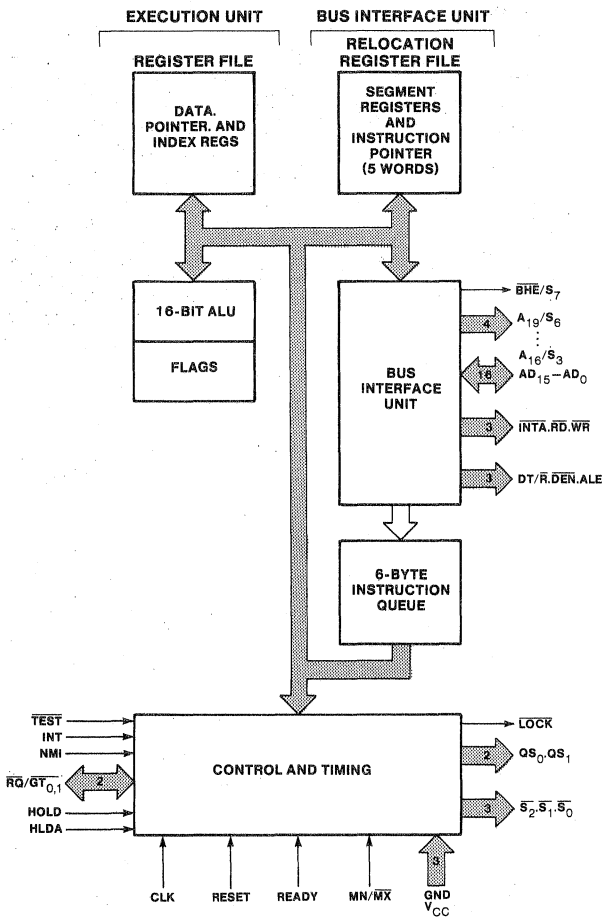
(a) Output of both MPU and FPU.
(b) Output of the MPU only.
(c) Output of the FPU only.

The MPU is a general-purpose microprocessor that has an address space of one megabyte. It operates at 4.9152 MHz and manipulates data in 8-bit or 16-bit word sizes (Figure 12-5).

The FPU is a specially designed co-processor which performs floating point numerical computations for the MPU. It also operates at 4.9152MHz (Figure 12-6).

CLOCK INPUT. DIRECT INPUT FROM CLK ON CLOCK GENERATOR. 33% DUTY CYCLE, 4.9152 MHz.	(19)	$\overline{LOCK}$ (29)	WHEN LOW, PREVENTS OTHER BUS MASTERS FROM CONTROLLING THE SYSTEM BUS. SOFTWARE ACTIVATED.
RESET, WHEN HIGH, MPU STOPS FOR FOUR CLOCK CYCLES, WHEN LOW, RESTARTS EXECUTION.	RST (21)	$\overline{RD}$ (32)	WHEN LOW, INDICATES A MEMORY READ IS IN PROGRESS, IF S2 IS HIGH, OR AN I/O READ, IF S2 IS LOW. FOR PROCESSOR BUS ONLY.
READY. WHEN HIGH, MEMORY OR I/O ACKNOWLEDGES IT WILL COMPLETE ITS DATA TRANSFER.	RDY (22)	QS0-1	NOT USED.
FIRMWARE "WAIT" INSTRUCTION LOOKS AT TEST. IF HIGH, MPU GOES INTO IDLE STATE. IF LOW, EXECUTION CONTINUES.	TEST (23)	$\overline{RQ/GT0-1}$	NOT USED.
		$\overline{S0-2}$ (26-28)	STATUS LINES. THESE INDICATE THE EIGHT STATES OF MPU DURING T1, T2, T4. PASSIVE (1,1,1) DURING T3, TW. BUS CONTROLLER USES THESE TO GENERATE ALL I/O AND MEMORY CONTROL SIGNALS. PROCESSOR BOARD USES THEM FOR LOCAL ACCESSES.
NON-MASKABLE INTERRUPT. LOW TO HIGH CAUSES AN INTERRUPT AT END OF CURRENT INSTRUCTION.	NMI (17)		
INTERRUPT. HIGH LEVEL TRIGGERED AND SAMPLED DURING LATEST CLOCK CYCLE OR EVERY INSTRUCTION. SOFTWARE MASKABLE.	INTR (18)		
		$\overline{BHE}$ (34)	BYTE HIGH ENABLE. USED TO ENABLE D8—15 ON READ, WRITE, AND INTERRUPT ACKNOWLEDGE CYCLES.
		A16—19 (39-35)	CARRY FOUR MOST SIGNIFICANT BITS OF A 20-BIT ADDRESS MEMORY OPERATION. INSIGNIFICANT IN A 16-BIT I/O OPERATION.
		AD0—15 (16-2,39)	TIME-MULTIPLEXED I/O/MEMORY ADDRESS (DURING T1) AND DATA (DURING T2, T3, TW, T4).

A. Pin Descriptions.



B. Functional Block Diagram.

(3820)4513-7

Figure 12-5. MPU Functional Block Diagram and Pin Descriptions.

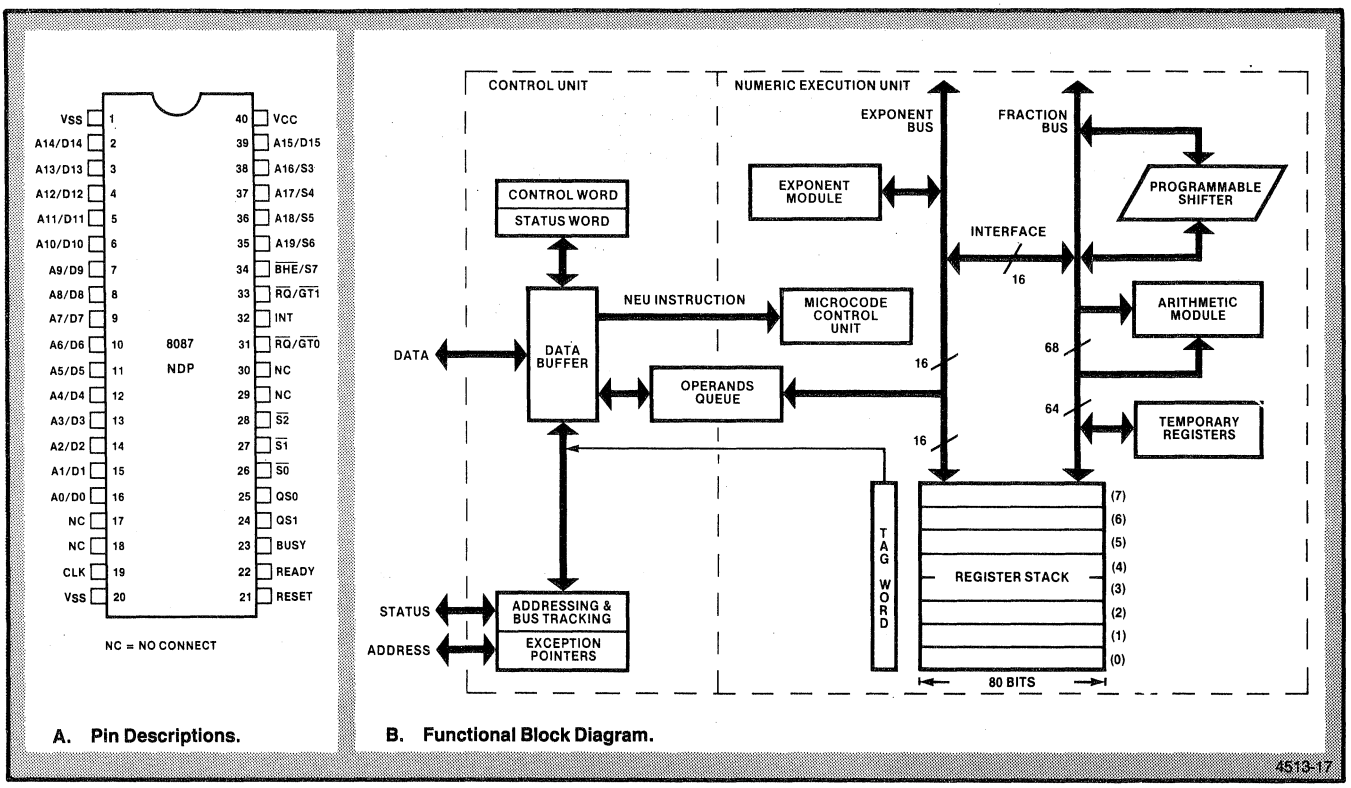


Figure 12-6. FPU Functional Block Diagram and Pin Descriptions.

The Bus Cycle

The MPU bus cycle consists of a variable number of clock states of approximately 200 ns each. The different kinds of clock states that can occur are T1, T2, T3, T4, Tw, and Ti. The minimum duration bus cycle is T1, T2, T3, and T4, in that order and one immediately after the other. However, some operations require that one or more clock states be inserted between T3 and T4. These states are Tw -- wait states. Also, sometimes states are inserted between bus cycles. These are Ti -- idle states. The MPU uses idle states for internal "housekeeping" operations. The FPU operates is such a manner as to enhance the operation of the MPU and greatly expand system capabilities and speed of operation. However, the FPU is not available to the programmer as an independent entity, but rather as a part of the CPU module.

MPU Internal States

During normal operation, the MPU performs one of eight types of bus cycles. These cycles are encoded in MPU outputs S0-0, S1-0, and S2-0. The cycle that corresponds with each of the eight bit patterns of S0-0, S1-0, and S2-0 is shown in Table 12-3, Status Word and Bus Controller Commands. This cycle status information is available during states T2, T3, and Tw. S0-0, S1-0, and S2-0 become binary 111 (inactive) during T4 and last through any Ti states. The FPU may perform two of the same bus cycles as the MPU, read memory and write memory. During these FPU cycles, the status of S0-0, S1-0, and S2-0 is the same as it would be during the identical MPU cycles.

The major operations of the MPU or FPU are reading and writing to memory or I/O address space, reset and initialization, and interrupt operations.

Memory and I/O Address Space Access

During T1, the MPU or FPU outputs an address. The MPU state information on S0-0, S1-0, and S2-0 becomes available and the address is latched during T2. Also, during T2, if the operation is a read operation, the direction of the data bus is changed. Data is then read from or written to memory or I/O locations during T2, T3, and Tw. The information on S0-0, S1-0, and S2-0 is used by different functional blocks of circuitry on the Processor board if the read or write access is local. Otherwise the S0-0, S1-0, and S2-0 information is converted directly into the I/O and memory access commands for the System bus by the Command Driver.

Reset and Initialization

MPU and FPU reset occurs when (RST-1) from the Microprocessor Control circuitry goes true. It must stay true for at least four clock cycles. The MPU executes no operations as long as (RST-1) is true. When (RST-1) goes false, an internal reset sequence is triggered that lasts about ten clock cycles. After this, the MPU and FPU fetch the instruction in location FFFF0. There must be at least 50<micro>s between power-up and the high to low transition of RST-1.

Interrupt Operations

There are two hardware interrupts to the MPU -- NMI-0 and INTR-1. NMI-0 is the non-maskable interrupt and INTR-1 is the maskable interrupt request input.

Non-Maskable Interrupt. If NMI-0 goes from low to high and Non-Maskable Interrupt goes false for more than two clock cycles, NMI-0 is latched by the MPU and is serviced immediately after the current instruction is completed. NMI-0 is only available on the processor Board Test Fixture Connector and is not part of the System bus.

Maskable Interrupt. When INTR-1 goes true during the clock Maskable Interrupt cycle preceding the end of the current instruction, the MPU is triggered into a response sequence. The MPU executes two consecutive interrupt acknowledge cycles (two bus cycles). (See Interrupt Controller description.) LOCK-0 is held true from T2 of the first cycle to T2 of the second cycle. During the second bus cycle, a byte is fetched from the PIC. This byte specifies what source requires the interrupt and also addresses a location in ROM from which the appropriate interrupt service routine can be determined.

INTR-1 can be disabled by resetting a status bit in the MPU by means of software instructions.

INTERRUPT CONTROLLER (SCHEMATIC A2-2)

The Interrupt Controller handles a maximum of eight System bus and Processor board interrupt signals (INT0-0 through INT7-0) and determines the priority of each interrupt signal in relation to the others. The Interrupt Controller also has the capability of addressing eight other optional Programmable Interrupt Controllers.

The input signals are:

- COMINT-0 (Communications Interrupt). Signals that the RS-232 Communications Interrupt has just received a character.
- TIMERINT-0 (Timer Interrupt). Indicates that at least one of these signals is active: TIMR1-1, (TXEMT-0), (TXRDY-0), an RS-232 status change interrupt, or OUT0-1.
- AWT-0 (Advanced Write). Advanced write signal for Processor board circuits.
- INT0-0 through INT7-0 (Interrupt 0 to 7). System bus interrupt signals. (COMINT-0 can generate INT0-0; KBINT-0 can generate INT4-0, BTINT-0 and TIMERINT-0 can generate INT5-0, and FPU Interrupt can generate INT7-0).
- BUSAEN-0 (Bus Address Enable). Shows that the MPU is bus master of the System bus.
- INTA-0 (Interrupt Acknowledge). Causes the Programmable Interrupt Controller to place an interrupt routine address ("vectoring information") on the System data bus.
- OBIO8&A-0 (On-Board I/O 8&A). When true enables the Programmable Interrupt Controller for communication with the MPU.
- RD-0 (Read). Read signal from the MPU.
- RST-0 (Reset). Initializes logic during power-up and system reset.
- BTINT-0 (Bus Timeout Interrupt). Indicates a Bus Timeout has occurred.

The output signals are:

- AD8-1, AD9-1, and AD10-10 (MPU Address/Data Bits 8 through 10). These carry the addresses of a maximum of eight other optional Programmable Interrupt Controllers.
- INTR-1 (Interrupt). Interrupt signal to the MPU.
- 1STINTA-0 (First Interrupt). Indicates to the MPU that no data will be transferred.
- OBINTA-0 (On-Board Interrupt). Indicates that the Programmable Interrupt Controller is placing data on D0-1 through D7-1. Also combines with other signals in the Address Decoding circuitry to produce OBDEN-0.

The heart of the Interrupt Controller is the Programmable Interrupt Controller (PIC) integrated circuit. The PIC receives interrupt signals (INT0-0 through INT7-0) and INTA-0 from the System bus, and control signals from other circuits on the board. It receives and transmits data from the Processor data bus, and it outputs INTR-1, OBINTA-0, and AD8-1 through AD10-1 (which can carry addresses for eight other optional Pics). The other circuitry in the Interrupt Controller inverts and buffers INT0-0 through INT7-0, provides enabling for the AD8-1 through AD10-1 drivers, and produces 1STINTA-0. Also, BUSAEN-0 and INTA-0 are NANDed for the INTA-0 input of the PIC.

Programmable Interrupt Controller

This integrated circuit consists of five main blocks of circuitry. They are the local data bus buffer, the interrupt registers and logic, the control logic, the read and write logic, and the cascade buffers and comparator (Figure 12-7 and Figure 12-8).

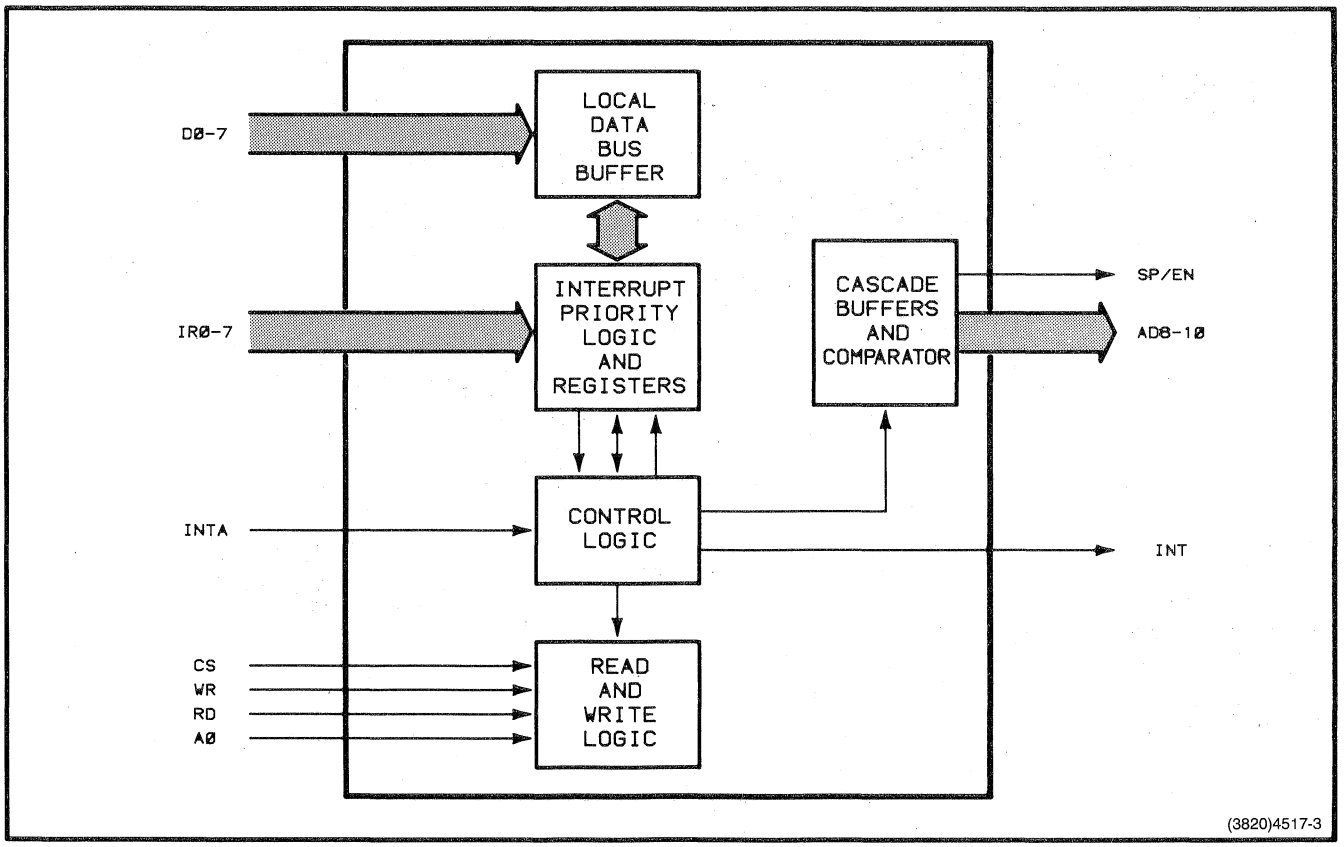


Figure 12-7. Programmable Interrupt Controller Block Diagram.

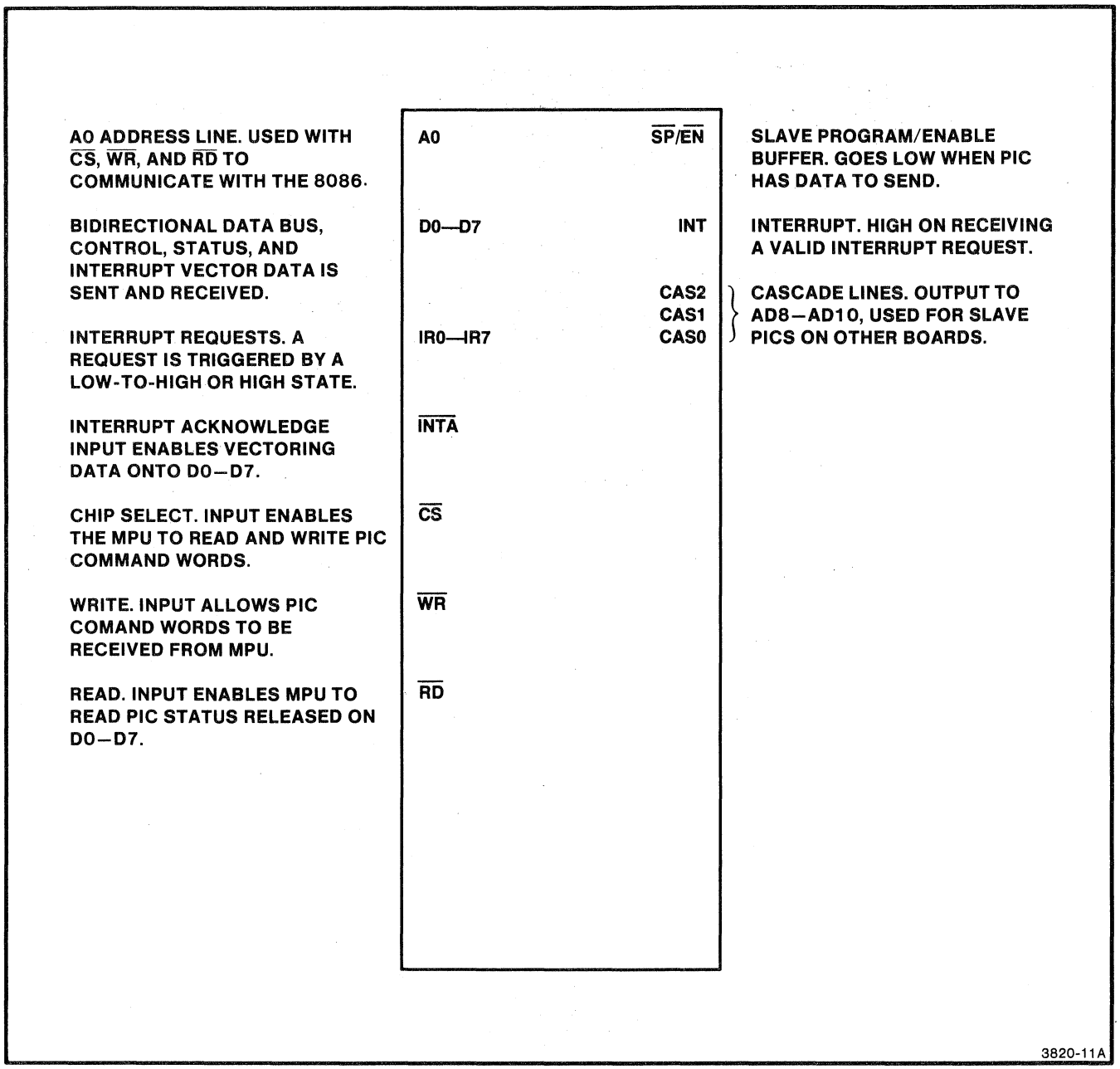


Figure 12-8. Programmable Interrupt Controller Pin Descriptions.

Local Data Bus Buffer. Eight bits, DO-1 through D7-1, are transmitted and received by this tri-state buffer. The system firmware supplies control words and status information to the PIC through this buffer. An internal bus connects the data bus buffer to the Interrupt Priority Logic Registers circuitry.

Interrupt Priority Logic and Registers. There are three registers in addition to the priority logic in this circuitry. The priority logic determines from the register holding INTO-O through INT7-O bits (that are requesting service) which bit has the highest priority. This bit is strobed into the corresponding bit position in the in-service register. Firmware uses the third register to mask the INTO-O through INT7-O bits in the requesting service register. This masking changes the priority of the INTO-O through INT7-O interrupt signals.

Control Logic. When the system signal INTA-O goes true, it causes the control logic to transmit interrupt service routine addresses to the System bus via the local data bus buffer. The control logic also issues INTR-1 to the MPU if it receives a signal from the priority logic.

Read and Write Logic. OBI08&A-O goes true to enable reading or writing access to the PIC. If RD-O goes true, any of the three registers and the interrupt level of the Interrupt Priority Logic and Registers circuitry can be output to the Processor data bus. If AWT-O goes true, firmware can write PIC control words to registers in the Read and Write Logic circuitry. AO-1 selects in conjunction with RD-O and AWT-O whether the interrupt registers or the command word registers are read from or written to.

Cascade Buffers and Comparator. If other PICs are added to the system, this circuitry addresses them. These slave PICs would be controlled by the master PIC on the Processor board. AD8-1 through AD10-1 would carry the address of any one of eight additional PICs.

There are two phases of operation in the PIC. These are initialization and normal operation. During initialization, soon after the 4170 is powered up, the system firmware sends set-up data and control words to the PIC. After initialization, the PIC is ready to receive and prioritize the interrupt request signals on INTO-O through INT7-O.

The Interrupt Sequence

The following sequence of events occurs in the normal operation of the Interrupt Controller circuitry.

1. One or more of the INT0-0 through INT7-0 bits goes true. This sets (because of the inverter) the corresponding bit(s) in the "interrupt request" register in the Interrupt Priority Logic and Registers circuitry in the PIC.

2. The PIC determines the priority of these INT0-0 through INT7-0 lines, and sends INTR-1 to the MPU if one of the INTx-0 lines is a higher priority interrupt request than any currently being processed.

3. The MPU receives the INTR-1 signal and responds with a low on its INTA line.

4. The PIC now sets the highest priority bit in its in-service register and the corresponding interrupt request register bit is reset. No signals are sent from the PIC at this time.

5. Now the MPU sends a second low on INTA-0. This causes the PIC to place an eight-bit interrupt-service-routine (vectoring) address on the Processor data bus, D0-1 through D7-1.

6. The MPU reads this vectoring address on the Processor data bus and begins the service routine. The Interrupt Controller is now done with this interrupt cycle.

If the duration of an interrupt request on INT0-0 through INT7-0 is not long enough to be true at step 4, the PIC automatically issues a vectoring address as if INT7-0 were true.

1STINTA-0 and Cascade Addresses for Slave PICs

During the interrupt sequence, the MPU sends two INTA-0 signals. At the time of the first INTA-0, 1STINTA-0 is generated by a toggling JK flip-flop and a NAND gate. At the time of the second INTA-0, the cascade address (AD8-1 through AD10-1) of the slave PIC is driven onto the Processor address data bus. This is accomplished by the same toggling JK flip-flop and another NAND gate.

ADDRESS DRIVERS (SCHEMATIC A2-2)

The Address Drivers latch 20 bits of address and BHEN-0, and then drive these onto the Processor address bus during a local read or write, or onto the System bus during a system read or write.

The input signals are:

- SO-0, S1-0, and S2-0 (Status 0, 1, & 2). Informs the Processor board whether the MPU is in an interrupt acknowledge, read I/O, write I/O, halt, code access, read memory, or write memory state; or if the FPU is in a read or write state.

- AD0-1 through AD19-1 (Address and Data Bits 0 through 19). Output directly from the MPU or FPU. AD0-1 through AD19-1 carries both address and data information in time-multiplexed mode.

- BHE-0 (Byte High Enable -- Unbuffered). Directly from the MPU and FPU.

- ALE-1 (Address Latch Enable). Enables the local address latches.

- BUSAEN-0 (Bus Address Enable). Enables the outputs of the system address latches.

The output signals are:

- LS0-0, LS1-0, and LS2-0 (Latched Status 0, 1, & 2). Informs the system what internal state the MPU or FPU is in. Refer to S0-0, S1-0, and S2-0.

- A0-1 through A19-1 (Processor Address Bus). Carries address information for the Processor board.

- ADR0-1 through ADR19-1 (System Address Bus). Carries address information for the system.

- LBHE-0 (Latched Byte High Enable). Processor board equivalent of BHEN-0.

- BHEN-0 (Byte High Enable). Enables DAT8-1 through DAT15-1 on read, write, or interrupt acknowledge cycles.

The Address Drivers circuitry consists of six packages of D-type latches. These form two sets of latches. One set outputs LS0-0 through LS2-0, LBHE-0, and the Processor bus address bits. The other set outputs BHEN-0 and the System bus address bits.

When the MPU or FPU outputs address information on AD0-1 through AD19-1 and ALE-1 makes a high-to-low transition, the Processor bus set latches this data. It also latches S0-0 through S2-0 and BHE-0. Since this set of latches is permanently enabled for output, the latched data appears immediately as A0-1 through A9-1, LS0-0 through LS2-0, and LBHE-0.

The System bus set of latches operates like the Processor bus set, except that the latches are not permanently enabled for output. BUSAEN-0 provides this enabling function.

DATA DRIVERS/RECEIVERS (SCHEMATIC A2-2)

The Data Drivers/Receivers transfer MPU or FPU data on AD0-1 through AD15-1 to or from either D0-1 through D15-1, the processor data bus, or DAT0-1 through DAT15-1, the System data bus.

The input signals are:

- AD0-1 through AD15-1 (Multiplexed Address and Data Bits). Address and data bits directly output from the MPU and FPU.
- DT/R-0 (Data Transmit/Receive). Data bus direction control.
- DEN-1 (Data Enable).
- SP/EN-0 (Enable Buffer). Enabling signal from the PIC.
- OBDEN-0 (On-Board Data Enable).

The output signals are:

- D0-1 through D15-1 (Processor Data Bus Bits).
- DAT0-1 through DAT15-1 (System Data Bus Bits).

Four bus transceivers and one NAND gate make up the Data Drivers/Receivers circuitry. The bus transceivers are divided into two sets. One handles D0-1 through D15-1, the Processor data bus, and the other set handles DAT0-1 through DAT15-1, the System data bus. Data bits are transmitted and received in each set.

If OBDEN-0 goes true and DT/R-0 is false, the data bits on AD0-1 through AD15-1 are transmitted to D0-1 through D15-1. But if OBDEN-0 is true and DT/R-0 goes true, data on D0-1 through D15-1 is transmitted back through the transceivers and becomes AD0-1 through AD15-1, which is received directly by the MPU and FPU.

Data is transmitted to the System data bus, DATO-1 through DAT15-1, when DT/R-O goes false, DEN-1 is true, and SP/EN-O is false. If DT/R-O is true during data transmission, the System bus bits become ADO-1 through AD15-1.

BUS COMMAND DRIVER (SCHEMATIC A2-2)

The Bus Command Driver decodes MPU and FPU status signals in order to generate System bus read, write, and interrupt commands. The Bus Command Driver also generates control signals for the Processor board address and data drivers and latches.

The input signals are:

- SO-O, S1-O, and S2-O (Status O through 2). Three lines that indicate what type of cycle the MPU is performing: I/O read, I/O write, memory read, memory write, halt, or interrupt acknowledge; or, in the case of the FPU, memory read or memory write.
- BUSAEN-O (Bus Address Enable). Shows that the MPU is System bus master, when true.
- OBADR-O (On-Board Address). When true, indicates that the Processor board circuitry is being accessed. When false, indicates that the System bus is being accessed.
- CLK-1 (Clock). Synchronizing clock signal for the Processor board.

The output signals are:

- DT/R-O (Data Transmit/Receive). Processor and System data bus transmit and receive.
- ALE-1 (Address Latch Enable). Enables the Processor and System bus address latches.
- (MCE-1) (Master Cascade Enable). Clocks a JK flip-flop in the Interrupt Controller circuitry that enables three line drivers to output AD8-1 through AD10-1 (an address for one of eight optional Interrupt Controllers).
- (DEN-1) (Data Enable). Enables system data drivers if OBINTA-O is false.
- INTA-O (Interrupt Acknowledge). Signals the Interrupt Controller to place vectoring data on the System data bus.
- IOWC-O (I/O Write Command).
- AIOWC-O (Advanced I/O Write Command).
- IORC-O (I/O Read Command).
- MRDC-O (Memory Read Command).
- MWTC-O (Memory Write Command).
- AMWC-O (Advanced Memory Write Command).

The Bus Command Driver circuitry consists solely of the Bus Controller. This device combines control logic, MPU status decoder, control signal generator, and Command Generator circuitry. Figure 12-9 shows the signal paths among these blocks of circuitry. See also Figure 12-10.

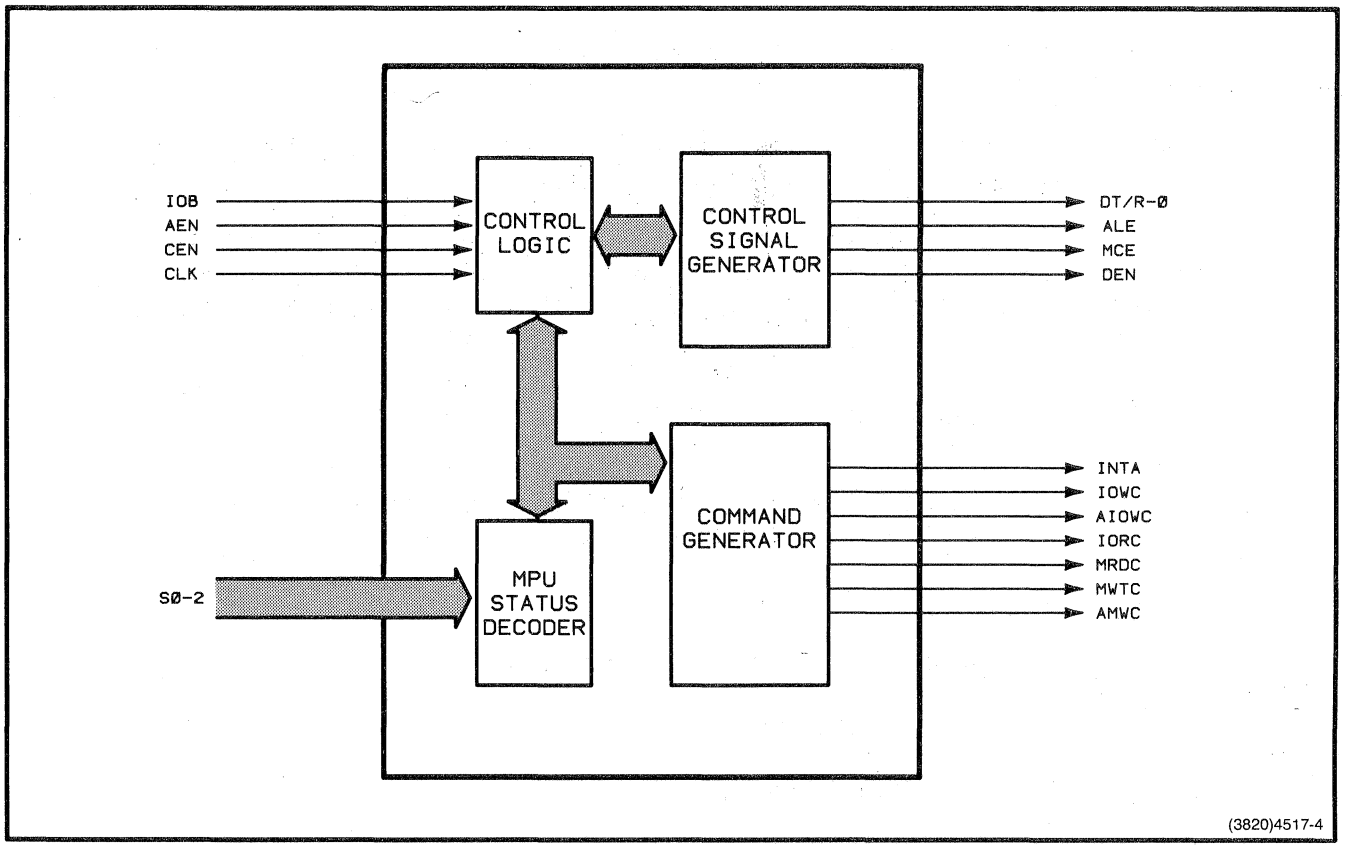


Figure 12-9. Bus Controller Block Diagram.

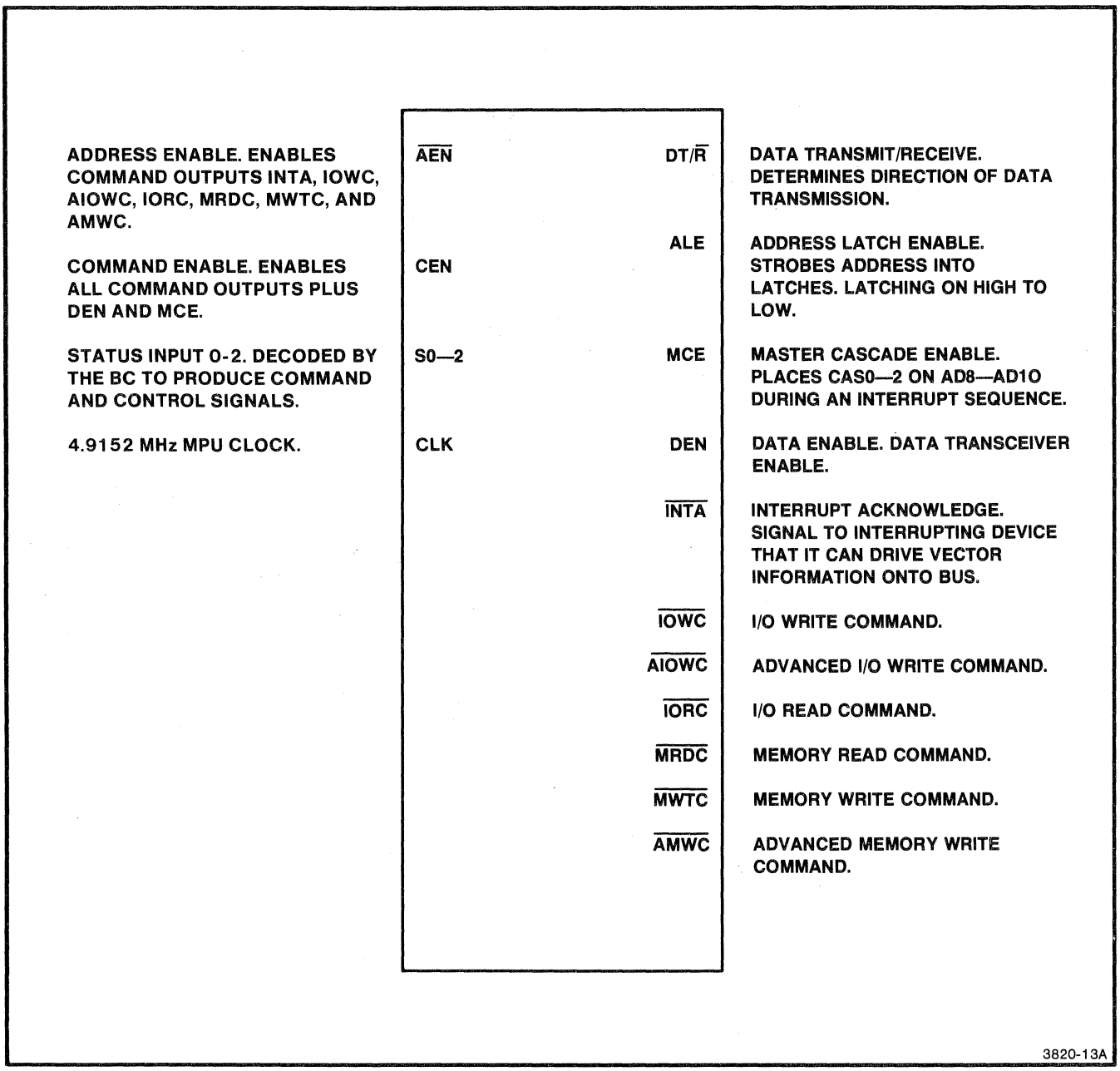


Figure 12-10. Bus Controller Pin Descriptions.

Control Logic

IOB-1 is permanently tied low. This sets up the control logic enabling the command generator to output command signals (AIOWC-0, MWTC-0, MRDC-0, etc.) no earlier than 105ns after BUSAEN-0 goes true. When BUSAEN-0 is false, the control logic tri-states (places in a high impedance condition) the command signal outputs.

OBADR-0 also affects command signal output. When OBADR-0 is true, all command signal outputs in addition to the (DEN-1) and (MCE-1) outputs are inactive. If OBADR-0 goes false, these same outputs are enabled.

CLK-1 synchronizes the operations within the Bus Controller.

Control Signal Generator

This part of the Bus Controller outputs DT/R-0, ALE-1, (MCE-1), and (DEN-1). These signals control, respectively, direction of data transmission, enabling of the address latches, enabling of AD8-1 through AD10-1 (additional Interrupt Controller addresses), and enabling of System bus data drivers if OBINTA-0 is false.

When BUSAEN-0 is true (MPU is bus master) and OBADR-0 is false (MPU is not accessing a device on the Processor board), the Control Signal Generator activates its control outputs according to whether S0-0 through S2-0 indicate a read, write, or interrupt cycle. See Figure 12-11 for the sequencing of these signals.

Notes. Notes 1 through 3 and A through V appear in Figure 12-11.

Figure 12-11 shows five different cycle types -- memory read, memory write, I/O read, I/O write, and interrupt acknowledge. Only one type can occur at any one time.

1. When this line is low, it indicates that one or more of the status lines (S0-0, S1-0, or S2-0) is true.

2. AD0-1 through AD19-1 are not used by the Bus Controller, but are drawn to show their relationship to ALE-1 (Address Latch Enable).

3. (MCE-1) (Master Cascade Enable) occurs only when all three status lines are true. This signals the Interrupt Acknowledge cycle.

A,B,C,D. When the MPU or FPU drives at least one status line true (S0-0 through S2-0), and CLK-1 is false, the Bus Controller generates ALE-1 and, during an interrupt acknowledge (INTA-0) cycle, (MCE-1) to strobe the address latches.

E,F,G ALE-1 is removed and the data direction is switched on write commands.

H,I,J,K (MCE-1) is removed, commands are driven onto the bus, and writing of data is enabled on a write command.

L,M Reading of data is enabled on read or INTA-0 cycles.

N,O A write command is initiated on memory and I/O write cycles. Note that the advanced write line (AIOWC-0 and AMWC-0) is already true.

P,Q,R,S All commands are moved and the reading of data is disabled.

T,U,V Writing of data is disabled and data reverts back to the MPU and FPU outputs in anticipation of the MPU or FPU address output for the next cycle.

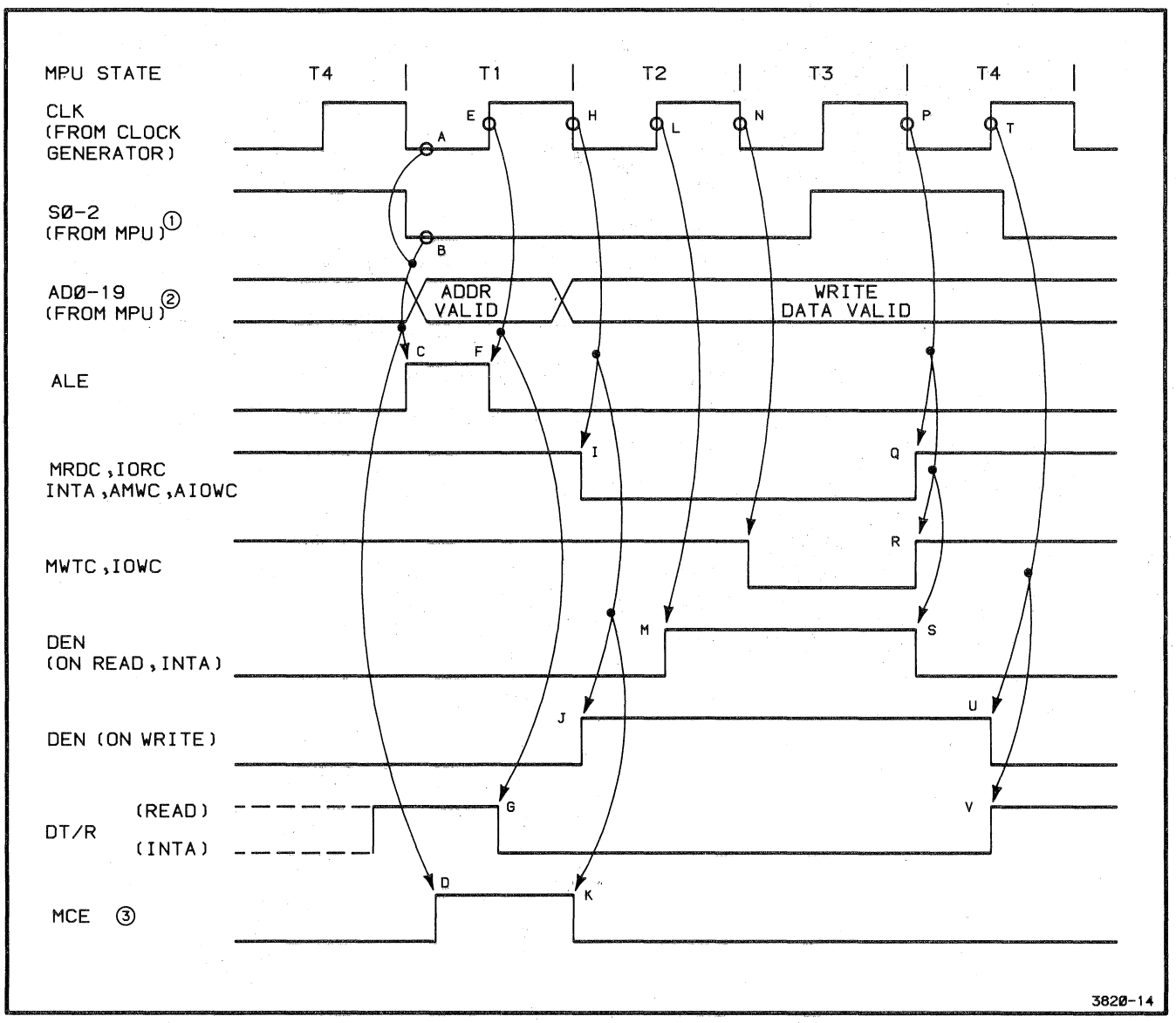


Figure 12-11. Bus Controller Timing.

MPU Status Decoder

This circuitry decodes the SO-0 through S2-0 signals from the MPU and FPU. These lines indicate the eight different states that the MPU can be in; or the three different states that the FPU can be in. Table 12-3 shows for each bit combination what MPU or FPU state corresponds to it. It also shows the corresponding command signal for each bit combination. These command signals are output only when the corresponding SO-0 through S2-0 bit combination is present.

Table 12-3

STATUS WORD AND BUS CONTROLLER COMMANDS

BUS			PROCESSOR STATE	CONTROLLER COMMAND
SO	S1	S2		
0	0	0	Interrupt Ack.	INTA
1	0	0	Read I/O Port	IORC
0	1	0	Write I/O Port	IOWC, AIOWC
1	1	0	Halt	none
0	0	1	Code Access	MRDC
1	0	1	Read Memory	MRDC (a)
0	1	1	Write Memory	MWTC, AMWC (a)
1	1	1	Inactive	none (a)

(a) Can be generated by the FPU.

Command Generator

This circuitry outputs the command signals: MRDC-0, MWTC-0, IORC-0, IOWC-0, AMWC-0, AIOWC-0, and INTA-0. When BUSAEN-0 is true and OBADR-0 is false, the Command Generator outputs the command signals corresponding to the three bits of SO-0 through S2-0. See Table 12-3, Status Words and Bus Controller Commands, for this correspondence.

READ ONLY MEMORY (ROMS) (SCHEMATIC A2-3)

The ROMs hold a maximum of 32K bytes of system firmware and outputs this to the Processor bus when addressed.

The input signals are:

- A1-1 through A112-1 (Processor Address Bits 1 through 14). A1-1 through A112-1 carry word addresses (A1-1 through A11-1) and ROM bank selection information (A12-1 through A112-1).
- ALE-1 (Address Latch Enable). ALE-1 is used to disable the ROMs when the address is changing.
- LS1-0 (Latched Status Bit 1). LS1-0 is a read/write status bit and is used to enable the ROMs during a read operation.

The output signals are:

- D0-1 through D15-1 (Processor Data Bits 0 through 15). These bits carry data information for the MPU and FPU.

ROM Configuration

The ROM circuitry is made up of eight ROM integrated circuits, arranged in four banks of two ROMs each. The four banks supply a total of 32K bytes. Each bank contains 4K by 16 bits of memory address space. The lower byte of each bank outputs D0-1 through D7-1 and the upper byte outputs D8-1 through D15-1.

Firmware

Starting from the leftmost bank on the schematic, the first three banks contain system firmware in erasable/programmable ROMs (EPROMs). The fourth bank contains a firmware jump table and patch code in EPROMs.

Straps

There are three kinds of straps that affect the operation of the ROMS circuitry: ROM type selection jumper straps, a ROM wait-states cut strap, and two ROM address-decoding cut straps.

The ROMS circuitry is designed to accept a variety of different ROM integrated circuit packages. A chart specifying these different ROM types is found on the schematic itself (Schematic A2-3). The chart also specifies the strap positions required for each type of ROM.

Cut-strap W475 selects the number of ROM wait states that the MPU inserts between state T3 and T4. If the strap is not altered, the MPU inserts one wait-state. This allows the use of ROMs with chip select access time of a maximum 580ns. However, if all the ROMs on the Processor board have an access time of 380ns or less, strap W475 may be altered so that the MPU inserts no wait-states between T3 and T4. This latter position increases the speed of the MPU when executing on-board ROM firmware.

The straps at W126 allow some or all of the ROM memory address space to be diverted from being used locally on the Processor board to being used by the system. There are three positions for the straps. See the appendix on straps to determine the settings for each of the three positions. If the straps are not altered, all 32K bytes of the Processor board ROMs are addressed in the range F8000 to FFFFF. However, the straps may be altered to two positions. In one position, only Processor board ROMs addressed in the range FC000 to FFFFF output data to the MPU or FPU. In the other position, there is no Processor board ROM space -- all memory space is accessed on the System bus.

The ROM circuitry receives an address from the MPU or FPU and puts the data for that address on the Processor bus. The sequence of events when the MPU or FPU reads from a ROM bank is the following.

1. An address from the MPU or FPU is latched by the Address Driver circuitry.

2. The Address Decoding circuitry determines whether it is a Processor board ROM address, which is normally any address in the range F8000 to FFFFF. If the address is in this range, Address Decoding drives OBROM-0 true.

3. The bank decoder selects one ROM bank in response to A13-1 and A112-1 in addition to ALE-1, LS1-0, LS2-0, and A15-1 through A19-1. See Figure 12-12, ROM Bank Decoder Logic, and Table 12-4, Selection Bits for ROM Banks for an exact description of how the bank decoder works.

4. The two ROMs in the selected ROM bank place their data on the Processor data bus lines, D0-1 through D1-15.

5. The Data Drivers/Receivers pass this data on to the MPU or FPU.

Table 12-4

SELECTION BITS FOR ROM BANKS

BANKS	ADDRESS BITS							
	A19-1	A18-1	A17-1	A16-1	A15-1	A112-1	A13-1	A12-1
F8000	1	1	1	1	1	0	0	X(a)
FA000	1	1	1	1	1	0	1	X(a)
FC000	1	1	1	1	1	1	0	X(a)
FE000	1	1	1	1	1	1	1	X(a)

(a) "x" is a "don't care" condition.

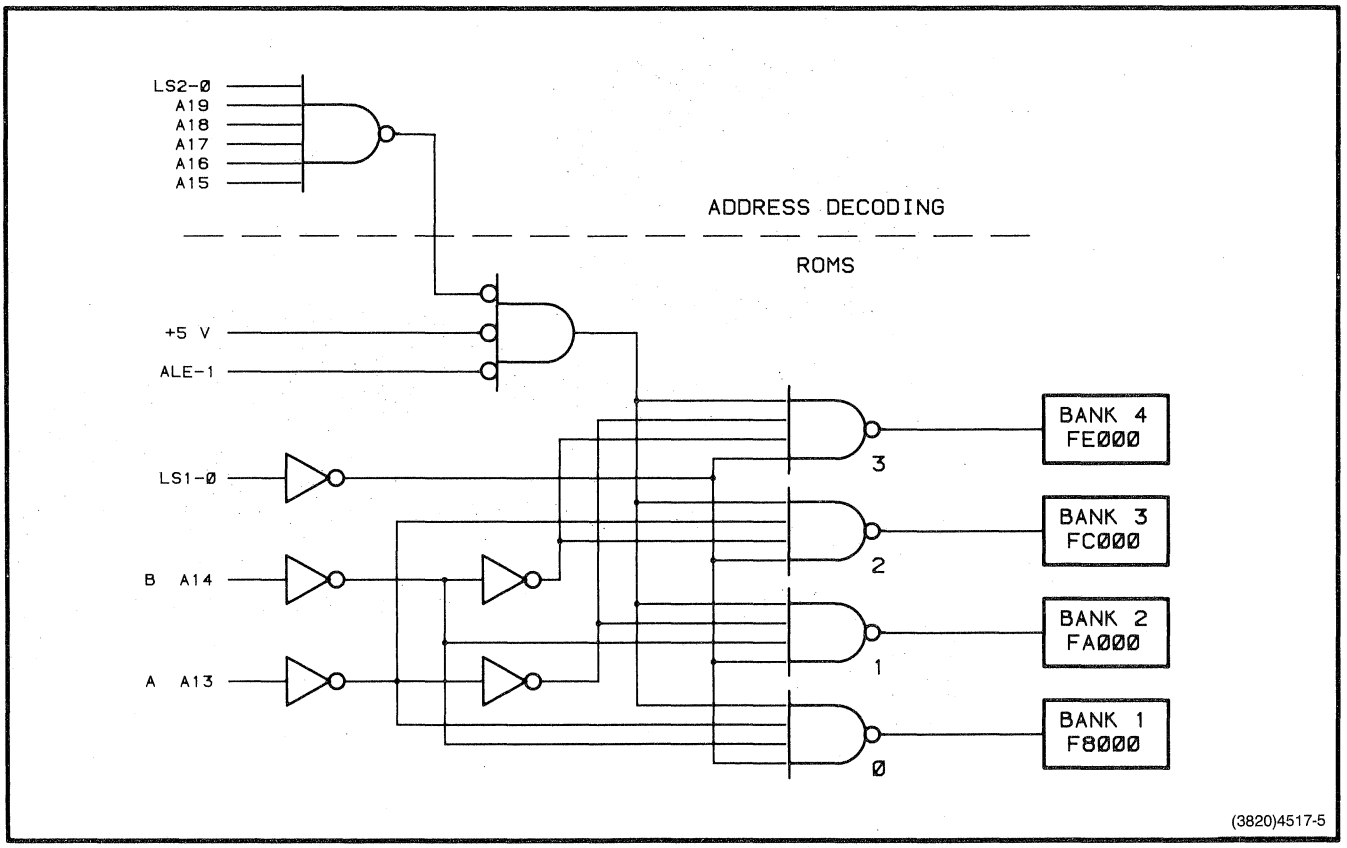


Figure 12-12. ROM Bank Decoder Logic.

NON-VOLATILE RAM (SCHEMATIC A2-4)

The Non-Volatile RAM stores various initialization and system parameters in 512 bytes of CMOS RAM.

The input signals are:

- RD-0 (Read). Allows the MPU to read from the CMOS RAM.
- AO-1 through A8-1 (Processor Address Bits 0 through 8).
- LBHE-0 (Latched Byte High Enable). Allows the MPU to access data bits D8-1 through D15-1.
- AWT-0 (Advanced Write). Allows the MPU or FPU to write to the CMOS RAM.
- DT/R-0 (Data Transmit or Receive) Causes the CMOS RAM to accept data when false, and to output data when true.
- OBRAM-0 (On-Board RAM). When RAM on the Processor board is addressed, OBRAM-0 goes true.

The output signals are:

- DO-1 through D15-1 (Processor Data Bus Bits 0 through 15).
- CMD-1 (Command). Created in the Non-Volatile RAM circuit block by the logical OR of AWT-0 and RD-0.

Four 10212-bit by 12-bit CMOS RAMs (of which only a 256-bit by 12-bit portion is used) combine to give a total memory capacity of 512 bytes. Also included in this circuitry are six logic gates that decode various input signals that control reading and writing access.

When power to the 4170 is removed, the CMOS RAM retains its contents. A 2.4 V rechargeable NiCad battery provides the voltage to make this possible. A trickle charge circuit assures that the battery remains fully charged.

Read Access

This RAM occupies FE00 to FFFF in I/O address space. (See the Address Decoding description to see how CMOS can be placed in memory address space.) When the MPU or FPU presents an address in this range to the Address Decoding circuitry, OBRAM-0 goes true. The MPU also causes RD-0 to go true. At this point, A0-1 goes false, if only the low order byte, D0-1 through D7-1, of the Processor data bus is needed. These conditions on OBRAM-0, RD-0, and A0-1 enable the two RAMs that output D0-1 through D7-1. Now DT/R-0 goes true, and the RAMs output D0-1 through D7-1 onto the Processor data bus.

A read access of the high order byte, D8-1 through D15-1, is accomplished in the same manner as the low order byte, except that LBHE-0 goes true.

Write Access

During a write access, AWT-0 goes true instead of RD-0, and DT/R-0 goes false. OBRAM-0, LBHE-0, and A0-1 operate as they do in a read access.

Note that INIT-0 disables all RAMs if it goes true.

MICROPROCESSOR CONTROL (SCHEMATIC A2-1)

The Microprocessor Control generates the 4.9152 MHz clock signal for the MPU and FPU. Also, it synchronizes the reset and ready lines.

The input signals are:

- o INIT-O (Initialize). Used by the Clock Generator to generate (RST-1) for the MPU and FPU.
- o ACK1-O (Acknowledge 1). When true, indicates that a device on the System data bus is ready to receive data or has data ready to transmit.
- o ACK2-O (Acknowledge 2). Has the same meaning as ACK1-O, but allows the MPU and FPU to insert one wait-state (Tw) into the bus cycle.
- o TIMR1-1 (Timer 1). A timer output from the Programmable Timer & Baud Rate Generator circuitry.
- o TEST-O (Test). Enables the MPU to terminate an on-board or off-board read or write operation. Also used by the MPU to monitor the FPU BUSY-1 signal.
- o RDYAND-O (Ready AND). Disables the (RDY-1) line to the MPU and FPU and causes MPU and FPU to enter a wait-state (Tw). Input from Test Connector J104 and is used for testing only.
- o RDYOR-O (Ready OR). Enables the (RDY-1) line to the MPU and FPU and causes MPU and FPU to exit a wait state (Tw). Input from Test Connector J104 and is used for testing only.
- o 1STINTA-O (First Interrupt Acknowledge). Goes true when the MPU sends the first of its two INTA-O signals in the interrupt sequence.
- o OBINTA-O (On-Board Interrupt Acknowledge). When true, indicates that the PIC is placing data on the Processor data bus.
- o OBROM-O (On-Board ROM). May cause (RDY1-1) to activate if the cut strap W475 is altered, resulting in no ROM wait-states.

The output signals are:

- (OSC-1). TTL level, 14.7456 MHz square wave.
- (CLK-1). 4.9152 MHz, 33% duty cycle clock signal which is fed to MPU and FPU.
- (RST-1). Resets the MPU and FPU when true and has similar timing to INIT-0.
- (RDY-1). When true, acknowledges that an addressed memory or I/O device will complete its data transfer.

This circuitry consists of one D-type flip-flop, some logic gates, and the Clock Generator and Driver (CGD). The CGD performs three functions. It uses a frequency derived from an external crystal to generate the (CLK-1) and (OSC-1) outputs. It synchronizes INIT-0 to produce (RST-1) and derives the (RDY-1) output from (RDY1-1) and (AEN1-0) or from (RDY2-1) and (AEN2-0) (Figure 12-13).

The crystal connected between (X1-1) and (X2-1) on the CGD is a series resonant, fundamental mode type and has a frequency of 14.7456 MHz. The output of the oscillator circuitry in the CGD is buffered and output on (OSC-1). The (F/C-0) input is gated low to select the crystal as the source of the (CLK-1) output frequency. Hard wiring (CSYNC-1) to ground is necessary to use the internal oscillator of the CGD.

The CGD reduces noise on the INIT-0 signal by feeding it through a Schmitt trigger and then synchronizes it with (CLK-1) by means of a flip-flop.

Inside the CGD, (AEN1-0) is inverted and ANDed with (RDY1-1) as is (AEN2-0) with (RDY2-1). The outputs of both ANDed pairs are ORed and synchronized with (CLK-1) by a D-type flip-flop. The Q output of the flip-flop is (RDY-1). So, if (AEN1-0) and (RDY1-1) are true, (RDY-1) goes true. The same is true of (AEN2-0) and (RDY2-1).

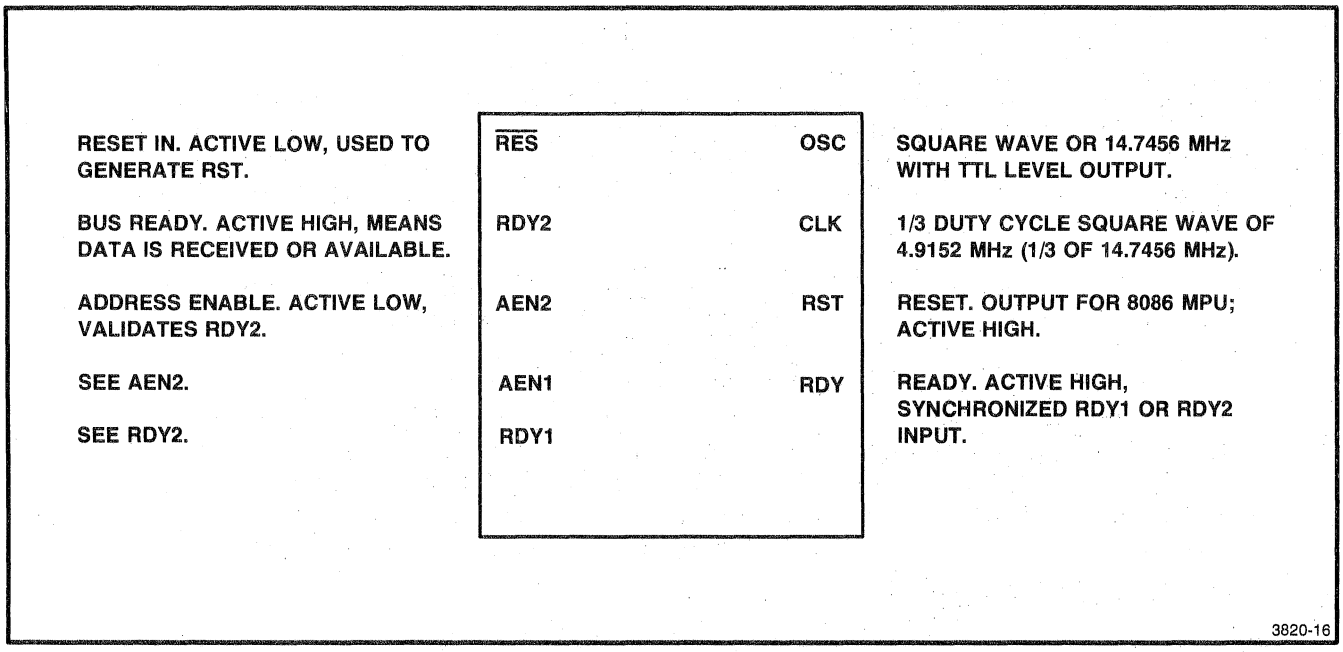


Figure 12-13. Clock Generator and Driver Pin Descriptions.

Clock and Reset

When the 4170 is powered-up, the CGD begins operation and generates the (OSC-1) and (CLK-1) signals until the 4170 power is removed.

INIT-0 originates in the power supply and is initially true after power-up. INIT-0 goes false about 40 ms after power up. At the next falling edge of (CLK-1), the (RST-1) output of the CGD goes true, resetting the MPU and FPU.

Ready Signals

Essentially, (AEN1-0) and (RDY1-1) are the Processor board ready signals and (AEN2-0) and (RDY2-1) are the off-board, or System bus, ready signals. The (AENx-0) signals act as qualifying or validating signals for the (RDYx-1) signals.

Processor Board Ready. The MPU and FPU receive a (RDY-1) input if any of the following conditions occur:

- RDYAND-0 is false and OBINTA-0 goes true.
- RDYAND-0 is false and 1STINTA-0 goes true.
- RDYAND-0 is false and, if cut-strap W475 is altered, OBROM-0 goes true.
- RDYAND-0 is false and OBADR-0 is true while T3W12-1 goes true.
- RDYAND is false and TEST-0 goes false while, TIMR1-1, and T3W12-1 go true.
- RDYAND is false and RDYOR-0 goes true.

However, if RDYAND-0 goes true all the before-mentioned conditions are disabled and neither the MPU or FPU receive a (RDY-1) input.

System Bus Ready. For the MPU to receive a (RDY-1) input from the System bus, the (AEN2-0) input must be true. This occurs under the following conditions -- RDYAND-0 is false and BUSAEN-0 is true and at least one of the following is true: INTA-0, AIOWC-0, IORC-0, MRDC-0, or AMWC-0. Once (AEN2-0) is true, an ACK1-0 going true triggers a (RDY-1) input to the MPU and FPU. Or, if the device sends an ACK2-0 signal instead, the MPU and FPU receive a (RDY-1) input on the next positive edge of the (CLK-1) signal after ACK2-0 goes true.

BUS TRANSFER LOGIC (SCHEMATIC A2-1)

Bus Transfer Logic allows the Processor board to gain control of the System bus to perform data transfers, and also to relinquish control to other bus masters.

The input signals are:

- OBADR-0 (On-Board Address). When false, shows that a system data transfer is to be done.
- BUSY-0 (System Bus Busy). When true, BUSY-0 indicates that another bus master is using the System bus for data transfer.
- CBRQ-0 (Common Bus Request). CBRQ-0 is used by a lower priority bus master to assert that it has data to transfer on the bus.
- BUSGRT-0 (Bus Grant). Input from Test Connector J104 used to force Processor board control of the System bus for testing.
- BCLK-0 (Bus Clock). BCLK-0 clocks all three flip-flops in this circuitry.
- BPRN-0 (Bus Priority In). The response from bus arbitration logic which grants bus mastership to the Processor board.
- LOCK-0 (Lock). LOCK-0 is a firmware-initiated signal, directly from the MPU, that prevents BUSY-0 from going false so that no other bus masters can gain control of the bus.

The output signals are:

- T4I-O (State T4 and Idle). Used here to signal when the MPU or FPU leaves state T4 or the idle state.
- BRQ-O (Bus Request). BRQ-O is immediately decoded according to the priority of the Motherboard slot into which the Processor board is inserted. BPRN-O goes true if no higher priority BRQ-O signal is asserted at the time the Processor board BRQ-O is asserted.
- CBRQ-O (Common Bus Request). (See input signals.)
- BUSGRT-1 (Bus Grant). Indicates that the Processor board has control of the bus when true.
- BUSY-O. (See input signals.)
- BUSAEN-O (Bus Address Enable). BUSAEN-O triggers the Bus Command Driver to output read or write signals on the System bus. It also enables the System bus address drivers.

The Bus Transfer Logic enables the Processor board to respond to the bus transfer protocol that all potential bus masters on the System bus must obey. (See the description of bus transfer protocol.) The Processor board is just one system device that can control the System bus.

There are three JK flip-flops whose Q or Q-not outputs are the main output signals of this circuitry. (For the purpose of this description, these flip-flops are called the BRQ-O, BUSAEN-O, and BUSY-O flip-flops according to their Q or Q-not outputs.)

Note that three signals, CBRQ-O, BUSGRT-O, and BUSY-O, are both inputs and outputs of this circuitry. This circumstance is made necessary by the bus transfer protocol.

End-of-Data-Transfer Strap

Strap W455 in the Microprocessor Timing Generator selects one of two signals that indicate the end of a current MPU data transfer. W455 should be strapped to pins 2 and 3 only in a multi-processor board system where the MPU is not the microprocessor supplying the bus clock signal. (Note that strap W456 (BCLK-O) should be open if the MPU is not supplying the bus clock signal.)

When the MPU begins a memory, I/O, or interrupt cycle, the Address Decoding circuitry drives OBADR-0 true if the cycle is transferring data on the Processor board internal bus. Since the MPU is not accessing the System bus, BRQ-0 is driven or remains false.

However, if OBADR-0 is driven high, BRQ-0 goes true and the bus transfer logic is enabled. At this point the MPU is in one of two states; it either controls or does not control the System bus. If it controls the bus, then, according to the protocol, the MPU has already caused BUSY-0 to go true and it proceeds with its data transfer immediately, since it is currently bus master.

However, if the MPU does not currently control the bus, the following sequence of events occurs:

1. The BRQ-0 flip-flop toggles high and drives BRQ-0 and CBRQ-0 both true onto the bus.

2. When BUSY-0 goes false (bus is available) and BPRN-0 goes true (bus is granted to Processor board), the Bus Transfer Logic makes BUSGRT-0 true and BUSY-0 active low (bus is busy and unavailable).

3. At the same time, the BUSAEN-0 flip-flop is preset by BUSY-0. This enables the bus controller to begin driving control signals onto the System bus.

4. BUSY-0 remains true until the MPU is finished transferring data. Then BUSY-0 goes false if BPRN-0 is false. There are two ways to cause BPRN-0 to go false:

    - o A higher priority bus master requests bus control by driving its BRQ-0 signal true. If this happens, the bus priority logic on the Motherboard causes the BPRN-0 of the Processor board to go false.

    - o A lower priority bus master requests bus control by making its CBRQ-0 go true. This causes the BRQ-0 flip-flop to toggle, which makes BRQ-0 go false. Thus, the Processor board no longer requests the System bus.

If no other bus master requests the bus, the Processor board maintains control. This eliminates the time delay that would be caused by having to get control of the bus for every data transfer.

ADDRESS DECODING (SCHEMATIC A2-3)

Address Decoding decodes an MPU address, consisting of A12-1 through A19-1, to determine which Processor board circuitry (or the System bus) is currently being addressed.

The input signals are:

- MDEN-1 (Master Data Enable). Disables all MPU data drivers when true. MDEN-1 is used for testing.
- T3W12-1 (State T3 Wait 4). When true, indicates that the MPU is in state T3, T4, or Tw (WAIT). T3W12-1 enables on-board data transceivers.
- T4I-0 (State T4 Idle). True during the MPU T4 and TI (IDLE) states, turning off the data transceivers during T4 in on-board memory and I/O reads.
- OBINTA-0 (On-Board Interrupt Acknowledge). When true, indicates that the Priority Interrupt Controller is generating an interrupt vector address.
- LS0-0, LS1-0, LS2-0 (Latched Status 0 through 2). When taken together, these signals indicate which of eight states the MPU is in -- interrupt acknowledge, I/O read, I/O write, halt, instruction fetch, memory read, memory write, or no bus cycle.
- A12-1 through A7-1 and A9-1 through A9-1. (Processor Address Bus Bits). These address the different areas of Processor board circuitry.

The output signals are:

- OBDEN-O (On-Board Data Enable). Enables the data transceivers so that Processor board data is put on the Processor data bus lines, DO-1 through D15-1.

- OBADR-O (On-Board Address). When true, OBADR-O disables the Bus Command Driver circuitry. No read, write, or interrupt signals are output to the System Bus, allowing the MPU to access only circuitry on the Processor board.

- OBRAM-O (On-Board RAM). In conjunction with LBHE-O, AO-1, and INIT-O, OBRAM-O enables the Non-volatile RAM block of the Processor board.

- OBROM-O (On-Board ROM). When true, OBROM-O enables the 32K of ROM on the Processor board.

- OBIOX&X-O. (On-Board (two byte) Input/Output Locations). These eight signals basically act, sometimes in conjunction with other signals, as enabling signals for two-byte I/O space locations on the Processor board. For example, OBIO0&2 and OBIO4&6 together with CMD are chip enables for the PCI in the RS-232 communications interface circuitry.

The Address Decoding circuitry consists of a number of logic gates that decode the ranges of addresses shown in Table 12-5, Processor Board Address Enabling Signals. A 3 to 8 line decoder also forms part of the circuitry, and outputs the eight OBIOX&X-O signals.

Note that two sets of two multiple-input AND gates are marked off by dashed lines on the schematic. Address Decoding is designed to work with one or the other set installed, but not both. Both of these sets of gates feed a three-input NAND gate which directly produces OBRAM-O. Ordinarily, the set with the four-input AND gates is present. This set decodes addresses in the range FEOO to FFFF in I/O ADDRESS SPACE. Non-volatile RAM is located in this range. If another MPU is added to the system, the set with five-input AND gates would be present. This set decodes addresses in the range OOOOO to OO1FF in MEMORY ADDRESS SPACE. This gives a secondary MPU local RAM.

Table 12-5

PROCESSOR BOARD ADDRESS ENABLING SIGNALS (a)

ADDRESSED CIRCUIT	SIGNAL	SPACE	ADDRESS
32K ROM	OBROM-O	MEMORY	F8OOO-FFFFF
Non-volatile RAM	OBRAM-O	I/O	FEOO & FFFF
PCI	OBIOO&2-O	I/O	OOEO & OOE2
PCI	OBIO4&6-O	I/O	OOE4 & OOE6
PIC	OBIO8&A-O	I/O	OOE8 & OOEA
FPC MPU	OBIOC&E-O	I/O	OOEC & OOEE
PIT	OBIO1&3-O	I/O	OOE1 & OOE3
PIT	OBIO5&7-O	I/O	OOE5 & OOE7
RS-232 Interrupt enable and Status A	OBIO9&B-O	I/O	OOE9 & OOEB
Bus Timeout Reset and Status B	OBIOD&F-O	I/O	OOED & OOEF
System Bus	OBADR(-1)	MEMORY	OOOOO-F7FFF
System Bus	OBADR-O	I/O	OOOO-OODF
System Bus	OBADR-O	I/O	OOFO-FDFF
System Bus	OBADR-O	I/O	8OOO-FDFF

(a) With all straps set to their normal positions.

The MPU enters bus cycle T1 during which it outputs an address on lines AD0-1 through AD19-1 and the MPU status signals S0-0, S1-0, and S2-02. The Address Drivers latch these signals on the trailing edge of ALE-1. If the address falls in one of the ranges of Processor board circuitry, the Address Decoding circuitry generates one and only one of the output signals in Table 12-5 -- Processor Board Address Enabling Signals -- in addition to OBADR-0 true. Once OBADR-0 goes true, it generates OBDEN-0 in conjunction with the timing signals T3W12-1 and T4I-0. T3W12-1 essentially enables OBDEN-0 and T4I-0 disables it. Note that a Processor board interrupt by means of OBINTA-0 can also cause OBDEN-0 to go true.

All of the OBIOX&X-0 signals are generated by the 3 to 8 decoder. Since all of the addresses for these signals fall in the range 00E0 to 00EF, the decoding gates examine bits A12-1 through A7-1 which carry the binary number 1110, which is hexadecimal E. The 3 to. 8 decoder has bits A0-1, A2-1, and A3-1 as inputs, but NOT bit A1-1. For example, the decoder cannot distinguish between 00E0 and 00E2 and outputs the same OBIOX&X-0 signal for either address.

If MDEN-1 goes true, it disables both OBDEN-0 and OBADR-0. This implies that the MPU cannot access either Processor board or System bus locations.

BUS TIMEOUT DETECTOR (SCHEMATIC A2-1)

The Bus Timeout Detector detects when a System bus slave device fails to respond with an acknowledge signal (ACK1-0) to a command from any master device. It drives ACK1-0 onto the System bus and sets an error status bit (D8), which the MPU can read by polling the status. To alert the MPU that a bus timeout has occurred without the necessity of polling, the Bus Timeout Detector also causes an interrupt to be generated through the Interrupt Controller on level 7.

The input signals are:

- OUT0-1 (Output 0). OUT0-1 is the output of a 16-bit down counter in the Programmable Timer and Baud Rate Generator circuitry.
- OBIOD&F-0 (On-Board I/O D and F). Along with an output from the MPU Timing Generator, OBIOD&F-0 resets the bus timeout error flip-flop.
- INTA-0 (Interrupt Acknowledge). The MPU interrupt acknowledge.
- AIOWC-0 (Advanced I/O Write Command).
- IORC-0 (I/O Read Command).
- MRDC-0 (Memory Read Command).
- AMWC-0 (Advanced Memory Write Command).
- BTIEN-1 (Bus Timeout Interrupt Enable). This signal enables the Bus Timeout Interrupt circuit whenever one of the bus command lines goes true during a normal data transfer between any bus master and any slave device.

The output signals are:

- ACK1-0 (Acknowledge 1). ACK1-0 is one of two acknowledge signals in the System bus data transfer protocol.
- D8-1 (Processor Data Bus Bit 8). Used to carry error status information.
- BTINT-0 (Bus Timeout Interrupt).

This circuitry consists of a 12-bit binary counter, a D-type flip-flop, and some logic gates. The QD output of the binary counter drives ACK1-0 through an open-collector driver. The Q output of the error flip-flop drives the Bus Timeout Interrupt signal, then becomes D8-1 after it passes through an output-controlled line driver.

During a normal data transfer between any bus master and any slave device, one of the bus command lines goes true, enabling the 12-bit counter to begin counting. Since the programmable clock input signal, OUTO-1, has a period of 20ms, the bus command would have to be held low continuously for 160ms before QD of the counter would go true. If this bus timeout condition does occur, QD sets the error flip-flop and drives ACK1-0 by means of an open-collector driver. This completes the handshake sequence for the non-responding or non-existent slave device.

If the bus commands are all less than 160ms, the counter is always reset before it can drive QD high and no bus timeout occurs.

Note that this circuit detects timeouts even when some other bus master is controlling the bus. In a multiple-processor board system, it may be desirable to disable the timeout function on one or more of the Processor boards by cutting cut-strap W561 and the BTINT-0 strap at W470.

STATUS INPUT (SCHEMATIC A2-5)

Status Input allows the MPU to read Processor board status signals -- BUSGRT-1, STEST-0, TIMR1-1, and bus timeout error.

The input signals are:

- BUSGRT-1 (Bus Grant). Similar to BUSAEN-0 but timed differently.
- STEST-0 (Self Test). Goes true when the self-test switch on the 4170 is pressed.
- LS0-0 (Latched Status 0). One of three status signals (S0-0, S1-0, and S0-2) output by the MPU to show the current state.
- OBIOD&F-0 (On-Board I/O D&F). OBIOX&X-0 signals inform the MPU what areas of Processor board circuitry are addressed.
- CMD-1 (Command). Shows whether a local read (RD-0) or write (AWT-0) operation takes place.

The output signals are:

- STATEN-1 (Status Enable). Enables a tri-state input to a buffer in the Bus Timeout Detector.

The Status Input circuitry drives STATEN-1 true if CMD-1 is true, LSO-0 is false, and OBIOD&F-0 is true. If STATEN-1 goes true, this enables two buffers, which output BUSGRT-1 as D11-1, STEST-0 as D10-1, Output 1 of the Programmable Interval Timer as D9-1, and bus timeout error as D8-1 on the Processor data bus, DO-1 through D15-1.

BUS CLOCK GENERATOR (SCHEMATIC A2-1)

The Bus Clock Generator generates the 4.9152 MHz, 50% duty cycle bus clock signal, BCLK-0, for the Processor board and System bus.

The input signals are:

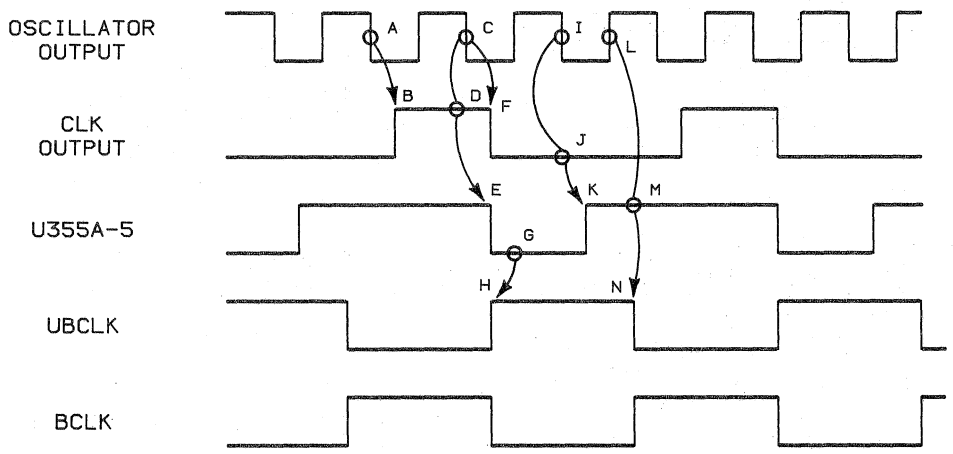
- OSC-0 (Oscillator). Output of the Clock Generator in the Microprocessor Control circuitry, which is a 14.7456 MHz square wave.
- CLK-1 (Clock). Output of the Clock Generator in the Microprocessor Control circuitry, which is a 4.9152 MHz square wave with a 33% duty cycle.

The output signals are:

- BCLK-0 (Bus Clock). A 4.9152 MHz square wave with a 50% duty cycle. It synchronizes bus mastership transfers and provides a stable clock for various functions on other circuit boards.
- UBCLK-1 (Unbuffered Bus Clock). Same as BCLK-0, but is used only on the Processor board to drive the clock inputs of the FPC MPU and the PCI.

The Bus Clock Generator consists of two JK flip-flops and two logic gates. A four-input NAND gate which is a 50 ohm line driver used to give extra drive to BCLK-0 for the System bus. The NOR gate is simply an inverter for the clock input to one of the JK flip-flops. (Cut-strap W456, if open, enables only one Processor board to generate BCLK-0 in a multiple Processor board system.)

This circuitry transforms a 50% duty cycle, 14.7456 MHz signal (OSC-0 output) and a 33% duty cycle, 4.9152 MHz signal (CLK-1 output) into two 50% duty cycle, 4.9152 MHz signals, UBCLK-1 and BCLK-0. Figure 12-14, Bus Clock Generator Timing, shows the timing relationships among the signals in the circuitry.



NOTES:

A,B The trailing edge of OSC causes CLK to go high.

C,D,E The trailing edge of OSC and a high on CLK cause Pin 5 of the U355A JK flip-flop to go low.

C,F The trailing edge of OSC causes CLK to go low.

G,H A low level on U355A-5 causes UBCLK to go high.

I,J,K The trailing edge of OSC and a low level on CLK causes U355A-5 to go high.

L,M,N The leading edge of OSC and a high level on U355A-5 causes UBCLK to go low.

3820-17

Figure 12-14. Bus Clock Generator Timing.

MPU TIMING GENERATOR (SCHEMATIC A2-1)

The MPU Timing Generator provides MPU timing information for other circuitry on the Processor board.

The input signals are:

- S0-0,S1-0, and S2-0 (MPU Status Bits 0 through 2). These signals show the eight possible states of the MPU -- interrupt acknowledge, I/O read, I/O write, halt, instruction fetch, memory read, memory write, and no bus cycle.
- ALE-1 (Address Latch Enable). Clears Timing Generator D-type flip-flop.
- LS1-0 (Latched Status Bit 1). Refer to Address Driver section for detailed information.

The output signals are:

- T3W12-1 (State T3, TW, & T4). True when the MPU is in a T3, TW (WAIT), or T4 state. T3W12-1 enables on-board data transceivers.
- T4I-0 (State T4, TI (Idle)). True during MPU T4 and TI (IDLE) states. It turns off data transceivers during T4 in Processor board memory and I/O reads.
- AWT-0 (Advanced Write). Similar to the System Bus AIOWC-0 signal, but used for Processor board circuitry only.

The MPU performs memory and I/O transfers by going through bus cycles. Each bus cycle consists of at least four clock states, which are called T1, T2, T3, and T4. Sometimes, TW, or WAIT states, are inserted between T3 and T4. Also, TI, or IDLE states, are sometimes inserted between bus cycles. The MPU Timing Generator makes this information available to other circuitry on the Processor board.

The MPU Timing Generator comprises four logic gates and one quadruple D-type flip-flop. The interconnections among the four D-type flip-flops are shown in Figure 12-15.

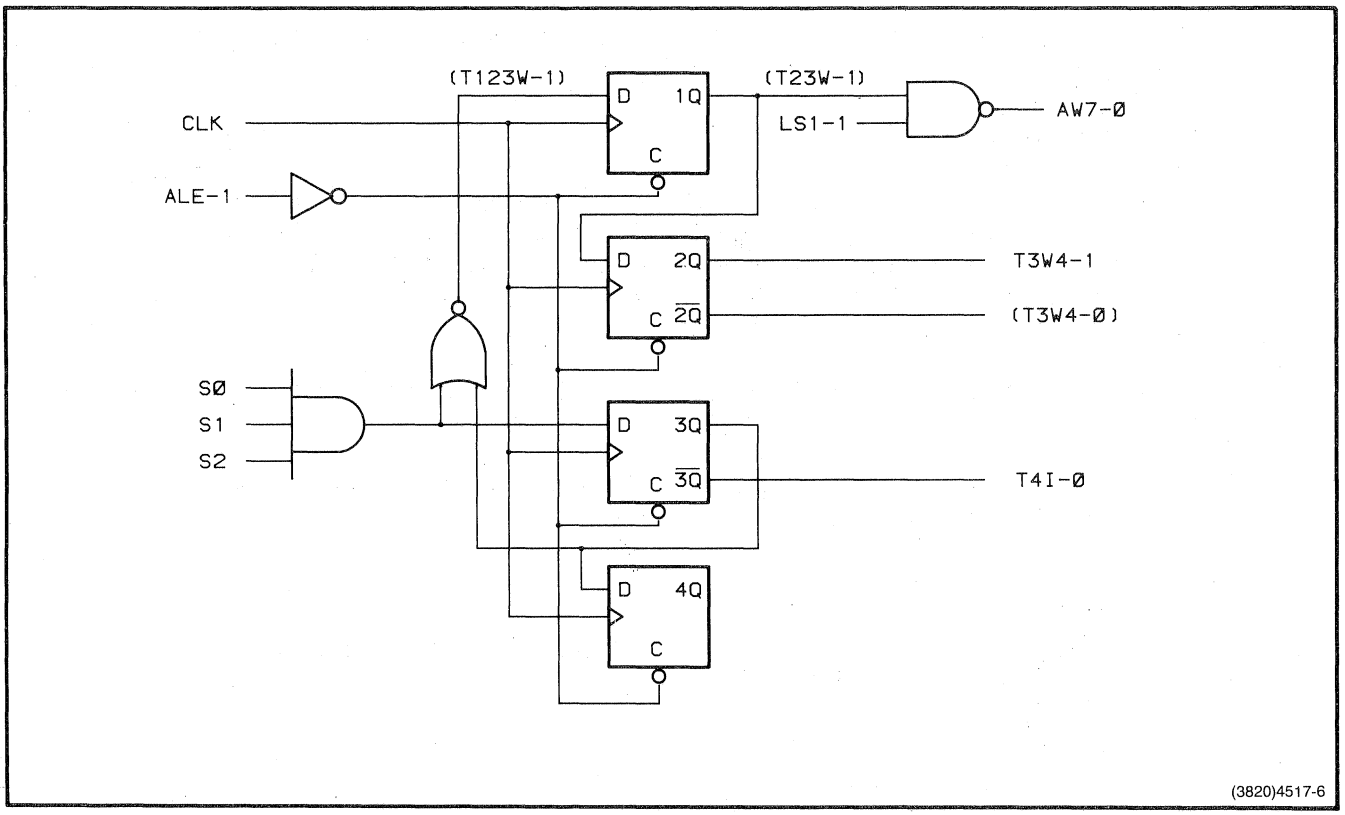


Figure 12-15. MPU Timing Generator Circuitry.

The circuit is basically a shift register. At the end of a bus cycle, the D input of flip-flop #1 is a logical one. When the next bus cycle begins, ALE-1 resets all four flip-flops during T1. Next, T2 is entered and the Q output of flip-flop #1 becomes logical one. This condition drives AWT-0 true. During T3, the Q output of flip-flop #2, which is T3W12-1, goes true, since it is connected directly to the Q output of flip-flop #1. After T3, it enters the TW state unless or until MPU status bits S0-0, S1-0, and S2-0 all become false. Then, in state T4, the Q-not output of flip-flop #3, which is T4I-0, goes true. The MPU Timing Generator stays in state TI until the MPU issues another ALE-1 during T1.

Table 12-6

MPU STATE AND MPU TIMING GENERATOR SIGNALS

SIGNAL NAME	MPU CLOCK STATE					
	T1	T2	T3	TW(a)	T4	TI(a)
(T1,2,3,W)	True	True	True	True	False	False
(T2,3,W)	False	True	True	True	False	False
T3W4-1	False	False	True	True	True	False
T4I-0	False	False	False	False	True	True

(a) Note that there may be multiple TW and TI states.

PROGRAMMABLE TIMER AND BAUD RATE GENERATOR (SCHEMATIC A2-5)

The Programmable Timer and Baud Rate Generator has three programmable counters contained in the Programmable Interval Timer (PIT) that generate timing signals to produce:

- Transmit baud rate in the RS-232 Communications Interface logic.
- TIMR1-1 that can generate an interrupt signal to the MPU.
- OUT0-1 -- the primary firmware interrupting timer and timing source for the bus timeout detector.

The input signals are:

- A1-1 and A2-1 (Processor Address Bits 1, 2). They address the three 16-bit down counters in the PIT and an internal address control word register.
- AWT-0 (Advanced Write). If true, enables the PIT to output counter data.
- BBCLK-1 (Buffered Bus Clock). A 4.9152 MHz clock signal that clocks a 12-bit counter.
- OBI01&3-0 and OBI05&7-0 (On-Board I/O). Either signal, when true, enables the PIT for read or write operation.
- RD-0 (Read). Processor bus read signal. The PIT inputs data from DO-1 through D7-1 when RD-0 is true.
- RST-1 (Reset). Local reset signal derived from System bus INIT-0. When true, clears the 12-bit counter.

The output signals for this block are:

- DO-1 through D7-1 (Processor Data Bus Bits 0 through 7). Carry three kinds of information:
  - Values to load into the PIT counters.
  - Values read from the PIT counters.
  - Data that programs the various modes of the PIT.
- OUT0-1 (Output 0). Output of counter 0 in the PIT.
- TIMR1-1 (Timer 1). Output of counter 1 in the PIT.
- TXC-0 (Output 2). Output of counter 2 in the PIT, inverted.

This circuitry consists primarily of the Programmable Interval Timer (PIT). The PIT includes three internal 16-bit counters (numbered 0, 1, and 2), a data buffer, and read/write circuitry (Figure 12-16).

This circuitry also has one AND gate and one 12-bit binary counter. The AND gate functions as an OR gate using OBI05&7-0 and OBI01&3-0 to operate the chip select input of the PIT. The counter produces two signals by dividing the buffered bus clock signal (BBCLK-1) by two and eight. BBCLK-1 divided by two is input to the number 2 counter of the PIT and BBCLK-1 divided by eight is input to the number 0 and 1 counters of the PIT.

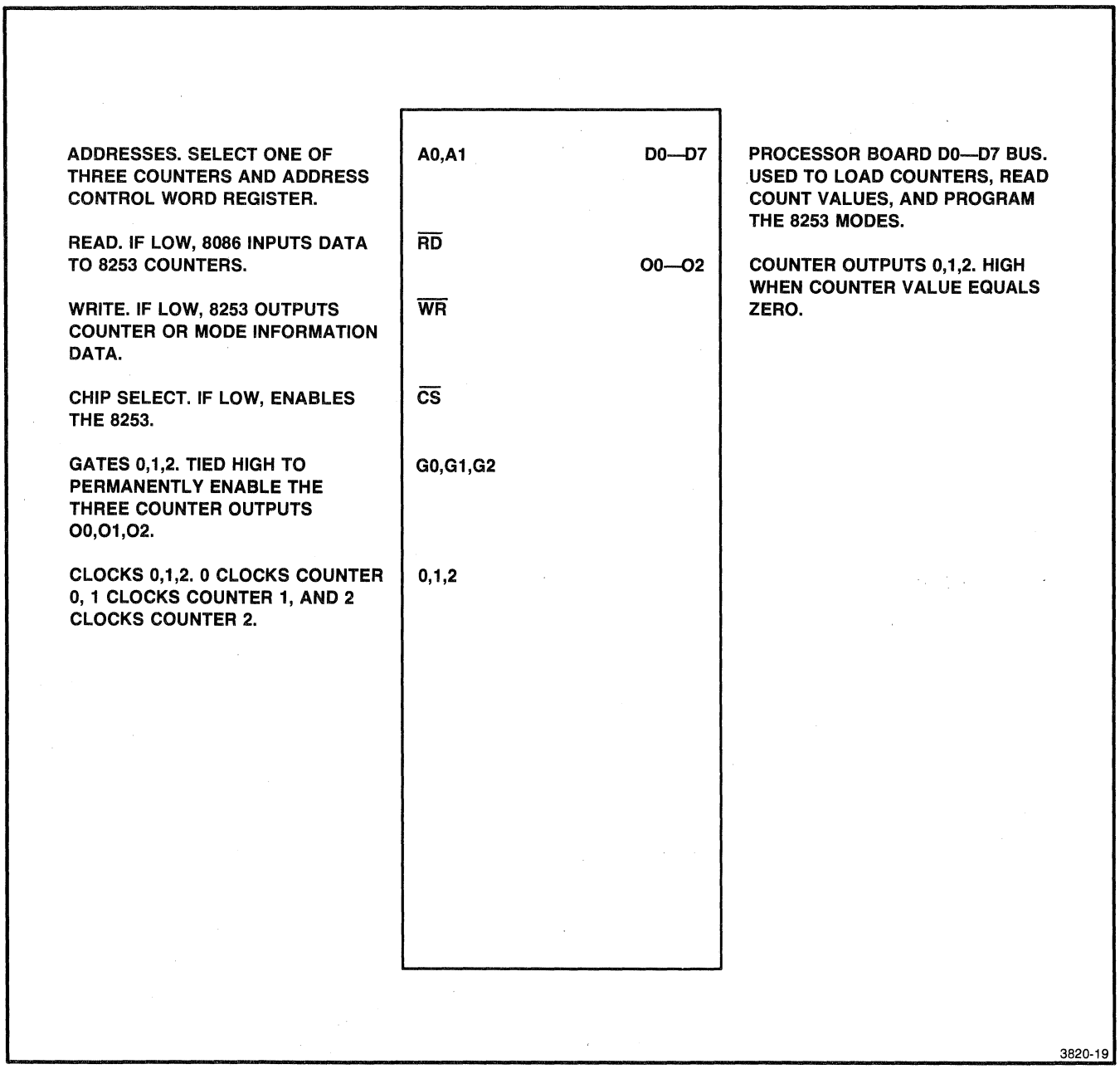


Figure 12-16. Programmable Interval Timer Pin Descriptions.

Programmable Interval Timer

There are three main blocks of circuitry in the PIT:

- o Three 16-bit down counters
- o An 8-bit data bus buffer
- o Read/write logic

The three counters operate independently and each has its own output (O0, O1, O2). How the counters operate is controlled completely by commands from the system (or possibly other) firmware. Each counter is loaded with an initial value supplied by firmware. Firmware commands can also read the counter value of a counter at any time during its down count.

The data bus buffer is connected to D8-1 through D15-1, part of the Processor data bus. Firmware commands, initial counter values, and read-out counter values pass through this buffer.

The read/write logic is enabled by the chip select input to the PIT. The local Processor signals, RD-0 and AWT-0, control the direction of data flow in the data bus buffer. Part of the Processor address bus, A1-1 and A2-1, select which of the three counters that commands and counter values are directed to.

The PIT has two phases of operation: initialization and normal operation. Initialization occurs shortly after power-up, though initialization commands may be given after this time.

After power-up, the MPU initializes the PIT as follows: either OBIO1&3-0 or OBIO5&7-0 goes true and enables the PIT, and A1-1 and A2-1 select counter 0, 1, or 2. AWT-0 goes true, enabling firmware commands and counter values to program the selected counter via D8-1 through D15-1. After all three counters are programmed, initialization is done.

Note that there is no reset pin on the PIT, and that after the System bus INIT-0 signal goes false and before the MPU initializes the PIT, outputs O1, O2, and O3 are undefined and may be stable high, stable low, or pulsing.

In normal operation, a count value may be read from one of the counters. In this case RD-0 goes true after the PIT and the appropriate counter are selected. The counter value then appears on D8-1 through D15-1.

Also in normal operation, output 2 (02) produces the TXC-0 baud rate signal for the RS-232 Communications Interface, output 1 (01) produces TIMR1 that is used by the MPU Control circuitry in combination with a number of other interrupt-type signals, and output 0 (00) produces OUT0 that serves as both the primary interrupting firmware timer and as the clock source for the bus timeout detector.

RS-232 STATE CHANGE DETECTOR (SCHEMATIC A2-5)

The RS-232 State Change Detector detects state changes on the incoming RS-232 status lines (DSR-1, DCD-1, SDCD-1, CTS-1, and RING-1) and generates an interrupt (TIMERINT-0) when any of these signals change state.

The input signals are:

- DSR-1 (Data Set Ready). Informs the RS-232 Communications Interface that the local modem is ready to operate.
- DCD-1 (Data Carrier Detector). Informs the RS-232 Communications Interface that the carrier wave is being received by the modem.
- SDCD-1 (Secondary Data Carrier Detector). Informs the RS-232 Communications Interface that the secondary carrier wave is being received by the modem.
- CTS-1 (Clear To Send). Informs the RS-232 Communications Interface that the modem is ready to transmit. This is a response to RTS (Request To Send) from the RS-232 Communications Interface circuitry.
- RING-1 (Ring Indicator). Informs the RS-232 Communications Interface that the local modem is receiving a ringing signal from a remote modem.
- RST-0 (Reset). Clears the last state latch.

The output signals are:

- (Y-0) (State Change Detected Signal). Produces TIMERINT-0 in conjunction with other signals in the RS-232 Communications Interface.

The circuitry consists of a 12-input line receiver, a 6-input D-type flip-flop, a bus driver, a 6-bit comparator, and part of another 12-input line receiver. (The five capacitors connected to the line receivers prevent voltage spikes (glitches) on the RS-232 status lines from being transmitted to the RS-232 State Change Detector circuitry.)

The MPU interacts with the RS-232 State Change Detector in two ways. It first receives an interrupt and then determines what caused the interrupt.

The RS-232 status signals (DSR-1, DCD-1, SDCD-1, CTS-1, and RING-1) are input to the last state latch and to the comparator. (Note also that the OO output of the Programmable Interval Timer in the Programmable Timer and Baud Rate Generator block makes up a sixth input to the comparator and latch.) The six outputs of the latch are routed to the bus driver and to the other half of the magnitude comparator. The comparator compares the inputs and outputs of the latch, and if any of the RS-232 status lines (or the OO output of the PIT) changes state, the comparator then drives Y-0 true which activates a NAND gate in the RS-232 Communications Interface. This causes TIMERINT-0 to go true. After TIMERINT-0 is processed by the Interrupt Controller, the MPU receives an interrupt.

At this point the MPU needs to determine which RS-232 status line (or OO from the PIT) caused the interrupt. To accomplish this the MPU reads either the current status or the new status. To read the current status which exists at the outputs of the latch, the MPU addresses the RS-232 State Change Detector, which causes OBIO9&B-0 to go true. Also, A1-1 must be true. RD-0 now strobes, causing the bus driver to drive the status line data onto the Processor bus. To read the new status, OBIO9&B-0 goes true again, but this time A1-1 goes false. Now, the latch is clocked, and the bus driver drives the inputs (new status data) of the latch onto the Processor data bus. At this point, firmware determines which status line caused the TIMERINT-0 interrupt.

RS-232 COMMUNICATIONS INTERFACE (SCHEMATIC A2-5)

The RS-232 Communications Interface transmits and receives RS-232 characters and control (handshake) signals. It contains the receive Baud Rate Generator.

The input signals are:

- RDATA-0 (Receive Data). Serial data from the host computer.
- RCLK-1 (Receive Clock). External clock signal generated by the local modem or other source. Can be used to clock data into the Programmable Communications Interface (PCI).
- TCLK-1 (Transmit Clock). External clock signal generated by the local modem or other source. Can be used to clock data out of the PCI.
- DO-1 through D15-1 (Processor Data Bus Bits 0 through 15).
- A1-1 and A2-1 (Processor Address Bus Bits 1 and 2). Carry addresses that select internal registers in the PCI.
- OBI09&B-0 (On-Board I/O 9 and B). True when data appears on the Processor bus for the interrupt enable latch (with inputs D8-1 through D15-1).
- UBCLK-1 (Unbuffered Bus Clock). Clocks the Internal Baud Rate Generator in the PCI.
- LS1-0 (Latched Status Bit 1). When LS1-0 is true, the MPU is in an interrupt acknowledge, I/O read, memory read, or instruction fetch bus transaction.
- OBI00&2-0 and OBI04&6-0 (On-Board I/O Locations 0 & 2, and 4 & 6). Either signal going true enables the PCI if CMD-1 is true.
- CMD-1 (Command). When true, it indicates that the MPU is doing a Processor board read or write.

The output signals are:

- SRTSA-1 or SRTSC-1 (Secondary Request To Send A or C). Half-duplex RS-232 handshaking signal. Can be strapped to A or C depending on the type of modem used.
- DTR-1 (Data Terminal Ready). Informs the modem that the 4170 is operational.
- RTS-1 (Request To Send). Informs the modem that the 4170 is ready to transmit data.
- TDATA-0 (Transmit Data). Data is transmitted serially on this line to the modem or directly to the host computer.
- COMINT-0 (Communications Interrupt). COMINT-0 goes true when a character is received by the PCI from the host computer.
- TIMERINT-0 (Timer Interrupt). When TIMERINT-0 goes true, any one of the following interrupts has occurred: PIT timer #1 (output 01), PCI TXEMT-0, PCI TXRDY-0, RS-232 status change, or a PIT OUT0-1 (via RS-232 State Change Detector circuitry).

This circuitry consists of a line driver, a line receiver, a number of logic gates, and the Programmable Communications Interface (PCI). The PCI is the heart of this circuitry (Figure 12-17).

Programmable Communications Interface

This integrated circuit performs the parallel-to-serial data conversion for data sent to the host computer and also the serial-to-parallel conversion for data sent to the 4170. The PCI internal Baud Rate Generator can also be programmed. In order to perform these functions, the PCI has the circuitry shown in Figure 12-18.

Data Bus Buffer. DO-1 through D7-1 are input and output via this 8-bit buffer. Firmware commands, status information, and data are transferred through the buffer.

Modem Control. Two handshaking signals, DTR-1 and RTS-1, are sent directly to the host computer or modem from this block. (TXEMT-O/DSCHG-O) also originates here, and when true, indicates that the transmitter has completed the parallel-to-serial conversion of the last character loaded by the MPU. Note that DSR-O, CTS-O, and DCD-O, which are generated in this circuitry, are permanently tied low.

Control Functions. By responding to A1-1, A2-1, LS1-O, OBI00&2-O, OBI04&6-O, and CMD-1, this circuitry controls when the PCI is written to or read from, and controls the overall PCI internal operation. In addition, there are internal registers whose contents can be manipulated by firmware commands.

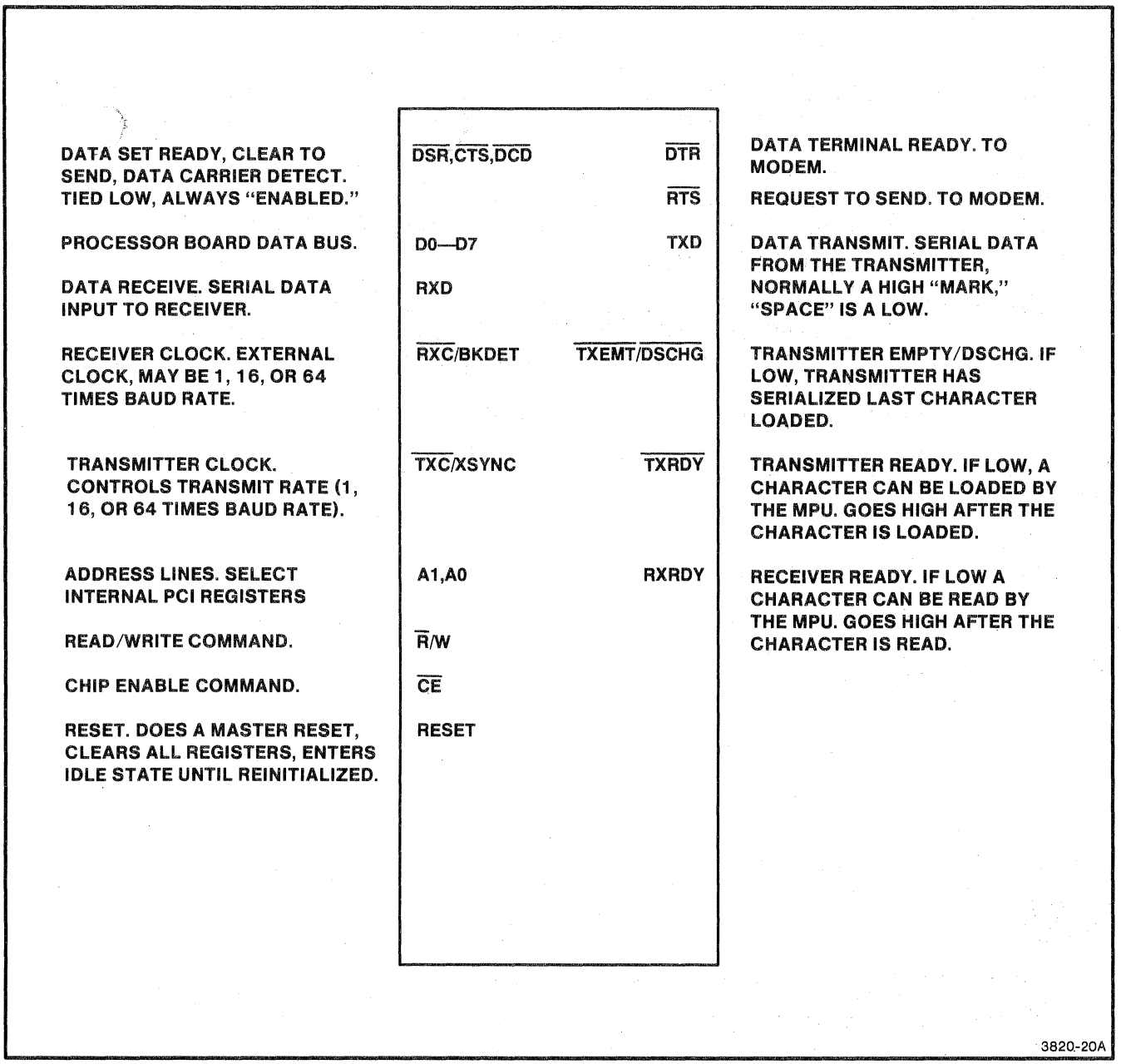


Figure 12-17. Programmable Communications Interface Pin Descriptions.

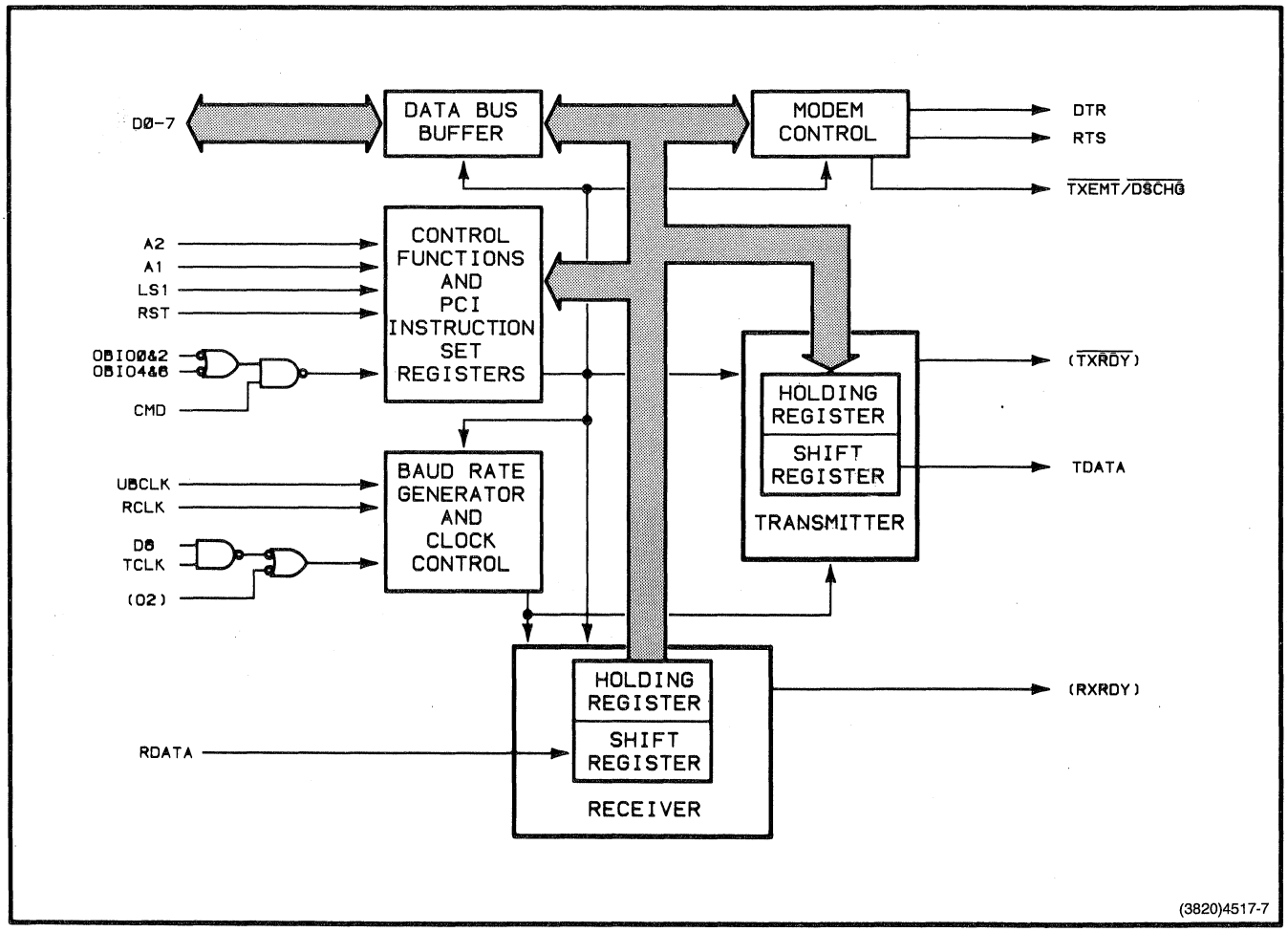


Figure 12-18. Programmable Communications Interface Block Diagram.

Baud Rate Generator and Clock Control. UBCLK provides the source frequency for the internal baud rate generator. Usually, the transmit baud rate is generated by output O2 from the PIT and the receive baud rate is generated from the internal Baud Rate Generator and a software-selectable register value. However, 1X baud rate clocks from the modem may be selected for either the receive rate, transmit rate, or both. For external transmit clocks, TCLK-1 is converted to TTL levels, multiplexed with the PIT O2, and input to (TXC-O). For external receive clocks, RCLK-1 is converted to TTL levels, and input directly to (RXC-O), where the signal is multiplexed with the Internal Baud Rate Generator.

Transmitter. The holding register receives data from the MPU and passes it to the shift register. Start, stop, and parity bits are added to the data according to the current communication parameters. The data is then output serially and becomes TDATA-O.

Receiver. The shift register receives serial data on RDATA. This passes to the holding register and bits or characters are checked according to the current communication parameters. The data is then output to the MPU during a read of the data register.

Before the RS-232 Communications Interface can send and receive data, its operating parameters must be set by a combination of firmware and software commands and values. Communications parameters like synchronous or asynchronous mode, receive baud rate, parity, and number of bits per character, are sent to the PCI shortly after the 4170 is powered up. Once this programming of values is completed, normal data communication begins.

MPU Control

The MPU controls the PCI by activating A1-1, A2-1, LS1-O, OBIOO&2-O, OBIO4&6-O, and CMD-1. The states of A1 and A2 determine which of four registers in the control function circuitry will be selected to be read from or written to. The MPU first selects the PCI by causing OBIOO&2-O or OBIO4& 6-O to go true, and CMD-1 to go true. LS1-O is true for a read operation and false for a write operation.

Data Communication

Serial data on RDATA-0 is input on the RS-232 cable (via P102). The data is then converted to TTL levels by the line receiver and fed to the (RXD-1) input of the PIC. Serial data is output to the line driver and becomes TDATA-0 on the RS-232 cable (via P102). The PIC determines when to output DTR-1 and RTS-1 for modem control. These signals are also output on P102. SRTS-1 is controlled by the MPU directly and is latched, inverted, and level-converted before it reaches P102. (SRTSC-1 is the usual strap setting.It may be strapped to produce SRTSA-1 if the modem requires it.)

Interrupts

The signals (TXEMT-0), (TXRDY-0), and (RXRDY-0) are status signals from the PCI that can generate interrupts to the MPU. All three signals are inverted and ANDed with interrupt enable-type signals from the latch. This allows the MPU to disable the PCI interrupts.

FRONT PANEL CONTROLLER (SCHEMATIC A2-4)

The Front Panel Controller scans the front panel port for key change data and informs the FPC MPU if there is any activity. The Front Panel Controller also handles the LED and bell signals.

The input signals are:

- A1-1 (Processor Address Line 1). Tells the FPC MPU whether data on DO-1 through D7-1 is data or a command.
- AWT-O (Advanced Write). Allows the MPU to write to the FPC MPU.
- RD-O (Read). Allows the MPU to read data from the Peripheral Interface MPU.
- OBIOC&E-O (On-Board I/O C and E). Enables the Peripheral Interface MPU.
- RST-O (Reset). Resets the FPC MPU.
- UBCLK-1 (Unbuffered Bus Clock). 4.9152 MHz square wave input to the FPC MPU.
- KBT1-1 (Keyboard Test 1). Test input to the Peripheral Interface MPU.
- KDO-1 through KD7-1 (Keyboard Data O to 7). Key change information appears on these lines from the front panel.

The output signals are:

- KWR-O (Keyboard Write). Strobes KAO-1 through KA3-1 into the LED and Bell Logic circuitry.
- KSTRB-O (Keyboard Strobe). Latches KAO-1 through KA3-1 into the LED and Bell Logic and Character Decoder circuits.
- KBDINT-O (Keyboard Interrupt). Interrupt to FPC MPU via the Programmable Interrupt Controller.
- KAO-1 through KA3-1 (Keyboard Address O to 3). Carry key matrix column addresses for the Character Decoder and LED on-off data for the LED and Bell logic.

The main component in this circuitry is an 8-bit Front Panel Controller Microcomputer (FPC MPU). Firmware written in the instruction set of the FPC MPU enables it to transmit key change data to the Processor data bus. Also, front panel LED and bell signals are transmitted from the Processor data bus to the front panel.

Other components are: an 8-bit buffer, three inverters, and seven open-collector buffer gates. The 8-bit buffer is always enabled and buffers KD0-1 through KD7-1 for the FPC MPU.
Two of the Inverters are placed between UBCLK-1 and the clock inputs, X1-1 and X2-1, to provide extra drive for the inputs. The seven buffer gates drive KWR-0, KSTRB-0, KBDINT-0, and KA0-1 through KA3-1.

Front Panel Controller Microcomputer

This microcomputer consists of the following blocks of circuitry: control logic, timing, test logic, I/O Ports 1 and 2, status register, data-out buffer, data-in buffer, and one large block that consists of all block that do not have direct inputs from or outputs to the Front Panel Controller (Figure 12-19 and Figure 12-20).

Control Logic. This logic produces internal instructions which accomplish a variety of internal control functions. The signals that determine these internal control functions are A1-1, AWT-0, RD-0, OBIOC&E-0, and RST-0. A1-1 enables the MPU to indicate to the FPC MPU that information on D0-1 through D7-1 is data or a command. When AWT-0 is true, the MPU is enabled to write data to the data in buffer. When D-0 is true, the MPU is enabled to read data from the status buffer or data-out buffer. OBIOC&E-0 enables the FPC MPU. RST-0 resets various internal counters and status flip-flops. (Note that the (SS-0) and (EA-1) inputs: though they are inputs to the control logic, they are disabled since they are tied high and grounded, respectively.)

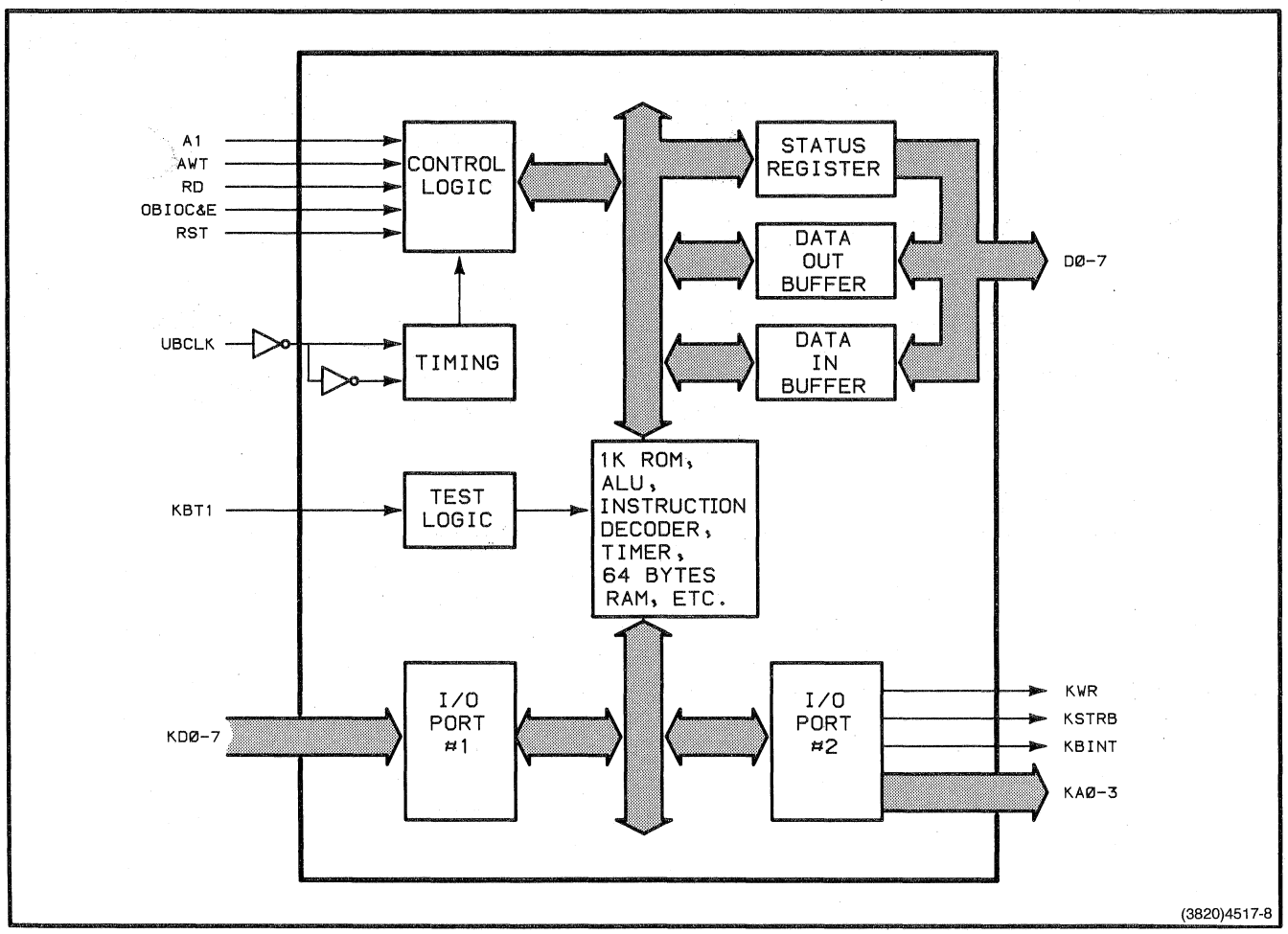


Figure 12-19. FPC MPU Block Diagram.

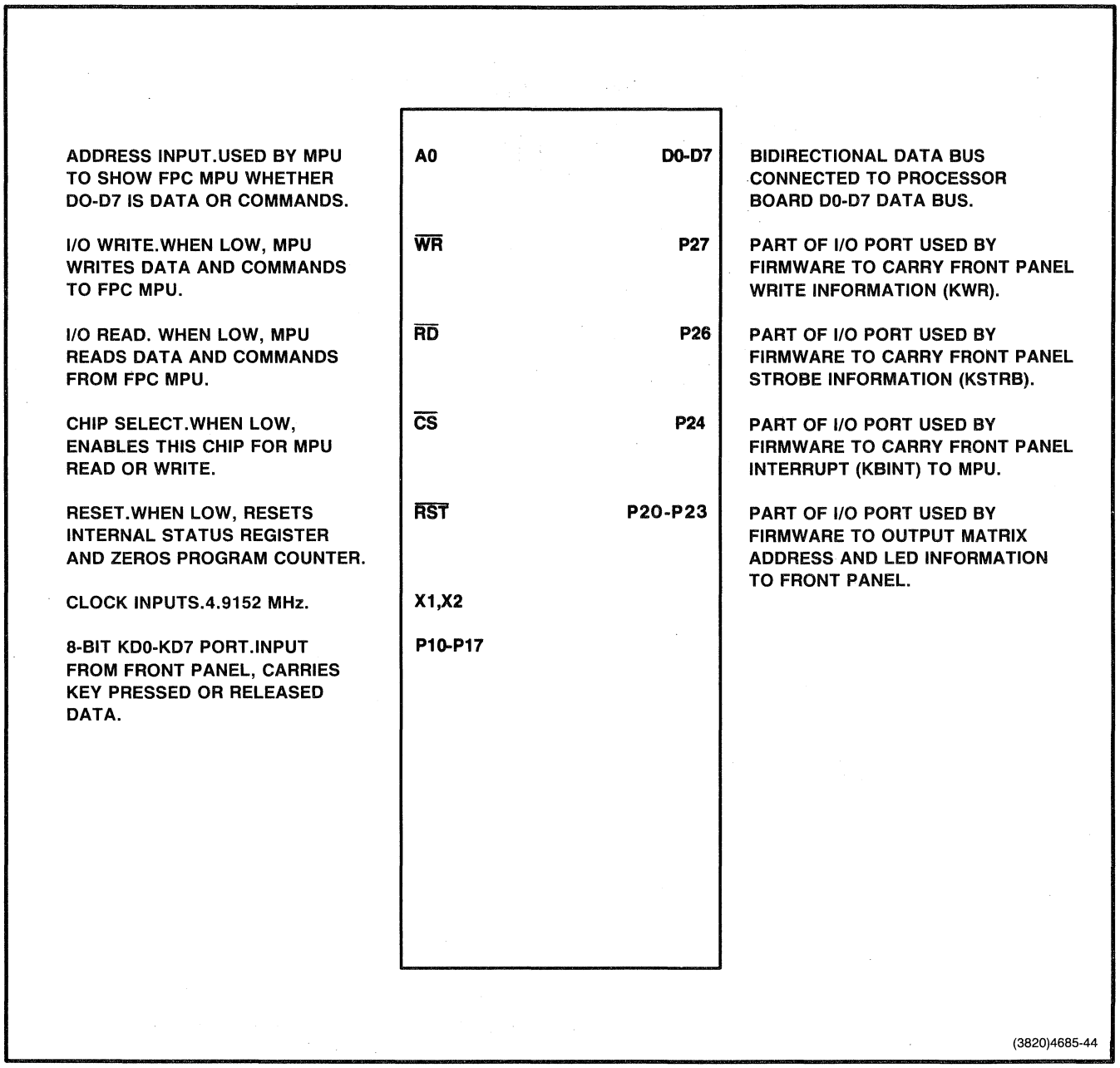


Figure 12-20. FPC MPU Pin Descriptions.

Timing. UBCLK-1 provides the 4.9152 MHz clock for an internal oscillator.

Test Logic. (Test 0) has been disabled by tying it high, but (Test 1) is controlled by KBT1-1. If KBT-1 is true, firmware can read this condition in the FPC MPU during testing.

I/O Ports 1 and 2. Although these ports can function as input and output ports, I/O Port 1 functions only as the input port for KD0-1 through KD7-1, and I/O Port 2 functions only as the output port for KWR-0, KSTRB-0, KBDINT-0, and KA0-1 through KA3-1.

Status Register. This register, which is accessed by the MPU through D0-1 through D7-1, carries status information informing the MPU what kind of data is in the data buffers and whether it should read from or write to the FPC MPU. The MPU reads the status register at I/O address 00EE.

Data-Out Buffer. The MPU reads all data (key codes and thumbwheel position data) from the FPC MPU from this buffer. The data-out buffer I/O address is 00EC.

Data-In Buffer. The MPU writes data and commands to the FPC MPU through this buffer. The data-in buffer I/O address is 00EC.

The FPC MPU has firmware routines masked into its 1K of ROM. These routines cause the FPC MPU to operate independently of the MPU. In addition to carrying out these routines, the FPC MPU responds to a set of commands that the MPU issues to it. These commands cause operations such as ringing the terminal bell, turning LEDs on and off, enabling front panel interrupts, enabling and disabling the timer interrupt, and resetting the FPC MPU.

The FPC MPU passes three kinds of data to the MPU: keycodes, thumbwheel count values, and keyboard identification data. The interactions between the MPU and FPC MPU are somewhat different for each kind of data.

Keycode Data

The FPC MPU has internal firmware that causes KA0-1 through KA3-1 to scan the character decoder. If the MPU has enabled the FPC MPU interrupt and if a key is pressed or released, the Front Panel Controller MPU reads a keycode from KD0-1 through KD7-1. The keycode is placed in the data-out buffer and the FPC MPU issues an interrupt on KBINT-0. The MPU then reads the status register to determine which of the three kinds of data the FPC MPU has for it. At this point, the MPU initiates a routine which reads the 8-bit keycode from the data-out buffer.

If a key is held down, The MPU recognizes whether this is a valid repeating key and the MPU sends the command to enable the timer if a repeat is sensed. The FPC MPU then starts a 500 ms delay. If the MPU, within this delay, has not sent the command to disable the timer, the FPC MPU interrupts the MPU every 100 ms until the MPU sends the disable command (after the key has been released).

LED and Bell

The LED on-off state and bell-sounding are controlled by sending commands to the FPC MPU. In order to turn an LED on or off, the MPU sends to the FPC MPU the address of the LED and the on command or the off command. The FPC MPU controls the timing once it gets the address and on or off command. The bell sounds when its address and the bell command is sent. The bell sounds once for every bell command that is sent to the FPC MPU.

ECC RAM MEMORY

The term ECC RAM Memory describes an ECC RAM Controller with one or two ECC RAM Array boards installed.

The ECC RAM Memory board has the following features:

- Error checking and correction (ECC). The ECC RAM Memory board checks the accuracy of the data stored in memory and corrects any single-bit errors that occur. Double and multiple-bit errors are detected and can be logged by firmware routines.

- Dual port access. The ECC RAM Memory board has the standard 4170 Processor bus access port (Port B), and also includes an additional synchronous port (Port A) which can interface directly to a Processor board having additional port capability.

- Memory density. The ECC RAM Memory board uses either 64K or 256K byte RAMs.

For further information, refer to the ECC RAM Service Manual.

THREE PORT PERIPHERAL INTERFACE (AND OPTION 10 3PPI)

The Three Port Peripheral Interface (3PPI) provides three RS-232-C ports for attaching peripheral devices to the 4170.

One 3PPI is standard in the 4170 and one optional 3PPI (Option 10) can be added for a total of six peripheral ports.

The 3PPI consists of:

- A 3PPI circuit board

- Three standard RS-232-C connectors (mounted on the rear panel)

- Cables from the 3PPI board to the RS-232-C connectors

For further information, refer to the Three Port Peripheral Interface Service Manual.

OPTION 44 DISK CONTROLLER BOARD

The first part of this discussion shows how the disk drive units work as a part of the 4170 Local Graphics Processing Unit for storing and retrieving data on a flexible disk. The next part of the discussion explains the disk formatting and disk data organization scheme. An overview of the hardware used for typical read and write operations follows. The section concludes with a description of the major logic functions of the Disk Controller board.

Signal names used in this theory section are identical to those used on the schematics. These names are abbreviations of the signal function.

SYSTEM CONFIGURATION

The Disk Subsystem consists of the disk drive units and the Disk Controller board (installed in the 4170 card-cage). The major logic functions contained on the Disk Controller board are the following:

- o Disk Controller (FDC)
- o Clock Generation
- o Write Control
- o Read Recovery
- o Disk Drive Control
- o Terminal Interface
- o ROMs

Figure 12-21 shows how the Disk Subsystem and the 4170 interact. The diagram shows the general data flow from the RAM through the Disk Controller board to the disk drive unit. A DMA control block is shown, which allows the Disk Controller board to act as a Processor bus master, accessing system RAMs directly. The heart of the Disk Controller board is a Flexible Disk Controller IC (FDC).

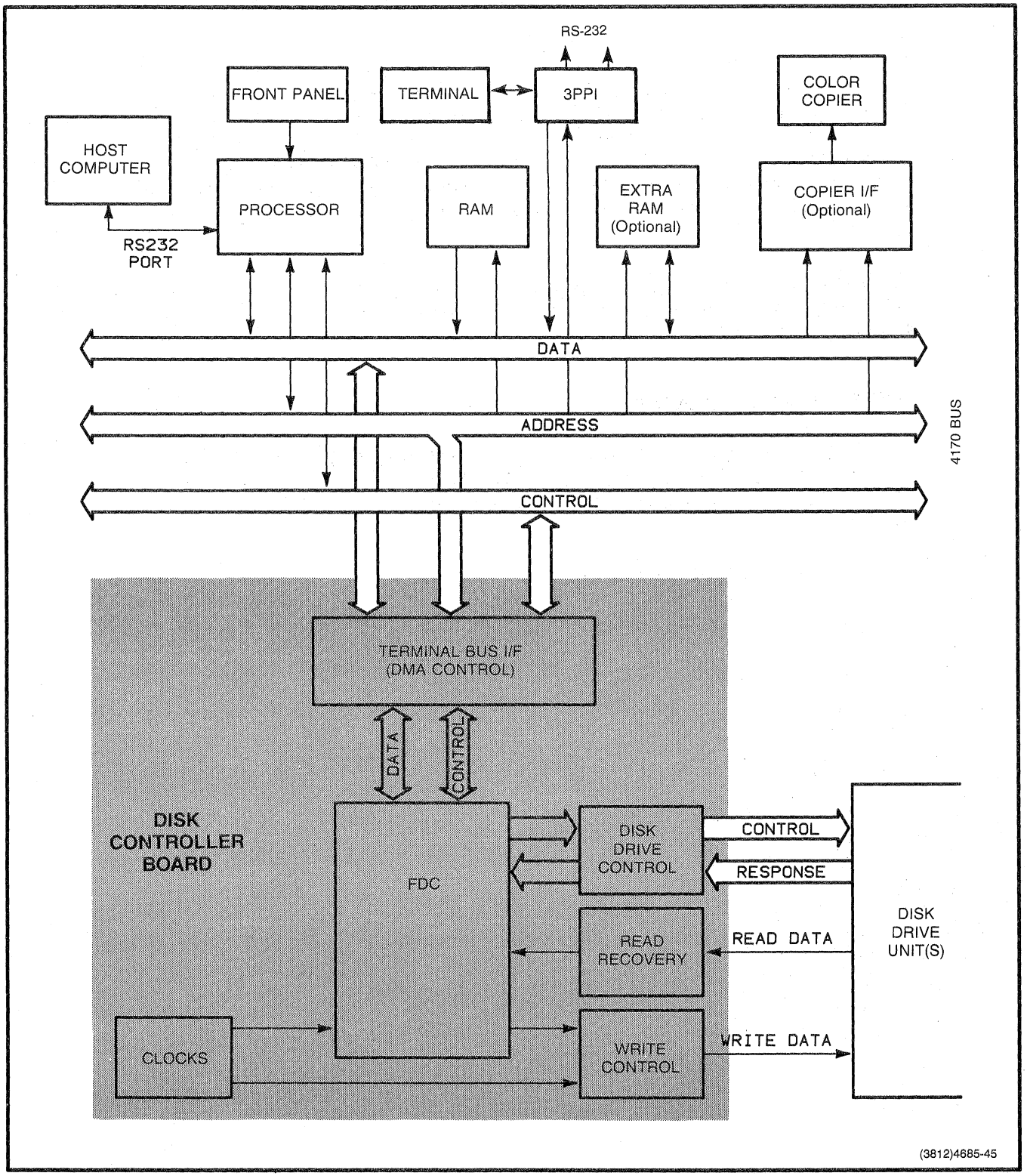


Figure 12-21. System Configuration Block Diagram.

BACKGROUND INFORMATION

Some background information (such as terminology and basic concepts) is required in order to understand the actual theory of operation. This discussion focuses on the type of media used by disk drive units and examines disk data organization and formatting.

Media

The flexible disk media, also called a floppy disk or diskette, contains a thin, flexible plastic disk coated with magnetic oxide. This disk is permanently enclosed and protected by a square, hard paper jacket. The magnetic disk is visible through holes in the jacket provided for indexing and read/write access.

Media Formatting

The diskette is divided into tracks and sectors (Figure 12-22). As the diskette spins under the read/write (R/W) head, the head traces a closed circular path around the disk. This path is where the data is written and is called a track. As the R/W head steps toward or away from the center of the disk, it traces different tracks. There are 40 concentric tracks on one side of a diskette. The lowest numbered track, 00, is on the outside of the disk; the highest numbered track, 39, is closest to the center. An index hole that is read by a photocell (the Index Detector) marks the beginning of each track. (1)

(1) The FDC also uses the index hole to tell when an entire track has been scanned when seeking a particular data record. If the data record is not found after a complete revolution of the disk, the FDC aborts the search, notes an error, and goes on to the next command. This prevents the FDC from being trapped in an endless search.

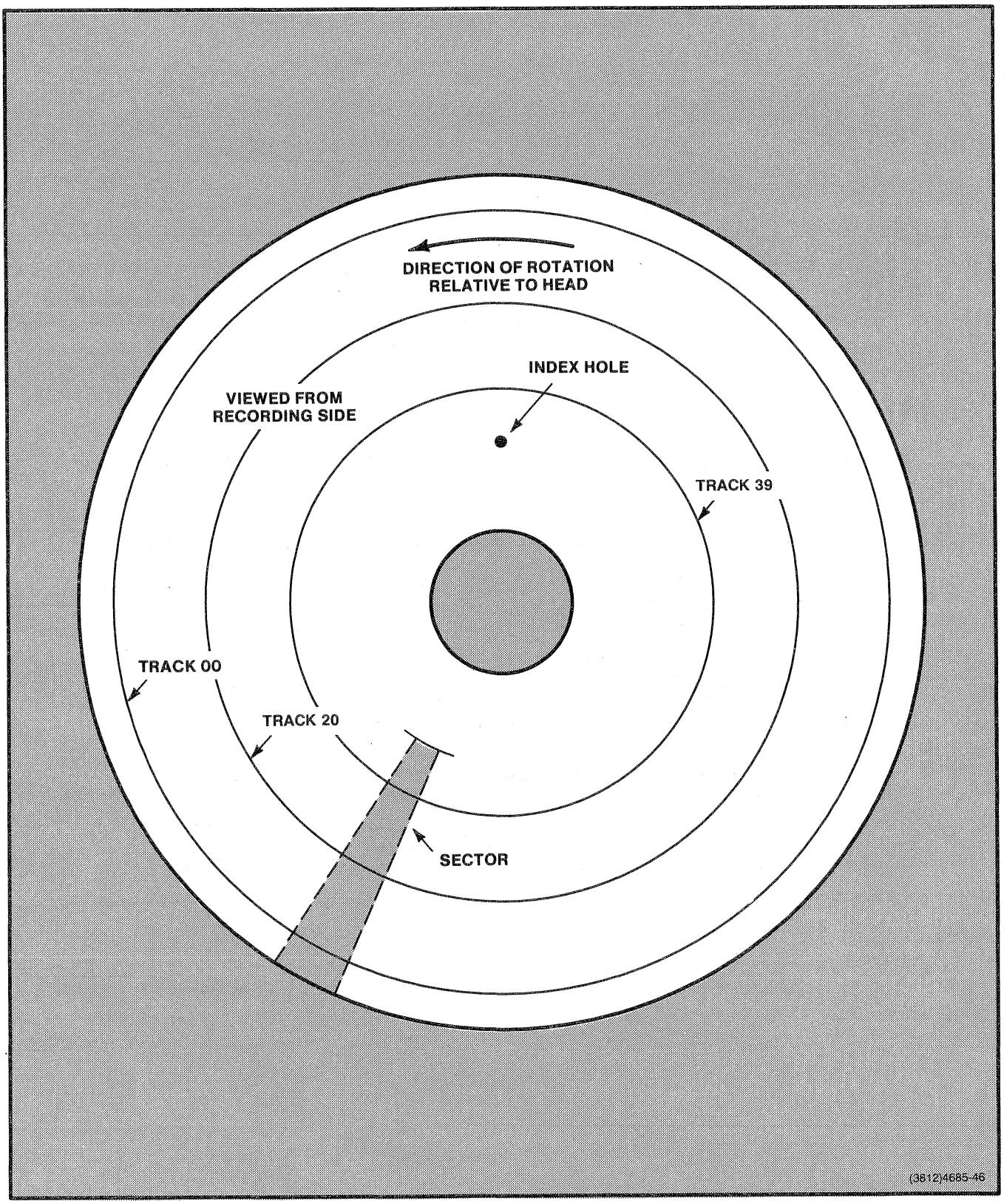


Figure 12-22. Flexible Disk Track and Sector Locations.

The index hole also serves as a drive ready monitor. When the Index Detector sees the hole twice, it knows a disk is in place and ready.

Each track is divided into 8 separate data fields, called sectors. Each is identified by its own address label. This label (called an Identifier Address Mark or ID AM) is written and read by the same head that reads and writes the data. This system of addressing by magnetic data is referred to as soft-sectoring. An alternate system, not used by the 4170, is called hard-sectoring. The hard-sectored diskette contains holes, similar to the index hole, to mark the boundaries between sectors.

Sectors having the same bit length vary in physical size according to their distance from the center of the disk. Data bits are naturally closer together on the higher numbered tracks. Since a sector contains 512 bytes, an 8-sector track contains 4096 data bytes. Using 40 tracks and both sides of the diskette, there is room for 327,680 data bytes per disk. These numbers assume the standard double-sided, double-density encoded data recording.

Sector Data Structure

During the formatting of a new diskette, the R/W head writes a unique address label at the beginning of each track and sector. At the same time, other bytes are written that allow the read circuitry to synchronize with the sector and locate data reliably.

Figure 12-23 shows how a sector is composed of identification (address), synchronization, data, and separator fields. The top line of the figure shows the pulse produced by the disk media index hole passing under the Index Detector. The track header follows this index mark, starting with a total of 92 bytes that allow the read circuitry to synchronize for reading an address or data. This header block contains GAP 1, which separates the track header from the sector address (ID) field.

A similar gap (GAP 2) separates the track header from the data field. Another gap (GAP 3) follows the data field and precedes the next sector ID field. However, if the sector is the last sector (sector 8), GAP 4, which fills the space up to the index mark, replaces GAP 3.

At this point, the arrangement of gaps and fields is repeated, starting around the next track (Figure 12-23). Each major segment in the track-sector organization is shown as a rectangular box. Each of these segments is then broken down further with its subparts listed under the box. Refer also to Table 12-7 for more information.

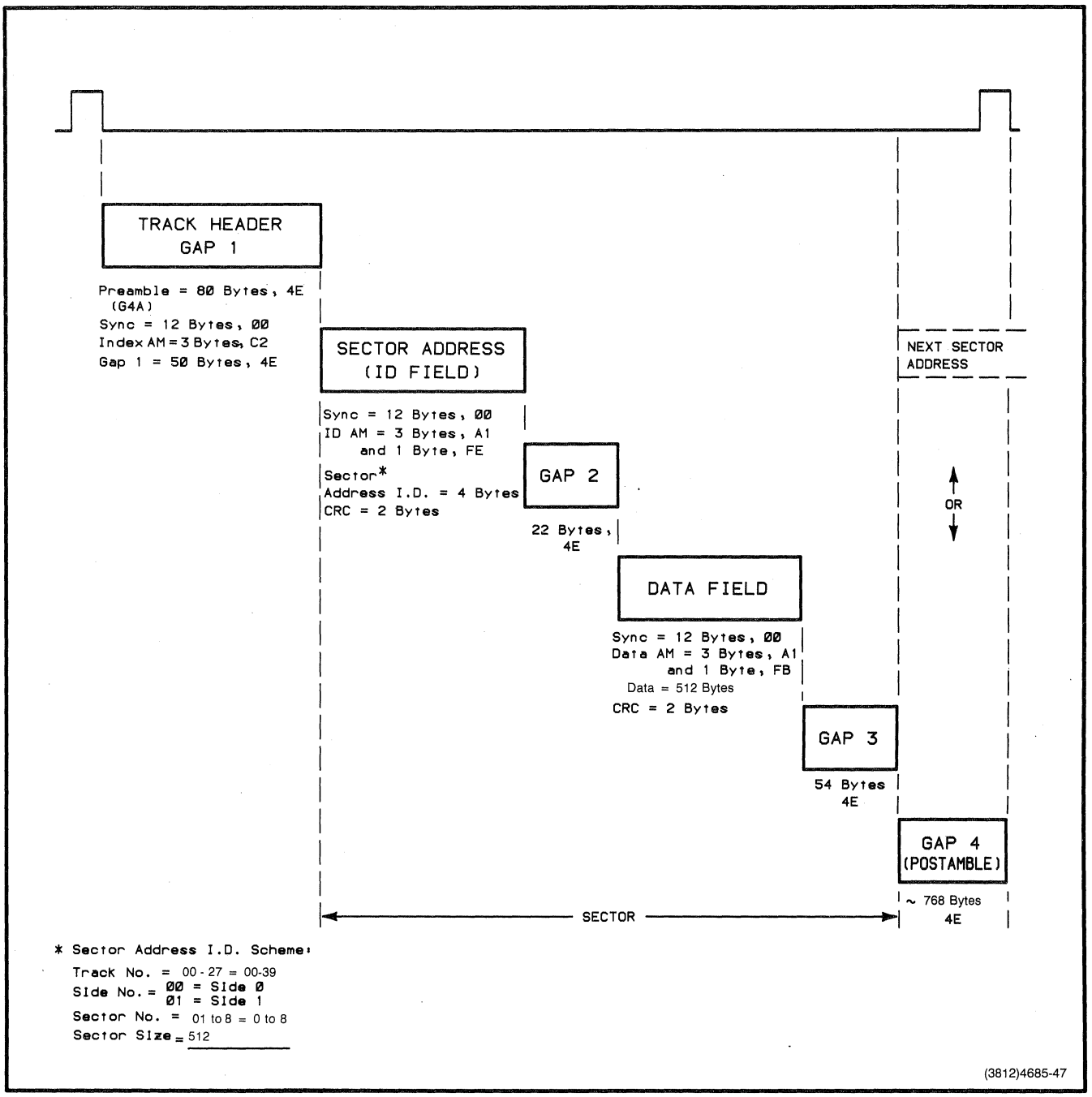


Figure 12-23. Track-Sector Format.

Table 12-7

IBM-COMPATIBLE MFM (a) FORMAT DESCRIPTION

Major Segments	Segment Parts	Description
TRACK HEADER	G4A	Goes from physical index mark sync allowing for physical index and speed variations, and interchange between disk drives.
	Sync	Is a fixed number of bytes for separator (b).
	Index AM	Is a unique byte that identifies index field and is written with special encoding rules (Figure 12-24b).
	GAP 1	Goes from index address mark to ID field address mark sync.
ID FIELD (Sector Address)	Sync	Fixed number of bytes for separator sync prior to the address mark. (b)
	ID AM	A unique byte identifying the ID field and written with special encoding rules (Figure 12-24c).
	ID	Is a four-byte address containing track number, head number (two-sided only), sector number, and sector length.
	CRC	Two bytes for CRC (cyclic redundancy check).
GAP 2	---	Gap from ID CRC to data AM sync; allows for speed variation, oscillator variation, and erase core clearance of ID CRC bytes prior to write gate turn-on for an update write.

Table 12-7 (cont)

IBM-COMPATIBLE MFM (a) FORMAT DESCRIPTION

Major Segments	Segment Parts	Description
DATA FIELD	Sync	12 bytes for Separator synchronization prior to an AM. (b)
	AM	A unique byte to identify the data field and written according to special encoding rules (see Figure 12-24d).
	Data	The area for user data storage.
	CRC	Two bytes for cyclic redundancy check.
GAP 3	---	Gap from CRC to next ID AM sync. Allows for: the erase core to clear the data field CRC bytes; speed and write oscillator variation; read preamp recovery time; and system turn-around time to read the following ID field.
GAP 4	---	G4B is the last gap before physical index and allows for speed and write oscillator variation (during format) and physical index variation.

(a) MFM stands for modified frequency modulation; see the MFM Disk Encoding Method discussion.

(b) Includes a minimum of two bytes plus worst case separator sync up requirements.

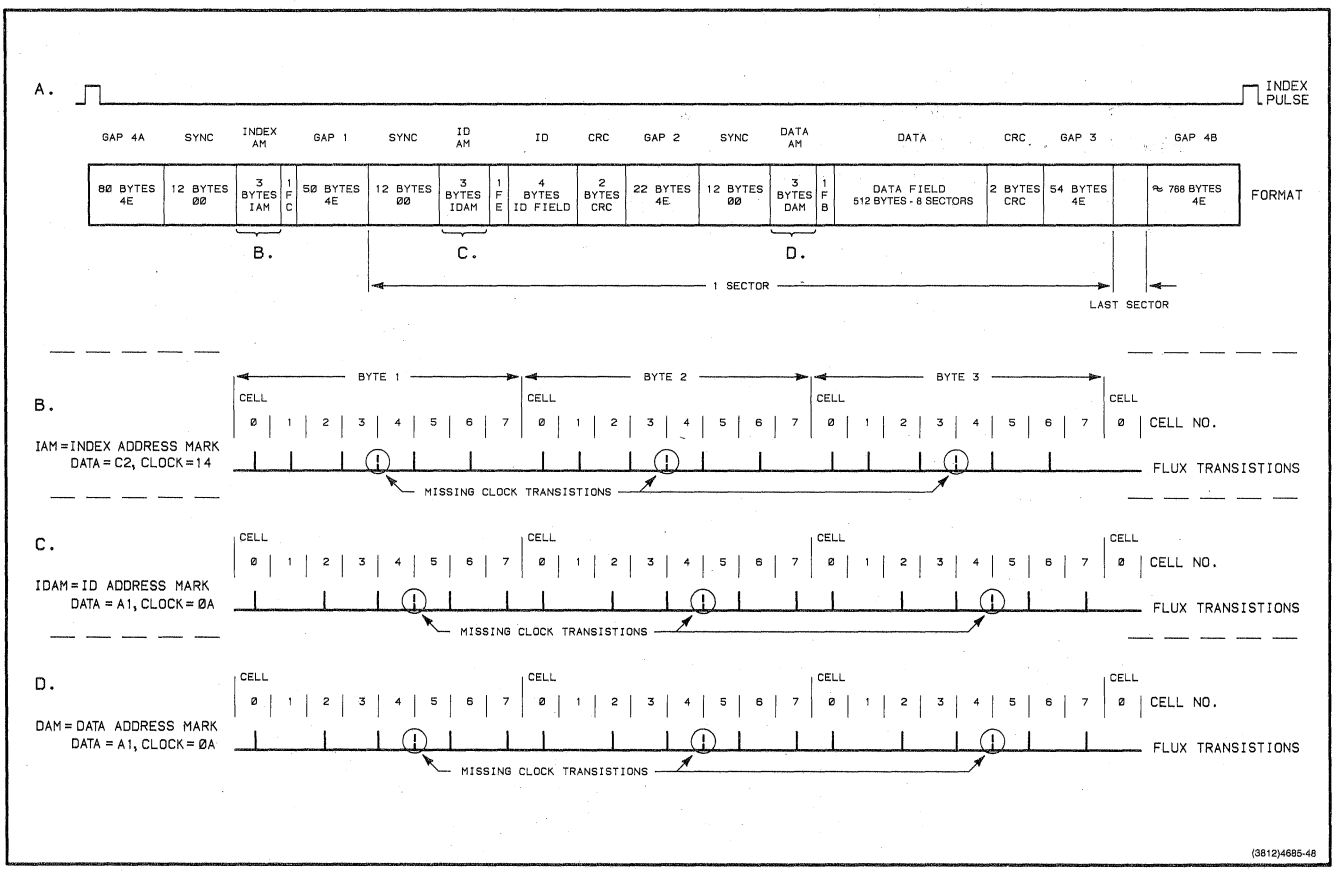


Figure 12-24. Specially Encoded Field Separators.

MFM Disk Encoding Method

The disk unit uses MFM (modified frequency modulation) recording, which allows twice the storage capacity of the usual (FM or frequency modulation) encoding method. The drive unit produces flux reversals on the diskette by changing the polarity (north-to-south vs. south-to-north) of the field applied by the write head.

Looking at the FM system (Figure 12-25a) a flux reversal marks the beginning of each bit cell, which is 4<micro>s long. An additional flux reversal between clocks indicates a data bit "one" in this cell. Conversely, a "zero" bit cell has no flux reversals between clock pulses.

By comparison, the MFM system (Figure 12-25b) retains the zero/one coding but omits the automatic clocks in each cell. Instead, a clock reversal occurs only when two or more zeros follow in succession; clocks are inserted at the boundaries of these adjacent zero cells. Also, as in FM, a reversal occurs for every one cell. By eliminating clocks where data reversals exist, the bit cells can be 2<micro>s long -- half the space needed for FM encoding. This effectively doubles the storage capacity of the MFM-encoded diskette.

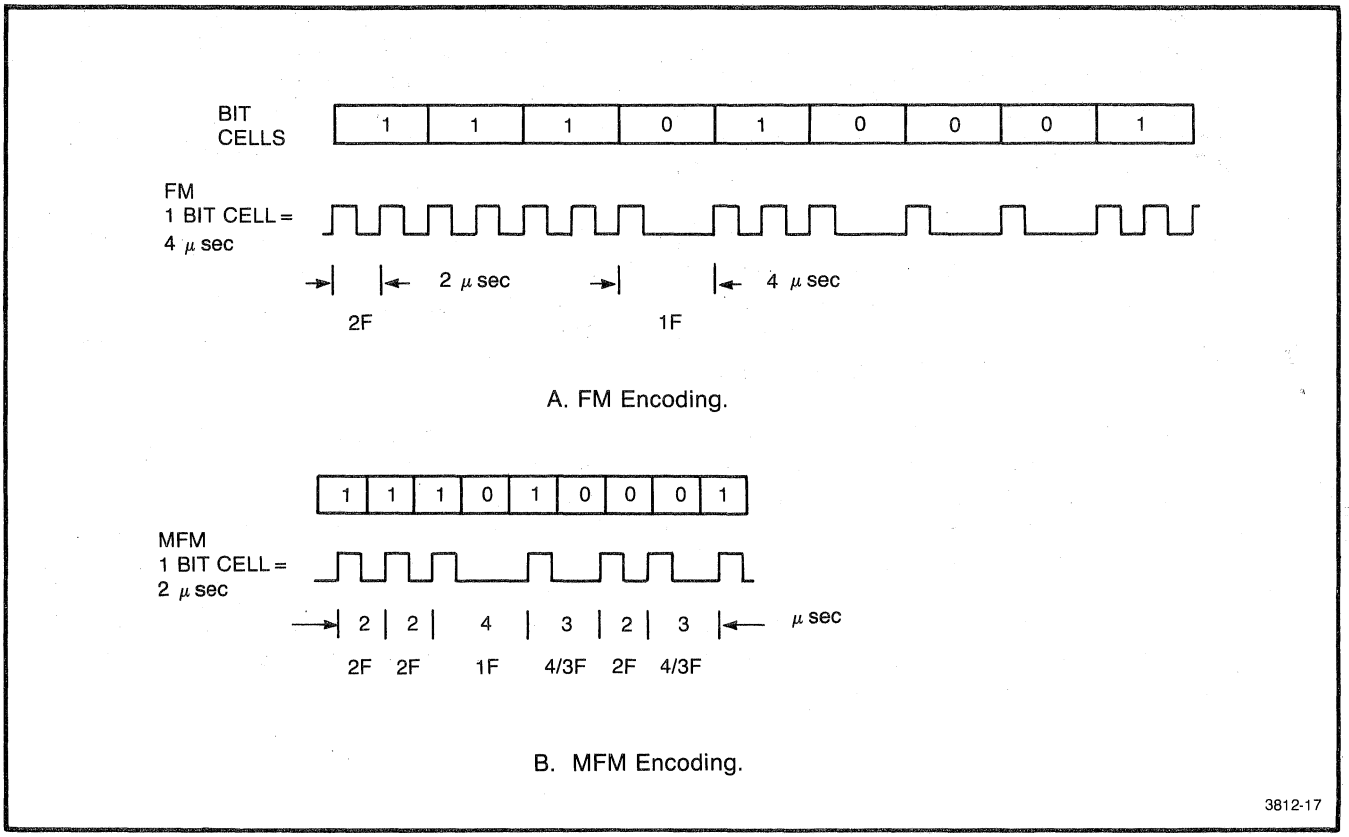


Figure 12-25. Flexible Disk Encoding Formats.

Write Precompensation

Flux reversal locations are adjusted to maintain a uniform timing of read pulses; this system is called write precompensation. It is normal for magnetic flux reversals (data) to relax (move) into a stable location shortly after they are written on the disk. This is caused by interaction of magnetic field forces in close proximity. The extent and direction of these reversal movements depends on the pattern of adjacent zero/one bits in the write stream. There are many combinations that cause a shift, but only the worst cases require compensation in order for the read head to sense the reversal where expected.

The write precompensation circuitry within the FDC looks at the serial bit patterns before they are written on the disk. Then it adjusts the write times as necessary up to +/-125 ns, depending on the bit pattern. The resulting data on the diskette has uniform bit cell spacing. This is called precompensation because a correction is entered before the data is written, rather than trying to correct the data stream when it is read.

OPERATION OVERVIEW

During operation, the main processor and the FDC talk to each other. Of these, the terminal processor is the more intelligent and can send commands asking the FDC to do certain jobs. The FDC is slower and needs access to the system bus, so the FDC responds to most commands from the processor by requesting the bus for small time slices. The processor grants the FDC bus control and waits for the FDC to complete the job. The FDC generates an interrupt when it finishes the task. The interrupt tells the processor that the FDC no longer needs the bus. The FDC then waits for the next command from the processor.

Hardware operations (reads, writes, etc.) are based on the three principle operating phases of the FDC chip:

- Status/Command
- Execution
- Results

Figures 12-26 through 12-28 compare the various input signals during the three operating phases.

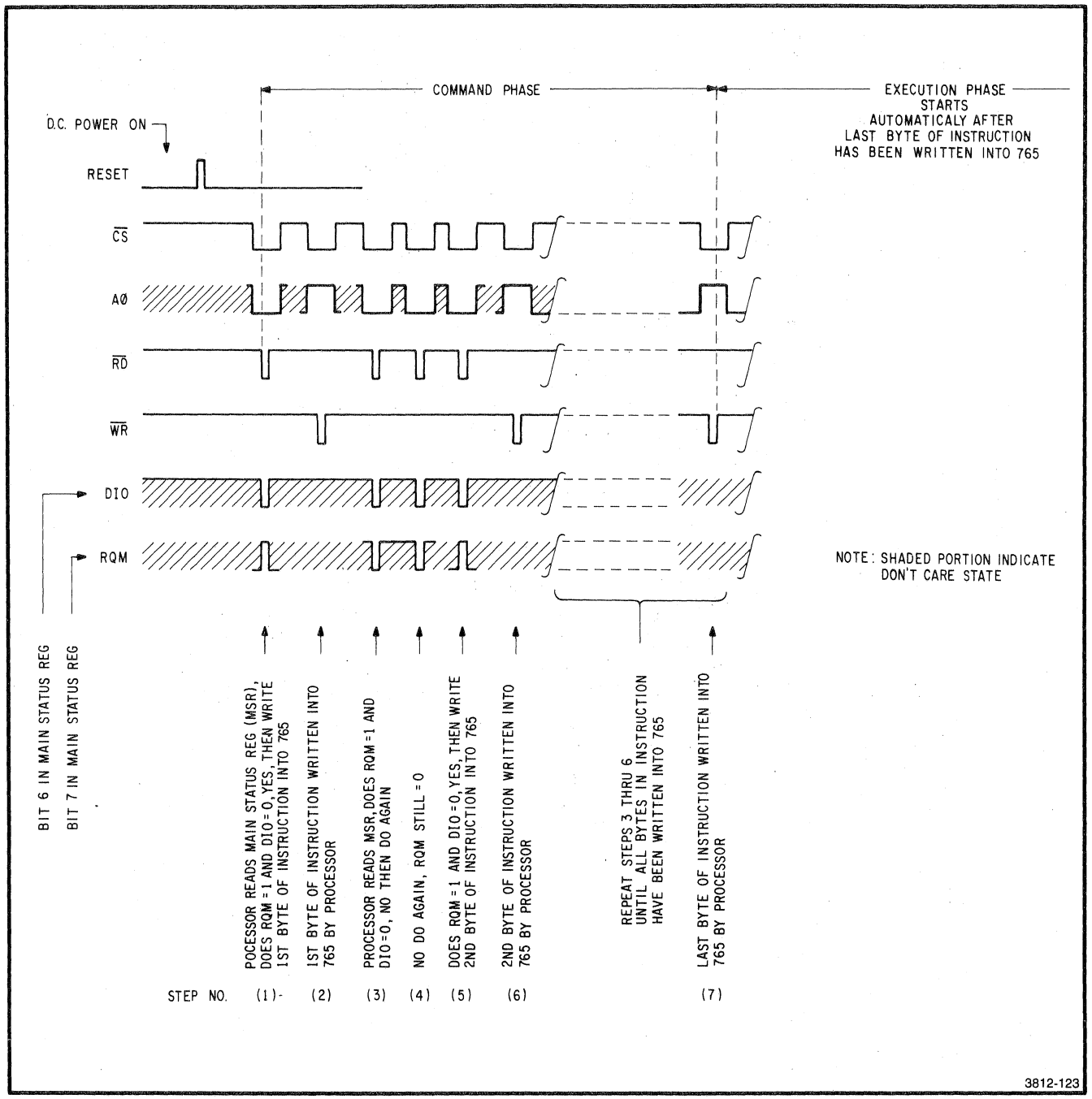


Figure 12-26. Command Phase Timing.

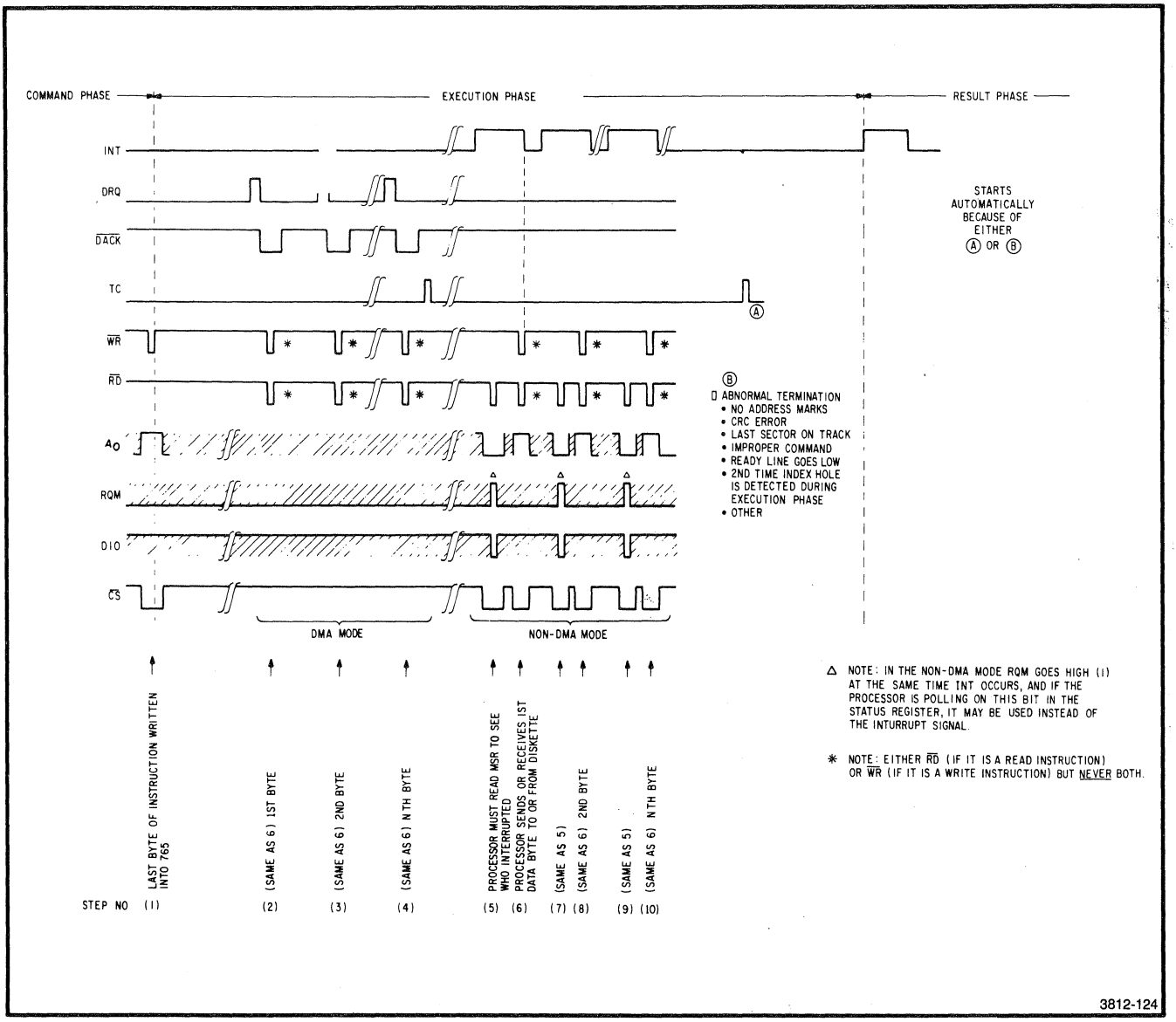


Figure 12-27. Execution Phase Timing.

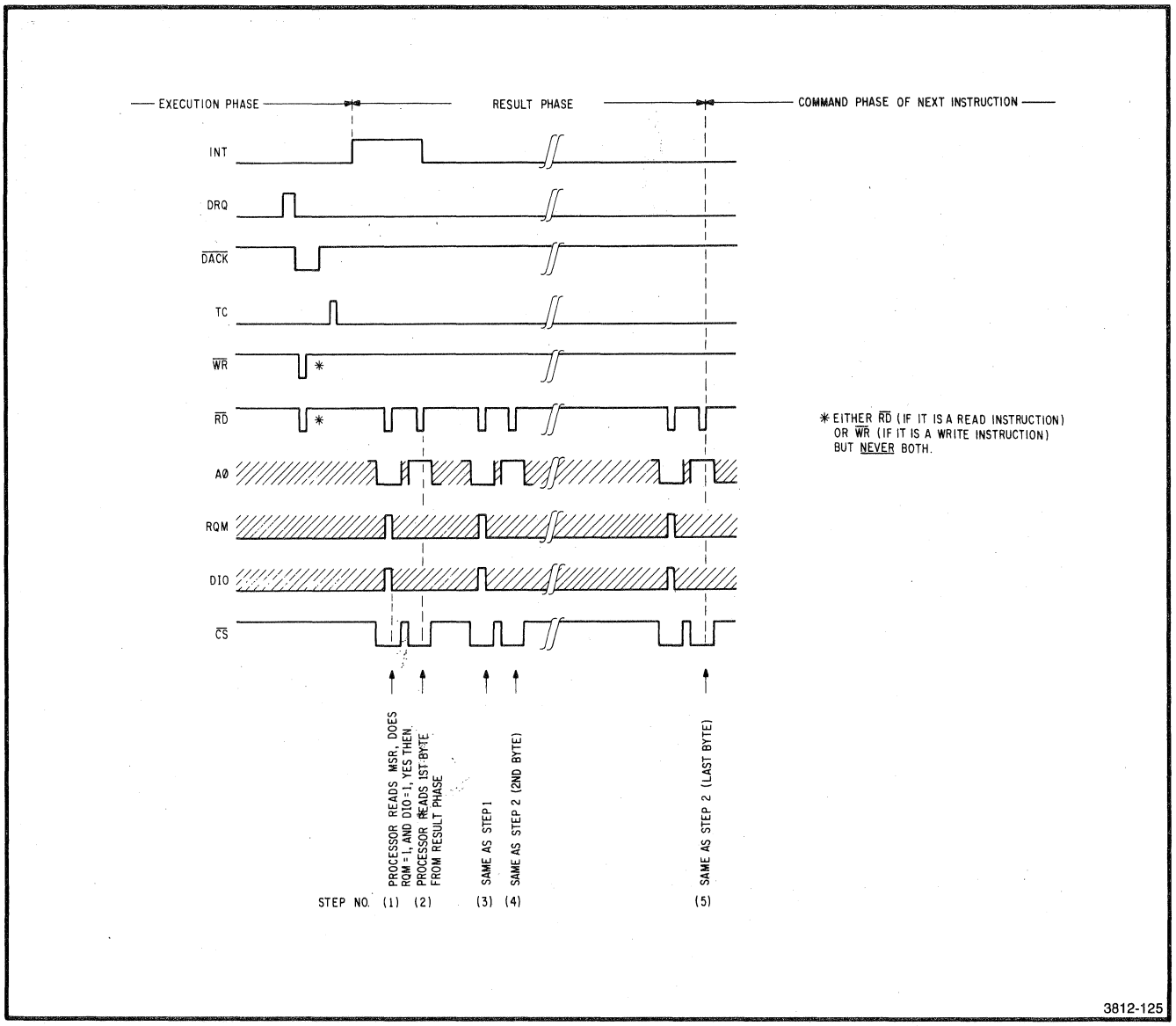


Figure 12-28. Result Phase Timing.

Status/Command Phase

The Status/Command phase begins with the processor reading the disk firmware. The processor uses the information found in the firmware to program the Disk Controller board hardware. The processor then receives a request for a disk seek, read, or write and assembles the set of instructions required for the operation. All processor-to-disk communication is transmitted via the disk I/O port in I/O address space FC00 to FC09 (Table 12-8).

Table 12-8

I/O PORT MEMORY MAP

I/O Address	Register
Base address+0	Board Status
Base address+1	Bits A16 to A19 of DMA starting address
Base address+2	Bits A8 to A15 of DMA starting address
Base address+3	Bits A0 to A7 of DMA starting address
Base address+6	FDC abort
Base address+7	Step DMA
Base address+8	FDC status
Base address+9	FDC Command/Result data

Figure 12-29 illustrates the steps the processor takes during the Status/Command phase. The processor first programs the Board Status Register (FC00). Its contents determine the following conditions:

- o The interrupt enable (INT on the Control Bus) is set
- o The DMA direction bit is set (read/write)

(The complete contents of this register are shown in Table 12-11, under the Board Status Register circuit description.)

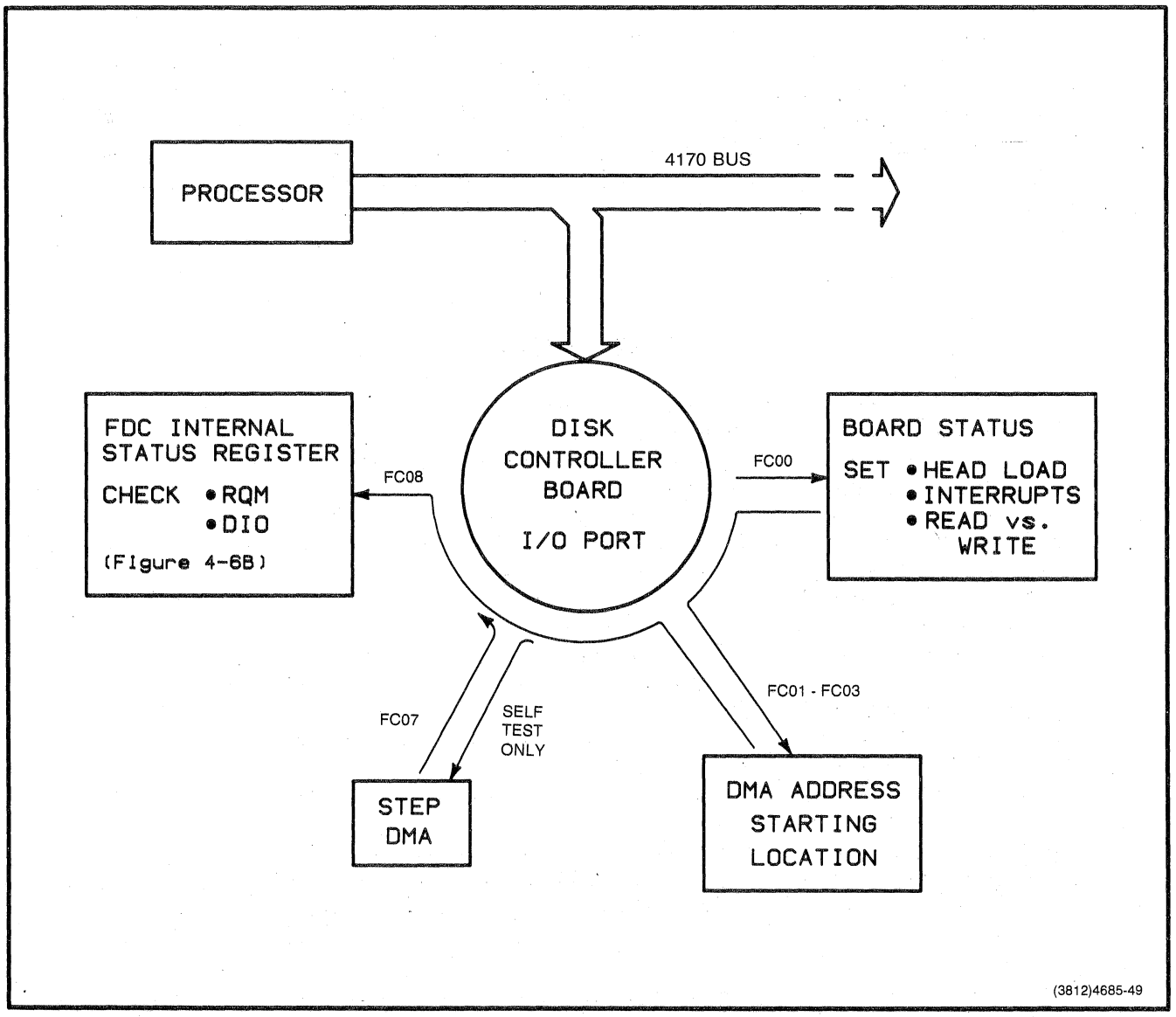


Figure 12-29. Status Flow Sequence.

The processor then goes to I/O address space FC01 -- FC03 where it writes the memory location of the first Direct Memory Access (between disk and RAM). Next, the processor reads a status register located inside the FDC chip (Table 12-9) and polls two bits of this register. The following information is expected:

- RQM-1 (Bit 8) which means FDC is not busy and is listening
- DIO-0 (Bit 7) which means the direction of data flow is into the FDC

The processor can now write the first instruction byte to the FDC Data Register. This corresponds to the process shown in Figure 12-30.

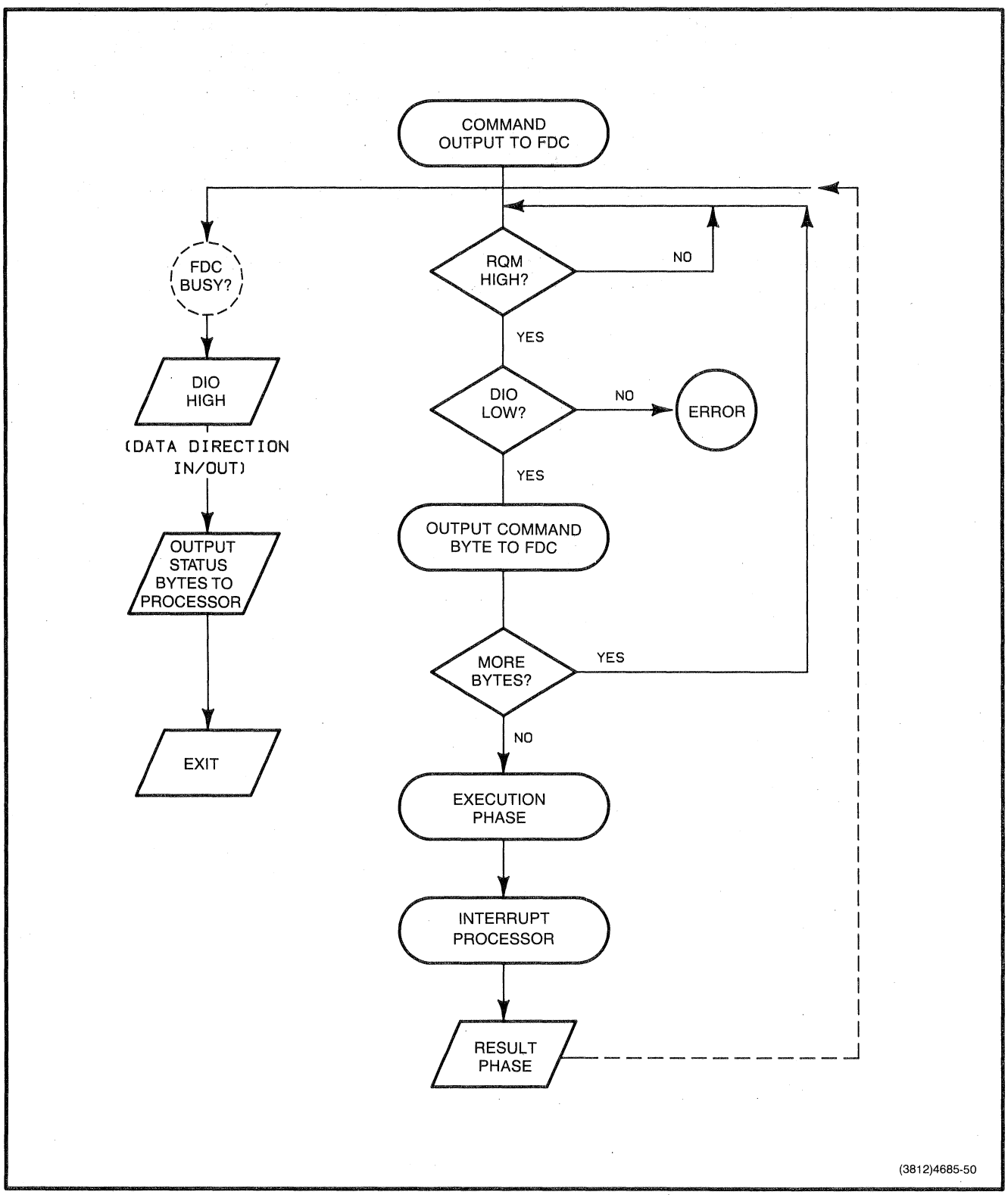


Figure 12-30. Command/Execution/Result Flowchart.

Table 12-9

FDC STATUS BYTE

Bit	Mnemonic	Type	Description
D7	RQM-1	Read Only	This bit indicates that the data register is ready to send data to or receive data from the processor. Both the RQM-1 and DIO-1 bits should be used to determine "ready" and "direction."
D6	DIO-1	Read Only	This bit indicates the direction of the data transfer for the data register. A zero indicates that data should be written to the FDC and a one indicates that data should be read from the FDC. This bit is only valid when RQM is true.
D5	NDM-1	Read Only	This bit indicates whether the FDC is in DMA mode or non-DMA mode. The interface must use the DMA mode; therefore, the bit must be a zero.
D4	CB-1	Read Only	This command indicates that the FDC is busy with a read or write command.
D3	D3B-1	Read Only	This bit indicates that Drive 3 is in the seek mode.
D2	D2B-1	Read Only	This bit indicates that Drive 2 is in the seek mode.
D1	D1B-1	Read Only	This bit indicates that Drive 1 is in the seek mode.
D0	D0B-1	Read Only	This bit indicates that Drive 0 is in the seek mode.

Execution Phase

The execution phase consists of performing one of the following operations:

- o Recal/Seek
- o Read
- o Write
- o Format

Most operations are a variation of a read or write. Therefore, the execution phase is explained in terms of a typical disk write operation. Later, you will see how the execution phase is different for a read operations. In all operations but seek (read and write), the hardware employs DMA (Direct Memory Access) to the system RAMs. (1)

(1) Using DMA means the Disk Controller board becomes a bus master, addressing RAM directly without involving the main processor.

Write Operation. The write operation begins with the FDC sending a DRQ (DMA Request) to the DMA State Machine. The DMA State Machine sends a bus request (BREQ and CBRQ), and examines the BUSY and BPRN lines to see if the bus is busy serving another bus master. If the bus is busy, the State Machine waits until the bus is available, then asserts its own BUSY on the bus. The Disk Controller board then becomes the temporary bus master.

At this time, the State Machine asserts DACK (DMA Acknowledge). DACK is sent to the FDC, confirming that DMA is granted; the FDC then removes DRQ. The DACK line also controls a Strobe switch, and enables the Address Counter/Buffer.

When the Address Counter receives a DACK, the Counter places its RAM address on the bus. This address with BHEN (Byte High Enable) remains on the bus until the data is read by the FDC and DACK is removed. When DACK is removed, the address is incremented (its counter is decremented).

During the command phase, the DMA direction bit is set. This has the effect of setting the R/W line (which enters the Control Strobes), causing the DMA READ line to connect to the bus via the MRDC line; this connects the DMA WRITE line to the FDC WR input. A WR causes the FDC to execute a single internal write operation. The R/W line also enters the DMA State Machine; when the machine receives W (R/W-O), it examines and waits for an ACK1 or ACK2 to be asserted by the RAMs. (When the RAMs are finished with a read or write cycle, they will respond with the proper acknowledge, ACK1 or ACK2.) After the RAMs write the data onto the bus, the WR is removed. At this transition the data byte is written into the FDC Data Register; then the MRDC is removed and, as stated earlier, DACK goes away. Thus the RAM address is cleared from the bus, and the address is incremented (Figure 12-31).

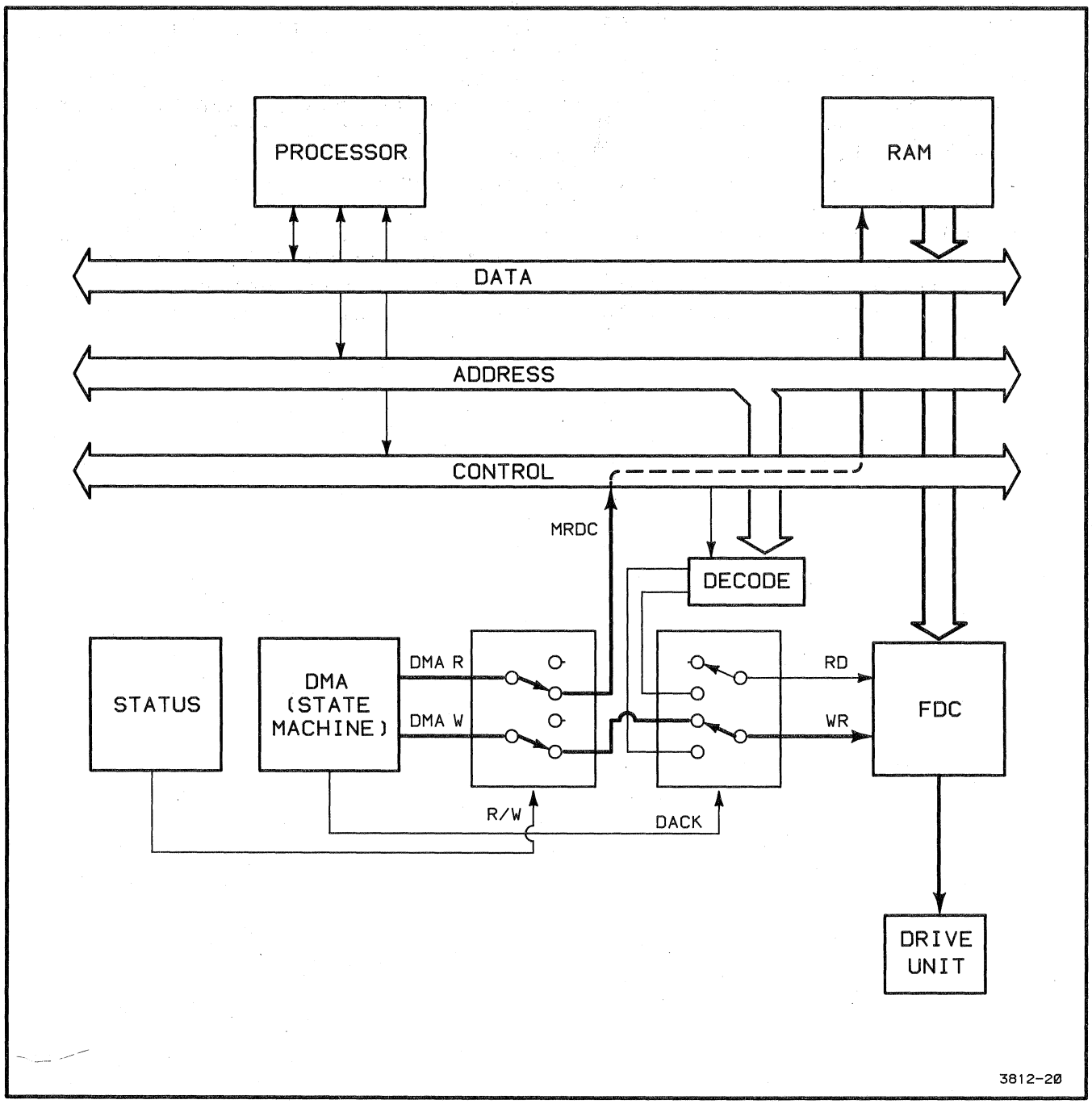


Figure 12-31. Write to Disk Signal Flow.

The data byte is then processed by the FDC internal processors as follows:

- o Transformed from parallel to serial data.
- o Adjusted for write precompensation.
- o Written to the track-sector on the disk.

The execution process just described is repeated for each additional byte of I/O data. When FDC instructions call for writing data on a complete sector (in double-density), 512 DMA byte transfers are executed per sector. Multiple sectors may be transferred in one command. One byte is processed every 32<micro>s. The actual time when the DMA transfer is passing data over the terminal bus is (approximately) 1<micro>s.

When the last byte transfer in the instruction set is completed, the FDC generates an interrupt (EOC). This interrupt passes through the Board Status Register and is placed on the Control bus. The interrupt tells the processor that the FDC has completed its work and is ready for the result phase to begin.

Read Operation. The previous discussion covered the DMA control circuitry during a write operation. Here the same circuitry is studied during a read operation.

During the command phase, DACK is set, and the R/W direction line is set to READ. The R/W line then sets the Control Strobe switches so the DMA WRITE line (from the State Machine) connects to the terminal bus MWTC and AMWTC lines. Also, the DMA READ line connects to the FDC RD operation. The same timing and addressing scheme used for FDC writes applies to FDC reads; thus disk data is placed on the bus by the FDC and then accepted by the RAMs (Figure 12-32).

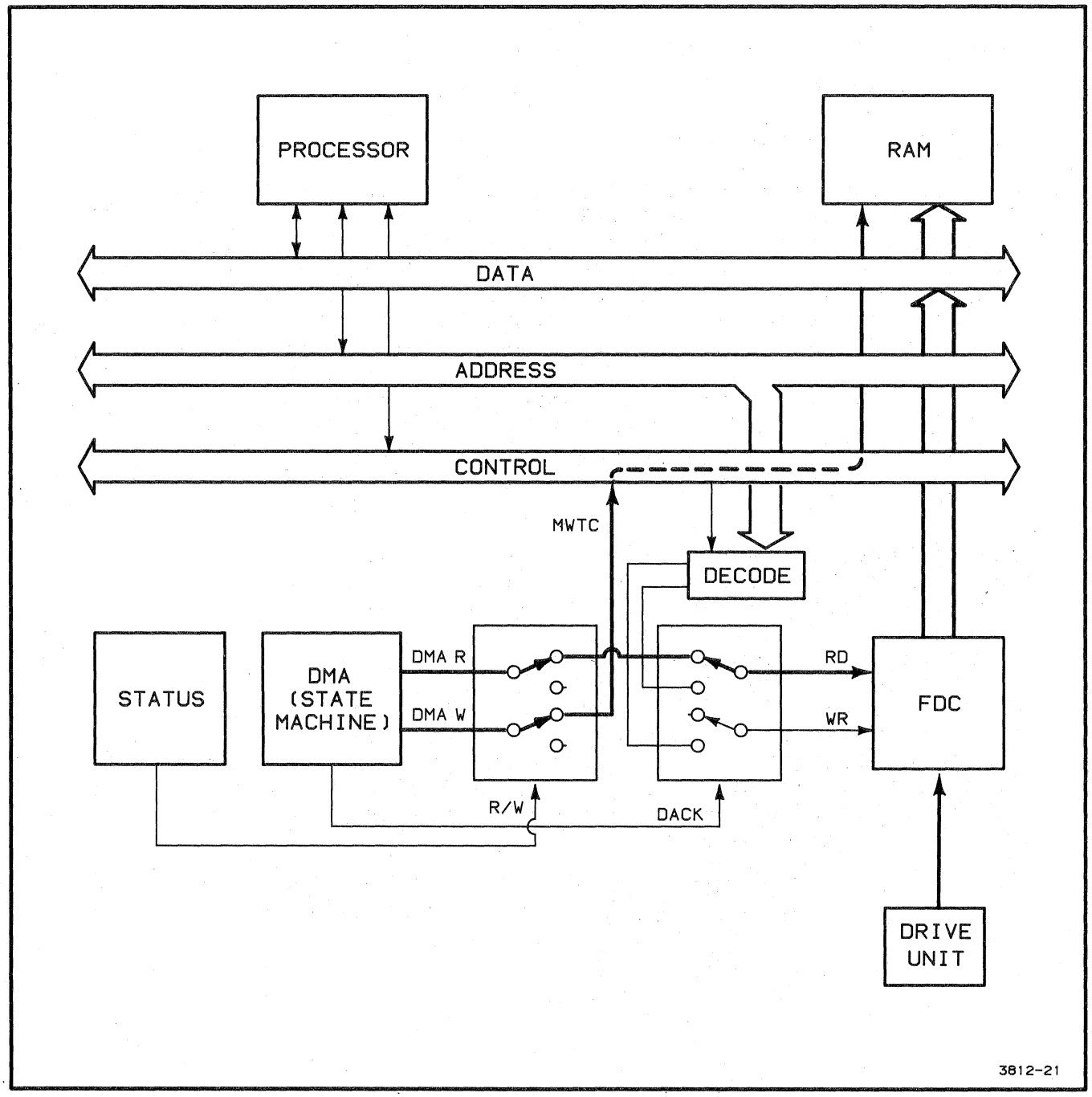


Figure 12-32. Read from Disk Data Flow.

Result Phase

During the result phase the processor reads the Main Status Register and waits for RQM and DIO to go high. The processor then reads up to seven bytes of response/status data from the FDC and waits for RQM and DIO to go high after reading each byte. Each status byte is read, one at a time, from an internal data register. The number and selection of bytes read depends on the instruction just executed. When the last byte is read, the Result phase is completed, and the FDC is ready to service the next set of instructions (disk operation) from the processor.

After the Result phase is completed, the processor again reads the Main Status Register and waits for RQM to go high. When DIO goes low, the processor then writes the first byte of the next set of instructions, and the entire cycle starts over again.

After the status bytes are read, the Motor On timer is set (enabled). This keeps the motor on for two seconds in case the next operation also needs the motor to be on; this avoids motor-turn-on delays.

CIRCUIT DESCRIPTIONS

This section focuses on the general operation of the hardware rather than on program-dependent operating states. Figure 12-33 is a functional block diagram of the Disk Controller board logic.

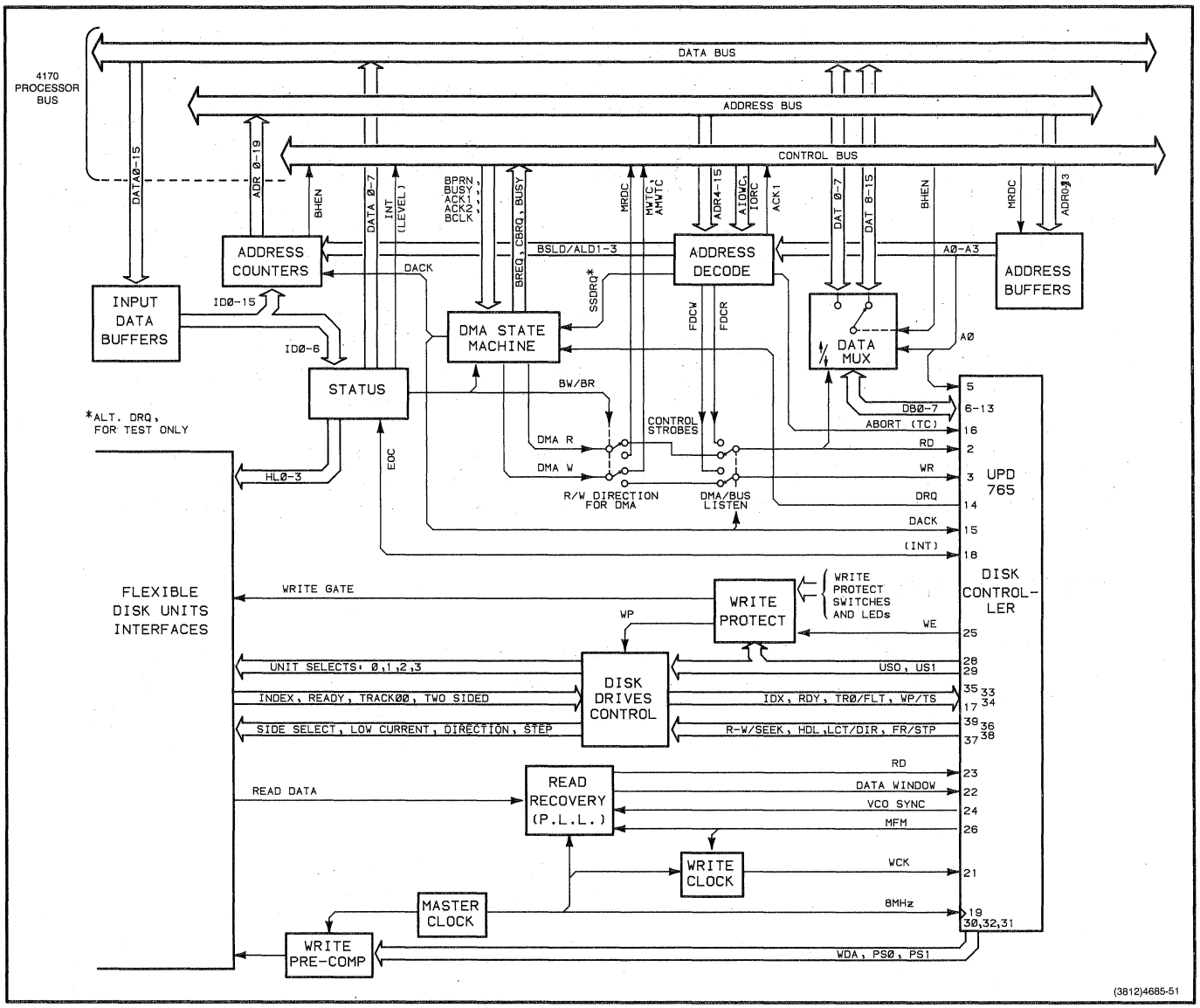


Figure 12-33. Disk Controller Board Block Diagram.

Disk Drive Control

The Flexible Disk Controller (FDC) is a single IC that is the main component of the Disk Drive Control (Schematic A3-4). The FDC handles the following functions and operations:

- Serial-to-Parallel Write data conversion from the 4170 Data Bus to the Disk Drive Unit(s).
- Calculation of the amount of Write Precompensation for each combination of serial data bits and adjustment of the write data stream accordingly.
- Serial-to-Parallel Read data conversion from the Disk Drive Unit(s) to the 4170 bus.
- Disk Drive Unit(s) function controls: Head Select, Unit Select, RW/Seek.
- Disk Drive Unit(s) status: Ready, Write-Protect, Index, Track 00.
- DMA State Machine control.

To handle these functions, the FDC has two internal processors, several data and status registers, and several buffers. The FDC also has separate READ and WRITE control lines that come from the Control Strobes.

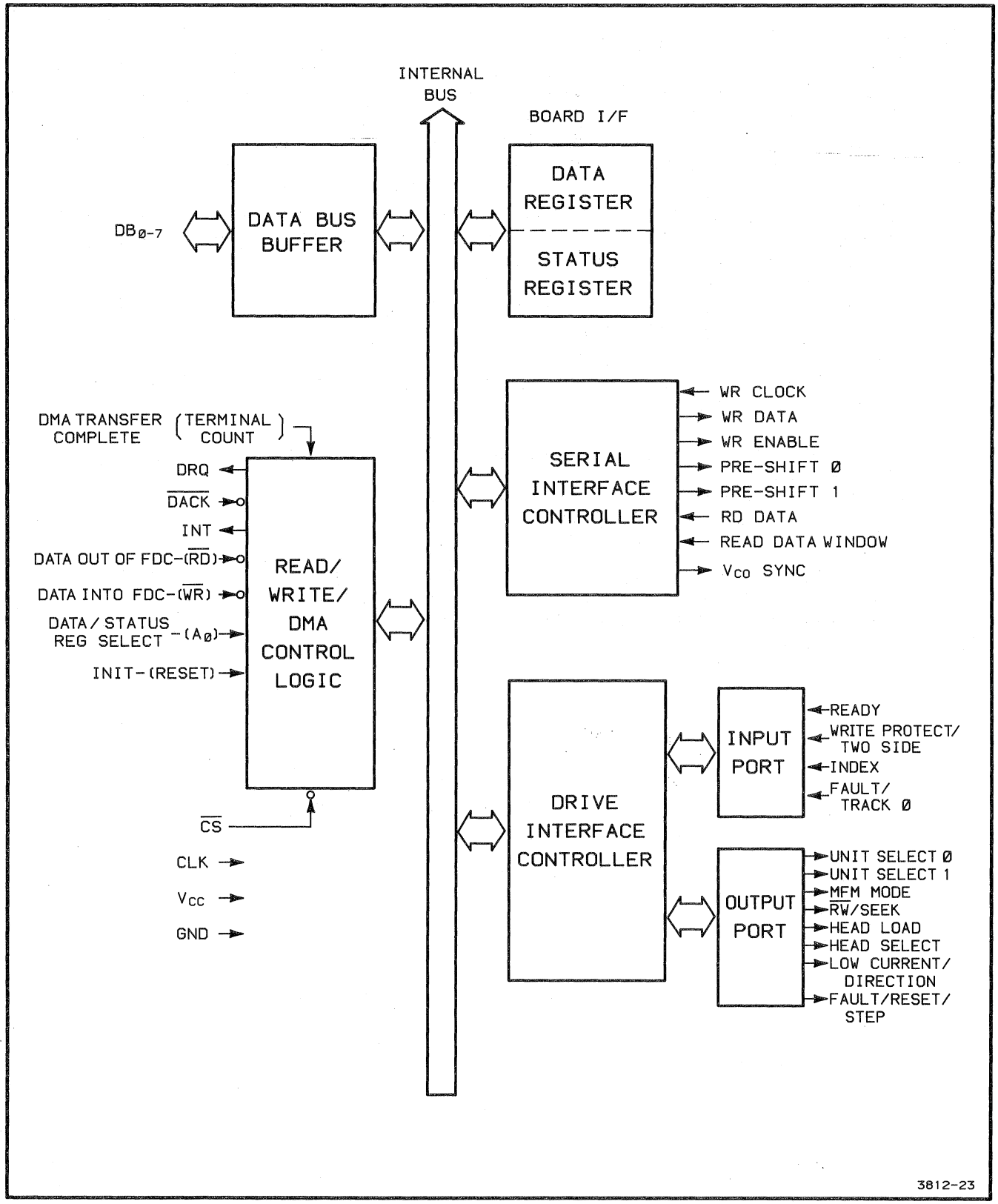


Figure 12-34. FDC Simplified Block Diagram.

The FDC is programmable and is controlled by the terminal processor. The firmware determines its operating modes and parameters. Commands and instructions enter the FDC through the Data Register (FC09). The FDC has an 8-bit data bus and data registers, while the processor uses 16-bit words. An external multiplexer converts the 16-bit processor instructions into 8-bit bytes, which the FDC can properly store and execute. A detailed description of the data multiplexer appears in the FDC Data MUX description.

I/O Registers. The FDC keeps its eight bits of status information in a one-byte internal register. Only the FDC itself can write into this status register. The processor must read this register before it can send or receive instructions to and from the Data Registers. Table 12-9 defines the contents of the status word contained in this register, accessed via FC08.

The Address Decoder translates addresses FC08 and FC09 into three control lines: A0, WR-0, and RD-0; these lines determine whether the data or status registers are enabled. Table 12-10 shows which combinations are needed to access these internal registers.

Table 12-10

3-TO-1 DECODER

Select Inputs		Input Selected (X=Don't Care)				Output
A	B	C0	C1	C2	C3	Write Data
L	L	L	X	X	X	L
L	L	H	X	X	X	H
H	L	X	L	X	X	L
H	L	X	H	X	X	H
L	H	X	X	L	X	L
L	H	X	X	H	X	H
H	H	X	X	X	H	H

Timing Information. The timing of the FDC operations is controlled by two external clocks which enter Pin 19 (4 MHz CLK) and Pin 21 (WR CLOCK). The 4 MHz clock times all internal operations except the write data output. The write data stream to the disk is timed, synchronized, and set for MFM by the WR CLOCK.

A combination of FDC interrupts and processor timing determines the rate of data flow to and from the 4170 bus. During disk data transfers between the FDC and the processor, the FDC is connected to the terminal bus and does a DMA transfer every 32<micro>s. If the FDC is not serviced in time, it sets the OR (Over Run) flag in the Status Register and terminates the current command.

Clocks

The Master Clocks and Write Clock circuits (Schematic A3-4) generate the Master Clock and Write Clock signals (Figure 12-35).

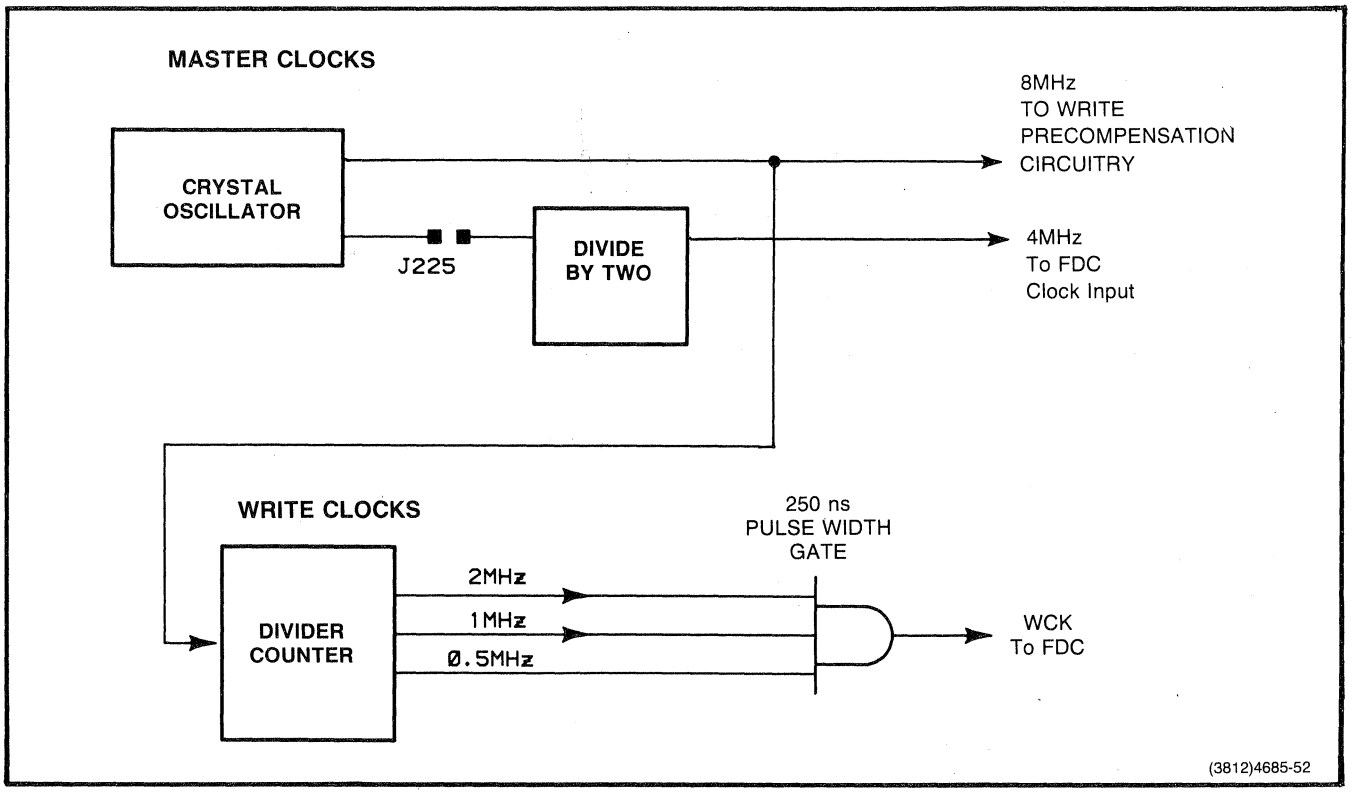


Figure 12-35. Clock Circuits.

Master Clocks. This circuit block generates the following clock signals:

- o 8 MHz for the Write Precompensation
- o 4 MHz for FDC timing
- o 1 MHz for Write Clock

An 8 MHz, crystal-controlled oscillator circuit generates the Master Clock signals. The oscillator output controls the Write Precompensation logic and also generates the Write Clocks.

Write Clock. The Write Clock logic provides the timing signal WCK (Write Clock) for the FDC write data stream to the disk. A divider/counter takes the Master Clock 8 MHz signal and divides it by 4, 8, and 16. The outputs of the divider (QB, QC, and QD) become 2 MHz, 1 MHz, and 0.5 MHz respectively. The divider outputs are ANDed, producing a 0.5 MHz clock with a pulse width of 250 ns -- determined by the 2-MHz input. The output signal is called WCK. Refer to the Write Clock waveforms in Figure 12-36.

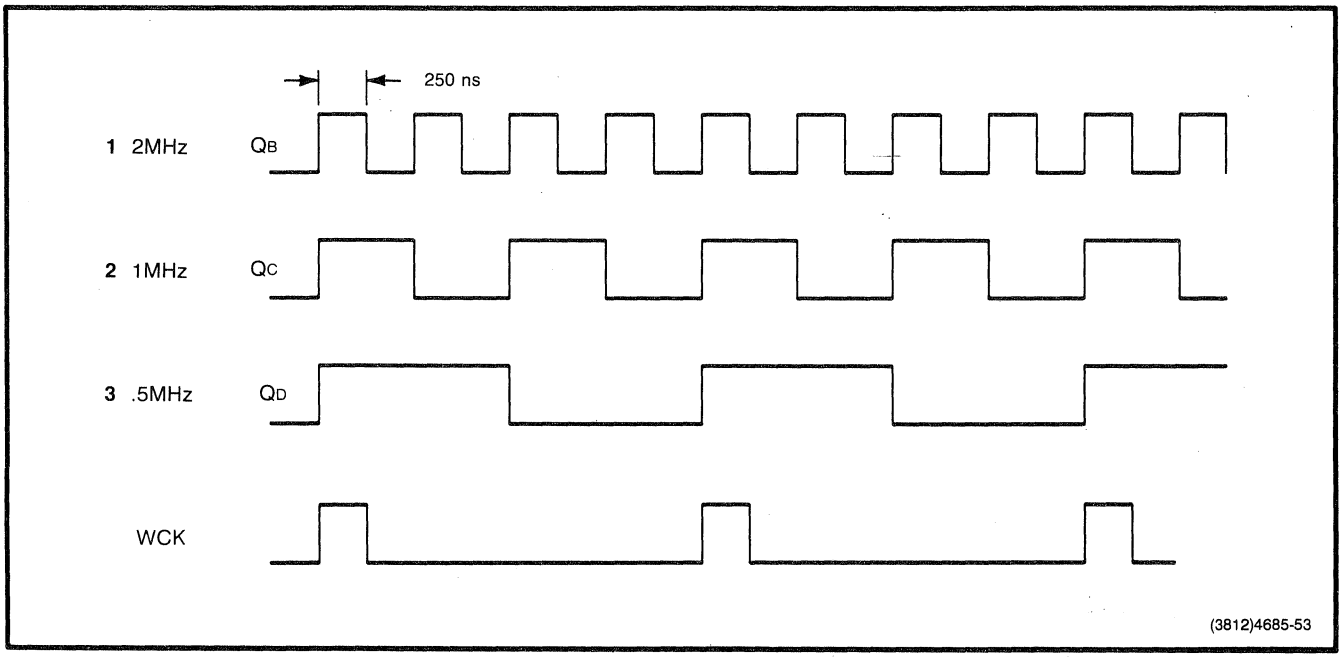


Figure 12-36. Write Clock Waveforms.

Write Control

The precompensation and write protection circuits are the only blocks that affect the path of data written to the disk.

Write Precompensation. The Write Precompensation logic (Schematic A3-4) takes the serial stream of write data (WDA) and adjusts each bit as needed (up to +/-125 ns) to compensate for magnetic interactions on the disk. After the data is written and relaxes into its stable position, the bit cells appear more uniformly spaced on the disk, making the disk easier to read. Refer to the Write Precompensation explanation earlier in this section.

The FDC computes the compensation shift needed for each bit pattern. It then tells this circuit block to adjust the bit spacing accordingly. (Some bit patterns will not need Write Precompensation.)

The write precompensation logic consists of an 8-bit Shift Register and a 3-to-1 decoder. The FDC sends a stream of write data (WDA) to both inputs of the shift register; thus no input comparison is performed by this chip. Only three outputs from this chip are used: the B output represents no relative shift, and the A and C outputs represent late and early shifts relative to B. The 8-MHz clock input (from the Master Clock block) determines the time value of these shifts. The standard bit shift (between normal and early or normal and late) is 125 ns (Figure 12-37).

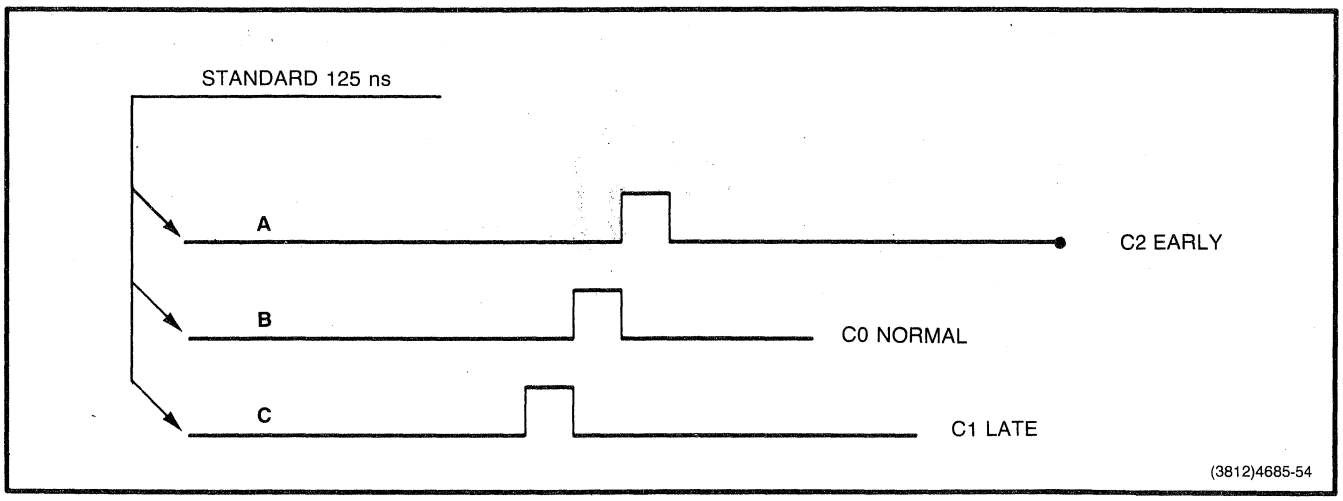


Figure 12-37. Bit Shift Selection.

The Write Control logic also chooses between the EARLY, NORMAL, and LATE lines and decodes them back into a serial bit stream. To do this, the 3-to-1 decoder accepts the Shift Register outputs and selects the proper one as directed by the FDC.

The FDC determines whether a shift is needed (and whether it is plus or minus). It then requests the shift via its PSO (Pre-Shift 0) and PS1 lines, which enter the decoder A and B select inputs. Table 12-10 shows the combinations of A and B that are needed to select EARLY, NORMAL, and LATE lines. The output is called WRITEDATA and is based on the timing of the NORMAL bit line (Figure 12-38).

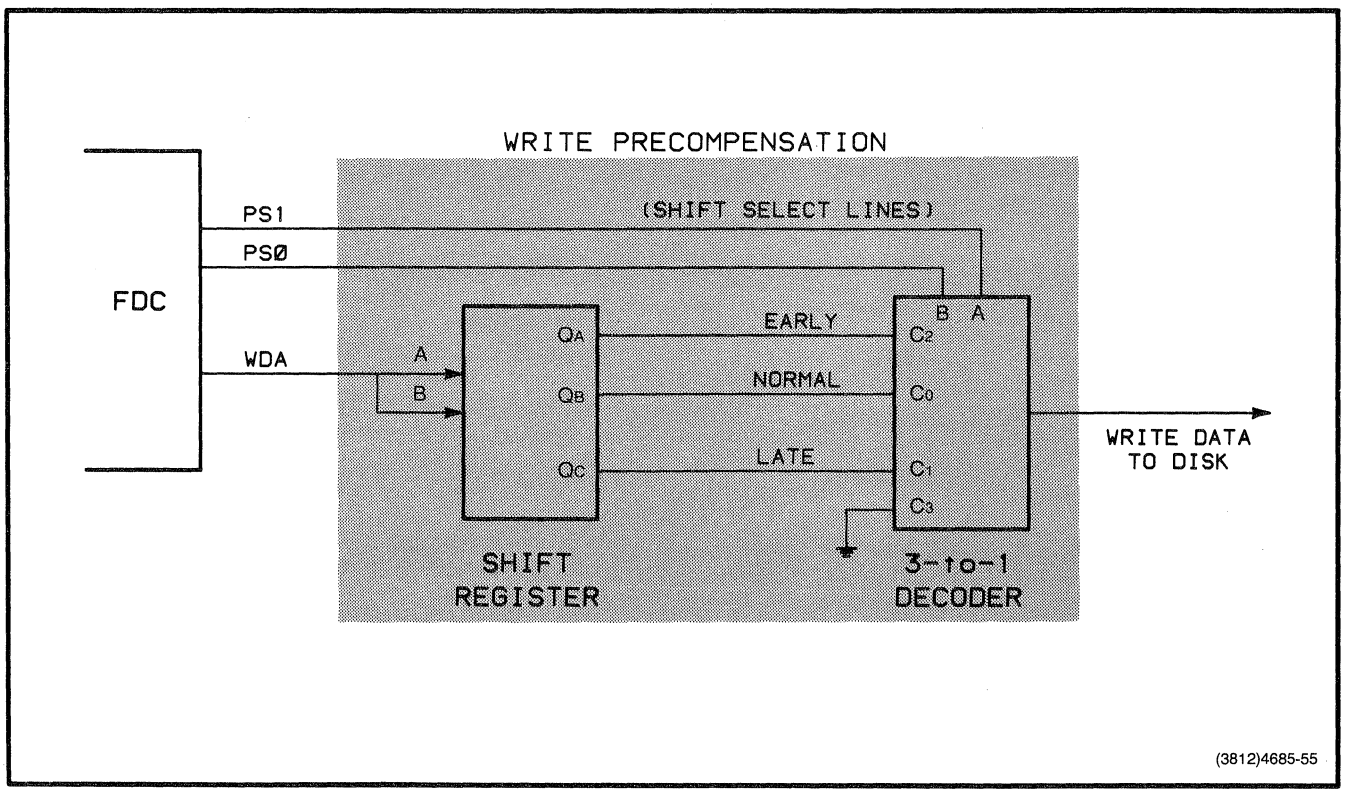


Figure 12-38. Write Precompensation Block.

Write-Protect Block. The Write-Protect (WP) logic (Schematic A3-4) gathers all WP inputs (protected disk media and WP switches), reports each drive unit WP status to the FDC, and controls the Write Enable line (to the write heads)(Figure 12-39).

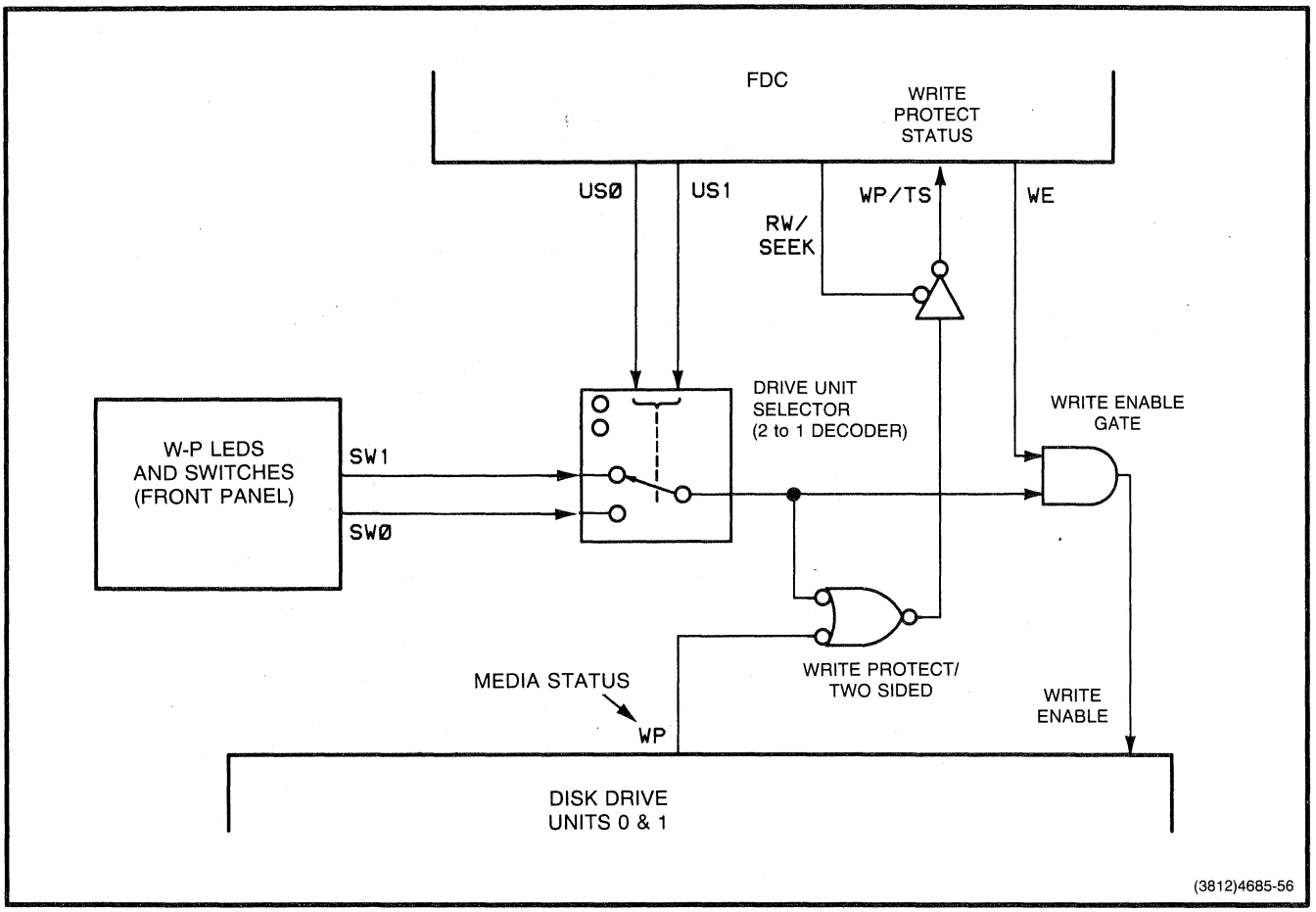


Figure 12-39. Write-Protect Circuit.

The presence of tape on the write-protect slot is detected by the LED-photocell. Any disk media can be temporarily protected by setting the WP switch (on the disk unit containing that media) to "protected." The adjacent LED will then light.

The WP switches are connected to a drive unit selector. The 2-to-1 decoder functions as a drive unit selector. The decoder is switched by the FDC USO (Unit Select 0) and US1 lines. The protection status of a particular drive unit and its media is sent to the Write Enable gate. This gate ANDs the FDC Write Enable line with the decoder output and makes the WRITE GATE line. This output operates with the Drive Select control lines to enable the selected drive ONLY IF THAT DRIVE IS NOT WRITE-PROTECTED.

Read Recovery. The Read Recovery circuit (Schematic A3-4) takes read data from the disk, stretches the pulses to a uniform 100 ns, sends the data to the FDC, and sends clock pulses to the FDC RDD input during a write to the disk.

Part of this logic also takes data or clock pulses and synthesizes a constant pulse frequency; it creates a time window that the FDC uses to discriminate between data and clock pulses on the incoming READ DATA stream from the disk.

The Read Recovery logic consists of Clock MUX (Multiplexer), Data MUX, Timing, Frequency Synthesizer, and Counter circuitry (Figure 12-40).

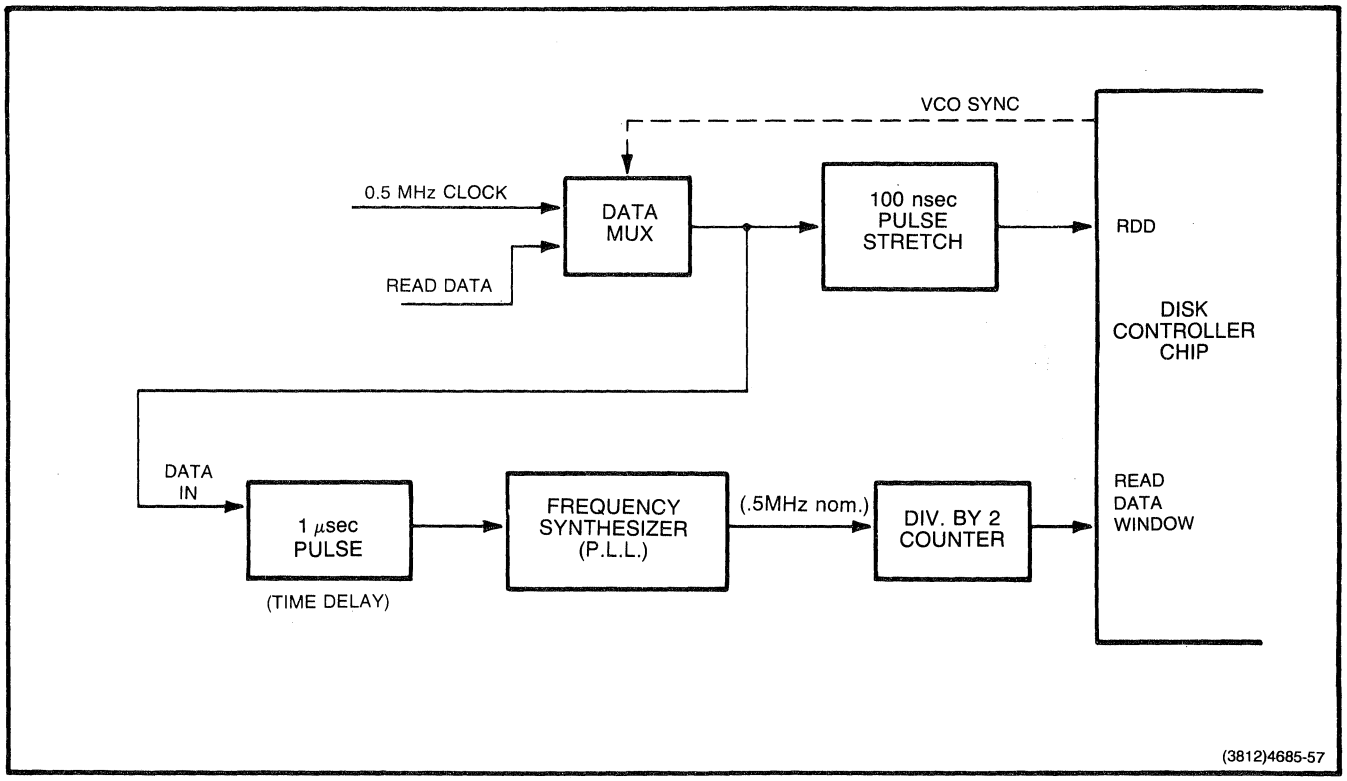


Figure 12-40. Read Recovery Circuit Block.

The Data MUX switches between the CLOCK input and the READ DATA line (from the Drive Unit's read/write head). When reading valid data from the disk, the Data MUX is in the READ DATA position. During disk writes, "idle time," or while unwanted data or data gaps are read, the incoming data frequency may vary drastically. This means that when valid data reappears, it may take longer for the PLL (phase-locked loop) to come back to the right frequency and sync on it. To overcome this problem, the Data MUX switches the input to a constant 0.5 MHz, the CLOCK input. The 0.5 MHz is very close to the normal data frequency.

The Data MUX output connects to both the READ DATA path and the Read Data Window path. The READ DATA path passes through a one-shot that stretches all incoming pulses to a uniform 100 ns duration. This makes it easier for the FDC to synchronize on the incoming data/clock pulses.

The other half of the Data MUX output enters another one-shot. This 1000 ns one-shot gives a time delay before pulses reach the frequency synthesizer circuit. The time delay simulates the normal pulse duration for read data. The pulse duration is half the period for the normal input frequency. The shaped data stream then passes into the phase-locked loop frequency synthesizer.

Phase-Locked Loop. The phase-locked loop allows the decoder to track directly on the raw data stream, thus eliminating the jitter caused by magnetic interactions on the disk.

Figure 12-41 shows the phase-locked loop diagram. This circuit functions as a frequency synthesizer and stabilizer. The incoming data stream first passes through a phase detector where it is compared against an error signal on a feedback input. The resulting difference is translated into an analog signal by the charge pump. This signal is then integrated, to give a picture of trends occurring over time, before entering the voltage-controlled oscillator (VCO, not to be confused with the VCO SYNC line from the FDC). The oscillator produces a digital clock signal that is shifted (plus or minus) according to the error present (if any). This error signal is fed back into the phase comparator, so the whole circuit acts like a frequency-regulating servo. A counter in the feedback path divides the oscillator frequency down to 0.5 MHz and sends it back to the Detector Enable.

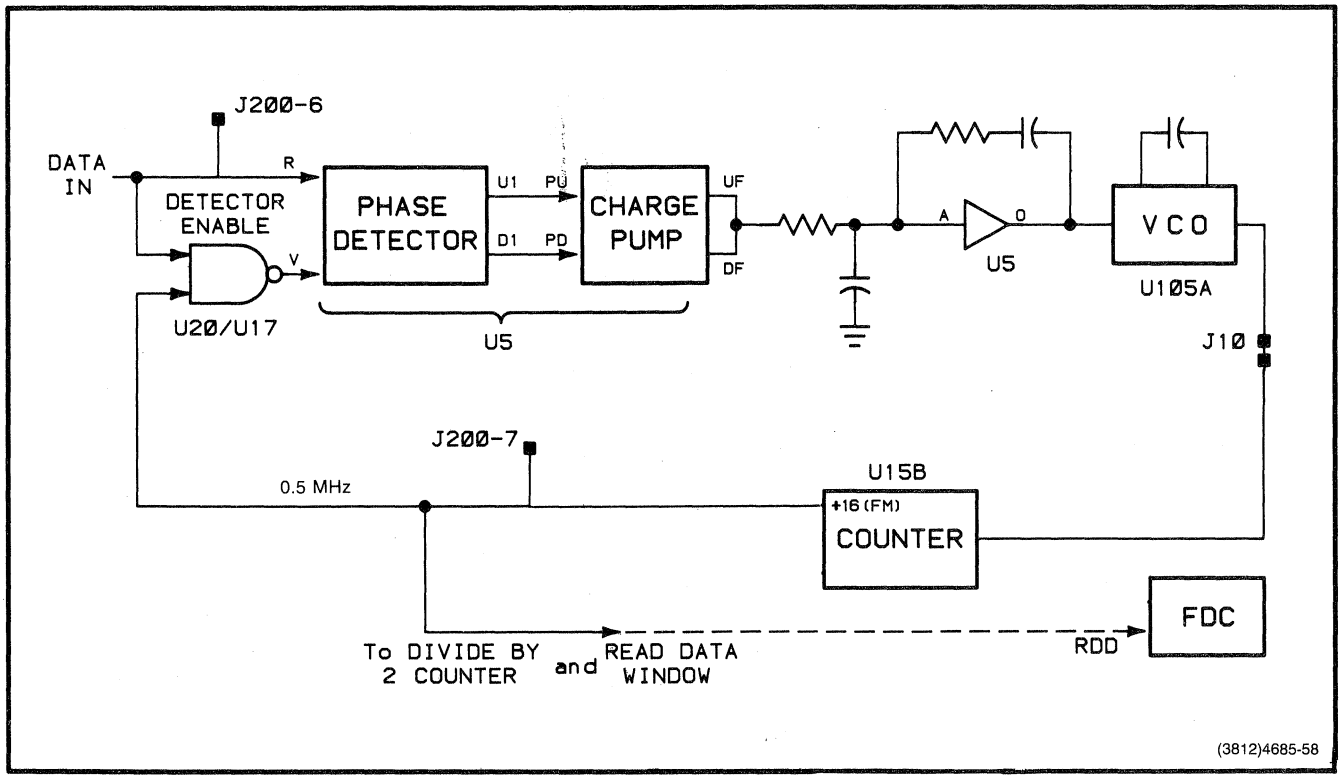


Figure 12-41. Phase-Locked Loop.

The Detector Enable circuit is comprised of a flip-flop array. Ordinarily, the VCO/counter is feeding constant 0.5 MHz pulses into the feedback input on the phase comparator. At the same time data stream pulses, at approximately 250 KHz, are entering the main input of the Phase Detector. Furthermore, there are times when this MFM data stream goes several feedback pulses without producing a clock or data pulse. This means that the feedback VCO pulses are not always accompanied by a data/clock pulse; the phase-locked loop misreads this as a drastic change in input frequency. Consequently, the VCO tries to shift its output accordingly, producing a large and unnecessary error correction.

The Detector Enable circuit solves this problem by interrupting the feedback VCO, unless there is a data pulse present for a valid comparison. The circuit, comprised of flip-flops and inverters, acts basically like a NAND gate. The purpose of the inverters is to provide the necessary pulse width for the flip-flops.

After this timing signal has made a complete loop through the PLL, it is stabilized and extracted just prior to the Detector Enable. This signal then passes through one more counter (a divide-by-two) before entering the FDC as RDW (Read Data Window). The Read Data Window line provides the timing window for data/clock separation.

Disk Drive Control

The Disk Drive Control circuitry (Schematic A3-4) buffers and strobes the control lines going between the FDC and the Disk Drive Unit(s)(Figure 12-42).

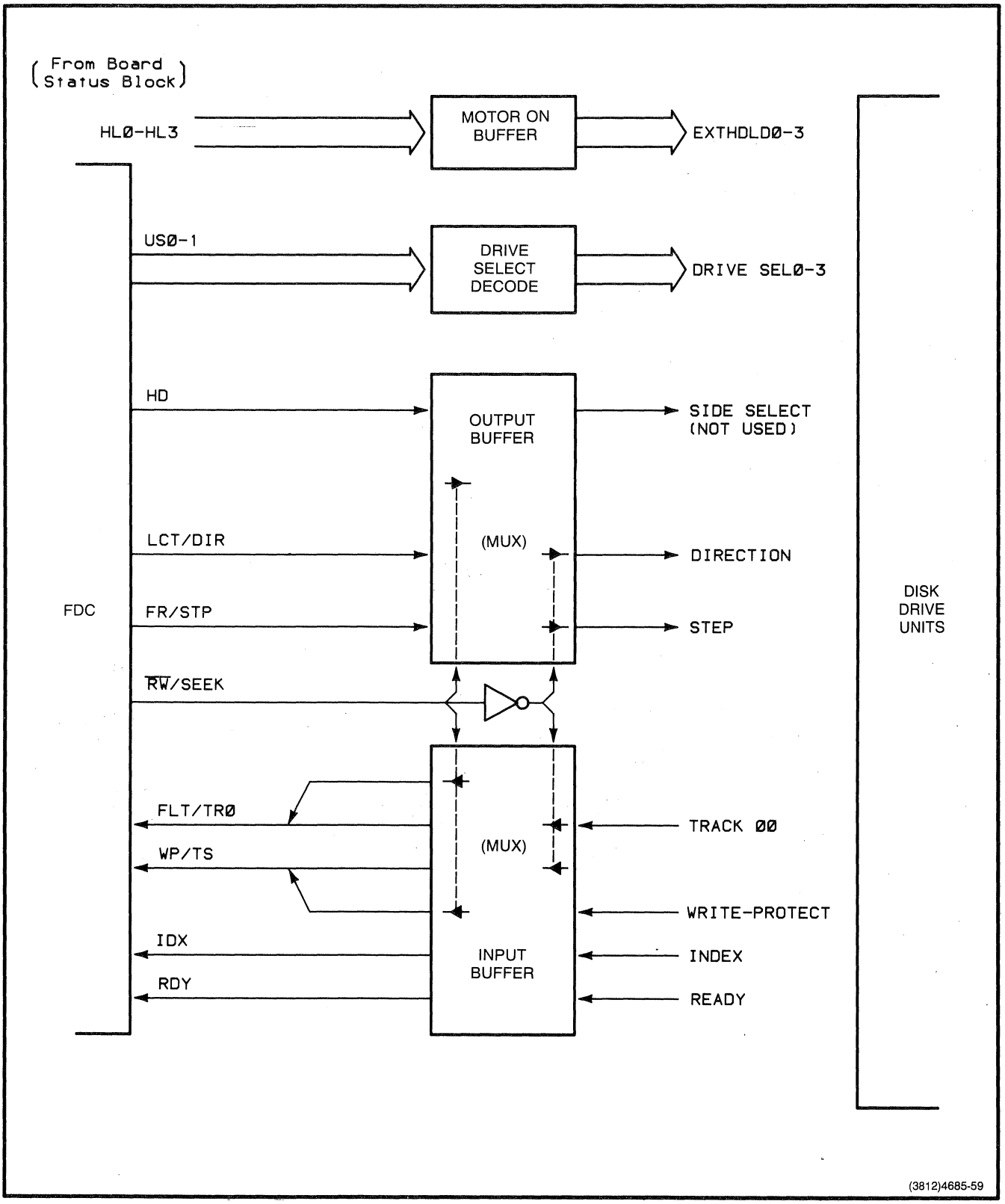


Figure 12-42. Disk Drive Control Block.

The control lines are:

- o Motor On (managed by the processor instead of the FDC)
- o Unit Select
- o Direction (of the step, toward or away from the center)
- o Step (from one track to the next)
- o Side Select

The response or feedback lines from the drive unit(s) are:

- o Track 00 (location)
- o Write-Protect
- o Index (marker response)

The four Motor On lines (Figure 12-42) come from the Board Status Register, are buffered, and go to the drive units.

The drive units are selected by the FDC via its Unit Select lines. The 2-to-4 decoder makes US0 and US1 into DRIVSEL-0 through DRIVSEL-3.

The remaining lines control and monitor the operation of the drive units. These lines pass through buffers where multiplexing also takes place. The RW/SEEK line and its complement alternately strobe two sets of buffered control lines. The Side Select line is only used with two-sided drive units. The Direction and Step lines are multiplexed together.

Of the five monitor lines (from the disk), two are inverted but not multiplexed: Index and Ready. The Write-Protect line is multiplexed by RW, and the Track 00 and Two-Sided lines are multiplexed by SEEK. Multiplexing these lines is necessary because the FDC does not have enough input pins.

Terminal Bus Interface

The following logic controls handle the flow of data, addresses, DMA control, FDC control, and disk control signals between the 4170 bus and the Disk Controller board:

- Data MUX (Schematic A3-2)
- Input Data Buffers (Schematic A3-3)
- Address Counters (Schematic A3-3)
- Address Decoder (Schematic A3-3)
- Board Status (Schematic A3-3)
- DMA State Machine (Schematic A3-2)
- Control Strobes (Schematics A3-2, -3)

Data MUX. The Data MUX (Multiplexer)(Schematic A3-2) takes the 16-bit data/instructions from the 4170 bus and converts them to 8-bit bytes for the FDC internal data bus and registers.

Two transceivers function as a tri-state buffer between the 4170 bus and the FDC data bus. To accomplish the 16-to-8 bit multiplexing, the decoder acts as a switch which enables either one or the other transceiver. BHEN on the decoder A input becomes output Y2 (to the high byte transceiver), and A0 on the B input becomes Y1. Either a Read (RD-0) or Write (WR-0) input to the FDC strobes this decoder/switch.

Input Data Buffer. The Input Data Buffers (Schematic A3-3) take processor data from the 4170 data bus and store it until it is loaded into the Address Counter Register and Board Status Register.

The sixteen data lines are buffered by three tri-state buffer/driver ICs. This circuit appears in the upper-left corner of the Disk Controller board block diagram (Figure 12-33).

Address Counters. The Address Counters (Schematic A3-3) transforms DMA addresses sent by the processor from 16-bit data bus format into 20-bit address bus format.

Figure 12-43 shows how the DMA address is first loaded into a set of counters. A 16-bit byte fills all but the top counter, so the next address byte only contains four bits and it is loaded into the top counter. The Address Decoder circuitry selectively enables these counters to achieve this.

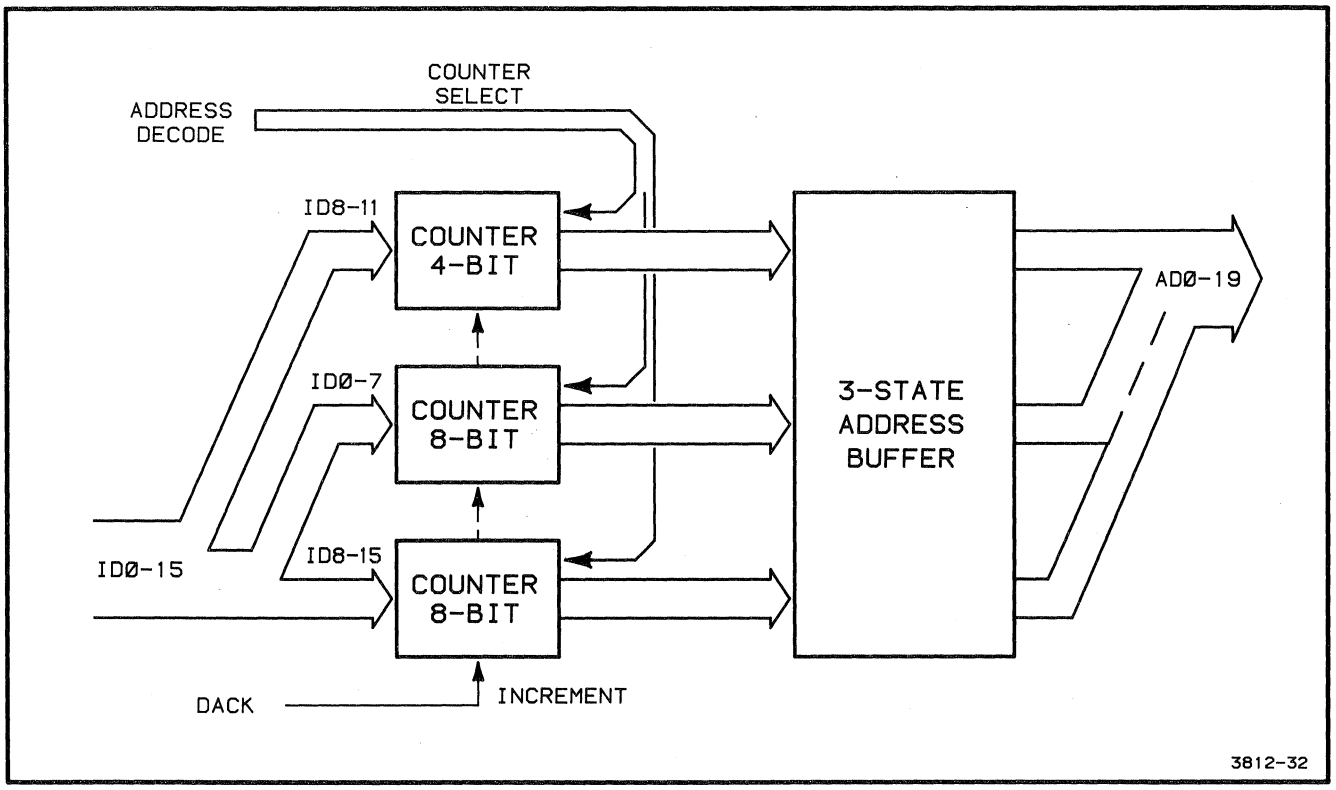


Figure 12-43. Address Counters.

This 20-bit address is then loaded into the tri-state buffers, which are read by the 4170 address bus at the time of the next DMA transfer. The address remains in the counters until the next DMA cycle, at which time the address is automatically incremented by the DACK line.

This means that the processor establishes the starting location in RAM for DMA transfers; then subsequent transfers automatically step through adjacent addresses in RAM.

Address Decoder. The Address Decoder (Schematic A3-3) generates the control signals that enable other circuit functions. The Address Decoder takes addresses in the processor Memory Address Space and converts them into:

- o ACK1 acknowledge to 4170 control bus
- o FDCR to FDC RD and WR inputs
- o FDCW to FDC RD and WR inputs
- o STROBE to Board Status output buffer
- o STATUS LATCH clock to the Board Status input latch
- o ADDRESS COUNTERS selects and enables to the three Address Counters
- o ABORT to FDC TC (abort) input
- o SSDRQ to DMA request (test only)

Figure 12-44 shows the functional groupings of this logic. An address of FC00 (1) passes the NAND circuit and produces its COMMON ENABLE output. This line is used along with specific addresses to enable the following function gates.

(1) May be strapped to FC80, but this strap setting is not normally used.

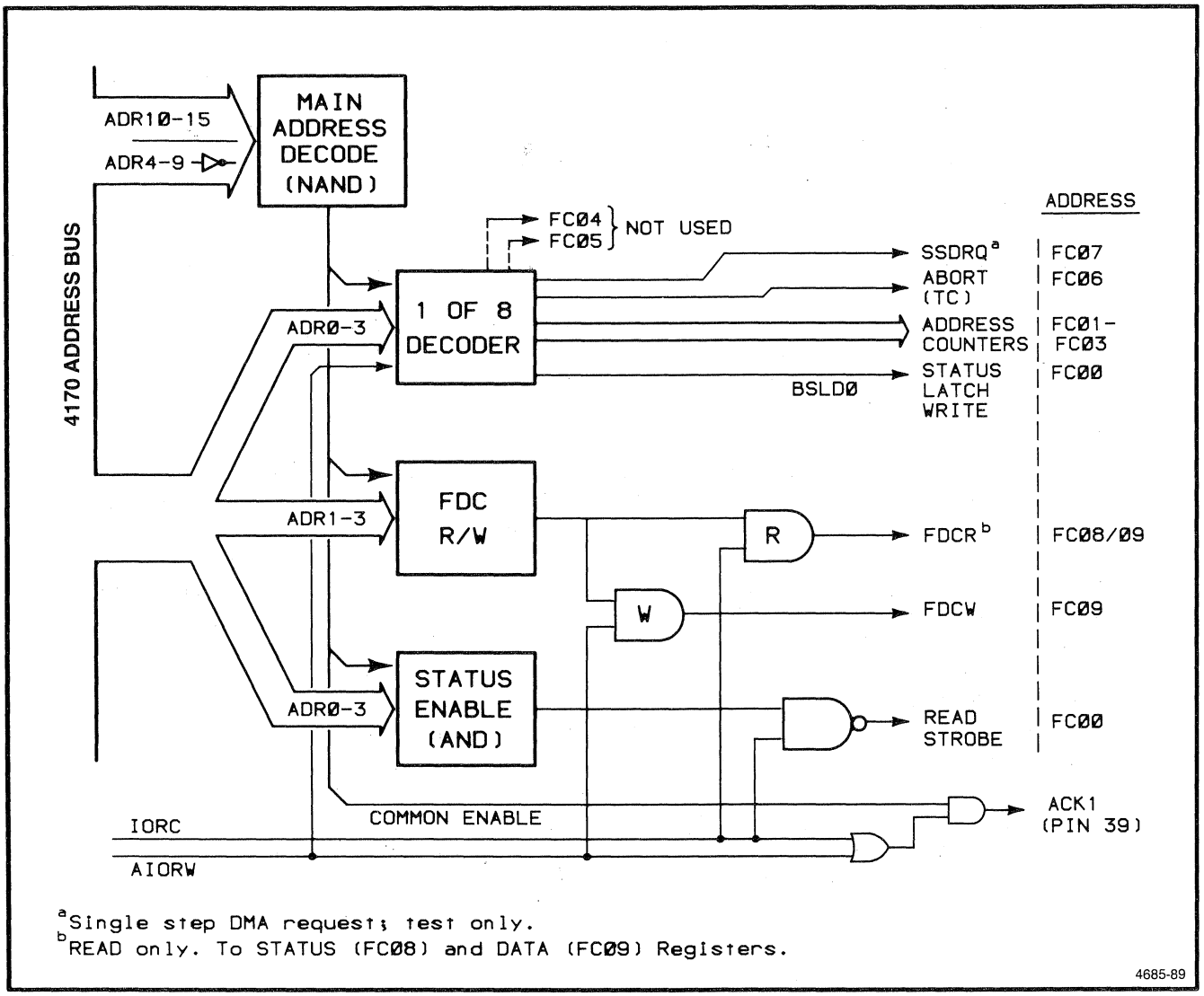


Figure 12-44. Address Decode Block Diagram.

The first is actually a 3-to-6 decoder. Addresses on the three input lines select the following outputs:

- o ADDRESS COUNTERS (3 lines)
- o STATUS LATCH WRITE
- o ABORT
- o SSDRQ

The second set of gates produces the FDC read and write line FDCW/R. This line is ANDed with both IORC (I/O Read Command) and AIOWC (Advanced I/O Write Command) to produce FDCR and FDCW, respectively. These lines become the FDC RD and WR inputs (after passing through Control Strobe switches).

The Status Enable function generates the strobe/gate that enables the Board Status output buffer. When address A0 - A3 equals 0000, it passes the five-input AND gate, whose output is then NANDed with the IORC line to finally generate STROBE.

THE ACK1 is generated by ORing IORC and AIOWC and ANDing this with the COMMON ENABLE. This acknowledge is placed on the 4170 control bus.

Board Status. The Board Status register (Schematic A3-3) stores board status information and head load status, which may be read by the processor. This information, combined with EOC (interrupt from the FDC), is used to generate the 4170 bus INT signal. One of the status lines is also separated out and called BW/BR, which is used to toggle the Control Strobe switches. See Table 12-11 for a description of the Board Status byte stored here.

Table 12-11

BOARD STATUS BYTE

Bit	Mnemonic	Type	Description
D7	EOC-O	Read Only	Active low signal which indicates the end of the execution phase of a FDC command.
D6	INTE-O	Read/Write	Active low signal which enables interrupts. This bit should not be set until the "specify" command has been issued to the FDC. This bit is set to one by a bus reset.
D5	BUSW-O	Read/Write	This signal specifies the direction of the flexible disk data. The bit should be set to a one for disk writes (bus reads) and a zero for disk reads (bus writes). This bit is set to one by a bus reset.
D4	ADR19-1	Read Only	This is the DMA address counter MSB. It is used for self-test purposes only.
D3	Not Used	---	---
D2	HDL2-O	Read/Write	Active low signal which turns on the motor of Disk Drive 2 and enables power to the step motor. The processor must provide the head settling delay (15 ms). The head settling delay must precede any disk read or write operation after a seek. This bit is set to one by a bus reset.

Table 12-11 (cont)

BOARD STATUS BYTE

Bit	Mnemonic	Type	Description
D1	HDL1-O	Read/Write	Active low signal which turns on the motor of Disk Drive 1 and enables power to the step motor. See the description of D2 (HDL2).
DO	HDLO-O	Read/Write	Active low signal which turns on the motor of Disk Drive O and enables power to the step motor. See the descriptions of D1 and D2

This logic consists mainly of an input latch/register and an output tri-state buffer. The processor places board and head load status bytes on IDO to ID6. The byte is written into the latch which is strobed by the Address Counter YO output. The processor can then read this status by placing a particular address, along with IORC, into the Address Decoder. The decoder ANDs these conditions, yielding the output buffer strobe signal.

HLO to HL3 are stored in this status buffer and are also sent to a Disk Drive Control buffer.

The EOC status is read by the status buffer and is ANDed with another status line which allows the INT to be turned on and off during testing.

The ADR19 line, (from the Address Decoder) which enters the status buffer as input 1A4, is used for self-test only.

DMA State Machine. The DMA State Machine (Schematic A3-2) takes a combination of input conditions and generates the resulting state/outputs; a DMA request will set up supporting circuits to produce these conditional outputs.

This circuit, represented in Figure 12-45, is based on a ROM which performs certain combinational logic functions. The inputs to the state machine are:

- o DRQ (DMA request)
- o BUSY (bus busy)
- o BPRN (bus priority in)
- o ACK1 (acknowledge 1)
- o ACK2 (acknowledge 2)
- o BW/BR (bus write or read)

The outputs from this circuit are:

1. To the 4170 control bus --
    - o BREQ (bus request)
    - o CBRQ (common bus request, for lower priority attention)
    - o BUSY (assert bus busy)
2. To the Address Counter and FDC --
    - o DACK (DMA acknowledge)
3. To toggle the Control Strobe switches --
    - o DMAR (DMA read)
    - o DMAW (DMA write)

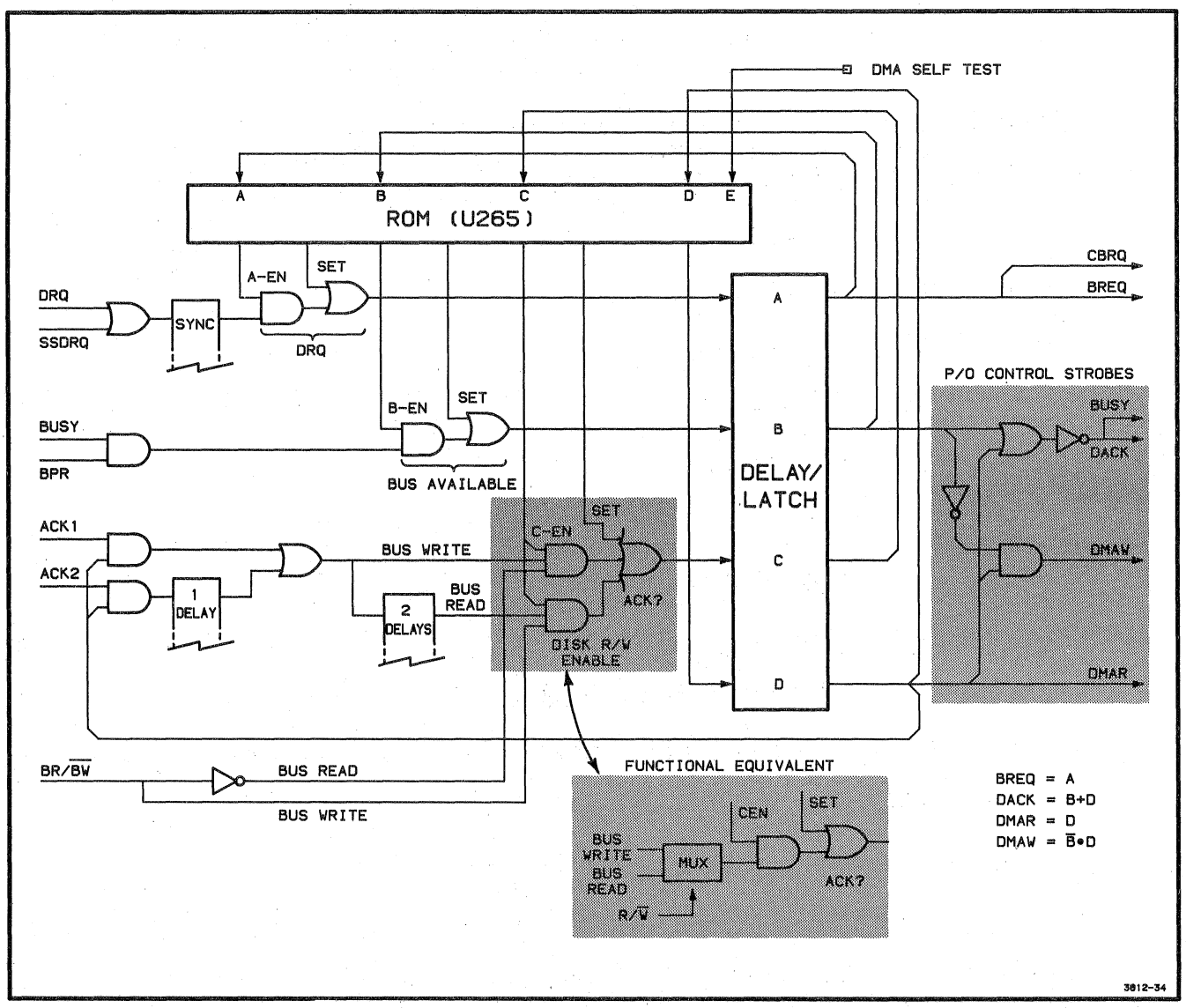


Figure 12-45. DMA State Machine.

The SSDRQ input line is only used for testing. This "single step DRQ" is an alternate DRQ input, allowing you to manually assert DRQ instead of waiting for the FDC to process a DMA request under firmware control.

Outside the ROM are three groups of signal lines that are controlled by gates and the ROM. First is the DMA request group. A DRQ is sent through a set of flip-flops that has the effect of synchronizing the 8-MHz frequency from which DRQ is derived with the 5-MHz frequency on which Bus Request controlled by the ROM. The ROM can also override and set the DRQ path (when no DRQ or SSDRQ is present) via the Rom A-set line on the OR gate. The ROM program determines the conditions when this override takes place. The flowcharts in Figures 12-46 and 12-47 show how the State Machine responds to input conditions by producing the corresponding output states.

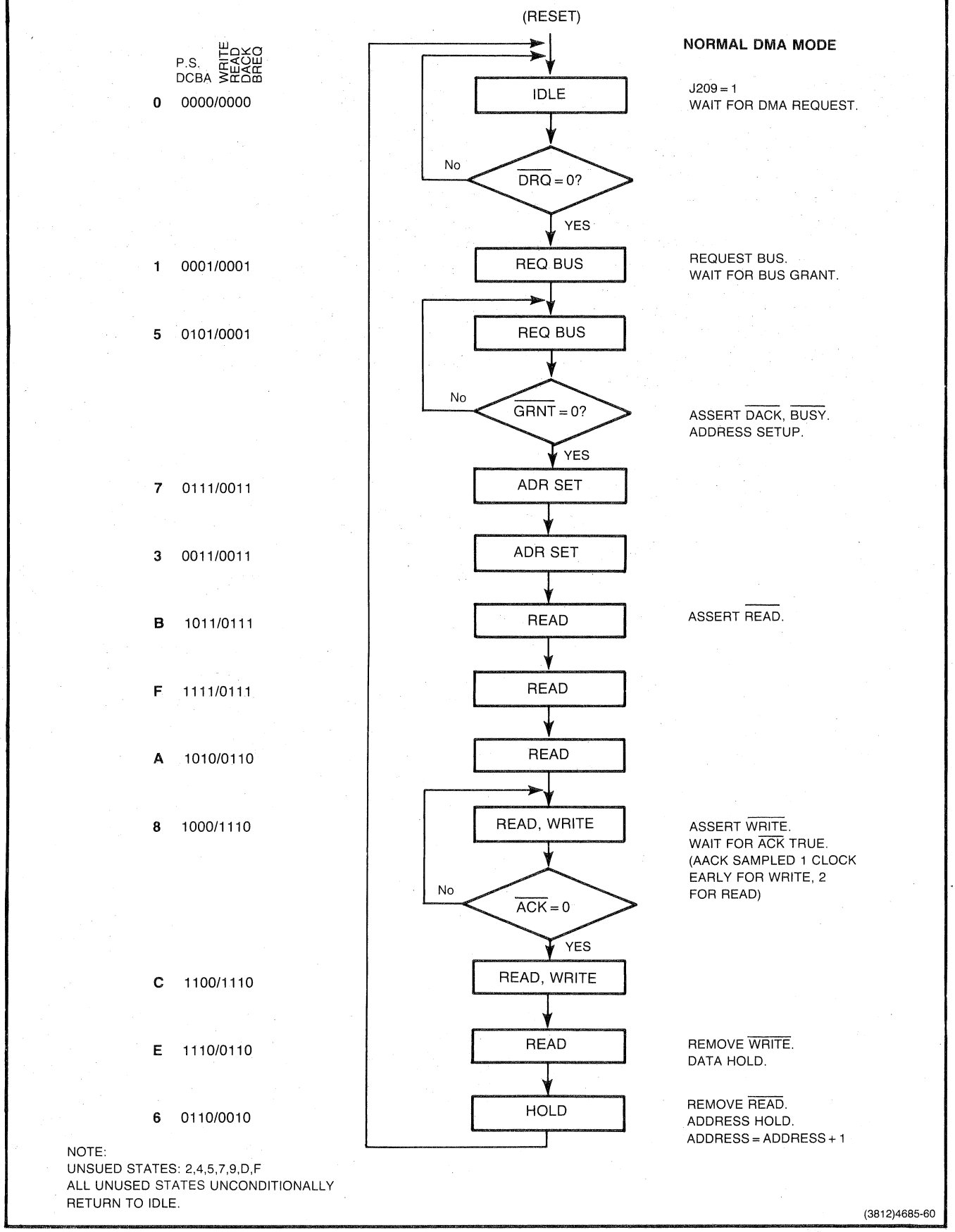


Figure 12-46. DMA State Diagram (Normal DMA Mode).

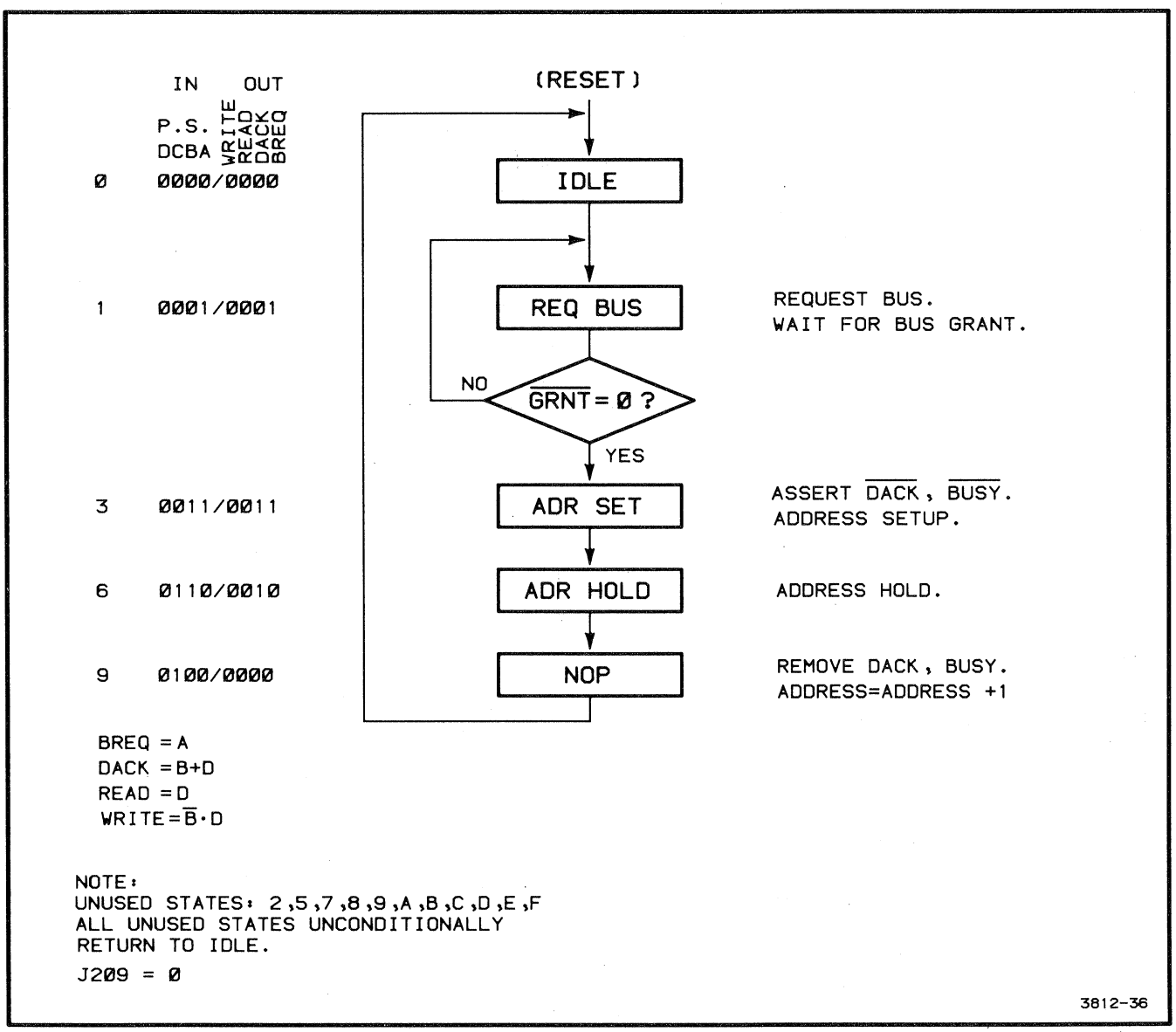


Figure 12-47. DMA State Diagram (Self-Test Mode).

The BUSY/BPR (bus available) and ACK1/ACK2 functions are handled similarly by ROM lines interacting with a set of AND and OR gates. The outputs are sent through a set of flip-flops (labeled A, B, C, D in Figure 12-45) which provide delays to synchronize the inputs back into the ROM.

The ACK1 or ACK2 circuit is first enabled by a DMA output from the ROM. After these AND gates, the ACK2 line is sent through a delay stage (one flip-flop) so it will be in phase with ACK1. These two ACKs are then ORed and split. One line of the split pair passes through an AND enable that looks for the BW/BR condition and the C-EN (C enable) from the ROM. The two delays in the WRITE GATE line keep the disk write function from being out of phase with the disk read function (with which they are timed). Finally, the two ACKs may be overridden by a "set" from the ROM on the last OR gate. The C-latch output only returns to the ROM to provide state information or feedback.

A DMA read or a busy produces a DACK/BUSY output. On the other hand, an inverted DACK is ANDed with the DMA READ line to produce the DMA WRITE output. Both of these (DMA READ/WRITE) set the bus and FDC to the proper read and write states. See the Control Strobes circuit description, next, for detailed explanation of what these outputs do.

Control Strobes. The Control Strobes circuitry (Schematics A3-2, -3) functions as a "steering device" or "switching network" to point the DMA control signals in the proper direction to or from the 4170 bus, Address Decoder, or FDC.

Figure 12-48 shows an equivalent circuit for the Control Strobes logic. The figure presents this combination of gates as two pairs of ganged switches. The left pair of switches (R/W Direction) directs the DMA READ and DMA WRITE control lines to either the 4170 bus or the FDC. In the case of a write to the disk, these switches connect the DMA READ line to MRDC (Figure 12-48A). The MRDC (Memory Read Command) is the control line on the 4170 bus that tells the RAM that a bus master (in this case, the Disk Controller) wants to read from its memory. The DMA WRITE line connects to the FDC WR input. This allows the DMA State Machine to initiate a write operation in the FDC. The right switch (DMA/Bus Listen) is in the DMA position.

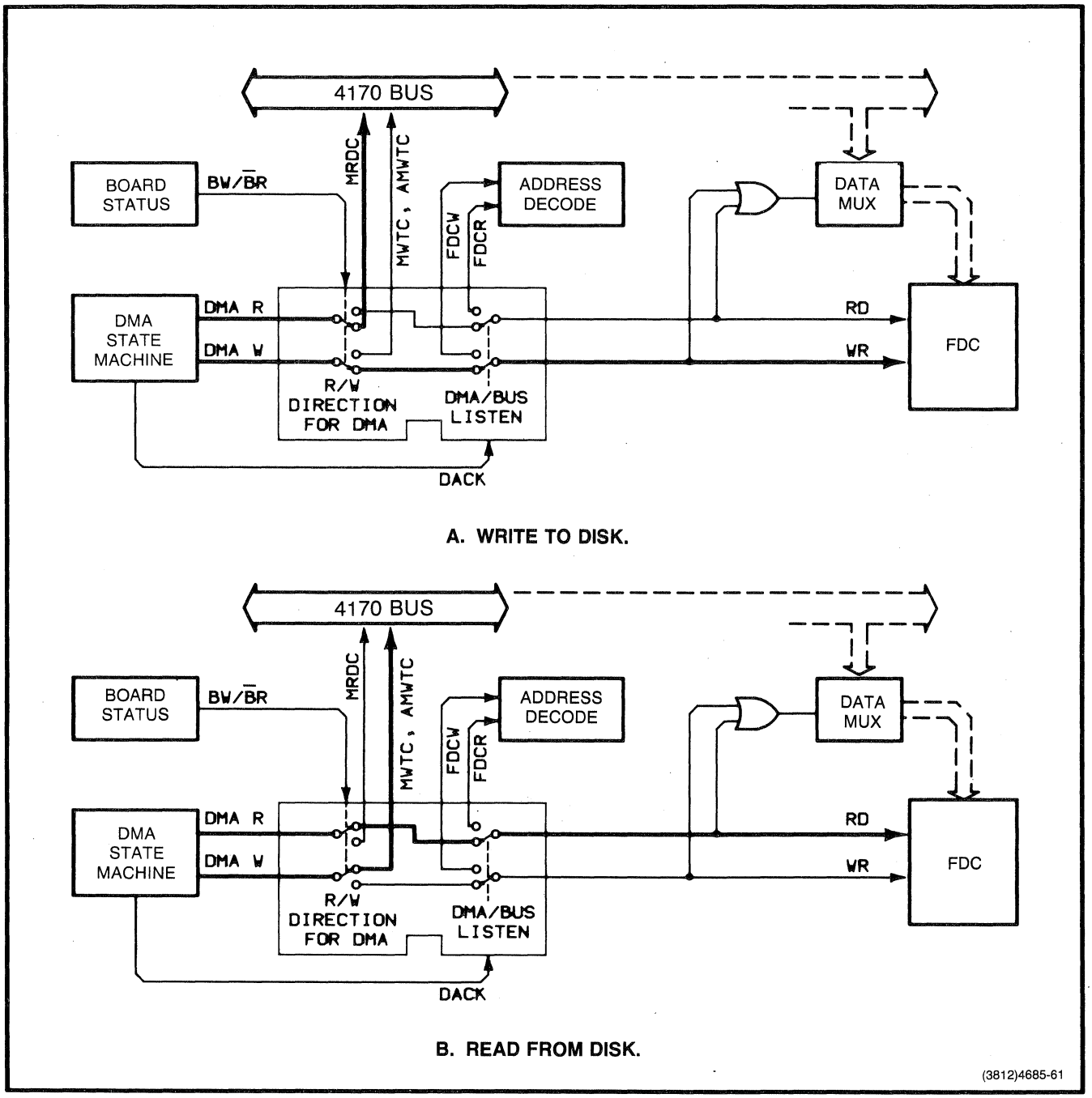


Figure 12-48. Control Strobes Equivalent Circuit.

For a read from the disk, the DMA/Bus Listen switch remains in the DMA position, but the other switch is toggled so it connects the DMA READ line to the FDC RD input. The DMA Write part of the switch is connected to the MWTC (Memory Write Command) line on the 4170 bus (Figure 12-48B). This allows the FDC to read from the disk and to write the data into the DMA area of system RAM.

The BW/BR line controls the DMA Read and Write Direction switches, and the DACK line controls the DMA/Bus Listen switches.

NOTE

The gates that make the DMA READ and DMA WRITE lines from DMA State Machine outputs are shown as a part of Figure 12-45.

ROM (Not Used)

The several blocks of circuitry which together comprise the ROM block (Schematic A3-1) are pictured in Figure 12-49.

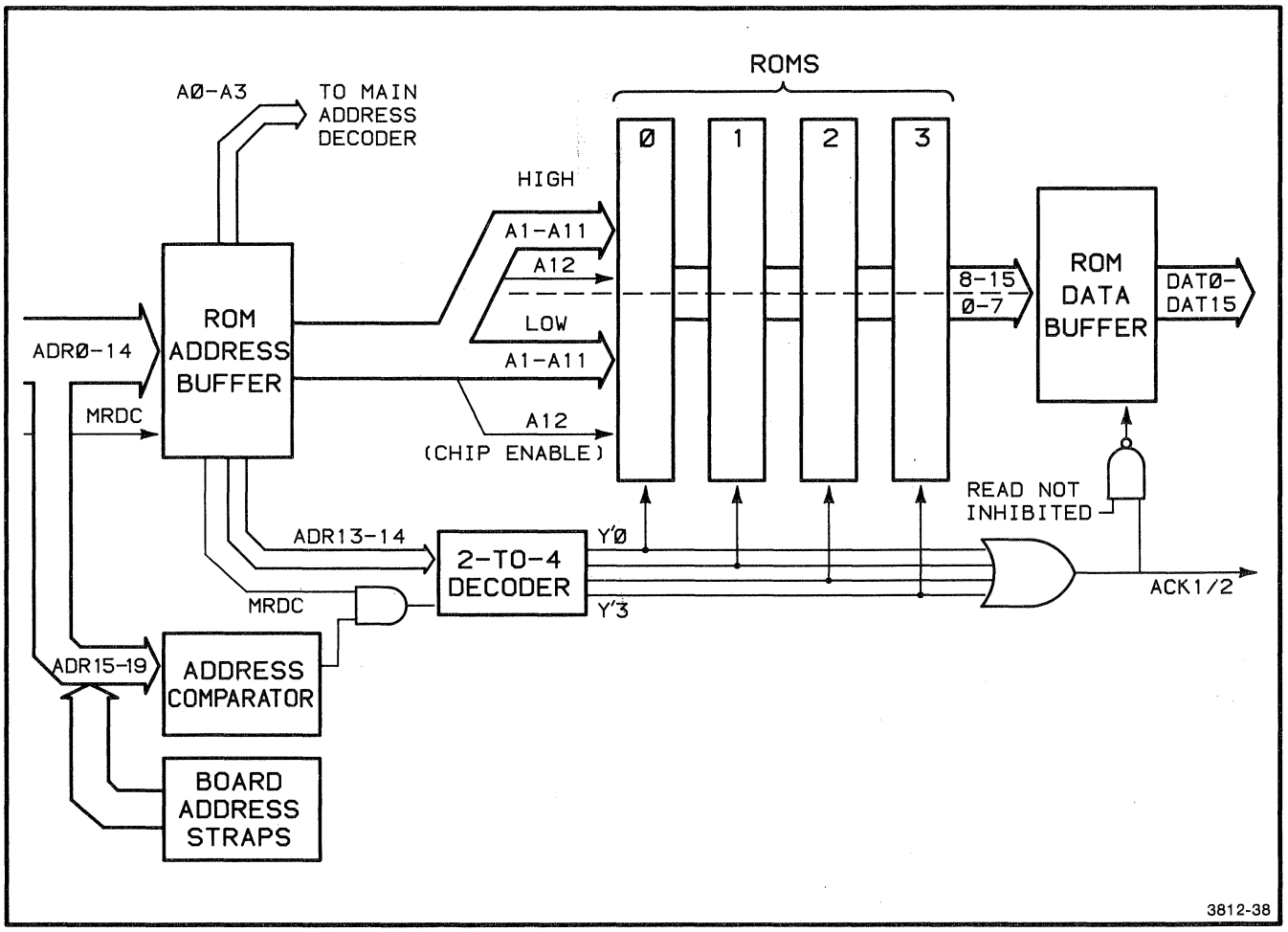


Figure 12-49. ROM Circuitry.

This section of circuitry can contain ROMs with 32K of firmware for FDC operations. This firmware is located in processor Memory Address Space between E8000 and EFFFF. The ROMs are accessible via the 4170 address bus (ADR0 - ADR19) and the MRDC (Memory Read Command) control line.

OPTION 45 MSIB CONTROLLER

The Option 45 Mass Storage Interface board (MSIB) is an interface that connects the 4170 to an internal Winchester hard disk unit or external mass storage devices on the mass storage interface bus.

Option 45 includes:

- Option 45 circuit board assembly
- 44-pin auxiliary edge connector and cable with MSIB connector

The MSIB board plugs into the 4170 main bus. For further information, refer to the 4110B Series Disk Options Service Manual.

HARD DISK CONTROLLER

The Hard Disk Controller contains the following functional blocks:

- Host Interface. The Host Interface connects the Hard Disk Controller internal data bus to the MSIB. The movement of data through the Host Interface is controlled by the State Machine.

- Processor. All functions within the Hard Disk Controller are under the general control of an 8-bit microprocessor.

- State Machine. The State Machine synchronizes the operation of the Host Interface, the Serializer/Deserializer, and the Sector Buffer.

- Serializer/Deserializer. The Serializer/Deserializer converts parallel data coming over the internal data bus to a NRZ (Non-Return to Zero) serial data stream suitable for the Data Separator. It converts serial data coming from the Data Separator to parallel format for transfer over the internal data bus.

- Data Separator. The Data Separator converts the serial NRZ data stream coming from the Serializer/Deserializer to serial MFM (Modified Frequency Modulation) encoded data suitable for the Hard Disk Drive. The Data Separator also converts MFM data coming back from the drive to NRZ format.

- The Sector Buffer. The Sector Buffer temporarily holds data during transfers between the disk drive and the host; its function is to prevent overrunning the host or the drive.

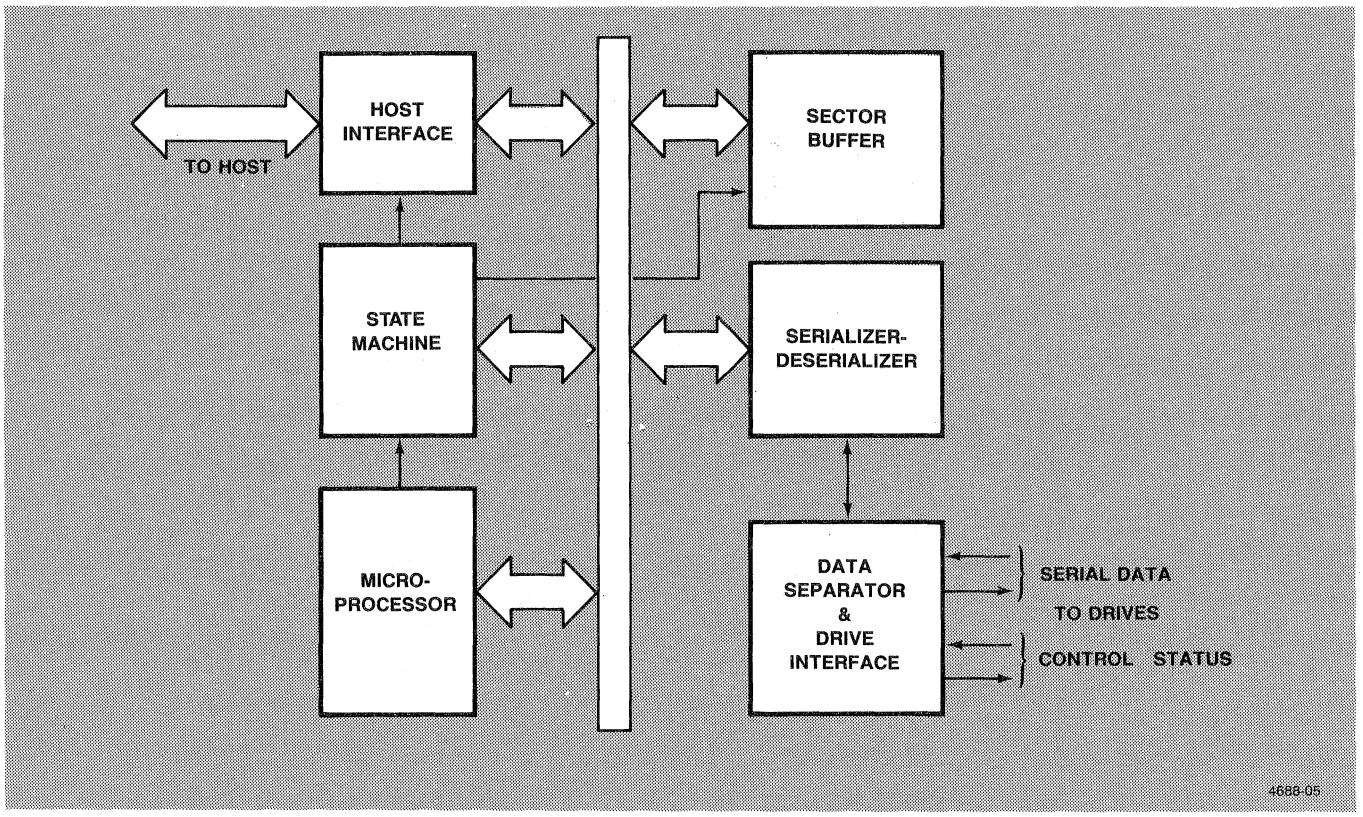


Figure 12-50. Hard Disk Controller Block Diagram.

Figures 12-51, 12-52, and 12-53 show MSIB signal timing for the Hard Disk Controller.

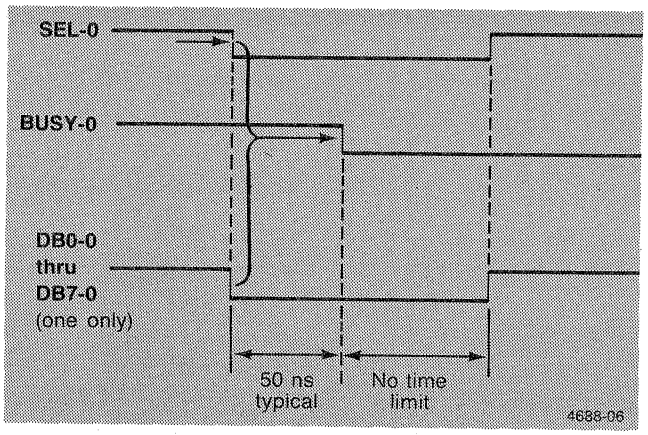


Figure 12-51. Hard Disk Controller Select Timing.

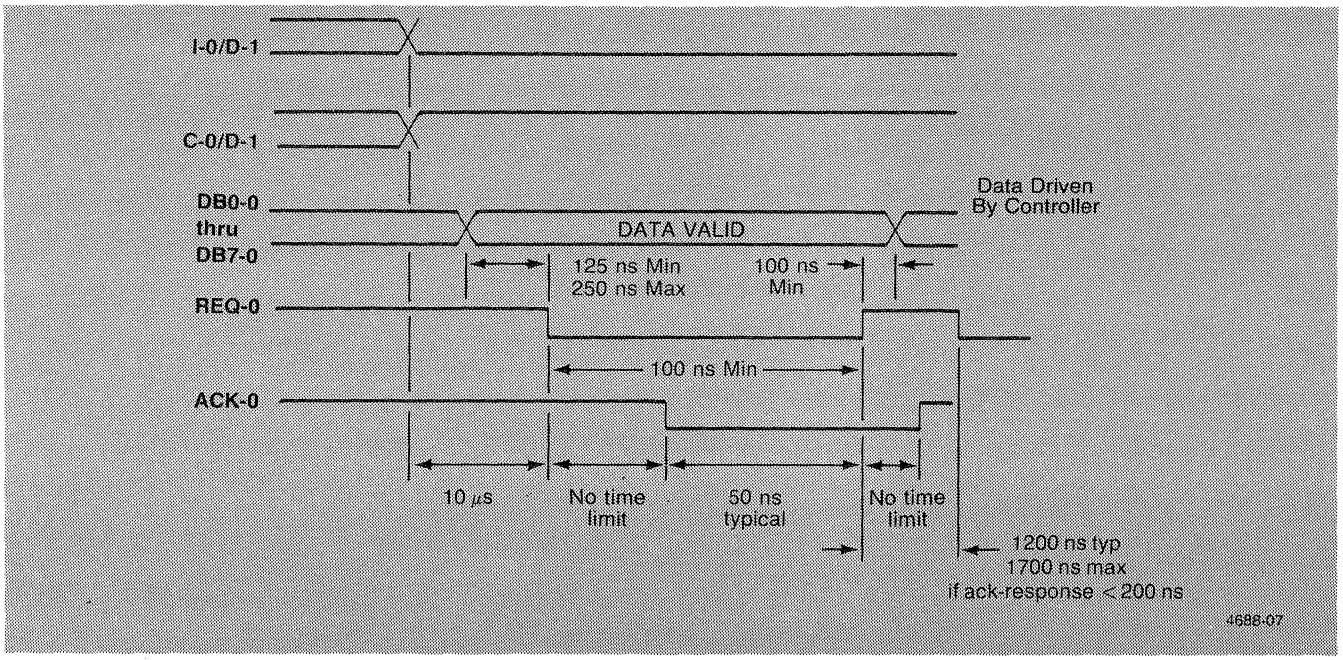


Figure 12-52. Hard Disk Controller Data Output Timing.

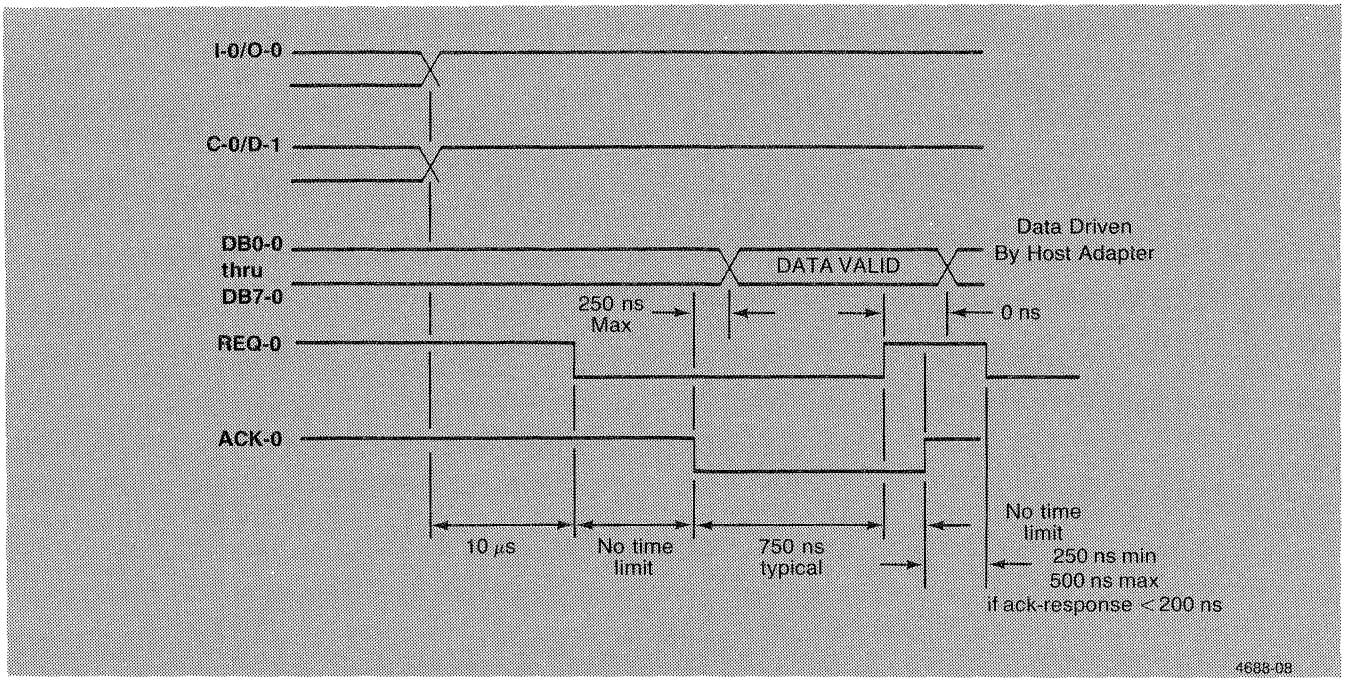


Figure 12-53. Hard Disk Controller Data Input Timing.

For further information, refer to the 4110B Series Disk Options Service Manual and the 119-1644-00 Hard Disk Drive Service Manual.

OPTION 9 COLOR COPIER INTERFACE

The Option 9 Interface allows a 4690 Series Color Graphics Copier to be connected to the 4170. The interface consists of:

- A circuit board with ROM firmware, located in the 4170 card cage.
- A connector panel holding one female 36-pin connector. This panel is mounted on the rear of the 4170 and is connected to the interface board by a cable and a 44-pin edge connector.

For further information, refer to the 4110F09 Interface for 4690 Series Color Graphics Copiers Service Manual.

FRONT PANEL

The Front Panel board consists of two major sections: the Display Section and the Keys Circuitry. Other portions of the board include a tone generator (bell), write-protect switches, board reset circuitry, and the system RESET and SELF-TEST buttons.

Refer to Figure 12-54 while reading the circuit descriptions.

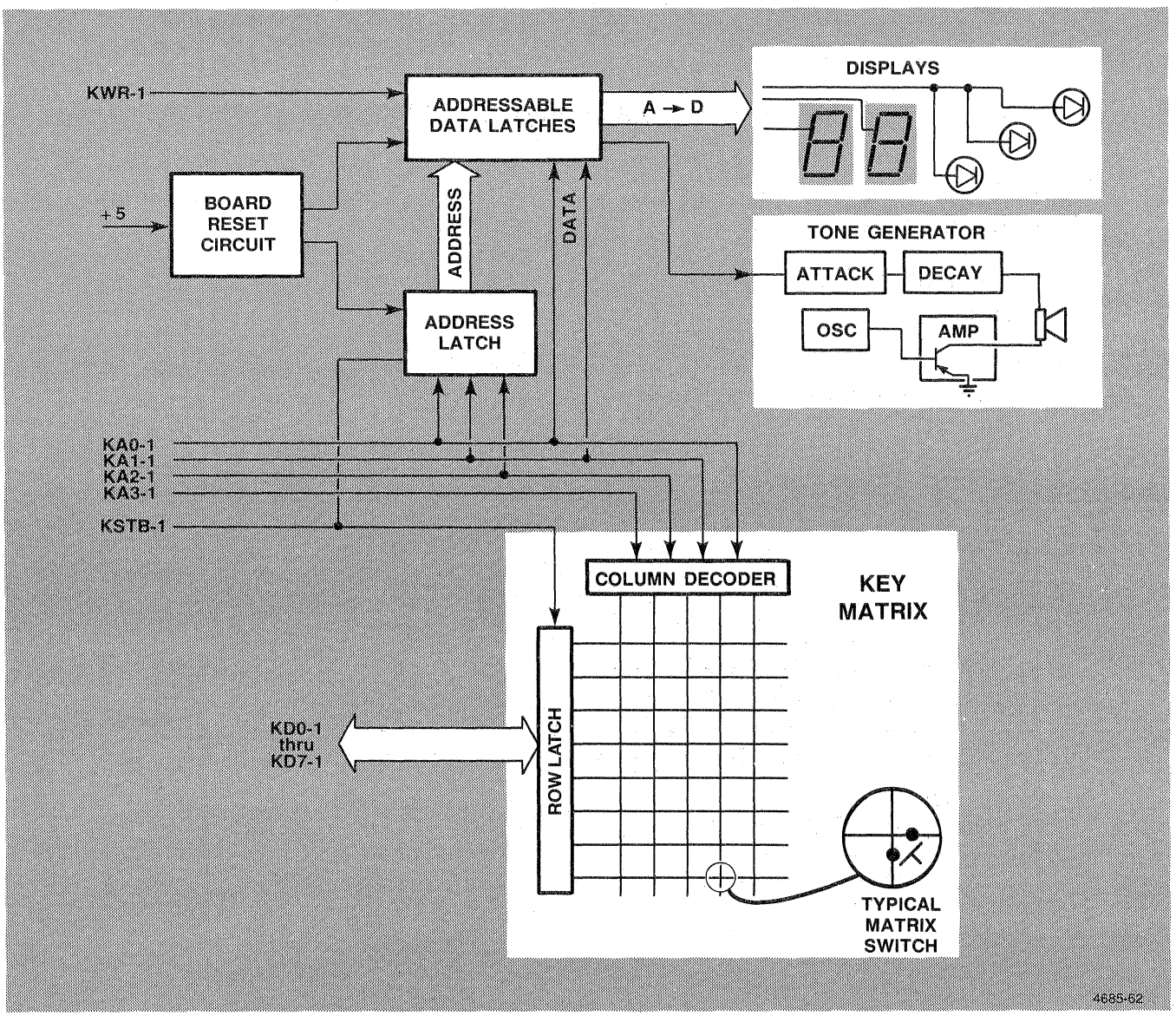


Figure 12-54. Front Panel Block Diagram.

KEYS CIRCUITRY

The Key Matrix (Schematic A4-1) is essentially the same as a computer display terminal keyboard, but with limited functions. The Front Panel Controller Microcomputer (FPC MPU) on the Processor board monitors and controls the Front Panel. To read the Key Matrix, the FPC MPU places the column address of the Key Matrix onto the KA0-1 -- KA3-1 lines. The column decoder uses these signals as inputs and energizes the appropriate column in the Key Matrix.

After the column address is established, the FPC MPU asserts KSTB-0. KSTB-0 is buffered, and, when asserted, gates the row latch. The FPC MPU then reads the key data from KD0-1 -- KD7-1 to determine if any key, corresponding to the selected column, has been pressed.

DISPLAY CIRCUITRY

The Display circuitry (Schematic A4-1) consists of two hexadecimal displays, three LEDs, and the addressable data latches that drive them. The eight flip/flops which make up the data latches are selected by an address asserted onto the latch inputs. Data is clocked into the latches by the KWR-0 signal. Signal lines KA0-1 -- KA2-1 carry both address information and data for the Display circuitry. A separate latch, U620, stores the address information until needed.

The FPC MPU places the address of the selected flip/flops onto KA0-1 -- KA2-1, then asserts KSTB-0, which enables the address latches. The FPC MPU then places data on KA0-1 and KA1-1. KWR-0 clocks the data into the appropriate flip/flops.

RESET CIRCUITRY

The Addressable Data Latches are reset upon power-up by the Reset circuitry (Schematic A4-1). The Reset circuitry provides a delay of approximately 100 ms after power-up before initializing the latches.

TONE GENERATOR

One of the Addressable Data Latches (U630) also drives the Tone Generator (Schematic A4-1). A 555-type oscillator generates the bell tone, which is amplified and sent to the speaker. Two transistors (Q130 and Q135) provide attack and decay control for the tone generator.

POWER SUPPLY MODULE

The Power Supply Module (Schematic A5-1) supplies power to all boards in the 4170. Table 12-12 gives the outputs at the socket.

Table 12-12

PIN ASSIGNMENTS

Pins	Signal
1	12 V
2-6	GROUND
7	GNDDR(+12V Return)
8	GND SENSE
9	RESTART
10	key
11-15	+5 VOLTS
16	+5 SENSE
17	+12 VDR (to disk drives)
18	+12 VOLTS

OVERVIEW

The Power Supply Module is based on a discontinuous mode, fly-back, high-efficiency, switching type design. This supply has the advantages of lower weight and smaller volume over conventional supplies.

The Power Supply Module consists of the following functional blocks:

- Line Input and Filter
- Line Select
- Rectifier and Filter
- Main Power Converter
- Snubber
- Control Loop Sense and Drive
- Over-Voltage and Protect
- Output

This section discusses each of these blocks.

DETAILED DESCRIPTIONS OF CIRCUIT BLOCKS

During the following discussion, refer to the Power Supply Module Schematic A5-1 and Figure 12-55. The block diagram shows how each block relates to the others while the schematics show how each block functions.

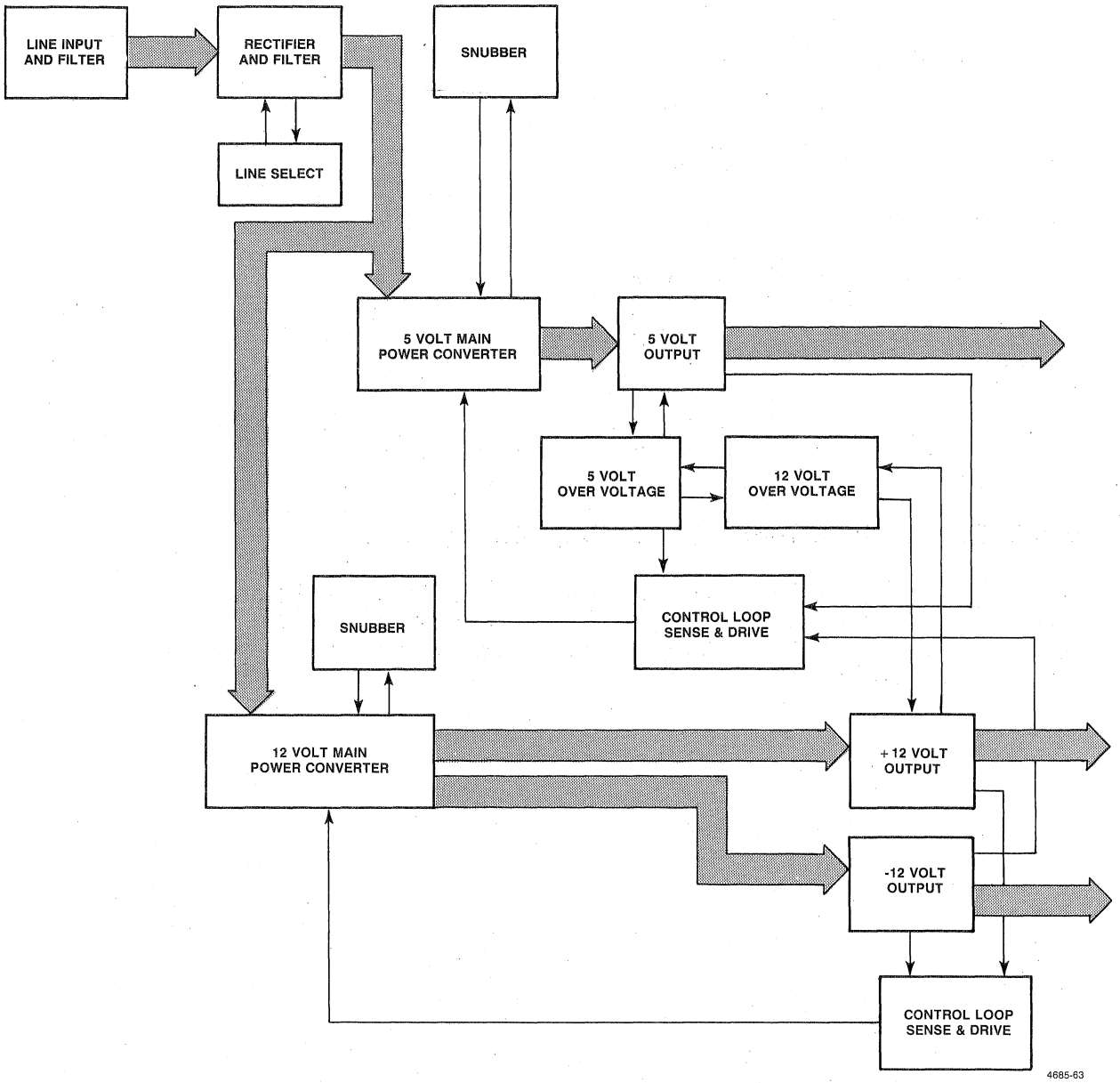


Figure 12-55. Power Supply Block Diagram.

Line Input and Filter

The Line Input and Filter consists of the ac plug, power switch, fuse, capacitor (high frequency line filter), thermal resistor, transient suppressor, and the EMI filtering section.

The EMI filtering section is used to keep high frequency noise (kHz range and up) created by the switching of the supply from getting back on line. The filtering, a common mode transformer and capacitor, does not affect the line voltage. When a high frequency signal attempts to enter the EMI transformer, that signal meets a high impedance. The signal then takes the alternate path through the capacitor (also tied across the line). The capacitor presents a low impedance at this frequency.

A 5-amp fuse protects this power supply. The fuse, not externally accessible, is mounted on the power supply circuit board.

The transient suppressor absorbs spikes on the ac line.

Line Select

The Line Select block consists of a switch and a capacitor, wired in parallel. The setting of this switch depends on whether the line voltage is 115 or 230 volts.

Rectifier and Filter

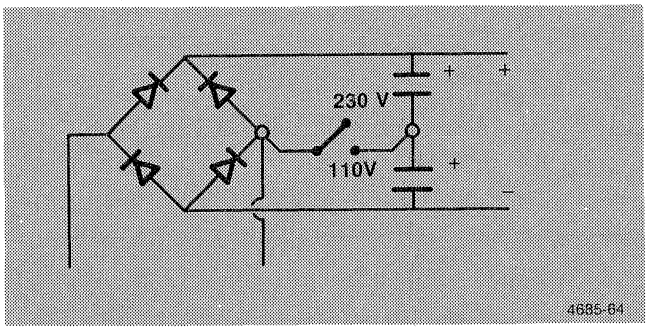


Figure 12-56. Rectifier and Filter

The ac inputs are two ranges: approximately 115 VAC OR 230 VAC. The lower range requires the line select switch to be in the closed (115V) position. This puts the power supply in the voltage doubler configuration. The higher range requires the line select switch to be in the open (230V) position. This puts the power supply in the mode of a standard bridge rectifier.

In any case, the main power converter will see a dc voltage range of between 255 and 367-volts. The two sets of two parallel electrolytic capacitors each have one parallel 240 K<ohm> resistor per set. The values of the resistors determine the discharge time of the capacitor after the power is turned off.

Main Power Converter

This circuit as shown in Figure 12-57 will oscillate due to the 180<degree> phase shift of the transformer primary and the base drive winding. An RC (resistor/capacitor) network (470 K<ohm> and 100 <micro>F (approximate)) in the base circuit of the main switching transistor, is outlined by a dotted line in Figure 12-57. This RC network is required to initiate oscillation.

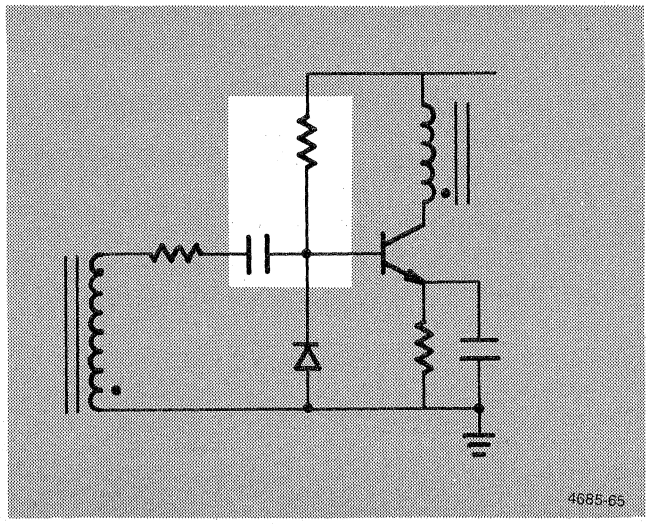


Figure 12-57. Main Power Converter (elementary form).

Once the supply is running, the RC network could be removed with no effect on the oscillation. As shown, the converter would operate with no regulating control.

A gating transistor (base drive transistor) is now added to control the on time of the main switching device. An appropriately timed pulse applied to the base of the gating transistor limits the conduction interval of the main transistor. This regulates the amount of energy stored in the transformer, hence the output power, on a cycle by cycle basis.

These pulses are generated using a unijunction transistor.

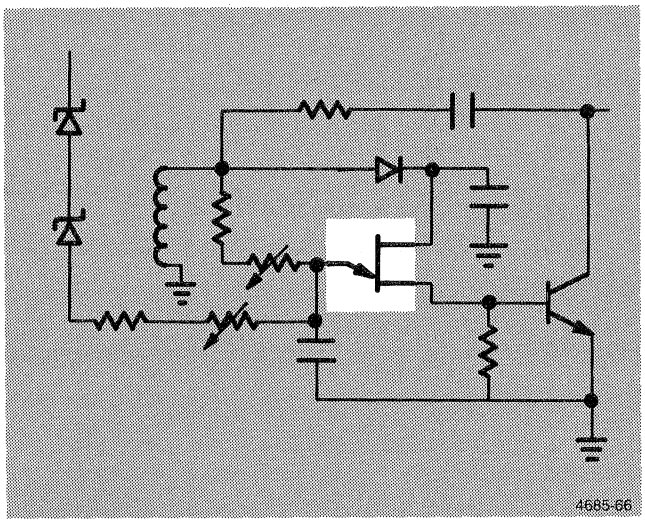


Figure 12-58. Pulse Generator.

The base drive winding voltage, swinging positive, turns on the main switching transistor. Coincidentally, the unijunction RC, (shaded in Figure 12-58) begins charging toward the unijunction trigger point. When the trigger voltage is exceeded, the unijunction fires. This injects a pulse into the base of the gating transistor. The base of the main switching transistor is then pulled to ground, thus terminating its conduction time.

A potentiometer in the base of the unijunction trims the unijunction RC time constant. This allows setting of the maximum allowable on time, hence the maximum available power. Two zener diodes, a resistor, and a potentiometer, all in series with the dc supply and the base of the unijunction, form a network which "measures" the primary input dc supply. A current proportional to the raw dc is injected into the timing capacitor at the base of the unijunction. This effectively reduces the transistor on time as the line voltage increases. The net effect is constant output power for all line voltages.

Abruptly terminating the current flow in the main transistor initiates the energy transfer interval. Current continues to flow in the transformer primary and the collector voltage of the main transistor begins to rise. This voltage rise will also appear across the secondary windings. The output diodes begin to conduct and the energy stored in the transformer core is transferred into the output capacitors.

Snubber

This energy transfer cannot occur instantaneously. This is due largely to less than perfect coupling in the transformer. The snubber network, an RC network contained in the collector circuit of the main switching transistor, forms an interim path for the primary current to flow. This network also limits the rate of rise at the collector so that the current in the main switching transistor is zero before the collector voltage begins to rise.

Control Loop Sense and Drive

The output structures are diode capacitor networks. A pi filter follows this on each output.

The error signal feedback is accomplished across an opto-coupler. Current flow in the secondary opto-diode is reflected in the primary side optical transistor. This optical transistor sources current in the timing capacitor (of the unijunction RC network). This again controls the main switching transistor on time and closes the regulating loop.

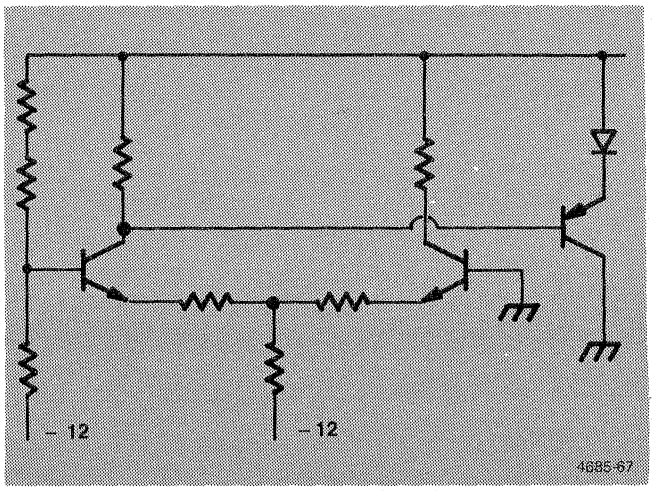


Figure 12-59. Control Loop, Sense and Drive.

The error amplifier itself is a differential pair with a voltage follower output (as shown next).

The 12 and 5-volt output voltages are compared to a fixed, minus 12-volt reference. The minus 12-volt regulator verifies the accuracy. The differential pair amplifiers vary the current in the opto-diodes, balancing the differential amplifiers and ensuring accuracy of the 12-volt and 5-volt supplies.

This minus 12-volt reference is also the minus 12-volt output (in addition to being the bias source for the differential amplifier).

Section 13

CHECKS AND ADJUSTMENTS

THE ADJUSTMENTS CONTAINED HEREIN ARE MEANT FOR QUALIFIED TECHNICIANS ONLY. If no qualified technicians are available, consult your local TEKTRONIX service center.

This section contains functional check, performance check, and adjustment procedures for the 4170. The flexible-disk drive-units are the only assemblies in the 4170 that have adjustment points that can be changed in the field.

FUNCTIONAL CHECK PROCEDURE

The functional check procedure verifies that all major parts of the 4170 function as defined in the operating procedures in this manual. This functional check procedure is typically used to test the 4170 after repair, or for initial acceptance testing by the customer.

The adjustment procedures, later in this section, provide a means for setting the 4170's performance in accordance with the specifications in Section 2. Since the performance check procedure is more time consuming, the functional check procedure is often used to determine if the 4170 is operational.

The functional check procedure consists of two parts. The first part is the power-up routine that runs each time the 4170 is turned on. This routine does a cursory check of the major modules and functions of the 4170. The second part of the functional checks is the Self-test diagnostic program. If, after Self-test is invoked, it runs to completion without stopping to display any error codes, this means the 4170 has passed the full functional check procedure. If Self-test stops and displays an error message, observe that message and look it up in Section 8. This will indicate where the fault is located. Near the end of Self-test, you can cause it to go into Adjustment Self-test; skip over this, as it is only used during the Performance Check and the adjustment/calibration procedures. Upon completion of Self-test (without errors), you will have verified that your 4170 functions properly.

PERFORMANCE CHECK PROCEDURE

The performance check procedure verifies that the 4170 performs according to the specifications in Section 2. Use the functional check procedures, earlier in this section, to first determine the overall functionality of the unit. Then, use the Performance Check procedures to set the performance levels of the various modules in the 4170. This procedure also uses the internal Self-test diagnostic program, but it uses the adjustment routines, as well as the diagnostics.

PROCEDURE

1. Remove the 4170 side cover panel.

2. Check the 3PPI and the host ports at the I/O connectors

    a. Attach the loopback connector

    b. Enter Self-test

    c. Perform the 3PPI and Host Port tests of Adjustment Self-test program

FLEXIBLE DISK DRIVE ADJUSTMENTS

Introduction

The following adjustment procedures apply only to the flexible disk drives. To initiate these procedures, enter the Adjustment Self-test program and select the disk menu. When this is done, certain front-panel keys are programmed to exercise the drive units. Table 13-1 gives the function for each key when programmed for disk calibration.

Table 13-1

FUNCTION KEY DEFINITIONS

Key Name	Function/Key-Definition
F1	MSIB -- ID Verification Test
F2	Step up One Track
F3	Step Down One Track
F4	Seek Track 0
F5	Seek Track 1
F6	Seek Middle of Disk
F7	Select Head
F8	Seek Last Track
2nd F1	Run Motor
2nd F2	Read Current Track
2nd F3	Arm Write Mode
2nd F4	Writes Current Track With a 2F Pattern
2nd F5	Select Your Own Track
2nd F6	Change Device Address
2nd F7	Auto Load/Unload Head on Current Track
2nd F8	MSIB -- External Peripheral Test

Tools and Equipment Required

The following tools/equipment are needed for servicing and adjusting the flexible disk drives:

- Formatted scratch diskette (in good condition).
- Calibration/alignment diskette.
- Oscilloscope - (TEKTRONIX Model 465 or equivalent). Must have External Sync and two channels (one with Invert and one with ADD).
- Frequency counter with an accuracy of +/- 0.05 Hz at 5 Hz.
- #1 and #2 Phillips screwdrivers.

Refer to Accessories, Section 20, for the part number of the alignment diskette.

Test Points

The following procedure references test points on the disk drive unit printed circuit board. Table 13-2 lists these test points and their function. Figure 13-1 shows their location.

Table 13-2

TEST POINTS

Test Point Number	Signal Name
1	+ Read Data
2	- Read Data
5	Signal Ground
6	Digital Read Data
7	Index
8	Track 0
9	Write Protect
10	Ground
12	Step Pulse
13	Motor On
15	Ground

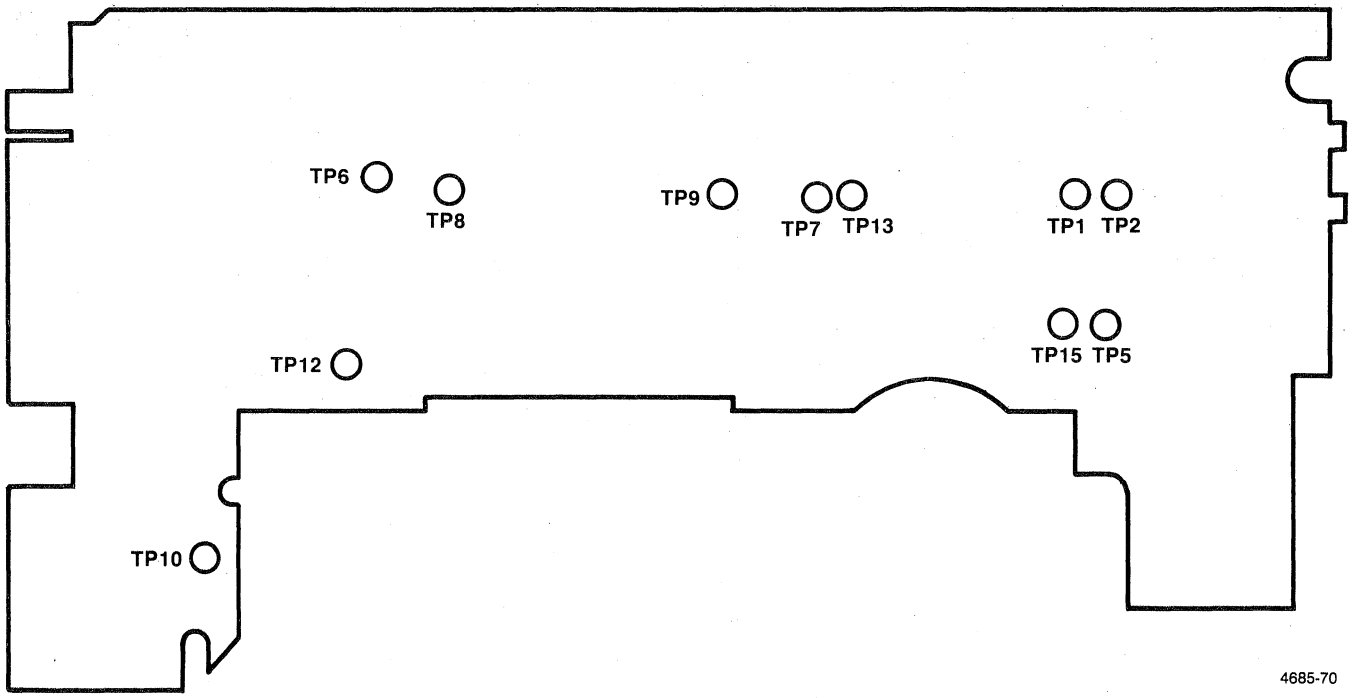


Figure 13-1. Test Points.

Procedure

NOTE

The alignment diskette should be left at room conditions for a period of at least 24 hours before alignment is attempted.

Remove the side/top cover panel from the 4170 for access to the inside of the disk drive units.

Refer to Section 14 of this manual to remove the drive units for adjustment access. Because the two drive units are mounted side-by-side, only one of the two drives is initially accessible. The drive units must be removed and separated for servicing. After removal and separation, the drive units can be operated by extending the cables from the 4170 to the drive unit under test. It is not necessary to reinstall the drive unit under test while adjusting.

Head Radial Alignment.

1. Press function keys 2nd and F1 to start the drive unit motor.
2. Install the alignment diskette.
3. Select drive and step the head to Track 16 (press function key F6).
4. Set up the oscilloscope as follows:
   a. Sync on test point TP-7, EXT, +. (This will display one revolution.)
   b. Set the time base to 20 ms/div.
   c. Connect one probe to TP-1 and the other to TP-2. (Connect both probe grounds to TP-5.)
   d. Set the vertical deflection to 100 mV/div.
   e. Set the inputs to ADD, AC, and invert one channel.
   f. Set the scope mode to ALT.

5. Alternately select heads 0 and 1 and verify the scope display with Figure 13-2. If the amplitude of the lobes on the scope display is not within 70% as shown, continue this procedure; if the amplitude is within 70% on both heads, proceed to the Read/Write Heads Azimuth Check in this section.

   a. Loosen the two mounting screws that hold the stepper motor to the base casting (Figure 13-3).

   b. Select the head whose radial alignment is farthest out of specification, and rotate the stepper motor to radially move the head in or out. If the left lobe is less than 70% of the right, turn the stepper motor clockwise, as viewed from the bottom of the drive. If the right lobe is less than 70% of the left lobe, turn the stepper motor counter-clockwise.

   c. When the lobes of the scope pattern are of equal amplitude, tighten the stepper motor mounting screws.

   d. Select the opposite head and verify that its alignment is still within specification. If not, perform Steps 5a through 5c until the alignment is in specification.

   NOTE

   If both heads cannot be aligned to the specification at the same time, their alignment in relation to each other is incorrect. This condition is not field-correctable -- replace the disk drive unit.

   e. Check the adjustment by stepping off track in both directions and returning. (Press function keys F2 and F3.)

   f. Whenever the head radial alignment has been adjusted, the Track 0 detector adjustment must be checked. (Refer to the Track 0 Detector Assembly Adjustment in this section.)

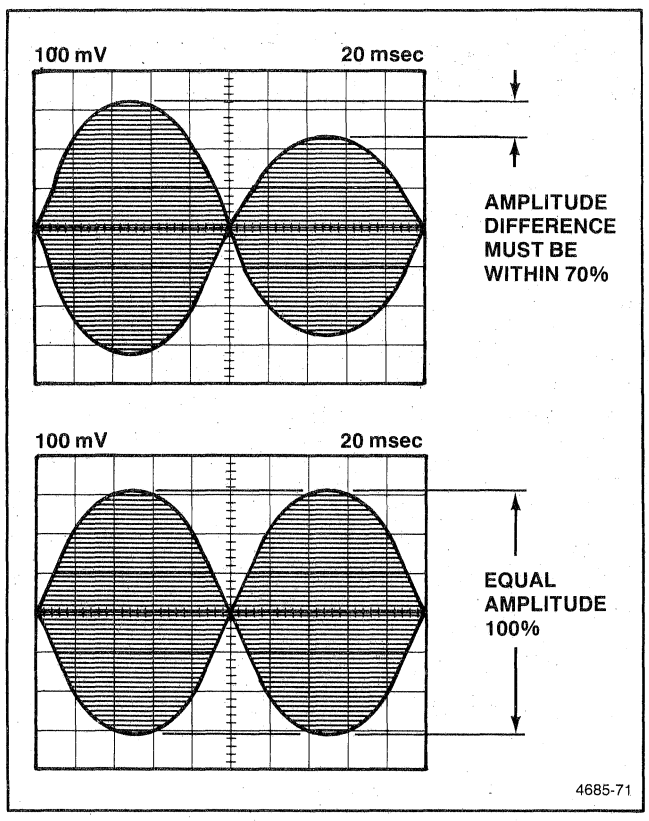


Figure 13-2. Alignment Waveforms.

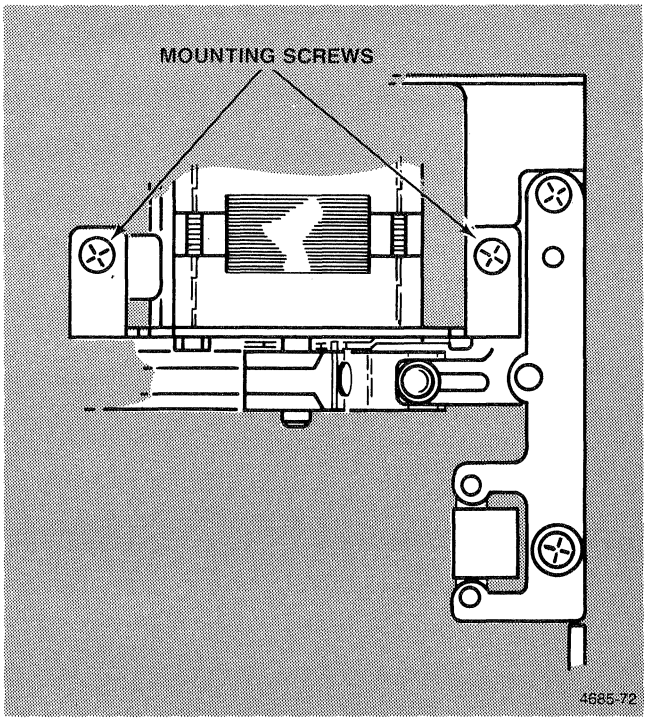


Figure 13-3. Stepper motor mounting Screws.

Read/Write Heads Azimuth Check. The head azimuth is not field adjustable. If this check shows the azimuth is not within +/- 21 minutes, replace the disk drive unit.

1. Install the alignment diskette.

2. Select the drive and press function keys 2nd F5 to step to track 34.

3. Set up the oscilloscope as follows:

   a. Sync on test point TP-7, EXT, +.

   b. Set the time base to 0.5 msec/div.

   c. Connect one probe to TP-1, and the other to TP-2. (Connect both probe grounds to TP-5.)

   d. Invert one channel.

   e. Set the vertical deflection to 100 mV/div and the inputs to AC, ADD.

4. Select head 0, and compare the waveform to Figure 13-4.

5. When observing any of the three azimuth burst sets, a head alignment of 0<degrees>0' is achieved when burst #1 and burst #4 are of equal amplitude. The inspection range limits are depicted at -12', -15', and -18' within the appropriate set. The inspection range limits are depicted at +12', +15', and +18' when burst #4 is equal in amplitude to burst #3 within the appropriate set.

   Tables 13-2 and 13-3 give the allowable limits of burst amplitude ratios. Table 13-2 gives the limits for -21 minutes; Table 13-3 gives the limits for +21 minutes. The tables are used as follows:

   - The first column gives the burst set (see Figure 13-4).
   - The second column gives the maximum ratio of burst #1 to burst #2 allowable.
   - The third column gives the maximum allowable ratio of burst #4 to burst #3.

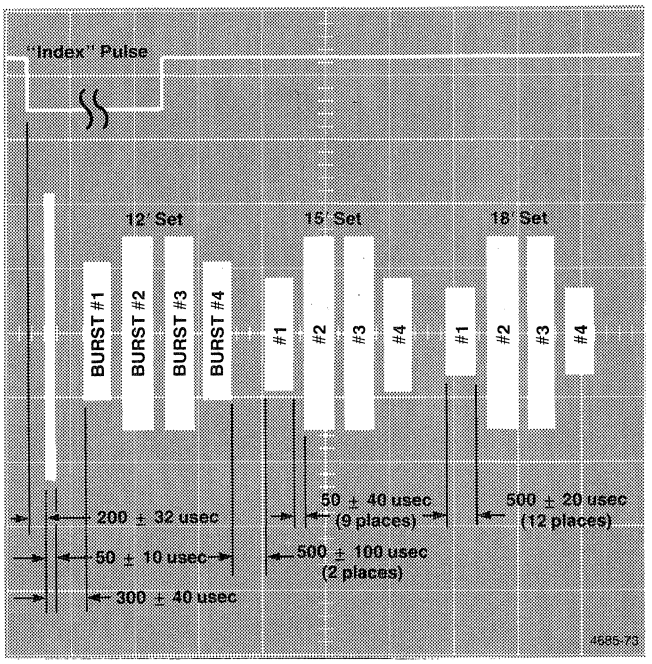


Figure 13-4. Azimuth Burst Waveforms.

Table 13-3

-21 MINUTES

Set	Ratio=(Burst #1/Burst#2)	Ratio=(Burst #4/Burst #3)
12' Set	>1.3261	<0.3150
15' Set	>1.2253	<0.2405
18' Set	>1.1159	<0.1689

Table 13-4

+21 MINUTES

Set	Ratio=(Burst #1/Burst#2)	Ratio=(Burst #4/Burst #3)
12' Set	<0.3150	>1.3261
15' Set	<0.2405	>1.2253
18' Set	<0.1689	>1.1159

6. Repeat for the other head.

7. If either head produces a waveform that is not within the range shown, replace the disk drive unit.

Head Amplitude Check.

Make certain that the scratch diskette used for this check is not "worn" or otherwise shows evidence of damage on either side.

1. Install the scratch diskette.
2. Press function keys 2nd and F1 to start the drive unit motor.
3. Press function key F8 to seek the last track.
4. Set up the oscilloscope as follows:
   a. Sync to TP-7 (+INDEX), EXT
   b. Connect one probe to TP-1 and the other to TP-2. (Connect both probe grounds to TP-5).
   c. Set the vertical deflection to 100 mV/div.
   d. Set the time base to 20 ms/div.
   e. Set the inputs to ADD, and invert one channel.
5. Press function key F7 to select Head 0. Press keys 2nd and F3 to arm the write mode. Press keys 2nd and F4 to write a 2F pattern on the track.
6. Read back the head amplitude. The average minimum peak-to-peak amplitude should be 100 mV.
7. Repeat Steps 5 and 6 for Head 1.
8. If either head fails to meet the amplitude specification, continue with Step 9. If both heads meet the specification, proceed to the Track 0 Detector Assembly Adjustment in this section.
9. Install a different diskette and recheck.
10. Check the motor speed. (Refer to the Motor Speed Adjustment in this section.)

11. With the oscilloscope in CHOP mode, verify that an output exists at both TP-1 and TP-2. If one test point has no output (or significantly less output than the other), turn the head cable connector over at J4. If the same test point has little or no signal, repair the circuit board or replace the disk drive unit. If the opposite test point shows the problem, the head assembly is faulty and the disk drive unit must be replaced.

Track Zero Detector Assembly Adjustment.

1. Install the alignment diskette.

2. Select drive and press function key F4 to seek Track 0.

3. Set up the oscilloscope as follows:

   a. Sync to TP-7, EXT, +.

   b. Set the time base to 20 msec/div.

   c. Connect one probe to TP-1 and the other to TP-2. (Connect both probe grounds to TP-5.)

   d. Set the vertical deflection to 200 mV/div.

   e. Set the inputs to AC, ADD, and invert one channel.

4. A 125 kHz signal should be observed on the scope at this time. (If the signal is not present, press function key F2 a maximum of five times or press F3 as many times as necessary until the signal is located.) When the 125 kHz signal is present on the oscilloscope, the drive carriage is at Track 0.

5. Disconnect one scope probe and connect to TP-8 -- Set input to DC, vertical deflection to 2 V/div, and trigger from the channel being used.

6. Press function key F5 to seek Track 1 and verify that TP-8 goes to 0 V.

7. If TP-8 does not go to 0 V, loosen the Track 0 Detector bracket (Figure 13-5). Adjust until TP-8 goes to 0 V when function key F5 is pressed and goes to +5 V when F4 is pressed.

8. Tighten the Track 0 Detector bracket and recheck.

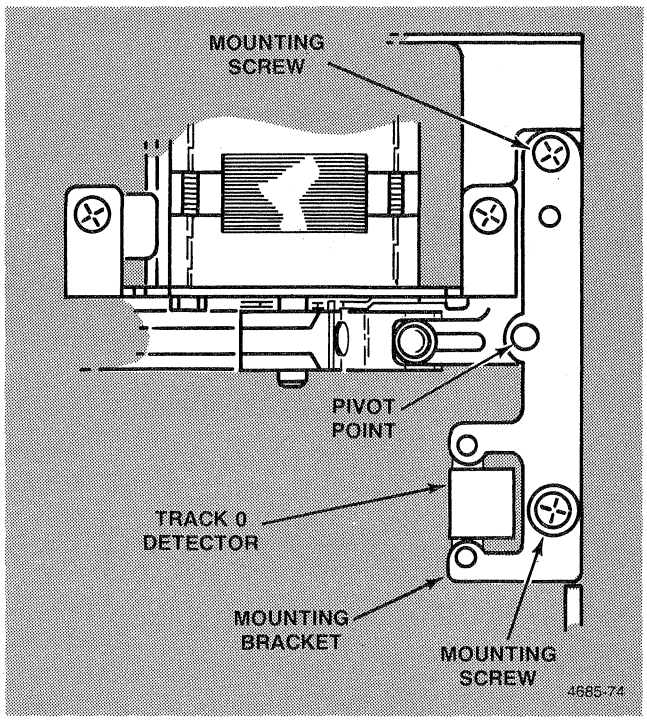


Figure 13-5. Track 0 Detector Bracket Mounting.

Index/Sector Timing Adjustment.

1. Insert the alignment diskette.
2. Press function key F7 to select Head 0.
3. Press F5 to seek Track 1.
4. Set up oscilloscope as follows:
    a. Sync to TP-7, EXT, +.
    b. Set time base to 50 <micro>sec/div.
    c. Connect one probe to TP-1 and the other to TP-2. (Connect both probe grounds to TP-5.)
    d. Set the vertical deflection to 100 mV/div.
    e. Set the inputs to AC, ADD, and invert one channel.
5. Verify that the timing between the start of the sweep and the first data pulse is 200 (+200/-100) <micro>sec (Figure 13-6). If the timing is not within this specification, proceed with Step 6. If the timing is within specification, proceed to the Write-Protect Switch Adjustment in this section.
6. Loosen the mounting screw in the index detector block until the block can just be moved (Figure 13-7).
7. Adjust the index detector block until the specification is met. Make certain that the detector assembly is against the registration surface on the hub frame.
8. Tighten the mounting screw.
9. Recheck the timing.
10. Press function key F7 to select Head 1 and press F5 to seek Track 1. Repeat the procedure for Head 1.

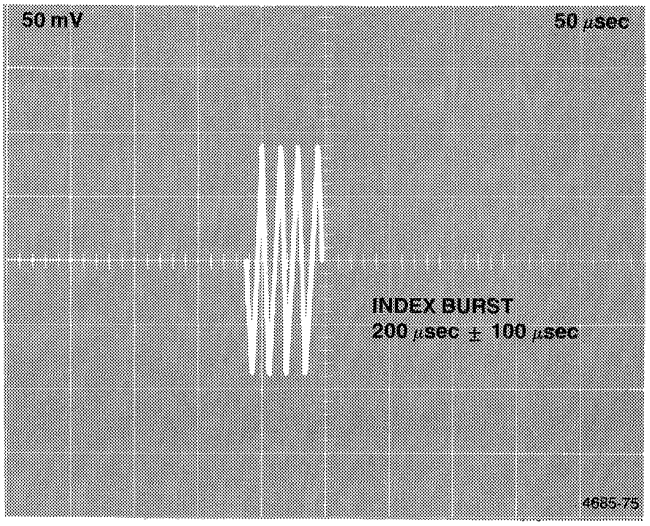


Figure 13-6. Index Timing.

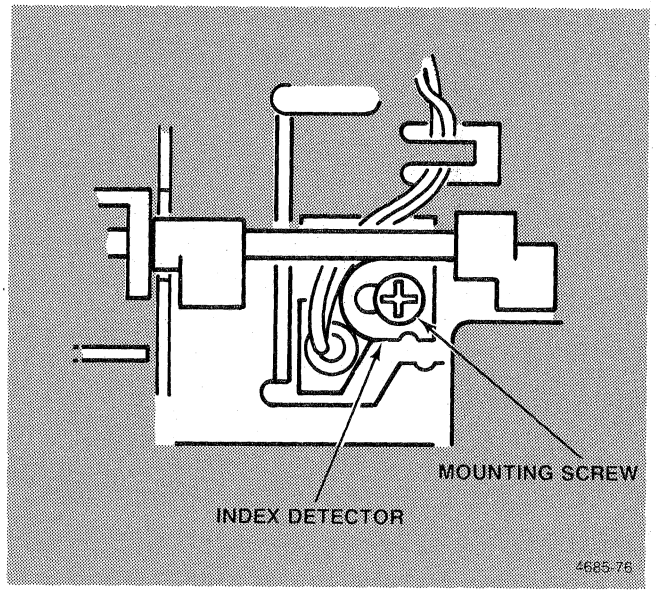


Figure 13-7. Index Detector Mounting.

Motor Speed Adjustment.

NOTE

It is recommended that this adjustment not be performed in the field.

1. Install a blank diskette in the drive unit.
2. Press function key F5 to seek Track 1.
3. Connect the frequency counter to TP-7.
4. Adjust the potentiometer on the motor circuit board to obtain a reading of 5 +/- 0.05 Hz (Period 200 +/- 2 ms).

Write-Protect Switch. The write-protect switch can be checked for proper operation by monitoring the voltage on TP-9 while inserting/removing a diskette that has the write-protect notch open. The voltage on TP-9 should go from 0 to 5 V as the diskette is inserted/removed.

Section 14

MAINTENANCE

This section contains the disassembly and reassembly procedures required for troubleshooting, adjustment, and repair of the 4170. This section also contains preventive maintenance procedures for the flexible disk drive units.

SAFETY SUMMARY

The operator section of this manual contains general safety warnings. Many such warnings are obvious to the trained technician or do not apply here (such as warnings against removing doors and access panels). The following are general service warnings that should be observed while working on this instrument.

DO NOT SERVICE ALONE

Do NOT perform internal service or adjustment of this product unless another person capable of rendering first aid and resuscitation is present.

USE CARE WHEN SERVICING WITH POWER ON

Disconnect power before removing circuit boards, soldering, or replacing components.

Exercise care so that tools and probes do not complete a connection other than that intended. All jewelry that might interfere or complete a circuit should be removed to prevent injury.

No hazardous voltages are exposed within the 4170 cabinet. The voltages present are +5, +12 and -12.

POWER SOURCE

The 4170 draws power from a line voltage source and brings AC voltages into the power supply and fan. Be sure the product is grounded properly via a ground conductor in the power cord or an insulated ground cable.

PREVENTIVE MAINTENANCE

The 4170 is designed to require a minimum of maintenance and servicing. Once a year you should remove the side panel and look for and correct the following:

- Remove any dust accumulations
- Loose screws, connectors, and switches
- Inspect the flexible disk drive heads for dirt deposits and clean if necessary (the optional Hard disk drive unit is a sealed module; therefore, its design does not allow field maintenance)
- Calibrate flexible disk drives (see checks and adjustment section)
- Excessive play in the fan motor shaft

CAUTION

Do not attempt to oil any of the motors in the disk drive units. These motors have sealed bearings and therefore do not require lubrication. Any oil added will only catch dust and create a greater wear problem.

PREVENTIVE MAINTENANCE PROCEDURES

DISK UNIT READ/WRITE HEAD CLEANING

The read/write head should be cleaned after each 12 months of normal use. Use a cleaning disk to clean the disk unit read/write head.

DISASSEMBLY/REASSEMBLY PROCEDURES

REMOVING THE FRONT COVER

The card cage is accessible via a cover on the front of the 4170. This cover is attached by two screws (one in each bottom corner) and a lip (that secures the top of the cover). First, unscrew the two thumbscrews. Then, swing the cover out at the bottom. Finally, allow the cover panel to drop free, releasing the retainer tab at the top. See Figure 14-1.

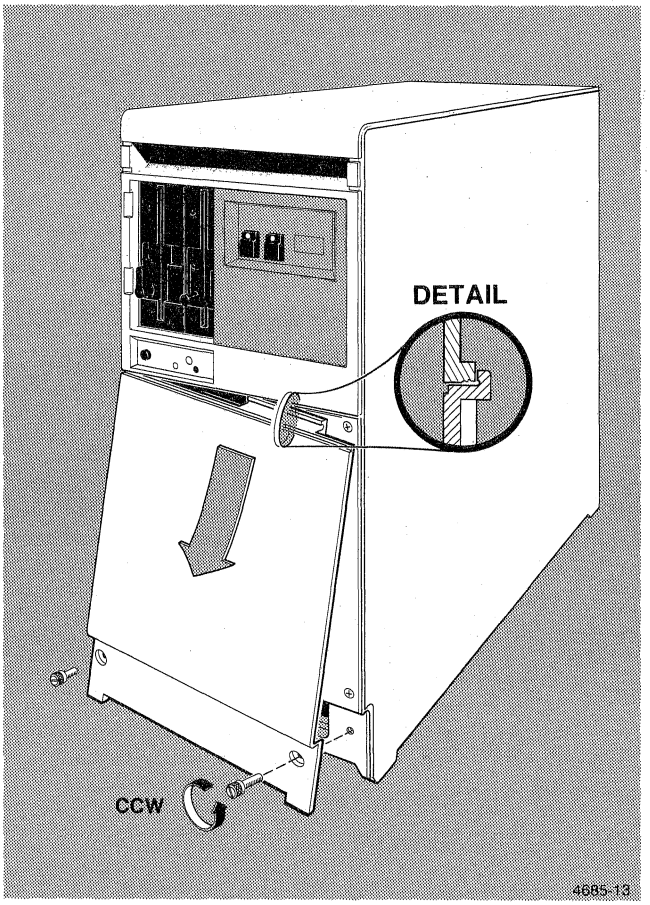


Figure 14-1. Removing the Front Cover.

REMOVING CIRCUIT BOARDS FROM THE CARD CAGE

Figure 14-2 shows the recommended arrangement of circuit boards in the card cage. Use this diagram to determine the location of the board(s) needing removal.

- Remove the Front Cover (described earlier).
- Determine which cables and wires (connecting to the front of the circuit boards) need to be unplugged. Then, label these cables, so you can easily reconnect them to their proper connectors. Unplug the cables.
- Read the name/label on the circuit board to be sure you are removing the intended board.
- Grasp the upper and lower ejectors as shown in Figure 14-3. Pull the ejectors out toward you.
- Pull the board straight out of the slot.

Before reinserting any circuit boards, verify that you are placing it in the proper slot according to Figure 14-2.

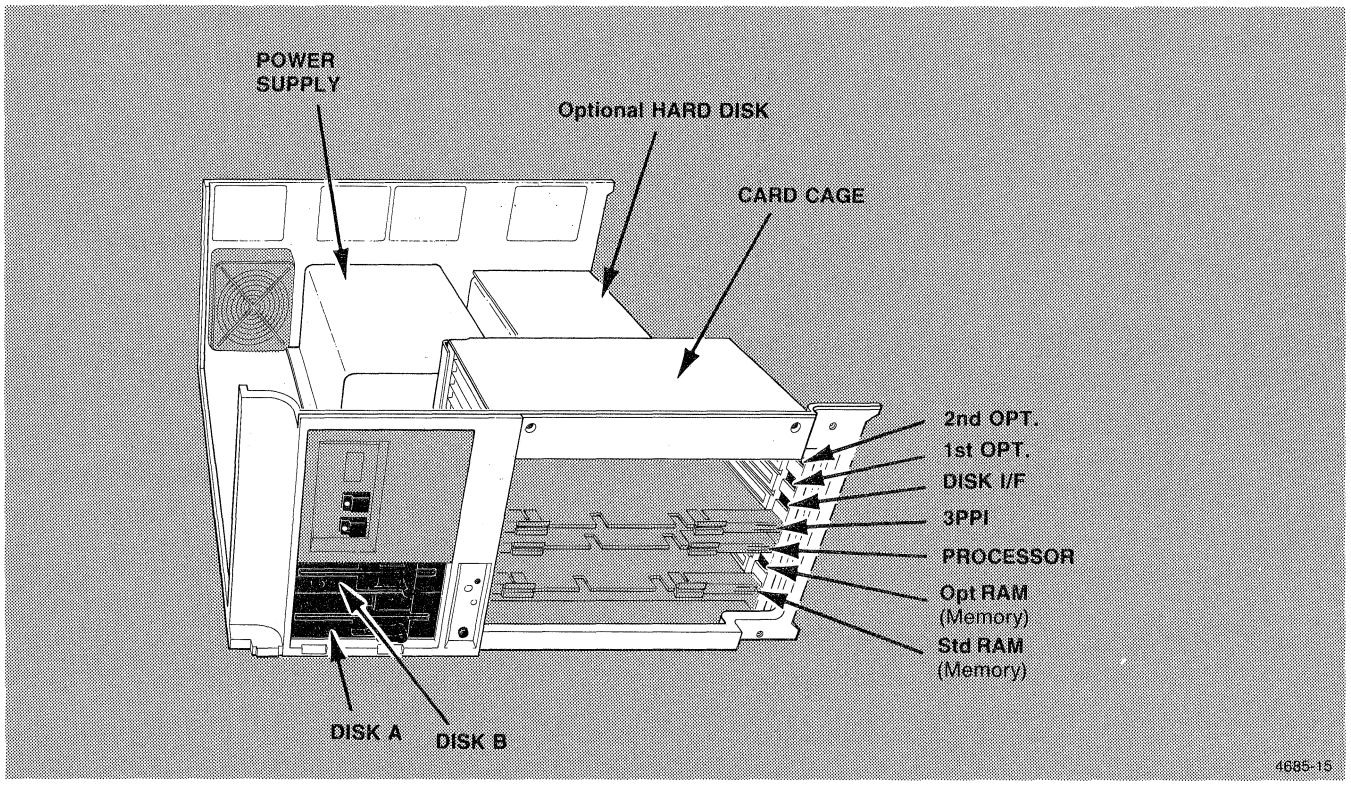


Figure 14-2. 4170 Circuit Board Locations.

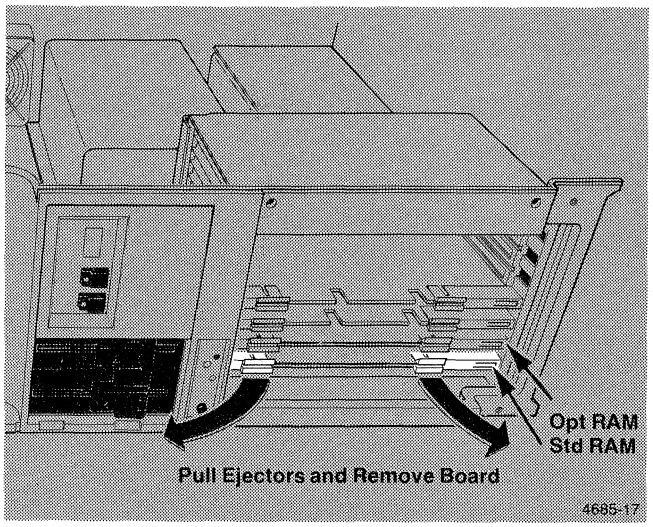


Figure 14-3. Removing Card Cage Circuit Boards.

REMOVING THE SIDE COVER

The major modules in the 4170 (flexible drive units, power supply, hard disk unit, control panel board, and fan) are accessible via the side cover. To remove this cover, refer to Figure 14-4 and do this procedure:

- Remove the front cover (described earlier).
- Place the 4170 on its left side.
- Unscrew the two screws at the right edge of the front access opening.
- Unscrew the two screws from the left edge of the rear panel.
- Lift the side cover away from the instrument (the panel wraps over the top of the 4170).

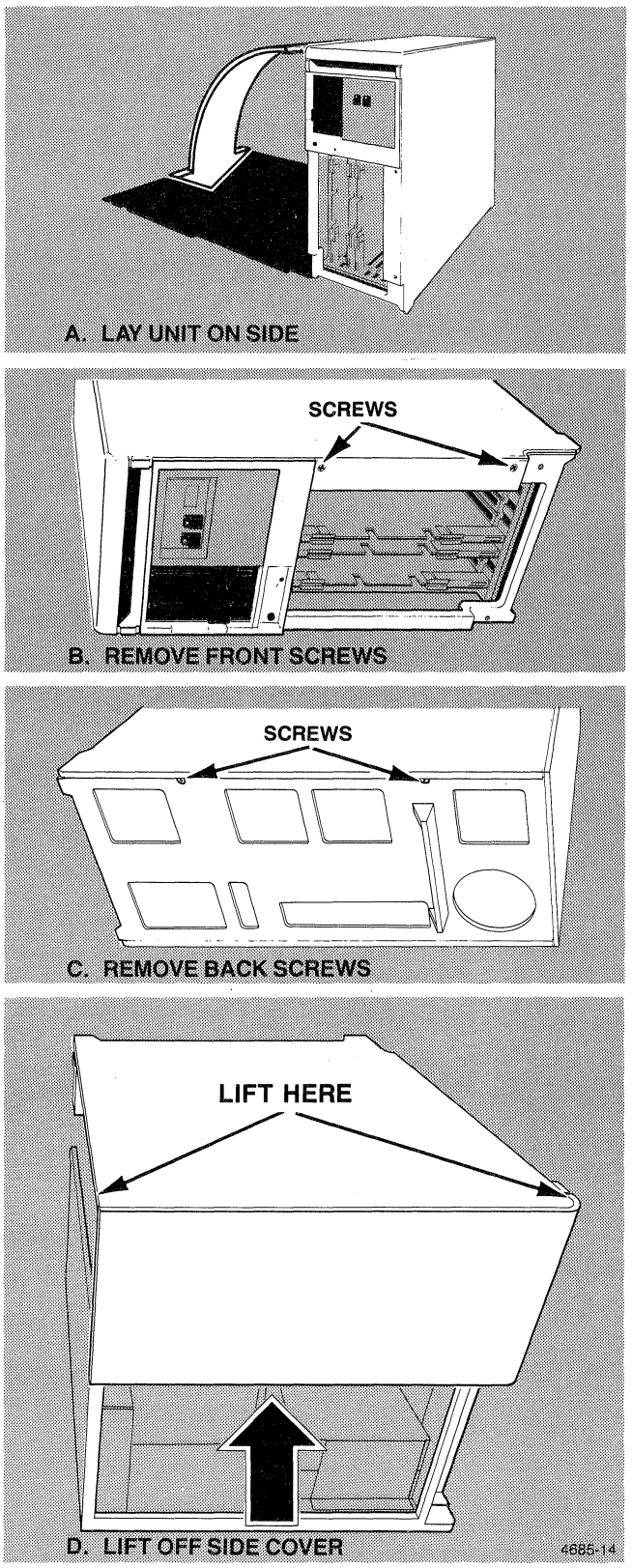


Figure 14-4. Side Cover Removal.

REMOVING THE FLEXIBLE-DISK DRIVE UNIT(S)

The two flexible-disk drive units mount together on two common brackets that are fastened to the side of the 4170. The common brackets are L-shaped. One is located on the top of the two drives and the other is located beneath the drives. Each common brackets is attached to the side panel with two screws. To remove the flexible-disk drives do the following:

- Remove the side cover (earlier procedure).
- Unplug the cables that go plug into the drive units.
- Loosen (do not remove) the four screws that secure the L-shaped brackets to the side panel.
- Slide the drive-units assembly (both drives and the two L-shaped brackets) to the rear until the screw heads are free.
- Lift the assembly away from the side panel.

DISASSEMBLING THE FLEXIBLE-DISK DRIVE ASSEMBLY

Since the two flexible-disk drive units are fastened together in a common mounting bracket, it is necessary to first separate them for most disk-unit repairs.

- Remove the drive-units assembly from the 4170.
- Remove the screws that mount the top and bottom L-brackets to the drive units. This separates the two drive units.

REMOVING THE POWER SUPPLY MODULE

The Power Supply Module is secured to the side panel with four screws. Remove the power supply by doing the following procedure:

- Remove the side cover (described earlier).
- Unplug P11 from the mother board.
- Unplug fan connector J62 from the power supply.
- Loosen (do not remove) the four screws that fasten the power supply to the 4170 side-panel.
- Slide the power supply forward in the key slots.
- As the screw heads reach the open end of the slot, lift the power supply away from the 4170.
- The dc power cable can then be easily unplugged from the power supply jack, J61.

Reverse procedure for reassembly.

REMOVING THE OPTIONAL HARD-DISK MODULE

The optional hard disk drive module consists of the drive unit (in a chassis-type enclosure), an interface board mounted on the outside of the chassis, the MSI Bus ribbon cable, and power cables. Refer to Figure 14-5 and perform the following procedure to remove the optional hard-disk module:

- Remove the side-cover.
- Unplug the MSI Bus ribbon cables from the interface board.
- Unplug the dc power cable (P9) from the mother board and plug P2 from the hard-disk drive.
- Loosen (do not remove) the four screws that mount the drive module to the side of the 4170.
- Slide the disk module forward in the screw slots, then lift out of the 4170.
- Remove the interface board as needed: disconnect the I/F board-to-drive unit cables, and unscrew the I/F board mounting screws.
- The hard disk unit is not a field-repairable item. This manual does not include any preventive maintenance or repair procedures for that unit. See TEKTRONIX manual 119-1644-00 Hard Disk Service Manual.

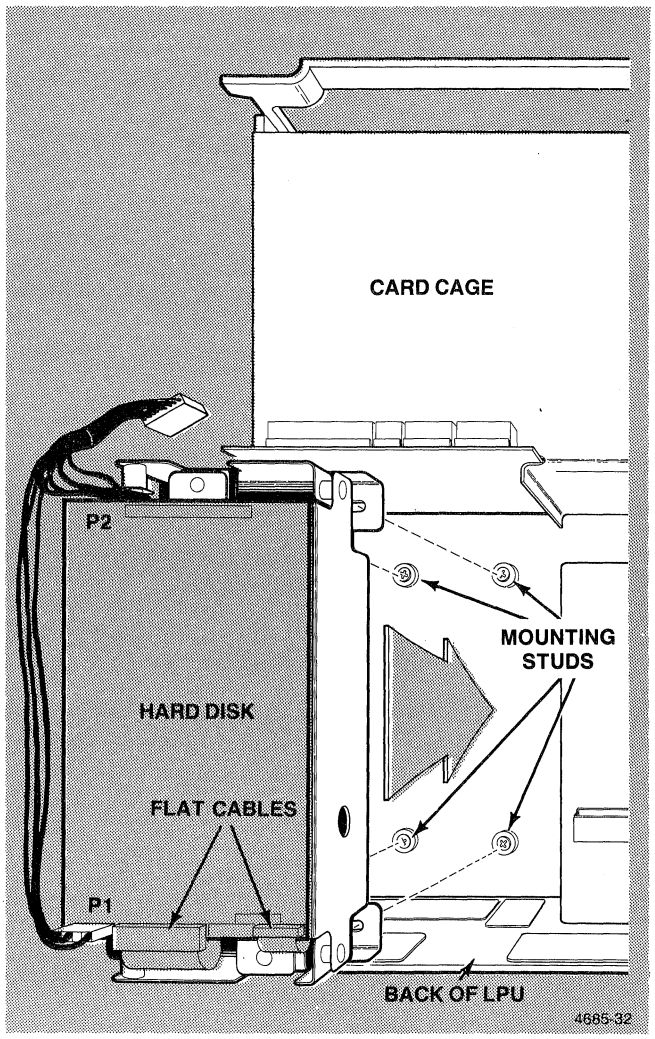


Figure 14-5. Removing Hard-Disk Module.

REMOVING THE CONTROL PANEL AND PANEL DOOR

The Control Panel contains several switches and two 7-segment displays. During normal operation the two displays are the only items visible to the operator; the switches and other components are hidden by a pop-off door. Figure 14-6 shows how to remove the pop-off door.

To remove the Control Panel circuit board, refer to Figure 14-7 and perform the following procedure:

- Remove the side cover; this provides access to the back side of the circuit board.
- Label and unplug the wires that connect to the Control Panel circuit board.
- Unscrew the four small screws that secure the board to the front panel.
- Remove board.

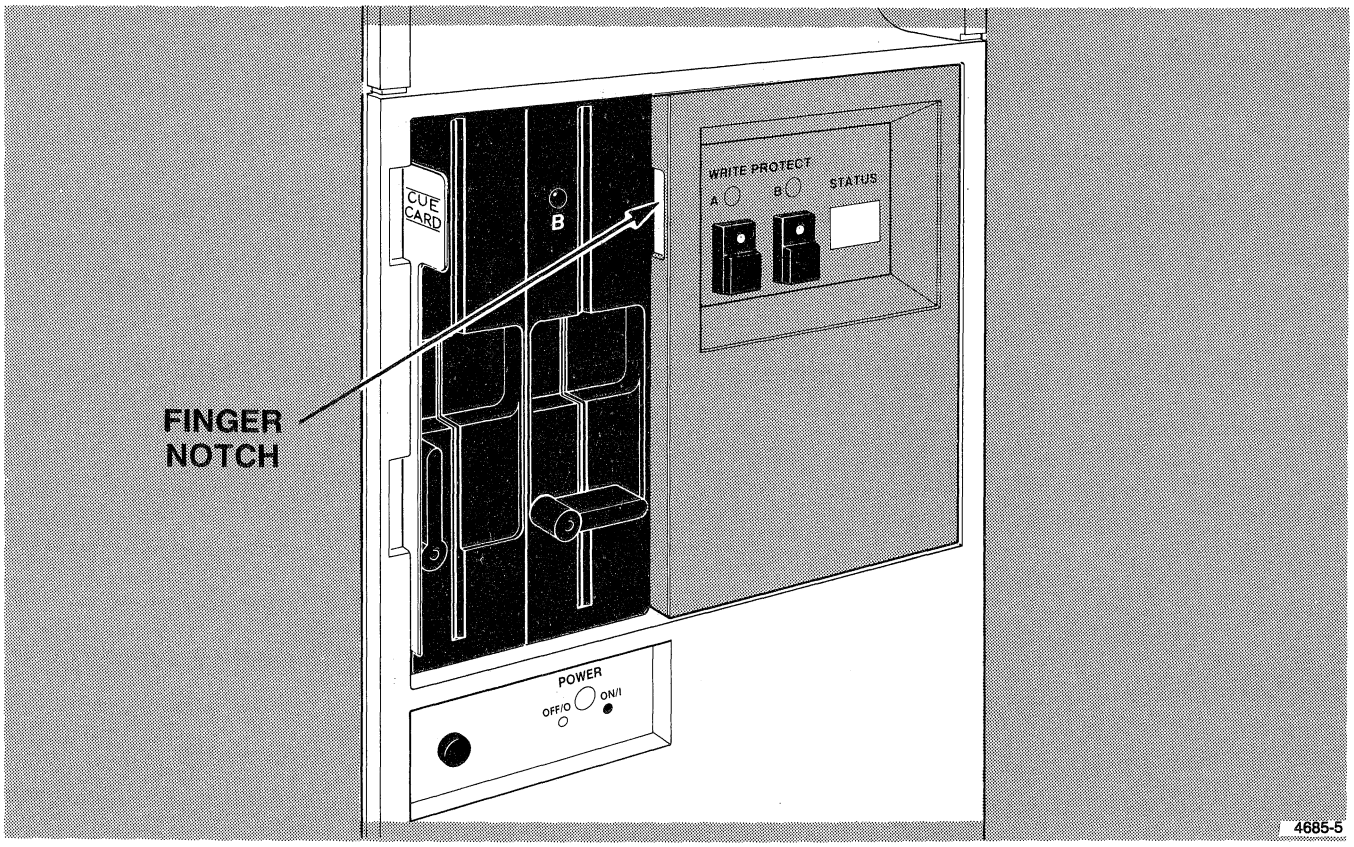


Figure 14-6. Removing the Pop-Off Control Panel Door.

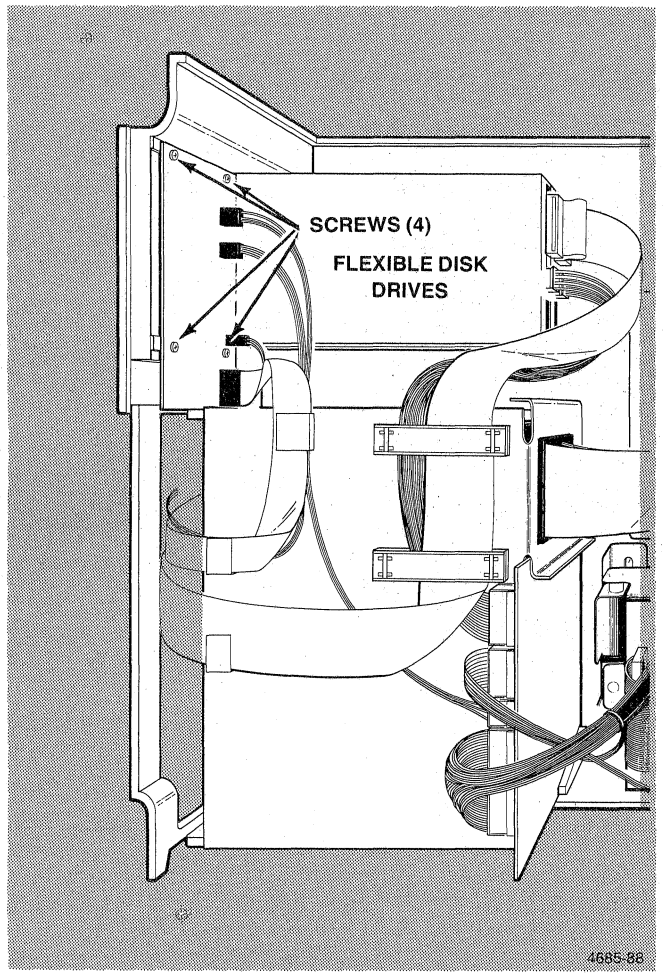


Figure 14-7. Removing Control Panel Circuit Board.

ACCESSING/REMOVING REAR-MOUNTED COMPONENTS

The 4170 rear-panel contains several types of connectors and the cooling fan. To replace any of these components, first remove the side cover. The connectors are mounted with screws from the outside, that thread into inserts in the connector panels. Label all connected cables that must be removed to replace these connectors.

To replace the cooling fan, first unplug its power cable, then, using a screwdriver and nut driver, unscrew the four mounting screws.

TROUBLESHOOTING AND CORRECTIVE MAINTENANCE

USING SELF TEST

The primary troubleshooting aid for this product is the Self Test diagnostic program that resides in firmware (or is loaded from the disk/diskette at power-up). Section 8 provides a complete description of all Self Test error messages and the module testing sequence.

INITIAL/VISUAL CHECKS

Self-Test will isolate most system problems; however, if it does not, go through the following procedures: A series of initial/visual checks may help isolate a problem to the module-level. First, try to determine whether the problem is hardware or software related. Does the disk work OK? If one disk is malfunctioning, try the other drive-unit; if the problem persists, the cause is either a bad diskette or a fault in the Disk Controller board. Some disk related problems are random and recoverable; such are called "soft-errors." Other disk problems persist after several tries, and are called "hard errors." Hard errors do not necessarily imply a hardware problem; it may mean the disk media is faulty. If the system is not host-connected and fails to boot properly, try a backup version of the boot diskette.

Section 15
REPLACEABLE ELECTRICAL PARTS

PARTS ORDERING INFORMATION

Replacement parts are available from or through your local Tektronix, Inc. Field Office or representative.

Changes to Tektronix instruments are sometimes made to accommodate improved components as they become available, and to give you the benefit of the latest circuit improvements developed in our engineering department. It is therefore important, when ordering parts, to include the following information in your order: Part number, instrument type or number, serial number, and modification number if applicable.

If a part you have ordered has been replaced with a new or improved part, your local Tektronix, Inc. Field Office or representative will contact you concerning any change in part number.

Change information, if any, is located at the rear of this manual.

LIST OF ASSEMBLIES

A list of assemblies can be found at the beginning of the Electrical Parts List. The assemblies are listed in numerical order. When the complete component number of a part is known, this list will identify the assembly in which the part is located.

CROSS INDEX-MFR. CODE NUMBER TO MANUFACTURER

The Mfr. Code Number to Manufacturer index for the Electrical Parts List is located immediately after this page. The Cross Index provides codes, names and addresses of manufacturers of components listed in the Electrical Parts List.

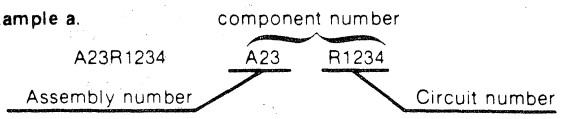
ABBREVIATIONS

Abbreviations conform to American National Standard Y1.1.

COMPONENT NUMBER (column one of the Electrical Parts List)

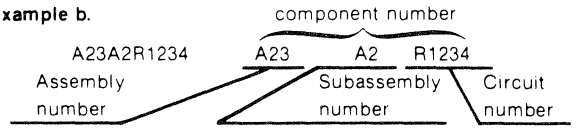
A numbering method has been used to identify assemblies, subassemblies and parts. Examples of this numbering method and typical expansions are illustrated by the following:

Example a.



Read: Resistor 1234 of Assembly 23

Example b.



Read: Resistor 1234 of Subassembly 2 of Assembly 23

Only the circuit number will appear on the diagrams and circuit board illustrations. Each diagram and circuit board illustration is clearly marked with the assembly number. Assembly numbers are also marked on the mechanical exploded views located in the Mechanical Parts List. The component number is obtained by adding the assembly number prefix to the circuit number.

The Electrical Parts List is divided and arranged by assemblies in numerical sequence (e.g., assembly A1 with its subassemblies and parts, precedes assembly A2 with its subassemblies and parts).

Chassis-mounted parts have no assembly number prefix and are located at the end of the Electrical Parts List.

TEKTRONIX PART NO. (column two of the Electrical Parts List)

Indicates part number to be used when ordering replacement part from Tektronix.

SERIAL/MODEL NO. (columns three and four of the Electrical Parts List)

Column three (3) indicates the serial number at which the part was first used. Column four (4) indicates the serial number at which the part was removed. No serial number entered indicates part is good for all serial numbers.

NAME & DESCRIPTION (column five of the Electrical Parts List)

In the Parts List, an Item Name is separated from the description by a colon (:). Because of space limitations, an Item Name may sometimes appear as incomplete. For further Item Name identification, the U.S. Federal Cataloging Handbook H6-1 can be utilized where possible.

MFR. CODE (column six of the Electrical Parts List)

Indicates the code number of the actual manufacturer of the part. (Code to name and address cross reference can be found immediately after this page.)

MFR. PART NUMBER (column seven of the Electrical Parts List)

Indicates actual manufacturers part number.

CROSS INDEX—MFR. CODE NUMBER TO MANUFACTURER

Mfr. Code	Manufacturer	Address	City, State, Zip
000HX	SAN-O INDUSTRIAL CORP.	170 WILBUR PLACE	BAHEMIA LONG ISLAND, NY 11716
000IZ	DALE ELECTRONICS CORP.	P.O. BOX 3164	TEMPE, AZ 85282
000JB	STAR MICRONICS INC.	200 PARK AVE SUITE 08	NEW YORK, NY 10166
00779	AMP, INC.	P.O. BOX 3608	HARRISBURG, PA 17105
00853	SANGAMO ELECTRIC CO., S. CAROLINA DIV.	P.O. BOX 128	PICKENS, SC 29671
01121	ALLEN-BRADLEY COMPANY	1201 2ND STREET SOUTH	MILWAUKEE, WI 53204
01295	TEXAS INSTRUMENTS, INC. SEMICONDUCTOR GROUP	P.O. BOX 5012	DALLAS, TX 75222
01807	PETERSEN RADIO COMPANY, INC.	2800 WEST BROADWAY	COUNCIL BLUFFS, IN 51501
03508	GENERAL ELECTRIC COMPANY, SEMI-CONDUCTOR PRODUCTS DEPARTMENT	ELECTRONICS PARK	SYRACUSE, NY 13201
04222	AVX CERAMICS, DIVISION OF AVX CORP.	P O BOX 867	MYRTLE BEACH, SC 29577
04713	MOTOROLA, INC., SEMICONDUCTOR PROD. DIV.	5005 E MCDOWELL RD,PO BOX 20923	PHOENIX, AZ 85036
05397	UNION CARBIDE CORPORATION, MATERIALS SYSTEMS DIVISION	11901 MADISON AVENUE	CLEVELAND, OH 44101
05574	VIKING INDUSTRIES, INC.	21001 NORDHOFF STREET	CHATSWORTH, CA 91311
07263	FAIRCHILD SEMICONDUCTOR, A DIV. OF FAIRCHILD CAMERA AND INSTRUMENT CORP.	464 ELLIS STREET	MOUNTAIN VIEW, CA 94042
13511	AMPHENOL CARDRE DIV., BUNKER RAMO CORP.		LOS GATOS, CA 95030
14433	ITT SEMICONDUCTORS	3301 ELECTRONICS WAY P O BOX 3049	WEST PALM BEACH, FL 33402
14752	ELECTRO CUBE INC.	1710 S. DEL MAR AVE.	SAN GABRIEL, CA 91776
15238	ITT SEMICONDUCTORS, A DIVISION OF INTER NATIONAL TELEPHONE AND TELEGRAPH CORP.	P.O. BOX 168, 500 BROADWAY	LAWRENCE, MA 01841
18324	SIGNETICS CORP.	811 E. ARQUES	SUNNYVALE, CA 94086
22526	BERG ELECTRONICS, INC.	YOUK EXPRESSWAY	NEW CUMBERLAND, PA 17070
27014	NATIONAL SEMICONDUCTOR CORP.	2900 SEMICONDUCTOR DR.	SANTA CLARA, CA 95051
27264	MOLEX PRODUCTS CO.	5224 KATRINE AVE.	DOWNERS GROVE, IL 60515
32997	BOURNS, INC., TRIMPOT PRODUCTS DIV.	1200 COLUMBIA AVE.	RIVERSIDE, CA 92507
34335	ADVANCED MICRO DEVICES	901 THOMPSON PL.	SUNNYVALE, CA 94086
34649	INTEL CORP.	3065 BOWERS AVE.	SANTA CLARA, CA 95051
50434	HEWLETT-PACKARD COMPANY	640 PAGE MILL ROAD	PALO ALTO, CA 94304
51642	CENTRE ENGINEERING INC.	2820 E COLLEGE AVENUE	STATE COLLEGE, PA 16801
54473	MATSUSHITA ELECTRIC, CORP. OF AMERICA	1 PANASONIC WAY	SECAUCUS, NJ 07094
55680	NICHICON/AMERICA/CORP.	6435 N PROESEL AVENUE	CHICAGO, IL 60645
55857	GOULD INC. PORTABLE BATTERY DIV.	2777 EAGANDALE BLVD	EAGAN, MN 55121
56289	SPRAGUE ELECTRIC CO.	87 MARSHALL ST.	NORTH ADAMS, MA 01247
57668	R-OHM CORP.	16931 MILLIKEN AVE.	IRVINE, CA 92713
59660	TUSONIX INC.	2155 N FORBES BLVD	TUCSON, AZ 85705
59821	CENTRALAB INC SUB NORTH AMERICAN PHILIPS CORP	7158 MERCHANT AVE	EL PASO, TX 79915
70903	BELDEN CORP.	2000 S BATAVIA AVENUE	GENEVA, IL 60134
71400	BUSSMAN MFG., DIVISION OF MCGRAW- EDISON CO.	2536 W. UNIVERSITY ST.	ST. LOUIS, MO 63107
72982	ERIE TECHNOLOGICAL PRODUCTS, INC.	644 W. 12TH ST.	ERIE, PA 16512
73138	BECKMAN INSTRUMENTS, INC., HELIPOT DIV.	2500 HARBOR BLVD.	FULLERTON, CA 92634
75915	LITTELFUSE, INC.	800 E. NORTHWEST HWY	DES PLAINES, IL 60016
80009	TEKTRONIX, INC.	P O BOX 500	BEAVERTON, OR 97077
82389	SWITCHCRAFT, INC.	5555 N. ELSTON AVE.	CHICAGO, IL 60630
82877	ROTRON, INC.	7-9 HASBROUCK LANE	WOODSTOCK, NY 12498
91637	DALE ELECTRONICS, INC.	P. O. BOX 609	COLUMBUS, NE 68601
96733	SAN FERNANDO ELECTRIC MFG CO	1501 FIRST ST	SAN FERNANDO, CA 91341
T1015	MUSASHI WORKS OF HITACHI LTD	1450 JOSUIHON-CHO	KODAIRA-SHI TOKYO, JAPAN
T1043	DALE ELECTRONICS CORP	PO BOX 3164	TEMPE, AZ 85282

Component No.	Tektronix Part No.	Serial/Model No. Eff	Dscont	Name & Description	Mfr Code	Mfr Part Number
				CIRCUIT BOARD ASSEMBLIES		
A1	670-8119-00			CKT BOARD ASSY:MOTHER	80009	670-8119-00
A2	672-1140-00			CKT BOARD ASSY:PROCESSOR	80009	672-1140-00
A3	670-8148-00			CKT BOARD ASSY:DISK CONTROLLER	80009	670-8148-00
A3	----- -----			(OPTION 44 ONLY)		
A4	670-8120-00			CKT BOARD ASSY:FRONT PANEL	80009	670-8120-00
A5	620-0324-00			CKT BOARD ASSY:POWER SUPPLY	80009	620-0324-00
A6	119-1617-00			CONTROLLER DISK:	80009	119161700
A6	----- -----			(OPTION 03 ONLY)		
A7	672-1163-00			CKT BOARD ASSY:INTERFACE	80009	672-1163-00
A7	----- -----			(OPTION 45 ONLY)		
A8	011-0090-00			TERMN,LINE:		
A8	----- -----			(OPTION 45 ONLY)		
A9	670-5291-XX			CKT BOARD ASSY:LOGIC EXTENDER		
A9	----- -----			(NOT AVAILABLE,USE 067-1005-00)		
	119-1644-00			DISK DRIVE:12.76M BYTES	80009	119-1644-00
	----- -----			(OPTION 03 ONLY)		
	----- -----			(SEE APPROP SERV MANUAL FOR PARTS LIST)		
	670-8139-00			CKT BOARD ASSY:RAM CONTROLLER	80009	670-8139-00
	----- -----			(SEE APPROP SERV MANUAL FOR PARTS LIST)		
	670-8138-00			CKT BOARD ASSY:RAM ARRAY	80009	670-8138-00
	----- -----			(SEE APPROP SERV MANUAL FOR PARTS LIST)		
	670-8304-00			CKT BOARD ASSY:3PPI	80009	670-8304-00
	----- -----			(STANDARD AND OPTION 10 ONLY)		
	----- -----			(SEE APPROP SERV MANUAL FOR PARTS LIST)		
	119-1636-00			CKT BOARD ASSY:FLEXIBLE DISK DRIVE	80009	119-1636-00
	----- -----			(SEE APPROP SERV MANUAL FOR PARTS LIST)		
	670-8392-00			CKT BOARD ASSY:COLOR COPIER I/F	80009	670-8392-00
	----- -----			(OPTION 09 ONLY)		
	----- -----			(SEE APPROP SERV MANUAL FOR PARTS LIST)		
	672-1173-00			CKT BOARD ASSY:MASS STORAGE I/O	80009	672-1173-00
	----- -----			(OPTION 45 ONLY)		
	----- -----			(SEE APPROP SERV MANUAL FOR PARTS LIST)		

Component No.	Tektronix Part No.	Serial/Model No. Eff	Dscont	Name & Description	Mfr Code	Mfr Part Number
				A1 MOTHER		
A1	670-8119-00			CKT BOARD ASSY:MOTHER	80009	670-8119-00
A1C34	290-0779-00			CAP.,FXD,ELCTLT:10UF,+50-10%,50VDC	56289	502D237
A1C41	283-0422-00			CAP.,FXD,CER DI:0.047UF,+80-20%,50V	04222	DG015E473Z
A1C42	283-0422-00			CAP.,FXD,CER DI:0.047UF,+80-20%,50V	04222	DG015E473Z
A1F11	159-0059-00			FUSE,WIRE LEAD:5A,FAST-BLOW	000HX	SPI-5A
A1F21	159-0059-00			FUSE,WIRE LEAD:5A,FAST-BLOW	000HX	SPI-5A
A1F24	159-0059-00			FUSE,WIRE LEAD:5A,FAST-BLOW	000HX	SPI-5A
A1F31	159-0059-00			FUSE,WIRE LEAD:5A,FAST-BLOW	000HX	SPI-5A
A1F32	159-0059-00			FUSE,WIRE LEAD:5A,FAST-BLOW	000HX	SPI-5A
A1F36	159-0059-00			FUSE,WIRE LEAD:5A,FAST-BLOW	000HX	SPI-5A
A1F42	159-0059-00			FUSE,WIRE LEAD:5A,FAST-BLOW	000HX	SPI-5A
A1J1	131-2059-01			CONN,RCPT,ELEC:EDGECARD,2 X 40,FEM,0.125	00779	3-530671-0
A1J2	131-2059-01			CONN,RCPT,ELEC:EDGECARD,2 X 40,FEM,0.125	00779	3-530671-0
A1J3	131-2059-01			CONN,RCPT,ELEC:EDGECARD,2 X 40,FEM,0.125	00779	3-530671-0
A1J4	131-2059-01			CONN,RCPT,ELEC:EDGECARD,2 X 40,FEM,0.125	00779	3-530671-0
A1J5	131-2059-01			CONN,RCPT,ELEC:EDGECARD,2 X 40,FEM,0.125	00779	3-530671-0
A1J6	131-2059-01			CONN,RCPT,ELEC:EDGECARD,2 X 40,FEM,0.125	00779	3-530671-0
A1J7	131-2059-01			CONN,RCPT,ELEC:EDGECARD,2 X 40,FEM,0.125	00779	3-530671-0
A1J8	131-2528-00			TERM SET,PIN:10 MALE CONTACTS,POLARIZED	27264	09-61-1106
A1J9	131-2484-00			TERM,SET,PIN:8 PIN,INSULATED	27264	09-61-1081
A1J10	131-2789-00			CONN,RCPT,ELEC:HEADER,1 X 4,0.156 SPACING	27264	09-61-1045
A1J11	131-3056-00			CONN,RCPT,ELEC:HEADER,1 X 18,0.156 SPACING		
A1R12	307-0675-00			RES NTWK,FXD FI:9,1K OHM,2%,1.25W	01121	210A102
A1R34	315-0272-00			RES.,FXD,CMPSN:2.7K OHM,5%,0.25W	01121	CB2725
A1R35	315-0273-00			RES.,FXD,CMPSN:27K OHM,5%,0.25W	01121	CB2735
A1R43	315-0201-00			RES.,FXD,CMPSN:200 OHM,5%,0.25W	01121	CB2015
A1R45	315-0102-00			RES.,FXD,CMPSN:1K OHM,5%,0.25W	01121	CB1025
A1R111	307-0596-00			RES NTWK,FXD FI:7,2.2K OHM,2%,1.0W	91637	MSP08A01222G
A1R142	315-0222-00			RES.,FXD,CMPSN:2.2K OHM,5%,0.25W	01121	CB2225
A1R143	315-0222-00			RES.,FXD,CMPSN:2.2K OHM,5%,0.25W	01121	CB2225
A1R145	307-0596-00			RES NTWK,FXD FI:7,2.2K OHM,2%,1.0W	91637	MSP08A01222G
A1R212	315-0221-00			RES.,FXD,CMPSN:220 OHM,5%,0.25W	01121	CB2215
A1R215	307-0675-00			RES NTWK,FXD FI:9,1K OHM,2%,1.25W	01121	210A102
A1R241	307-0675-00			RES NTWK,FXD FI:9,1K OHM,2%,1.25W	01121	210A102
A1R243	315-0102-00			RES.,FXD,CMPSN:1K OHM,5%,0.25W	01121	CB1025
A1R245	315-0511-00			RES.,FXD,CMPSN:510 OHM,5%,0.25W	01121	CB5115
A1R246	315-0511-00			RES.,FXD,CMPSN:510 OHM,5%,0.25W	01121	CB5115
A1R312	307-0596-00			RES NTWK,FXD FI:7,2.2K OHM,2%,1.0W	91637	MSP08A01222G
A1R342	307-0596-00			RES NTWK,FXD FI:7,2.2K OHM,2%,1.0W	91637	MSP08A01222G
A1R344	307-0596-00			RES NTWK,FXD FI:7,2.2K OHM,2%,1.0W	91637	MSP08A01222G
A1U22	156-1252-01			MICROCIRCUIT,DI:8-LINE-TO-3-LINE PRI ENCDR	01295	SN74LS148(NP3 OR
A1U25	156-0541-02			MICROCIRCUIT,DI:DUAL 2 TO 4 LINE DCDR	01295	SN74LS139NP3
A1U41	156-1600-00			MICROCIRCUIT,DI:DUAL RETRIG MONO MULTIVIB	01295	SN74LS123NP3
A1U45	156-0094-02			MICROCIRCUIT,DI:DUAL 2-INP NAND DRVR	01295	SN75451(NP3 OR J

Component No.	Tektronix Part No.	Serial/Model No. Eff	Dscont	Name & Description	Mfr Code	Mfr Part Number
				A2 PROCESSOR		
A2	672-1140-00			CKT BOARD ASSY:PROCESSOR	80009	672-1140-00
A2BT10	146-0040-00			BATTERY,STORAGE:2.4V,70MAH,AAA CELL	55857	MS0702405333-002
A2C11	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A2C31	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A2C75	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A2C135	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A2C155	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A2C170	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A2C202	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A2C216	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A2C231	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A2C270	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A2C355	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A2C360	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A2C365	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A2C401	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A2C455	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A2C501	283-0422-00			CAP.,FXD,CER DI:0.047UF,+80-20%,50V	04222	DG015E473Z
A2C502	283-0422-00			CAP.,FXD,CER DI:0.047UF,+80-20%,50V	04222	DG015E473Z
A2C503	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A2C504	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A2C505	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A2C511	283-0422-00			CAP.,FXD,CER DI:0.047UF,+80-20%,50V	04222	DG015E473Z
A2C512	283-0422-00			CAP.,FXD,CER DI:0.047UF,+80-20%,50V	04222	DG015E473Z
A2C513	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A2C514	283-0422-00			CAP.,FXD,CER DI:0.047UF,+80-20%,50V	04222	DG015E473Z
A2C525	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A2C526	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A2C536	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A2C541	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A2C551	290-0745-00			CAP.,FXD,ELCTLT:22UF,+50-10%,25V	54473	ECE-A25V22L
A2C555	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A2C560	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A2C562	290-0745-00			CAP.,FXD,ELCTLT:22UF,+50-10%,25V	54473	ECE-A25V22L
A2C565	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A2C570	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A2C575	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A2C576	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A2CR9	152-0141-02			SEMICOND DEVICE:SILICON,30V,150MA	01295	1N4152R
A2CR10	152-0141-02			SEMICOND DEVICE:SILICON,30V,150MA	01295	1N4152R
A2CR11	152-0141-02			SEMICOND DEVICE:SILICON,30V,150MA	01295	1N4152R
A2CR262	152-0141-02			SEMICOND DEVICE:SILICON,30V,150MA	01295	1N4152R
A2CR525	152-0141-02			SEMICOND DEVICE:SILICON,30V,150MA	01295	1N4152R
A2CR526	152-0141-02			SEMICOND DEVICE:SILICON,30V,150MA	01295	1N4152R
A2J61	131-0608-00			TERMINAL,PIN:0.365 L X 0.025 PH BRZ GOLD	22526	47357
A2J61	----- -----			(QUANTITY OF 2)		
A2J62	131-0608-00			TERMINAL,PIN:0.365 L X 0.025 PH BRZ GOLD	22526	47357
A2J62	----- -----			(QUANTITY OF 2)		
A2J63	131-0608-00			TERMINAL,PIN:0.365 L X 0.025 PH BRZ GOLD	22526	47357
A2J63	----- -----			(QUANTITY OF 2)		
A2J103	131-1789-00			CONN,RCPT,ELEC:RT-ANGLE,2/10 0.025 SQ PINS	22526	65268-008
A2J125	131-0608-00			TERMINAL,PIN:0.365 L X 0.025 PH BRZ GOLD	22526	47357
A2J125	----- -----			(QUANTITY OF 2)		

Component No.	Tektronix Part No.	Serial/Model No. Eff	Dscont	Name & Description	Mfr Code	Mfr Part Number
A2J150	131-0608-00			TERMINAL,PIN:0.365 L X 0.025 PH BRZ GOLD	22526	47357
A2J150	----- -----			(QUANTITY OF 2)		
A2J155	131-0608-00			TERMINAL,PIN:0.365 L X 0.025 PH BRZ GOLD	22526	47357
A2J155	----- -----			(QUANTITY OF 2)		
A2J226	131-0608-00			TERMINAL,PIN:0.365 L X 0.025 PH BRZ GOLD	22526	47357
A2J226	----- -----			(QUANTITY OF 5)		
A2J326	131-0608-00			TERMINAL,PIN:0.365 L X 0.025 PH BRZ GOLD	22526	47357
A2J326	----- -----			(QUANTITY OF 5)		
A2J426	131-0608-00			TERMINAL,PIN:0.365 L X 0.025 PH BRZ GOLD	22526	47357
A2J426	----- -----			(QUANTITY OF 5)		
A2J427	131-0608-00			TERMINAL,PIN:0.365 L X 0.025 PH BRZ GOLD	22526	47357
A2J427	----- -----			(QUANTITY OF 5)		
A2J522	131-0608-00			TERMINAL,PIN:0.365 L X 0.025 PH BRZ GOLD	22526	47357
A2J522	----- -----			(QUANTITY OF 3)		
A2R9	315-0271-00			RES.,FXD,CMPSN:270 OHM,5%,0.25W	01121	CB2715
A2R11	315-0361-00			RES.,FXD,CMPSN:360 OHM,5%,0.25W	01121	CB3615
A2R12	315-0751-00			RES.,FXD,CMPSN:750 OHM,5%,0.25W	01121	CB7515
A2R13	315-0751-00			RES.,FXD,CMPSN:750 OHM,5%,0.25W	01121	CB7515
A2R31	307-0502-00			RES NTWK,FXD,FI:(9) 1.8K OHM,20%,0.125W	91637	MSP10A01-182M
A2R55	307-0650-00			RES NTWK,FXD,FI:9,2.7K OHM,5%,0.150W	32997	4310R-101-272
A2R61	315-0472-00			RES.,FXD,CMPSN:4.7K OHM,5%,0.25W	01121	CB4725
A2R62	315-0472-00			RES.,FXD,CMPSN:4.7K OHM,5%,0.25W	01121	CB4725
A2R63	315-0472-00			RES.,FXD,CMPSN:4.7K OHM,5%,0.25W	01121	CB4725
A2R158	315-0511-00			RES.,FXD,CMPSN:510 OHM,5%,0.25W	01121	CB5115
A2R160	315-0511-00			RES.,FXD,CMPSN:510 OHM,5%,0.25W	01121	CB5115
A2R175	315-0272-00			RES.,FXD,CMPSN:2.7K OHM,5%,0.25W	01121	CB2725
A2R216	307-0637-00			RES NTWK,FXD,FI:5,2K OHM,2%,0.125W	01121	206A202
A2R250	307-0650-00			RES NTWK,FXD,FI:9,2.7K OHM,5%,0.150W	32997	4310R-101-272
A2R260	315-0272-00			RES.,FXD,CMPSN:2.7K OHM,5%,0.25W	01121	CB2725
A2R350	307-0650-00			RES NTWK,FXD,FI:9,2.7K OHM,5%,0.150W	32997	4310R-101-272
A2R440	307-0446-00			RES,NTWK,FXD FI:10K OHM,20%,(9) RES	91637	MSP10A01-103M
A2R525	315-0102-00			RES.,FXD,CMPSN:1K OHM,5%,0.25W	01121	CB1025
A2R526	315-0102-00			RES.,FXD,CMPSN:1K OHM,5%,0.25W	01121	CB1025
A2R550	307-0446-00			RES,NTWK,FXD FI:10K OHM,20%,(9) RES	91637	MSP10A01-103M
A2U25	156-0956-02			MICROCIRCUIT,DI:OCTAL BFR W/3 STATE OUT	01295	SN74LS244NP3
A2U35	160-0843-00			MICROCIRCUIT,DI:MICROCOMPUTER,PRGM,SCRN	80009	160-0843-00
A2U40	156-1643-00			MICROCIRCUIT,DI:HMOS,NUMERIC PROCESSOR EXT	34649	D OR C 8087-3
A2U45	156-1416-00			MICROCIRCUIT,DI:16 BIT UP,SCREENED	34649	D8086
A2U60	156-0145-02			MICROCIRCUIT,DI:QUAD 2-INP NAND BFR	01295	SN7438
A2U65	156-0385-02			MICROCIRCUIT,DI:HEX INVERTER	01295	SN74LS04
A2U70	156-0473-02			MICROCIRCUIT,DI:DUAL 5-INP NAND GATE,SCRN	27014	DM8092N/A+
A2U75	156-0479-02			MICROCIRCUIT,DI:QUAD 2-INP OR GATE	01295	SN74LS32NP3
A2U101	156-0382-02			MICROCIRCUIT,DI:QUAD 2-INP NAND GATE	01295	SN74LS00
A2U115	156-0041-05			MICROCIRCUIT,DI:DUAL D-TYPE FF,BURN-IN	01295	SN7474
A2U125	156-0385-02			MICROCIRCUIT,DI:HEX INVERTER	01295	SN74LS04
A2U130	156-0469-02			MICROCIRCUIT,DI:3/8 LINE DCDR	01295	SN74LS138NP3
A2U135	156-0140-02			MICROCIRCUIT,DI:HEX BUFFERS W/OC HV OUT	27014	DM8017NA+/JA+
A2U145	156-0478-02			MICROCIRCUIT,DI:DUAL 4 INP & GATE,BURN-IN	01295	SN74LS21NP3
A2U150	156-1065-01			MICROCIRCUIT,DI:OCTAL D TYPE TRANS LATCHES	34335	AM74LS373
A2U155	156-1428-02			MICROCIRCUIT,DI:CLOCK GENERATOR & DRIVER	34649	QD8284A
A2U165	156-0694-02			MICROCIRCUIT,DI:DCDR/3 LINE TO 8 LINE,SCRN	07263	74S138DCQR
A2U170	156-0481-02			MICROCIRCUIT,DI:TRIPLE 3 INP & GATE	27014	DM74LS11NA+
A2U175	156-1059-01			MICROCIRCUIT,DI:DUAL J-K EDGE TRIGGERED	01295	SN74LS109A
A2U201	156-0478-02			MICROCIRCUIT,DI:DUAL 4 INP & GATE,BURN-IN	01295	SN74LS21NP3

Component No.	Tektronix Part No.	Serial/Model No. Eff	Dscont	Name & Description	Mfr Code	Mfr Part Number
A2U202	156-0382-02			MICROCIRCUIT,DI:QUAD 2-INP NAND GATE	01295	SN74LS00
A2U215	156-0383-02			MICROCIRCUIT,DI:QUAD 2-INP NOR GATE	01295	SN74LS02
A2U216	156-0385-02			MICROCIRCUIT,DI:HEX INVERTER	01295	SN74LS04
A2U225	156-0383-02			MICROCIRCUIT,DI:QUAD 2-INP NOR GATE	01295	SN74LS02
A2U226	156-0865-02			MICROCIRCUIT,DI:OCTAL D-TYPE FF W/CLEAR	01295	SN74LS273NP3
A2U230	156-1036-01			MICROCIRCUIT,DI:PRGM INTERVAL TIMER	34649	QD8253
A2U235	156-1172-01			MICROCIRCUIT,DI:DUAL 4 BIT CNTR	01295	SN74LS393
A2U240	156-0392-03			MICROCIRCUIT,DI:QUAD LATCH W/CLEAR	01295	SN74S175NP3
A2U245	156-0385-02			MICROCIRCUIT,DI:HEX INVERTER	01295	SN74LS04
A2U250	156-1065-01			MICROCIRCUIT,DI:OCTAL D TYPE TRANS LATCHES	34335	AM74LS373
A2U255	156-1065-01			MICROCIRCUIT,DI:OCTAL D TYPE TRANS LATCHES	34335	AM74LS373
A2U260	156-0465-02			MICROCIRCUIT,DI:8 INP NAND GATE	01295	SN74LS30NP3
A2U262	156-0383-02			MICROCIRCUIT,DI:QUAD 2-INP NOR GATE	01295	SN74LS02
A2U265	156-0382-02			MICROCIRCUIT,DI:QUAD 2-INP NAND GATE	01295	SN74LS00
A2U270	156-0386-02			MICROCIRCUIT,DI:TRIPLE 3-INP NAND GATE	27014	DM74LS10N
A2U275	156-1258-01			MICROCIRCUIT,DI:DUAL J-K NEG-EDGE TRIG FF	01295	SN74LS112
A2U301	156-1460-00			MICROCIRCUIT,DI:ENHANCED PRGM COMM INTFC	18324	2661-2I/CP2752
A2U325	156-0467-02			MICROCIRCUIT,DI:QUAD 2-INP NAND BFR,SCRN	01295	SN74LS38
A2U350	156-1111-02			MICROCIRCUIT,DI:OCTAL BUS TRANSCEIVERS	01295	SN74LS245JP3
A2U355	156-0118-03			MICROCIRCUIT,DI:1 DUAL J-K FF,BURN-IN	01295	SN74S112JP3
A2U360	156-0690-03			MICROCIRCUIT,DI:QUAD 2 INP NOR GATE,BURN IN	01295	SN74S02
A2U365	156-0722-02			MICROCIRCUIT,DI:TPL 3-INPUT POS NAND GATE	04713	SN74LS12NDS
A2U370	156-1204-01			MICROCIRCUIT,DI:INTERRUPT CONTROLLER,SCRN	34649	QD8259A
A2U401	156-0391-02			MICROCIRCUIT,DI:HEX LATCH W/CLEAR	01295	SN74LS174
A2U415	156-0845-02			MICROCIRCUIT,DI:6 BIT COMPARATOR,BURN-IN	80009	156-0845-02
A2U425	156-0720-02			MICROCIRCUIT,DI:HEX DRVR,4 TO 2 LINE	01295	SN74LS368
A2U450	156-1111-02			MICROCIRCUIT,DI:OCTAL BUS TRANSCEIVERS	01295	SN74LS245JP3
A2U455	156-0419-02			MICROCIRCUIT,DI:DUAL 4 INP NAND LINE DRVR	07263	74S140
A2U460	156-0955-02			MICROCIRCUIT,DI:OCTAL BFR W/3 STATE OUT	04713	SN74LS241
A2U465	156-0385-02			MICROCIRCUIT,DI:HEX INVERTER	01295	SN74LS04
A2U470	156-0140-02			MICROCIRCUIT,DI:HEX BUFFERS W/OC HV OUT	27014	DM8017NA+/JA+
A2U475	156-0473-02			MICROCIRCUIT,DI:DUAL 5-INP NAND GATE,SCRN	27014	DM8092N/A+
A2U501	156-0878-01			MICROCIRCUIT,DI:QUAD LINE RCVR,SCRN	80009	156-0878-01
A2U515	156-0878-01			MICROCIRCUIT,DI:QUAD LINE RCVR,SCRN	80009	156-0878-01
A2U525	156-0879-01			MICROCIRCUIT,DI:QUAD LINE DRIVER,SCRN	80009	156-0879-01
A2U530	156-0479-02			MICROCIRCUIT,DI:QUAD 2-INP OR GATE	01295	SN74LS32NP3
A2U535	156-1786-00			MICROCIRCUIT,DI:CMOS,STATIC RAM, 1024 X 4	T1015	HM4334P-3
A2U536	156-1786-00			MICROCIRCUIT,DI:CMOS,STATIC RAM, 1024 X 4	T1015	HM4334P-3
A2U540	156-1786-00			MICROCIRCUIT,DI:CMOS,STATIC RAM, 1024 X 4	T1015	HM4334P-3
A2U541	156-1786-00			MICROCIRCUIT,DI:CMOS,STATIC RAM, 1024 X 4	T1015	HM4334P-3
A2U550	156-1111-02			MICROCIRCUIT,DI:OCTAL BUS TRANSCEIVERS	01295	SN74LS245JP3
A2U555	156-1111-02			MICROCIRCUIT,DI:OCTAL BUS TRANSCEIVERS	01295	SN74LS245JP3
A2U560	156-1427-01			MICROCIRCUIT,DI:BUS CONTROLLER,SCREENED	34649	QD8288
A2U565	156-1065-01			MICROCIRCUIT,DI:OCTAL D TYPE TRANS LATCHES	34335	AM74LS373
A2U570	156-1065-01			MICROCIRCUIT,DI:OCTAL D TYPE TRANS LATCHES	34335	AM74LS373
A2U575	156-1065-01			MICROCIRCUIT,DI:OCTAL D TYPE TRANS LATCHES	34335	AM74LS373
A2Y155	158-0135-00			XTAL UNIT,QTZ:14.7456 MHZ,0.01%,SERIES	01807	OBD
A2Y155	----- -----			(CRYSTAL REQUIRES FOAM ADHESIVE)		

Component No.	Tektronix Part No.	Serial/Model No. Eff	Dscont	Name & Description	Mfr Code	Mfr Part Number
				A3 DISK CONTROLLER		
A3	670-8148-00			CKT BOARD ASSY:DISK CONTROLLER	80009	670-8148-00
A3	----- -----			(OPTION 44 ONLY)		
A3C2	283-0065-00			CAP.,FXD,CER DI:0.001UF,5%,100V	59660	0835-591Y5EO102J
A3C5	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A3C7	285-1076-00			CAP.,FXD,PLSTC:0.2UF,5%,100V	14752	230B1B204J
A3C15	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A3C25	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A3C35	290-0746-00			CAP.,FXD,ELCTLT:47UF,+50-10%,16V	55680	ULA1C470TEA
A3C40	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A3C55	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A3C100	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A3C102	283-0333-00			CAP.,FXD,CER DI:35PF,5%,1000V	72982	838-534 A 350>
A3C105	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A3C108	283-0635-00			CAP.,FXD,MICA D:51PF,1%,100V	00853	D151E510F0
A3C110	283-0635-00			CAP.,FXD,MICA D:51PF,1%,100V	00853	D151E510F0
A3C120	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A3C125	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A3C135	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A3C140	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A3C155	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A3C160	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A3C170	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A3C205	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A3C220	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A3C230	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A3C240	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A3C250	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A3C260	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A3C270	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A3C325	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A3C330	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A3C350	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A3C360	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A3C375	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A3C405	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A3C415	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A3C435	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A3C450	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A3C460	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A3C475	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A3C505	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A3C515	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A3C525	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A3C535	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A3C550	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A3C560	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A3C570	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A3C640	290-0745-00			CAP.,FXD,ELCTLT:22UF,+50-10%,25V	54473	ECE-A25V22L
A3CR36	152-0066-00			SEMICOND DEVICE:SILICON,400V,750MA	14433	LG4016
A3CR38	152-0066-00			SEMICOND DEVICE:SILICON,400V,750MA	14433	LG4016
A3J10	131-0608-00			TERMINAL,PIN:0.365 L X 0.025 PH BRZ GOLD	22526	47357
A3J10	----- -----			(QUANTITY OF 2)		

Component No.	Tektronix Part No.	Serial/Model No. Eff	Dscont	Name & Description	Mfr Code	Mfr Part Number
A3J200	131-0608-00			TERMINAL,PIN:0.365 L X 0.025 PH BRZ GOLD	22526	47357
A3J200	----- -----			(QUANTITY OF 10)		
A3J205	131-0589-00			TERMINAL,PIN:0.46 L X 0.025 SQ	22526	48283-029
A3J206	131-0589-00			TERMINAL,PIN:0.46 L X 0.025 SQ	22526	48283-029
A3J220	131-0608-00			TERMINAL,PIN:0.365 L X 0.025 PH BRZ GOLD	22526	47357
A3J220	----- -----			(QUANTITY OF 10)		
A3J225	131-0608-00			TERMINAL,PIN:0.365 L X 0.025 PH BRZ GOLD	22526	47357
A3J225	----- -----			(QUANTITY OF 2)		
A3J260	131-0608-00			TERMINAL,PIN:0.365 L X 0.025 PH BRZ GOLD	22526	47357
A3J510	131-0608-00			TERMINAL,PIN:0.365 L X 0.025 PH BRZ GOLD	22526	47357
A3J510	----- -----			(QUANTITY OF 5)		
A3J511	131-0608-00			TERMINAL,PIN:0.365 L X 0.025 PH BRZ GOLD	22526	47357
A3J511	----- -----			(QUANTITY OF 5)		
A3J512	131-0608-00			TERMINAL,PIN:0.365 L X 0.025 PH BRZ GOLD	22526	47357
A3J512	----- -----			(QUANTITY OF 5)		
A3J513	131-0608-00			TERMINAL,PIN:0.365 L X 0.025 PH BRZ GOLD	22526	47357
A3J513	----- -----			(QUANTITY OF 5)		
A3R3	315-0821-00			RES.,FXD,CMPSN:820 OHM,5%,0.25W	01121	CB8215
A3R5	315-0202-00			RES.,FXD,CMPSN:2K OHM,5%,0.25W	01121	CB2025
A3R27	315-0121-00			RES.,FXD,CMPSN:120 OHM,5%,0.25W	01121	CB1215
A3R28	315-0272-00			RES.,FXD,CMPSN:2.7K OHM,5%,0.25W	01121	CB2725
A3R29	315-0272-00			RES.,FXD,CMPSN:2.7K OHM,5%,0.25W	01121	CB2725
A3R30	315-0272-00			RES.,FXD,CMPSN:2.7K OHM,5%,0.25W	01121	CB2725
A3R32	315-0121-00			RES.,FXD,CMPSN:120 OHM,5%,0.25W	01121	CB1215
A3R33	315-0121-00			RES.,FXD,CMPSN:120 OHM,5%,0.25W	01121	CB1215
A3R34	315-0121-00			RES.,FXD,CMPSN:120 OHM,5%,0.25W	01121	CB1215
A3R35	315-0121-00			RES.,FXD,CMPSN:120 OHM,5%,0.25W	01121	CB1215
A3R36	315-0121-00			RES.,FXD,CMPSN:120 OHM,5%,0.25W	01121	CB1215
A3R38	315-0121-00			RES.,FXD,CMPSN:120 OHM,5%,0.25W	01121	CB1215
A3R42	315-0272-00			RES.,FXD,CMPSN:2.7K OHM,5%,0.25W	01121	CB2725
A3R45	315-0363-00			RES.,FXD,CMPSN:36K OHM,5%,0.25W	01121	CB3635
A3R46	315-0272-00			RES.,FXD,CMPSN:2.7K OHM,5%,0.25W	01121	CB2725
A3R55	315-0363-00			RES.,FXD,CMPSN:36K OHM,5%,0.25W	01121	CB3635
A3R100	315-0102-00			RES.,FXD,CMPSN:1K OHM,5%,0.25W	01121	CB1025
A3R108	321-0329-00			RES.,FXD,FILM:26.1K OHM,1%,0.125W	91637	MFF1816G26101F
A3R109	131-0566-00			BUS CONDUCTOR:DUMMY RES,2.375,22 AWG	57668	JWW-0200E0
A3R110	321-0236-00			RES.,FXD,FILM:2.8K OHM,1%,0.125W	91637	MFF1816G28000F
A3R162	315-0121-00			RES.,FXD,CMPSN:120 OHM,5%,0.25W	01121	CB1215
A3R260	307-0502-00			RES NTWK,FXD,FI:(9) 1.8K OHM,20%,0.125W	91637	MSP10A01-182M
A3R265	315-0102-00			RES.,FXD,CMPSN:1K OHM,5%,0.25W	01121	CB1025
A3R335	307-0650-00			RES NTWK,FXD,FI:9,2.7K OHM,5%,0.150W	32997	4310R-101-272
A3R340	307-0650-00			RES NTWK,FXD,FI:9,2.7K OHM,5%,0.150W	32997	4310R-101-272
A3R435	307-0445-00			RES NTWK,FXD,FI:4.7K OHM,20%,(9) RES	91637	MSP10A01-472M
A3R500	315-0102-00			RES.,FXD,CMPSN:1K OHM,5%,0.25W	01121	CB1025
A3R508	315-0102-00			RES.,FXD,CMPSN:1K OHM,5%,0.25W	01121	CB1025
A3R512	307-0650-00			RES NTWK,FXD,FI:9,2.7K OHM,5%,0.150W	32997	4310R-101-272
A3U5	156-0124-02			MICROCIRCUIT,DI:PHASE/FREQ DETECTOR,SCRN	80009	156-0124-02
A3U10	156-0530-02			MICROCIRCUIT,DI:QUAD 2-INP MUX,SCRN	01295	SN74LS157P3
A3U15	156-0617-02			MICROCIRCUIT,DI:DUAL 4 BIT CNTR,SCRN	01295	SN74393NP3
A3U17	156-0385-02			MICROCIRCUIT,DI:HEX INVERTER	01295	SN74LS04
A3U20	156-0388-03			MICROCIRCUIT,DI:DUAL D FLIP-FLOP	07263	74LS74A
A3U30	156-0385-02			MICROCIRCUIT,DI:HEX INVERTER	01295	SN74LS04
A3U40	156-0093-02			MICROCIRCUIT,DI:HEX INV BUFFER,BURN-IN	27014	DM8016
A3U45	156-0140-02			MICROCIRCUIT,DI:HEX BUFFERS W/OC HV OUT	27014	DM8017NA+/JA+

Component No.	Tektronix Part No.	Serial/Model No. Eff	Dscont	Name & Description	Mfr Code	Mfr Part Number
A3U55	156-0093-02			MICROCIRCUIT,DI:HEX INV BUFFER,BURN-IN	27014	DM8016
A3U105	156-0121-02			MICROCIRCUIT,DI:DUAL VOLTAGE-CONT MV,SCRN	80009	156-0121-02
A3U110	156-0733-02			MICROCIRCUIT,DI:DUAL MONOSTABLE MV,SCRN	04713	SN74LS221N/J
A3U120	119-0413-00			OSC,XTAL CLOCK:8 MHZ	T1043	XO-33C8
A3U120	----- -----			(CRYSTAL REQUIRES FOAM ADHESIVE)		
A3U125	156-0480-02			MICROCIRCUIT,DI:QUAD 2 INP & GATE	01295	SN74LS08NP3
A3U130	156-0481-02			MICROCIRCUIT,DI:TRIPLE 3 INP & GATE	27014	DM74LS11NA+
A3U135	156-0388-03			MICROCIRCUIT,DI:DUAL D FLIP-FLOP	07263	74LS74A
A3U140	156-0541-02			MICROCIRCUIT,DI:DUAL 2 TO 4 LINE DCDR	01295	SN74LS139NP3
A3U145	156-0530-02			MICROCIRCUIT,DI:QUAD 2-INP MUX,SCRN	01295	SN74LS157P3
A3U150	156-0914-03			MICROCIRCUIT,DI:OCT SR BBR W/3 ST OUT	27014	N74LS240N
A3U155	156-0798-02			MICROCIRCUIT,DI:DUAL 14 TO 1 LINE SEL/MUX	01295	SN74LS153
A3U160	156-0798-02			MICROCIRCUIT,DI:DUAL 14 TO 1 LINE SEL/MUX	01295	SN74LS153
A3U170	156-0480-02			MICROCIRCUIT,DI:QUAD 2 INP & GATE	01295	SN74LS08NP3
A3U225	156-0391-02			MICROCIRCUIT,DI:HEX LATCH W/CLEAR	01295	SN74LS174
A3U230	156-0382-02			MICROCIRCUIT,DI:QUAD 2-INP NAND GATE	01295	SN74LS00
A3U235	156-0914-03			MICROCIRCUIT,DI:OCT SR BBR W/3 ST OUT	27014	N74LS240N
A3U240	156-0985-01			MICROCIRCUIT,DI:DUAL 5 INPUT NOR GATE,SCRN	04713	SN74LS260
A3U245	156-0469-02			MICROCIRCUIT,DI:3/8 LINE DCDR	01295	SN74LS138NP3
A3U250	156-0651-02			MICROCIRCUIT,DI:8 BIT PRL-OUT SER SHF RGTR	01295	SN74LS164(NP3 OR
A3U255	156-0480-02			MICROCIRCUIT,DI:QUAD 2 INP & GATE	01295	SN74LS08NP3
A3U260	156-0382-02			MICROCIRCUIT,DI:QUAD 2-INP NAND GATE	01295	SN74LS00
A3U265	160-2519-00			MICROCIRCUIT,DI:32 X 8 PROM,PRGM	80009	160251900
A3U270	156-0866-02			MICROCIRCUIT,DI:13 INP NAND GATES,SCRN	80009	156-0866-02
A3U325	156-0386-02			MICROCIRCUIT,DI:TRIPLE 3-INP NAND GATE	27014	DM74LS10N
A3U330	156-0383-02			MICROCIRCUIT,DI:QUAD 2-INP NOR GATE	01295	SN74LS02
A3U340	156-1412-00			MICROCIRCUIT,DI:SGL/DBL DENS FLOPPY DISC	34649	D8272A
A3U350	156-0479-02			MICROCIRCUIT,DI:QUAD 2-INP OR GATE	01295	SN74LS32NP3
A3U355	156-0865-02			MICROCIRCUIT,DI:OCTAL D-TYPE FF W/CLEAR	01295	SN74LS273NP3
A3U360	156-0385-02			MICROCIRCUIT,DI:HEX INVERTER	01295	SN74LS04
A3U365	156-0412-02			MICROCIRCUIT,DI:SYN 4 BIT UP/DOWN CNTR	01295	SN74LS193N3
A3U370	156-0412-02			MICROCIRCUIT,DI:SYN 4 BIT UP/DOWN CNTR	01295	SN74LS193N3
A3U375	156-0385-02			MICROCIRCUIT,DI:HEX INVERTER	01295	SN74LS04
A3U425	156-0956-02			MICROCIRCUIT,DI:OCTAL BFR W/3 STATE OUT	01295	SN74LS244NP3
A3U430	156-0385-02			MICROCIRCUIT,DI:HEX INVERTER	01295	SN74LS04
A3U435	156-0467-02			MICROCIRCUIT,DI:QUAD 2-INP NAND BFR,SCRN	01295	SN74LS38
A3U450	156-0385-02			MICROCIRCUIT,DI:HEX INVERTER	01295	SN74LS04
A3U455	156-0093-02			MICROCIRCUIT,DI:HEX INV BUFFER,BURN-IN	27014	DM8016
A3U460	156-0385-02			MICROCIRCUIT,DI:HEX INVERTER	01295	SN74LS04
A3U465	156-0412-02			MICROCIRCUIT,DI:SYN 4 BIT UP/DOWN CNTR	01295	SN74LS193N3
A3U470	156-0412-02			MICROCIRCUIT,DI:SYN 4 BIT UP/DOWN CNTR	01295	SN74LS193N3
A3U475	156-0412-02			MICROCIRCUIT,DI:SYN 4 BIT UP/DOWN CNTR	01295	SN74LS193N3
A3U505	156-0694-02			MICROCIRCUIT,DI:DCDR/3 LINE TO 8 LINE,SCRN	07263	74S138DCQR
A3U510	156-0956-02			MICROCIRCUIT,DI:OCTAL BFR W/3 STATE OUT	01295	SN74LS244NP3
A3U515	156-0464-02			MICROCIRCUIT,DI:DUAL 4 INP NAND GATE	01295	SN74LS20
A3U520	156-0956-02			MICROCIRCUIT,DI:OCTAL BFR W/3 STATE OUT	01295	SN74LS244NP3
A3U525	156-0956-02			MICROCIRCUIT,DI:OCTAL BFR W/3 STATE OUT	01295	SN74LS244NP3
A3U530	156-0382-02			MICROCIRCUIT,DI:QUAD 2-INP NAND GATE	01295	SN74LS00
A3U535	156-1111-02			MICROCIRCUIT,DI:OCTAL BUS TRANSCEIVERS	01295	SN74LS245JP3
A3U540	156-1111-02			MICROCIRCUIT,DI:OCTAL BUS TRANSCEIVERS	01295	SN74LS245JP3
A3U550	156-0462-02			MICROCIRCUIT,DI:HEX INVERTER,SCREENED	01295	SN7414
A3U555	156-0539-01			MICROCIRCUIT,DI:6 BIT UNIFIED BUS COMPTR	80009	156-0539-01
A3U560	156-0385-02			MICROCIRCUIT,DI:HEX INVERTER	01295	SN74LS04
A3U565	156-0914-03			MICROCIRCUIT,DI:OCT SR BBR W/3 ST OUT	27014	N74LS240N
A3U570	156-0914-03			MICROCIRCUIT,DI:OCT SR BBR W/3 ST OUT	27014	N74LS240N
A3U575	156-0914-03			MICROCIRCUIT,DI:OCT SR BBR W/3 ST OUT	27014	N74LS240N

Component No.	Tektronix Part No.	Serial/Model No. Eff	Dscont	Name & Description	Mfr Code	Mfr Part Number
				A4 FRONT PANEL		
A4	670-8120-00			CKT BOARD ASSY:FRONT PANEL	80009	670-8120-00
A4C103	283-0370-00			CAP.,FXD,CER DI:0.027UF,5%,100V	72982	8131N153X7R0273J
A4C104	283-0238-00			CAP.,FXD,CER DI:0.01UF,10%,50V	72982	8121N075X7R0103K
A4C124	283-0167-00			CAP.,FXD,CER DI:0.1UF,10%,100V	72982	8131N145X5R0104K
A4C133	283-0164-00			CAP.,FXD,CER DI:2.2UF,20%,25V	04222	SR402E225MAA
A4C510	283-0422-00			CAP.,FXD,CER DI:0.047UF,+80-20%,50V	04222	DG015E473Z
A4C520	283-0422-00			CAP.,FXD,CER DI:0.047UF,+80-20%,50V	04222	DG015E473Z
A4C530	283-0422-00			CAP.,FXD,CER DI:0.047UF,+80-20%,50V	04222	DG015E473Z
A4C605	290-0779-00			CAP.,FXD,ELCTLT:10UF,+50-10%,50VDC	56289	502D237
A4C608	290-0779-00			CAP.,FXD,ELCTLT:10UF,+50-10%,50VDC	56289	502D237
A4C620	283-0422-00			CAP.,FXD,CER DI:0.047UF,+80-20%,50V	04222	DG015E473Z
A4C625	283-0422-00			CAP.,FXD,CER DI:0.047UF,+80-20%,50V	04222	DG015E473Z
A4C630	283-0422-00			CAP.,FXD,CER DI:0.047UF,+80-20%,50V	04222	DG015E473Z
A4C635	283-0422-00			CAP.,FXD,CER DI:0.047UF,+80-20%,50V	04222	DG015E473Z
A4D1	150-1078-00			LT EMITTING DIO:GREEN,565NM,20MA	50434	HLMP 1502
A4D2	150-1078-00			LT EMITTING DIO:GREEN,565NM,20MA	50434	HLMP 1502
A4D3	150-1078-00			LT EMITTING DIO:GREEN,565NM,20MA	50434	HLMP 1502
A4D4	150-1013-00			LAMP,LED:READOUT	01295	TIL311
A4D5	150-1013-00			LAMP,LED:READOUT	01295	TIL311
A4DS109	150-1033-00			LT EMITTING DIO:YELLOW,585NM,40MA MAX	50434	HLMP 1401
A4DS124	150-1033-00			LT EMITTING DIO:YELLOW,585NM,40MA MAX	50434	HLMP 1401
A4J440	131-0608-00			TERMINAL,PIN:0.365 L X 0.025 PH BRZ GOLD	22526	47357
A4J440	----- -----			(QUANTITY OF 3)		
A4J441	131-0608-00			TERMINAL,PIN:0.365 L X 0.025 PH BRZ GOLD	22526	47357
A4J441	----- -----			(QUANTITY OF 2)		
A4J443	131-1789-00			CONN,RCPT,ELEC:RT-ANGLE,2/10 0.025 SQ PINS	22526	65268-008
A4J446	131-0608-00			TERMINAL,PIN:0.365 L X 0.025 PH BRZ GOLD	22526	47357
A4J446	----- -----			(QUANTITY OF 4)		
A4LS125	119-1427-00			XDCR,AUDIO:6V,30MA,1-4.2 KHZ	000JB	QMB-06
A4Q122	151-0103-00			TRANSISTOR:SILICON,NPN	80009	151-0103-00
A4Q130	151-0301-00			TRANSISTOR:SILICON,PNP	27014	2N2907A
A4Q135	151-0281-00			TRANSISTOR:SILICON,NPN	03508	X16P4039
A4R101	315-0223-00			RES.,FXD,CMPSN:22K OHM,5%,0.25W	01121	CB2235
A4R102	315-0183-00			RES.,FXD,CMPSN:18K OHM,5%,0.25W	01121	CB1835
A4R104	315-0151-00			RES.,FXD,CMPSN:150 OHM,5%,0.25W	01121	CB1515
A4R105	315-0151-00			RES.,FXD,CMPSN:150 OHM,5%,0.25W	01121	CB1515
A4R106	315-0151-00			RES.,FXD,CMPSN:150 OHM,5%,0.25W	01121	CB1515
A4R107	315-0151-00			RES.,FXD,CMPSN:150 OHM,5%,0.25W	01121	CB1515
A4R108	315-0271-00			RES.,FXD,CMPSN:270 OHM,5%,0.25W	01121	CB2715
A4R121	315-0222-00			RES.,FXD,CMPSN:2.2K OHM,5%,0.25W	01121	CB2225
A4R123	315-0271-00			RES.,FXD,CMPSN:270 OHM,5%,0.25W	01121	CB2715
A4R125	315-0821-00			RES.,FXD,CMPSN:820 OHM,5%,0.25W	01121	CB8215
A4R126	315-0821-00			RES.,FXD,CMPSN:820 OHM,5%,0.25W	01121	CB8215
A4R127	315-0821-00			RES.,FXD,CMPSN:820 OHM,5%,0.25W	01121	CB8215
A4R128	315-0821-00			RES.,FXD,CMPSN:820 OHM,5%,0.25W	01121	CB8215
A4R131	315-0271-00			RES.,FXD,CMPSN:270 OHM,5%,0.25W	01121	CB2715
A4R132	315-0224-00			RES.,FXD,CMPSN:220K OHM,5%,0.25W	01121	CB2245
A4R134	315-0200-00			RES.,FXD,CMPSN:20 OHM,5%,0.25W	01121	CB2005
A4R135	315-0102-00			RES.,FXD,CMPSN:1K OHM,5%,0.25W	01121	CB1025
A4R609	307-0422-00			RES.,FXD,FILM:15 RES. NETWORK	73138	898-1-R2.4K
A4R610	315-0333-00			RES.,FXD,CMPSN:33K OHM,5%,0.25W	01121	CB3335

Component No.	Tektronix Part No.	Serial/Model No. Eff	Dscont	Name & Description	Mfr Code	Mfr Part Number
A4S1	263-0019-09			SWITCH PB ASSY:MOMENTARY	80009	263-0019-09
A4S2	263-0019-09			SWITCH PB ASSY:MOMENTARY	80009	263-0019-09
A4S3	263-0019-09			SWITCH PB ASSY:MOMENTARY	80009	263-0019-09
A4S4	263-0019-09			SWITCH PB ASSY:MOMENTARY	80009	263-0019-09
A4S5	263-0019-09			SWITCH PB ASSY:MOMENTARY	80009	263-0019-09
A4S6	263-0019-09			SWITCH PB ASSY:MOMENTARY	80009	263-0019-09
A4S7	263-0019-09			SWITCH PB ASSY:MOMENTARY	80009	263-0019-09
A4S8	263-0019-09			SWITCH PB ASSY:MOMENTARY	80009	263-0019-09
A4S9	263-0019-09			SWITCH PB ASSY:MOMENTARY	80009	263-0019-09
A4S10	263-0019-09			SWITCH PB ASSY:MOMENTARY	80009	263-0019-09
A4S11	263-0019-09			SWITCH PB ASSY:MOMENTARY	80009	263-0019-09
A4S12	263-0019-09			SWITCH PB ASSY:MOMENTARY	80009	263-0019-09
A4S13	263-0019-09			SWITCH PB ASSY:MOMENTARY	80009	263-0019-09
A4S14	263-0019-09			SWITCH PB ASSY:MOMENTARY	80009	263-0019-09
A4S15	263-0019-09			SWITCH PB ASSY:MOMENTARY	80009	263-0019-09
A4S16	263-0019-09			SWITCH PB ASSY:MOMENTARY	80009	263-0019-09
A4S17	263-0019-09			SWITCH PB ASSY:MOMENTARY	80009	263-0019-09
A4S18	260-2163-00			SWITCH,PUSH:SPDT,10 MA,24V		
A4S19	260-2163-00			SWITCH,PUSH:SPDT,10 MA,24V		
A4U110	156-0402-02			MICROCIRCUIT,LI:TIMER,CHK	27014	LM555CN/A+
A4U510	156-1065-01			MICROCIRCUIT,DI:OCTAL D TYPE TRANS LATCHES	34335	AM74LS373
A4U520	156-0736-02			MICROCIRCUIT,DI:BCD TO DECIMAL DCDR	80009	156-0736-02
A4U530	156-0153-02			MICROCIRCUIT,DI:HEX INVERTER BUFFER	27014	DM8006
A4U620	156-0392-03			MICROCIRCUIT,DI:QUAD LATCH W/CLEAR	01295	SN74S175NP3
A4U625	156-0645-02			MICROCIRCUIT,DI:HEX INV ST NAND GATES,SCRN	01295	SN74LS14
A4U630	156-0874-02			MICROCIRCUIT,DI:8 BIT ADDRESSABLE LCH	04713	SN74LS259
A4U635	156-0874-02			MICROCIRCUIT,DI:8 BIT ADDRESSABLE LCH	04713	SN74LS259

Component No.	Tektronix Part No.	Serial/Model No. Eff	Dscont	Name & Description	Mfr Code	Mfr Part Number
				A5 POWER SUPPLY		
A5	620-0324-00			CKT BOARD ASSY:POWER SUPPLY	80009	620-0324-00
A5A101	118-3324-00			SEMICOND DVC,DI:SCR,SWITCHING		
A5A102	118-3324-00			SEMICOND DVC,DI:SCR,SWITCHING		
A5A103	118-3327-00			MICROCIRCUIT,DI:REG VOIT		
A5A201	118-3328-00			ISOLATOR,OPTO:		
A5A202	118-3328-00			ISOLATOR,OPTO:		
A5C101	118-3338-00			CAPACITOR:0.47UF,250V		
A5C102	118-3338-00			CAPACITOR:0.47UF,250V		
A5C103	118-3338-00			CAPACITOR:0.47UF,250V		
A5C104	118-3333-00			CAP.,FXD,ELCTLT:470UF,200V		
A5C105	118-3333-00			CAP.,FXD,ELCTLT:470UF,200V		
A5C106	118-3333-00			CAP.,FXD,ELCTLT:470UF,200V		
A5C107	118-3333-00			CAP.,FXD,ELCTLT:470UF,200V		
A5C108	118-3338-00			CAPACITOR:0.47UF,250V		
A5C109	118-3338-00			CAPACITOR:0.47UF,250V		
A5C110	118-3331-00			CAP.,FXD,ELCTLT:100UF,10V		
A5C111	118-3338-00			CAPACITOR:0.47UF,250V		
A5C112	118-3341-00			CAP.,FXD,CER DI:0.0047UF,3KV		
A5C113	118-3339-00			CAPACITOR:0.0033UF,1000V		
A5C114	118-3491-00			CAP,FXD,PLASTIC:0.0047UF,1KV		
A5C115	290-0525-00			CAP.,FXD,ELCTLT:4.7UF,20%,50V	56289	196D475X0050KA1
A5C116	118-3334-00			CAP.,FXD,ELCTLT:1000UF,10V		
A5C117	118-3334-00			CAP.,FXD,ELCTLT:1000UF,10V		
A5C118	118-3334-00			CAP.,FXD,ELCTLT:1000UF,10V		
A5C119	118-3334-00			CAP.,FXD,ELCTLT:1000UF,10V		
A5C120	118-3334-00			CAP.,FXD,ELCTLT:1000UF,10V		
A5C121	118-3334-00			CAP.,FXD,ELCTLT:1000UF,10V		
A5C122	118-3334-00			CAP.,FXD,ELCTLT:1000UF,10V		
A5C123	118-3334-00			CAP.,FXD,ELCTLT:1000UF,10V		
A5C124	118-3332-00			CAP.,FXD,ELCTLT:6800UF,6.3V		
A5C125	118-3332-00			CAP.,FXD,ELCTLT:6800UF,6.3V		
A5C126	118-3331-00			CAP.,FXD,ELCTLT:100UF,10V		
A5C127	118-3338-00			CAPACITOR:0.47UF,250V		
A5C128	118-3339-00			CAPACITOR:0.0033UF,1000V		
A5C129	290-0525-00			CAP.,FXD,ELCTLT:4.7UF,20%,50V	56289	196D475X0050KA1
A5C130	118-3334-00			CAP.,FXD,ELCTLT:1000UF,10V		
A5C131	118-3334-00			CAP.,FXD,ELCTLT:1000UF,10V		
A5C132	118-3334-00			CAP.,FXD,ELCTLT:1000UF,10V		
A5C133	118-3334-00			CAP.,FXD,ELCTLT:1000UF,10V		
A5C134	118-3334-00			CAP.,FXD,ELCTLT:1000UF,10V		
A5C135	118-3334-00			CAP.,FXD,ELCTLT:1000UF,10V		
A5C136	118-3334-00			CAP.,FXD,ELCTLT:1000UF,10V		
A5C137	118-3334-00			CAP.,FXD,ELCTLT:1000UF,10V		
A5C138	118-3332-00			CAP.,FXD,ELCTLT:6800UF,6.3V		
A5C139	118-3332-00			CAP.,FXD,ELCTLT:6800UF,6.3V		
A5C140	118-3335-00			CAP.,FXD,ELCTLT:470UF,25V		
A5C141	118-3335-00			CAP.,FXD,ELCTLT:470UF,25V		
A5C142	118-3335-00			CAP.,FXD,ELCTLT:470UF,25V		
A5C143	118-3335-00			CAP.,FXD,ELCTLT:470UF,25V		
A5C144	118-3329-00			CAP.,FXD,ELCTLT:2200UF,10-16V		
A5C145	118-3489-00			CAP.,FXD,MTLZD:330UF,35V		

Component No.	Tektronix Part No.	Serial/Model No. Eff	Dscont	Name & Description	Mfr Code	Mfr Part Number
A5C146	118-3330-00			CAP.,FXD,ELCTLT:220UF,35V		
A5C147	118-3490-00			CAP.,FXD,ELCTLT:TANTALUM,2.2UF,16V		
A5C148	118-3492-00			CAP.,FXD,CER DI:0.0047UF,250V		
A5C201	283-0081-00			CAP.,FXD,CER DI:0.1UF,+80-20%,25V	59821	2DDU69E104Z
A5C203	283-0081-00			CAP.,FXD,CER DI:0.1UF,+80-20%,25V	59821	2DDU69E104Z
A5C204	118-3496-00			CAP.,FXD,PLASTIC:0.01UF,63VDC		
A5C207	118-3337-00			CAPACITOR:0.0068UF,63-100V		
A5CR101	118-3488-00			SEMICOND DVC,DI:PWR,400V,3A		
A5CR102	118-3488-00			SEMICOND DVC,DI:PWR,400V,3A		
A5CR103	118-3488-00			SEMICOND DVC,DI:PWR,400V,3A		
A5CR104	118-3488-00			SEMICOND DVC,DI:PWR,400V,3A		
A5CR105	118-3320-00			SEMICOND DVC,DI:ZENER		
A5CR106	152-0400-00			SEMICOND DEVICE:SILICON,400V,1A	80009	152-0400-00
A5CR107	118-3319-00			SEMICOND DVC,DI:SCHOTKY,RECT		
A5CR108	118-3319-00			SEMICOND DVC,DI:SCHOTKY,RECT		
A5CR109	118-3319-00			SEMICOND DVC,DI:SCHOTKY,RECT		
A5CR110	118-3319-00			SEMICOND DVC,DI:SCHOTKY,RECT		
A5CR111	118-3319-00			SEMICOND DVC,DI:SCHOTKY,RECT		
A5CR112	118-3319-00			SEMICOND DVC,DI:SCHOTKY,RECT		
A5CR113	118-3319-00			SEMICOND DVC,DI:SCHOTKY,RECT		
A5CR114	118-3319-00			SEMICOND DVC,DI:SCHOTKY,RECT		
A5CR115	118-3319-00			SEMICOND DVC,DI:SCHOTKY,RECT		
A5CR115	152-0400-00			SEMICOND DEVICE:SILICON,400V,1A	80009	152-0400-00
A5CR116	118-3319-00			SEMICOND DVC,DI:SCHOTKY,RECT		
A5CR117	118-3320-00			SEMICOND DVC,DI:ZENER		
A5CR118	152-0691-00			SEMICOND DEVICE:ZENER,1.0W,10%,5.1V	80009	152-0691-00
A5CR119	118-3318-00			SEMICOND DVC,DI:PWR		
A5CR120	118-3409-00			SEMICOND DVC,DI:RECT,SCHOTKY		
A5CR201	152-0062-00			SEMICOND DVC,DI:SW,SI,100V,25NA — 20V		
A5CR202	152-0062-00			SEMICOND DVC,DI:SW,SI,100V,25NA — 20V		
A5CR203	118-3321-00			SEMICOND DVC,DI:ZENER		
A5CR204	118-3494-00			SEMICOND DVC,DI:ZENER,100V		
A5CR205	152-0066-01			SEMICOND DEVICE:SILICON,400V,1A	15238	LG4012
A5CR206	152-0062-00			SEMICOND DVC,DI:SW,SI,100V,25NA — 20V		
A5CR207	152-0062-00			SEMICOND DVC,DI:SW,SI,100V,25NA — 20V		
A5CR208	118-3321-00			SEMICOND DVC,DI:ZENER		
A5CR209	118-3321-00			SEMICOND DVC,DI:ZENER		
A5CR210	152-0066-01			SEMICOND DEVICE:SILICON,400V,1A	15238	LG4012
A5F101	159-0014-00			FUSE,CARTRIDGE:3AG,5A,250V,FAST-BLOW	71400	MTH5
A5F106	118-3493-00			FUSE:PICO,0.75A		
A5F107	118-3493-00			FUSE:PICO,0.75A		
A5F114	118-3493-00			FUSE:PICO,0.75A		
A5J61	118-3353-00			CONN,RCPT,ELEC:RT ANGLE,18 PIN,GOLD PL		
A5J63	118-3567-00			CONN,RCPT,ELEC:HEADER,RTANG,2 POSITION		
A5J101	131-2663-00			CONN,RCPT,ELEC:PWR,3 MALE,250 VAC,6A	70903	17265
A5J104	118-3352-00			CONN,RCPT,ELEC:D TYPE		
A5L103	118-3497-00			COIL:INDUCTOR,RADIAL		
A5L104	118-3497-00			COIL:INDUCTOR,RADIAL		
A5L105	118-3497-00			COIL:INDUCTOR,RADIAL		
A5L106	118-3355-00			INDUCTOR:12-1/2 TURN		
A5L106	118-3497-00			COIL:INDUCTOR,RADIAL		
A5L107	118-3497-00			COIL:INDUCTOR,RADIAL		
A5L108	118-3497-00			COIL:INDUCTOR,RADIAL		
A5L109	118-3354-00			INDUCTOR:		

Component No.	Tektronix Part No.	Serial/Model No. Eff	Dscont	Name & Description	Mfr Code	Mfr Part Number
A5P101	131-1234-00			CONN,RCPT,ELEC:PWR,FEMALE,250VAC,6A	82389	EAC-305
A5P203	118-3325-00			TRANSISTOR:UJT		
A5Q101	118-3322-00			TRANSISTOR:NPN,SELECTED		
A5Q102	118-3322-00			TRANSISTOR:NPN,SELECTED		
A5Q201	118-3326-00			TRANSISTOR:UJT		
A5Q202	118-3323-00			TRANSISTOR:NPN		
A5Q204	151-0103-00			TRANSISTOR:SILICON,NPN	80009	151-0103-00
A5Q205	151-0190-00			TRANSISTOR:NPN,SI,TO-92	04713	SPS7969
A5Q206	151-0190-00			TRANSISTOR:NPN,SI,TO-92	04713	SPS7969
A5Q207	151-0188-00			TRANSISTOR:SILICON,PNP	04713	SPS6868K
A5Q208	151-0190-00			TRANSISTOR:NPN,SI,TO-92	04713	SPS7969
A5Q209	151-0190-00			TRANSISTOR:NPN,SI,TO-92	04713	SPS7969
A5Q210	151-0188-00			TRANSISTOR:SILICON,PNP	04713	SPS6868K
A5Q211	151-0302-00			TRANSISTOR:NPN,SI,TO-18	04713	ST899
A5Q212	151-0190-00			TRANSISTOR:NPN,SI,TO-92	04713	SPS7969
A5R101	301-0241-00			RES.,FXD,CMPSN:240 OHM,5%,0.50W	01121	EB2415
A5R101	344-0326-00			CLIP,ELECTRICAL:FUSE,BRASS	75915	102071
A5R102	301-0241-00			RES.,FXD,CMPSN:240 OHM,5%,0.50W	01121	EB2415
A5R103	301-0113-00			RES.,FXD,CMPSN:11K OHM,5%,0.50W	01121	EB1135
A5R104	305-0390-00			RES.,FXD,CMPSN:39 OHM,5%,2W	01121	HB3905
A5R104	311-1555-00			RES.,VAR,NONWIR:100K OHM,20%,0.5W	73138	91-77-0
A5R104	----- -----			(SMALL CIRCUIT BOARD ONLY)		
A5R105	301-0471-00			RES.,FXD,CMPSN:470 OHM,5%,0.50W	01121	EB4715
A5R108	118-3568-00			RES.,FXD,WW:68 OHM,5%,5W		
A5R109	118-3343-00			RES.,FXD,FILM:1.1 OHM,5%,2W		
A5R110	301-0273-00			RES.,FXD,CMPSN:27K OHM,5%,0.50W	01121	EB2735
A5R110	311-1562-00			RES.,VAR,NONWIR:2K OHM,20%,0.50W	73138	91-84-0
A5R110	----- -----			(SMALL CIRCUIT BOARD ONLY)		
A5R111	301-0620-00			RES.,FXD,CMPSN:62 OHM,5%,0.50W	01121	EB6205
A5R112	----- -----			(PART NUMBER NOT AVAILABLE)		
A5R113	301-0471-00			RES.,FXD,CMPSN:470 OHM,5%,0.50W	01121	EB4715
A5R115	118-3344-00			RES.,FXD,WW:33 OHM,5%,5W		
A5R116	118-3432-00			RESISTOR:1.3 OHM,0.5W,5%		
A5R116	311-1555-00			RES.,VAR,NONWIR:100K OHM,20%,0.5W	73138	91-77-0
A5R116	----- -----			(SMALL CIRCUIT BOARD ONLY)		
A5R117	118-3345-00			RES.,FXD,WW:51 OHM,5%,5W		
A5R118	315-0100-00			RES.,FXD,CMPSN:10 OHM,5%,0.25W	01121	CB1005
A5R119	315-0150-00			RES.,FXD,CMPSN:15 OHM,5%,0.25W	01121	CB1505
A5R120	315-0100-00			RES.,FXD,CMPSN:10 OHM,5%,0.25W	01121	CB1005
A5R121	315-0150-00			RES.,FXD,CMPSN:15 OHM,5%,0.25W	01121	CB1505
A5R122	311-1562-00			RES.,VAR,NONWIR:2K OHM,20%,0.50W	73138	91-84-0
A5R122	315-0102-00			RES.,FXD,CMPSN:1K OHM,5%,0.25W	01121	CB1025
A5R126	311-1222-00			RES.,VAR,NONWIR:100 OHM,20%,0.50W	32997	3386F-T04-101
A5R126	----- -----			(SMALL CIRCUIT BOARD ONLY)		
A5R142	311-1222-00			RES.,VAR,NONWIR:100 OHM,20%,0.50W	32997	3386F-T04-101
A5R142	----- -----			(SMALL CIRCUIT BOARD ONLY)		
A5R201	315-0101-00			RES.,FXD,CMPSN:100 OHM,5%,0. 25W	01121	CB1015
A5R203	315-0470-00			RES.,FXD,CMPSN:47 OHM,5%,0.25W	01121	CB4705
A5R205	301-0101-00			RES.,FXD,CMPSN:100 OHM,5%,0.50W	01121	EB1015
A5R205	315-0101-00			RES.,FXD,CMPSN:100 OHM,5%,0. 25W	01121	CB1015
A5R207	301-0101-00			RES.,FXD,CMPSN:100 OHM,5%,0.50W	01121	EB1015
A5R207	315-0101-00			RES.,FXD,CMPSN:100 OHM,5%,0. 25W	01121	CB1015
A5R208	315-0101-00			RES.,FXD,CMPSN:100 OHM,5%,0. 25W	01121	CB1015
A5R209	301-0621-00			RES.,FXD,CMPSN:620 OHM,5%,0.50W	57668	NTR501E620E

Component No.	Tektronix Part No.	Serial/Model No. Eff	Dscont	Name & Description	Mfr Code	Mfr Part Number
A5R210	315-0102-00			RES.,FXD,CMPSN:1K OHM,5%,0.25W	01121	CB1025
A5R211	315-0102-00			RES.,FXD,CMPSN:1K OHM,5%,0.25W	01121	CB1025
A5R213	301-0151-00			RES.,FXD,CMPSN:150 OHM,5%,0.50W	01121	EB1515
A5R214	315-0151-00			RES.,FXD,CMPSN:150 OHM,5%,0.25W	01121	CB1515
A5R215	315-0151-00			RES.,FXD,CMPSN:150 OHM,5%,0.25W	01121	CB1515
A5R216	315-0241-00			RES.,FXD,CMPSN:240 OHM,5%,0.25W	01121	CB2415
A5R217	315-0102-00			RES.,FXD,CMPSN:1K OHM,5%,0.25W	01121	CB1025
A5R218	315-0102-00			RES.,FXD,CMPSN:1K OHM,5%,0.25W	01121	CB1025
A5R219	315-0102-00			RES.,FXD,CMPSN:1K OHM,5%,0.25W	01121	CB1025
A5R220	315-0102-00			RES.,FXD,CMPSN:1K OHM,5%,0.25W	01121	CB1025
A5R222	315-0681-00			RES.,FXD,CMPSN:680 OHM,5%,0.25W	01121	CB6815
A5R223	315-0151-00			RES.,FXD,CMPSN:150 OHM,5%,0.25W	01121	CB1515
A5R224	315-0151-00			RES.,FXD,CMPSN:150 OHM,5%,0.25W	01121	CB1515
A5R225	315-0820-00			RES.,FXD,CMPSN:82 OHM,5%,0.25W	01121	CB8205
A5R226	315-0302-00			RES.,FXD,CMPSN:3K OHM,5%,0.25W	01121	CB3025
A5R227	315-0103-00			RES.,FXD,CMPSN:10K OHM,5%,0.25W	01121	CB1035
A5R228	315-0681-00			RES.,FXD,CMPSN:680 OHM,5%,0.25W	01121	CB6815
A5R229	315-0302-00			RES.,FXD,CMPSN:3K OHM,5%,0.25W	01121	CB3025
A5RV101	118-3347-00			RES,V SENSITIVE:		
A5S101	118-3411-00			SWITCH:		
A5S102	118-3348-00			SWITCH:LINE SELECT		
A5T101	118-3412-00			TRANSFORMER:INDUCTOR,16.0 MHZ		
A5T102	118-3413-00			TRANSFORMER:150W		
A5T103	118-3570-00			TRANSFORMER:100W		
A5THR101	118-3346-00			RES.,THERMAL:		

Component No.	Tektronix Part No.	Serial/Model No. Eff	Dscont	Name & Description	Mfr Code	Mfr Part Number
				A6 CONTROLLER DISK		
A6	119-1617-00			CONTROLLER DISK:	80009	119161700
A6	----- -----			(OPTION 03 ONLY)		
A6C51	290-0745-00			CAP.,FXD,ELCTLT:22UF,+50-10%,25V	54473	ECE-A25V22L
A6C53	281-0820-00			CAP.,FXD,CER DI:680PF,10%,50V	05397	C114K681K1X5CA
A6R2	315-0101-00			RES.,FXD,CMPSN:100 OHM,5%,0. 25W	01121	CB1015
A6R3	315-0101-00			RES.,FXD,CMPSN:100 OHM,5%,0. 25W	01121	CB1015
A6R47	315-0103-00			RES.,FXD,CMPSN:10K OHM,5%,0.25W	01121	CB1035
A6R48	315-0563-00			RES.,FXD,CMPSN:56K OHM,5%,0.25W	01121	CB5635
A61F	156-0388-03			MICROCIRCUIT,DI:DUAL D FLIP-FLOP	07263	74LS74A
A61H	156-0480-02			MICROCIRCUIT,DI:QUAD 2 INP & GATE	01295	SN74LS08NP3
A62J	156-0405-03			MICROCIRCUIT,DI:DUAL RETRIG MONOSTABLE MV	07263	9602
A63A	307-0847-00			RES NTWK,FXD,FI:12X220 OHM,12X330 OHM,5%	01121	314E221331
A63B	156-0914-02			MICROCIRCUIT,DI:OCT ST BFR W/3 STATE OUT	01295	SN74LS240
A63J	156-0462-02			MICROCIRCUIT,DI:HEX INVERTER,SCREENED	01295	SN7414
A64A	156-0153-02			MICROCIRCUIT,DI:HEX INVERTER BUFFER	27014	DM8006
A64H	156-1025-02			MICROCIRCUIT,DI:INVERTING QUAD BUS XCVR	01295	SN74LS242
A65A	156-0153-02			MICROCIRCUIT,DI:HEX INVERTER BUFFER	27014	DM8006
A65H	156-1025-02			MICROCIRCUIT,DI:INVERTING QUAD BUS XCVR	01295	SN74LS242
A65J	307-0847-00			RES NTWK,FXD,FI:12X220 OHM,12X330 OHM,5%	01121	314E221331
A66J	156-0153-02			MICROCIRCUIT,DI:HEX INVERTER BUFFER	27014	DM8006
A67J	156-0385-02			MICROCIRCUIT,DI:HEX INVERTER	01295	SN74LS04
A69A	156-1681-00			MICROCIRCUIT,DI:QUAD DIFF LINE DRIVER	34335	AM26LS31DCB
A610A	156-1315-00			MICROCKT,INTFC:QUAD DIFFERENTIAL RECEIVER	34335	AM26LS32

Component No.	Tektronix Part No.	Serial/Model No. Eff	Dscont	Name & Description	Mfr Code	Mfr Part Number
				A7 INTERFACE		
A7	672-1163-00			CKT BOARD ASSY:INTERFACE	80009	672-1163-00
A7	----- -----			(OPTION 45 ONLY)		
A7C3	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A7C10	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A7C20	283-0194-00			CAP.,FXD,CER DI:4.7UF,20%,50V	56289	5C37Z5U475M050B
A7C23	283-0212-00			CAP.,FXD,CER DI:2UF,20%,50V	51642	400-050-Z5U205M
A7C26	283-0194-00			CAP.,FXD,CER DI:4.7UF,20%,50V	56289	5C37Z5U475M050B
A7C60	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A7C76	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A7C100	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A7C106	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A7C116	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A7C123	290-0755-00			CAP.,FXD,ELCTLT:100UF,+50-10%,10V	55680	ULA1A01TEA
A7C125	283-0339-00			CAP.,FXD,CER DI:0.22UF,10%,50V	72982	8131N075W5R224K
A7C126	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A7C140	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A7C150	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A7C153	290-0755-00			CAP.,FXD,ELCTLT:100UF,+50-10%,10V	55680	ULA1A01TEA
A7C170	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A7C203	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A7C213	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A7C226	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A7C236	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A7C250	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A7C266	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A7C300	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A7C303	283-0204-00			CAP.,FXD,CER DI:0.01UF,20%,50V	96733	R2676
A7C310	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A7C316	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A7C326	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A7C333	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A7C340	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A7C360	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A7C370	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A7C403	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A7C413	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A7C420	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A7C426	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A7C436	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A7C450	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A7C476	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A7C506	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A7C513	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A7C530	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A7C536	290-0755-00			CAP.,FXD,ELCTLT:100UF,+50-10%,10V	55680	ULA1A01TEA
A7C540	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A7C546	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A7C556	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A7C566	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A7C576	283-0421-00			CAP.,FXD,CER DI:0.1UF,+80-20%,50V	04222	DG015E104Z
A7F606	159-0114-00			FUSE,CARTRIDGE:1A,125VAC,FAST-BLOW	71400	GFA 1

Component No.	Tektronix Part No.	Serial/Model No. Eff	Dscont	Name & Description	Mfr Code	Mfr Part Number
A7J106	131-0608-00			TERMINAL,PIN:0.365 L X 0.025 PH BRZ GOLD	22526	47357
A7J106	----- -----			(QUANTITY OF 2)		
A7J110	131-0608-00			TERMINAL,PIN:0.365 L X 0.025 PH BRZ GOLD	22526	47357
A7J110	----- -----			(QUANTITY OF 6)		
A7J132	131-0608-00			TERMINAL,PIN:0.365 L X 0.025 PH BRZ GOLD	22526	47357
A7J132	----- -----			(QUANTITY OF 2)		
A7J160	131-0608-00			TERMINAL,PIN:0.365 L X 0.025 PH BRZ GOLD	22526	47357
A7J160	----- -----			(QUANTITY OF 2)		
A7J166	131-0608-00			TERMINAL,PIN:0.365 L X 0.025 PH BRZ GOLD	22526	47357
A7J166	----- -----			(QUANTITY OF 2)		
A7J167	131-0608-00			TERMINAL,PIN:0.365 L X 0.025 PH BRZ GOLD	22526	47357
A7J167	----- -----			(QUANTITY OF 2)		
A7J168	131-0608-00			TERMINAL,PIN:0.365 L X 0.025 PH BRZ GOLD	22526	47357
A7J168	----- -----			(QUANTITY OF 2)		
A7J169	131-0608-00			TERMINAL,PIN:0.365 L X 0.025 PH BRZ GOLD	22526	47357
A7J169	----- -----			(QUANTITY OF 2)		
A7J205	131-0589-00			TERMINAL,PIN:0.46 L X 0.025 SQ	22526	48283-029
A7J205	----- -----			(QUANTITY OF 5)		
A7J206	131-0589-00			TERMINAL,PIN:0.46 L X 0.025 SQ	22526	48283-029
A7J206	----- -----			(QUANTITY OF 5)		
A7J216	131-2222-00			CONN,RCPT,ELEC:CKT BD,34 CONT,MALE	00779	2-86479-1
A7J306	131-0608-00			TERMINAL,PIN:0.365 L X 0.025 PH BRZ GOLD	22526	47357
A7J306	----- -----			(QUANTITY OF 2)		
A7J320	131-0608-00			TERMINAL,PIN:0.365 L X 0.025 PH BRZ GOLD	22526	47357
A7J320	----- -----			(QUANTITY OF 2)		
A7J321	131-0608-00			TERMINAL,PIN:0.365 L X 0.025 PH BRZ GOLD	22526	47357
A7J321	----- -----			(QUANTITY OF 2)		
A7J322	131-0608-00			TERMINAL,PIN:0.365 L X 0.025 PH BRZ GOLD	22526	47357
A7J322	----- -----			(QUANTITY OF 2)		
A7J354	131-0608-00			TERMINAL,PIN:0.365 L X 0.025 PH BRZ GOLD	22526	47357
A7J354	----- -----			(QUANTITY OF 3)		
A7J376	131-0608-00			TERMINAL,PIN:0.365 L X 0.025 PH BRZ GOLD	22526	47357
A7J376	----- -----			(QUANTITY OF 2)		
A7Q520	151-0190-00			TRANSISTOR:SILICON,NPN	07263	S032677
A7R17	315-0272-00			RES.,FXD,CMPSN:2.7K OHM,5%,0.25W	01121	CB2725
A7R18	315-0272-00			RES.,FXD,CMPSN:2.7K OHM,5%,0.25W	01121	CB2725
A7R19	315-0105-00			RES.,FXD,CMPSN:1M OHM,5%,0.25W	01121	CB1055
A7R20	315-0824-00			RES.,FXD,CMPSN:820K OHM,5%,0.25W	01121	CB8245
A7R21	315-0121-00			RES.,FXD,CMPSN:120 OHM,5%,0.25W	01121	CB1215
A7R22	315-0272-00			RES.,FXD,CMPSN:2.7K OHM,5%,0.25W	01121	CB2725
A7R23	315-0105-00			RES.,FXD,CMPSN:1M OHM,5%,0.25W	01121	CB1055
A7R46	315-0363-00			RES.,FXD,CMPSN:36K OHM,5%,0.25W	01121	CB3635
A7R47	315-0363-00			RES.,FXD,CMPSN:36K OHM,5%,0.25W	01121	CB3635
A7R48	315-0272-00			RES.,FXD,CMPSN:2.7K OHM,5%,0.25W	01121	CB2725
A7R60	307-0824-00			RES NTWK,FXD,FI:4,150 OHM,2%,0.3W EACH	01121	208B151
A7R61	307-0824-00			RES NTWK,FXD,FI:4,150 OHM,2%,0.3W EACH	01121	208B151
A7R62	307-0650-00			RES NTWK,FXD,FI:9,2.7K OHM,5%,0.150W	32997	4310R-101-272
A7R115	315-0272-00			RES.,FXD,CMPSN:2.7K OHM,5%,0.25W	01121	CB2725
A7R119	315-0272-00			RES.,FXD,CMPSN:2.7K OHM,5%,0.25W	01121	CB2725
A7R120	315-0272-00			RES.,FXD,CMPSN:2.7K OHM,5%,0.25W	01121	CB2725
A7R121	315-0624-00			RES.,FXD,CMPSN:620K OHM,5%,0.25W	01121	CB6245
A7R170	315-0272-00			RES.,FXD,CMPSN:2.7K OHM,5%,0.25W	01121	CB2725
A7R205	315-0272-00			RES.,FXD,CMPSN:2.7K OHM,5%,0.25W	01121	CB2725
A7R303	315-0272-00			RES.,FXD,CMPSN:2.7K OHM,5%,0.25W	01121	CB2725

Component No.	Tektronix Part No.	Serial/Model No. Eff	Dscont	Name & Description	Mfr Code	Mfr Part Number
A7R320	307-0658-00			RES NTWK,FXD,FI:14,220 OHM,14,330 OHM,2%	01121	316E221331
A7R324	315-0272-00			RES.,FXD,CMPSN:2.7K OHM,5%,0.25W	01121	CB2725
A7R325	315-0272-00			RES.,FXD,CMPSN:2.7K OHM,5%,0.25W	01121	CB2725
A7R330	315-0272-00			RES.,FXD,CMPSN:2.7K OHM,5%,0.25W	01121	CB2725
A7R331	315-0272-00			RES.,FXD,CMPSN:2.7K OHM,5%,0.25W	01121	CB2725
A7R336	315-0102-00			RES.,FXD,CMPSN:1K OHM,5%,0.25W	01121	CB1025
A7R351	315-0272-00			RES.,FXD,CMPSN:2.7K OHM,5%,0.25W	01121	CB2725
A7R352	315-0272-00			RES.,FXD,CMPSN:2.7K OHM,5%,0.25W	01121	CB2725
A7R357	315-0511-00			RES.,FXD,CMPSN:510 OHM,5%,0.25W	01121	CB5115
A7R358	315-0511-00			RES.,FXD,CMPSN:510 OHM,5%,0.25W	01121	CB5115
A7R359	315-0272-00			RES.,FXD,CMPSN:2.7K OHM,5%,0.25W	01121	CB2725
A7R375	315-0272-00			RES.,FXD,CMPSN:2.7K OHM,5%,0.25W	01121	CB2725
A7R376	315-0272-00			RES.,FXD,CMPSN:2.7K OHM,5%,0.25W	01121	CB2725
A7R403	307-0658-00			RES NTWK,FXD,FI:14,220 OHM,14,330 OHM,2%	01121	316E221331
A7R426	315-0272-00			RES.,FXD,CMPSN:2.7K OHM,5%,0.25W	01121	CB2725
A7R502	315-0330-00			RES.,FXD,CMPSN:33 OHM,5%,0.25W	01121	CB3305
A7R503	315-0272-00			RES.,FXD,CMPSN:2.7K OHM,5%,0.25W	01121	CB2725
A7R510	315-0272-00			RES.,FXD,CMPSN:2.7K OHM,5%,0.25W	01121	CB2725
A7R516	315-0272-00			RES.,FXD,CMPSN:2.7K OHM,5%,0.25W	01121	CB2725
A7R520	315-0472-00			RES.,FXD,CMPSN:4.7K OHM,5%,0.25W	01121	CB4725
A7R530	315-0272-00			RES.,FXD,CMPSN:2.7K OHM,5%,0.25W	01121	CB2725
A7R533	315-0272-00			RES.,FXD,CMPSN:2.7K OHM,5%,0.25W	01121	CB2725
A7R573	315-0272-00			RES.,FXD,CMPSN:2.7K OHM,5%,0.25W	01121	CB2725
A7U1	156-0388-03			MICROCIRCUIT,DI:DUAL D FLIP-FLOP	07263	74LS74A
A7U3	156-0798-02			MICROCIRCUIT,DI:DUAL 14 TO 1 LINE SEL/MUX	01295	SN74LS153
A7U6	156-1172-01			MICROCIRCUIT,DI:DUAL 4 BIT CNTR	01295	SN74LS393
A7U10	156-0651-02			MICROCIRCUIT,DI:8 BIT PRL-OUT SER SHF RGTR	01295	SN74LS164(NP3 OR
A7U16	156-0145-02			MICROCIRCUIT,DI:QUAD 2-INP NAND BFR	01295	SN7438
A7U30	156-0798-02			MICROCIRCUIT,DI:DUAL 14 TO 1 LINE SEL/MUX	01295	SN74LS153
A7U46	156-0093-02			MICROCIRCUIT,DI:HEX INV BUFFER,BURN-IN	27014	DM8016
A7U53	156-0530-02			MICROCIRCUIT,DI:QUAD 2-INP MUX,SCRN	01295	SN74LS157P3
A7U60	156-0462-02			MICROCIRCUIT,DI:HEX INVERTER,SCREENED	01295	SN7414
A7U63	156-1111-02			MICROCIRCUIT,DI:OCTAL BUS TRANSCEIVERS	01295	SN74LS245JP3
A7U66	156-0865-02			MICROCIRCUIT,DI:OCTAL D-TYPE FF W/CLEAR	01295	SN74LS273NP3
A7U70	156-0914-02			MICROCIRCUIT,DI:OCT ST BFR W/3 STATE OUT	01295	SN74LS240
A7U76	156-1412-00			MICROCIRCUIT,DI:SGL/DBL DENS FLOPPY DISC	34649	D8272A
A7U100	156-1258-01			MICROCIRCUIT,DI:DUAL J-K NEG-EDGE TRIG FF	01295	SN74LS112
A7U103	156-0385-02			MICROCIRCUIT,DI:HEX INVERTER	01295	SN74LS04
A7U106	156-0385-02			MICROCIRCUIT,DI:HEX INVERTER	01295	SN74LS04
A7U110	156-1258-01			MICROCIRCUIT,DI:DUAL J-K NEG-EDGE TRIG FF	01295	SN74LS112
A7U113	156-0798-02			MICROCIRCUIT,DI:DUAL 14 TO 1 LINE SEL/MUX	01295	SN74LS153
A7U116	156-0388-03			MICROCIRCUIT,DI:DUAL D FLIP-FLOP	07263	74LS74A
A7U120	156-1335-00			MICROCIRCUIT,DI:DUAL RETRIG RESET MONO MV	07263	96LS02
A7U123	156-0479-02			MICROCIRCUIT,DI:QUAD 2-INP OR GATE	01295	SN74LS32NP3
A7U126	156-0651-02			MICROCIRCUIT,DI:8 BIT PRL-OUT SER SHF RGTR	01295	SN74LS164(NP3 OR
A7U130	156-1335-00			MICROCIRCUIT,DI:DUAL RETRIG RESET MONO MV	07263	96LS02
A7U133	156-1258-01			MICROCIRCUIT,DI:DUAL J-K NEG-EDGE TRIG FF	01295	SN74LS112
A7U136	156-0541-02			MICROCIRCUIT,DI:DUAL 2 TO 4 LINE DCDR	01295	SN74LS139NP3
A7U140	156-0140-02			MICROCIRCUIT,DI:HEX BUFFERS W/OC HV OUT	27014	DM8017NA+/JA+
A7U143	156-0382-02			MICROCIRCUIT,DI:QUAD 2-INP NAND GATE	01295	SN74LS00
A7U146	156-0481-02			MICROCIRCUIT,DI:TRIPLE 3 INP & GATE	27014	DM74LS11NA+
A7U150	156-0480-02			MICROCIRCUIT,DI:QUAD 2 INP & GATE	01295	SN74LS08NP3
A7U153	156-1888-00			MICROCIRCUIT,DI:MOS,FLOPPY DISK DATA SEP		
A7U160	156-0385-02			MICROCIRCUIT,DI:HEX INVERTER	01295	SN74LS04

Component No.	Tektronix Part No.	Serial/Model No. Eff	Dscont	Name & Description	Mfr Code	Mfr Part Number
A7U163	156-1111-02			MICROCIRCUIT,DI:OCTAL BUS TRANSCEIVERS	01295	SN74LS245JP3
A7U165	156-0956-02			MICROCIRCUIT,DI:OCTAL BFR W/3 STATE OUT	01295	SN74LS244NP3
A7U170	156-0480-02			MICROCIRCUIT,DI:QUAD 2 INP & GATE	01295	SN74LS08NP3
A7U200	156-0381-02			MICROCIRCUIT,DI:QUAD 2-INP EXCL OR GATE	01295	SN74LS86
A7U203	156-0388-03			MICROCIRCUIT,DI:DUAL D FLIP-FLOP	07263	74LS74A
A7U206	156-0388-03			MICROCIRCUIT,DI:DUAL D FLIP-FLOP	07263	74LS74A
A7U210	156-0383-02			MICROCIRCUIT,DI:QUAD 2-INP NOR GATE	01295	SN74LS02
A7U213	156-0392-03			MICROCIRCUIT,DI:QUAD LATCH W/CLEAR	01295	SN74S175NP3
A7U216	156-0865-02			MICROCIRCUIT,DI:OCTAL D-TYPE FF W/CLEAR	01295	SN74LS273NP3
A7U220	160-2021-00			MICROCIRCUIT,DI:512 X 8 EPROM,PRGM	80009	160-2021-00
A7U223	156-0865-02			MICROCIRCUIT,DI:OCTAL D-TYPE FF W/CLEAR	01295	SN74LS273NP3
A7U226	156-0985-01			MICROCIRCUIT,DI:DUAL 5 INPUT NOR GATE,SCRN	04713	SN74LS260
A7U230	156-0530-02			MICROCIRCUIT,DI:QUAD 2-INP MUX,SCRN	01295	SN74LS157P3
A7U233	156-0383-02			MICROCIRCUIT,DI:QUAD 2-INP NOR GATE	01295	SN74LS02
A7U236	156-0469-02			MICROCIRCUIT,DI:3/8 LINE DCDR	01295	SN74LS138NP3
A7U240	156-0469-02			MICROCIRCUIT,DI:3/8 LINE DCDR	01295	SN74LS138NP3
A7U243	156-0479-02			MICROCIRCUIT,DI:QUAD 2-INP OR GATE	01295	SN74LS32NP3
A7U246	156-1065-01			MICROCIRCUIT,DI:OCTAL D TYPE TRANS LATCHES	34335	AM74LS373
A7U250	156-0718-03			MICROCIRCUIT,DI:TRIPLE 3-INP NOR GATE	01295	SN74LS27
A7U253	160-2020-00			MICROCIRCUIT,DI:16384 X 8 EPROM,PRGM	80009	160-2020-00
A7U260	156-1734-00			MICROCIRCUIT,DI:8192 X 8 PSUEDO STATIC RAM		
A7U266	156-0382-02			MICROCIRCUIT,DI:QUAD 2-INP NAND GATE	01295	SN74LS00
A7U270	156-0388-03			MICROCIRCUIT,DI:DUAL D FLIP-FLOP	07263	74LS74A
A7U273	156-0383-02			MICROCIRCUIT,DI:QUAD 2-INP NOR GATE	01295	SN74LS02
A7U276	160-2153-00			MICROCIRCUIT,DI:HEX 12 INP AOI GATE ARRAY	80009	160-2153-00
A7U300	156-0735-02			MICROCIRCUIT,DI:4 BIT BISTABLE LCH,BURN-IN	01295	SN74LS75
A7U303	156-0382-02			MICROCIRCUIT,DI:QUAD 2-INP NAND GATE	01295	SN74LS00
A7U306	156-1065-01			MICROCIRCUIT,DI:OCTAL D TYPE TRANS LATCHES	34335	AM74LS373
A7U310	160-2022-00			MICROCIRCUIT,DI:512 X 8 PROM,PRGM	80009	160-2022-00
A7U313	156-0956-02			MICROCIRCUIT,DI:OCTAL BFR W/3 STATE OUT	01295	SN74LS244NP3
A7U316	156-0865-02			MICROCIRCUIT,DI:OCTAL D-TYPE FF W/CLEAR	01295	SN74LS273NP3
A7U323	156-0994-02			MICROCIRCUIT,DI:8 INPUT DATA SEL/MUX	01295	SN74LS151NP3
A7U326	156-0385-02			MICROCIRCUIT,DI:HEX INVERTER	01295	SN74LS04
A7U330	156-0388-03			MICROCIRCUIT,DI:DUAL D FLIP-FLOP	07263	74LS74A
A7U333	156-0385-02			MICROCIRCUIT,DI:HEX INVERTER	01295	SN74LS04
A7U336	156-0865-02			MICROCIRCUIT,DI:OCTAL D-TYPE FF W/CLEAR	01295	SN74LS273NP3
A7U340	156-0434-00			MICROCIRCUIT,DI:8 X 16 I/O PROCESSOR,SCRN		
A7U346	156-1111-02			MICROCIRCUIT,DI:OCTAL BUS TRANSCEIVERS	01295	SN74LS245JP3
A7U350	156-1428-02			MICROCIRCUIT,DI:CLOCK GENERATOR & DRIVER	34649	QD8284A
A7U360	156-0140-02			MICROCIRCUIT,DI:HEX BUFFERS W/OC HV OUT	27014	DM8017NA+/JA+
A7U363	156-0866-02			MICROCIRCUIT,DI:13 INP NAND GATES,SCRN	80009	156-0866-02
A7U370	160-2018-04			MICROCIRCUIT,DI:16384 X 8 EPROM,PRGM	80009	160-2018-04
A7U370	160-2015-00			MICROCIRCUIT,DI:16384 X 8 EPROM,PRGM	80009	160-2015-00
A7U376	160-2019-04			MICROCIRCUIT,DI:16384 X 8 EPROM,PRGM	80009	160-2019-04
A7U376	160-2017-00			MICROCIRCUIT,DI:16384 X 8 EPROM,PRGM	80009	160-2017-00
A7U400	156-1065-01			MICROCIRCUIT,DI:OCTAL D TYPE TRANS LATCHES	34335	AM74LS373
A7U406	156-0865-02			MICROCIRCUIT,DI:OCTAL D-TYPE FF W/CLEAR	01295	SN74LS273NP3
A7U410	156-0914-02			MICROCIRCUIT,DI:OCT ST BFR W/3 STATE OUT	01295	SN74LS240
A7U413	156-0914-02			MICROCIRCUIT,DI:OCT ST BFR W/3 STATE OUT	01295	SN74LS240
A7U416	156-0865-02			MICROCIRCUIT,DI:OCTAL D-TYPE FF W/CLEAR	01295	SN74LS273NP3
A7U420	156-0111-02			MICROCIRCUIT,DI:BCD TO DEC DCDR/DRVR,SCRN	01295	SN74145NP3
A7U423	156-0480-02			MICROCIRCUIT,DI:QUAD 2 INP & GATE	01295	SN74LS08NP3
A7U426	156-0480-02			MICROCIRCUIT,DI:QUAD 2 INP & GATE	01295	SN74LS08NP3
A7U430	156-0385-02			MICROCIRCUIT,DI:HEX INVERTER	01295	SN74LS04

Component No.	Tektronix Part No.	Serial/Model No. Eff	Dscont	Name & Description	Mfr Code	Mfr Part Number
A7U433	156-0469-02			MICROCIRCUIT,DI:3/8 LINE DCDR	01295	SN74LS138NP3
A7U436	156-0480-02			MICROCIRCUIT,DI:QUAD 2 INP & GATE	01295	SN74LS08NP3
A7U446	156-0956-02			MICROCIRCUIT,DI:OCTAL BFR W/3 STATE OUT	01295	SN74LS244NP3
A7U450	156-1111-02			MICROCIRCUIT,DI:OCTAL BUS TRANSCEIVERS	01295	SN74LS245JP3
A7U453	156-1065-01			MICROCIRCUIT,DI:OCTAL D TYPE TRANS LATCHES	34335	AM74LS373
A7U456	156-0956-02			MICROCIRCUIT,DI:OCTAL BFR W/3 STATE OUT	01295	SN74LS244NP3
A7U460	156-0956-02			MICROCIRCUIT,DI:OCTAL BFR W/3 STATE OUT	01295	SN74LS244NP3
A7U463	156-0865-02			MICROCIRCUIT,DI:OCTAL D-TYPE FF W/CLEAR	01295	SN74LS273NP3
A7U466	156-1273-01			MICROCIRCUIT,DI:8 BIT EQUAL TO COMPTR,SCRN	80009	156-1273-01
A7U470	160-2143-00			MICROCIRCUIT,DI:HEX 12 INP AOI GATE ARRAY	80009	160-2143-00
A7U473	156-0956-02			MICROCIRCUIT,DI:OCTAL BFR W/3 STATE OUT	01295	SN74LS244NP3
A7U476	156-0866-02			MICROCIRCUIT,DI:13 INP NAND GATES,SCRN	80009	156-0866-02
A7U500	156-0914-02			MICROCIRCUIT,DI:OCT ST BFR W/3 STATE OUT	01295	SN74LS240
A7U503	156-0145-02			MICROCIRCUIT,DI:QUAD 2-INP NAND BFR	01295	SN7438
A7U506	156-0145-02			MICROCIRCUIT,DI:QUAD 2-INP NAND BFR	01295	SN7438
A7U510	156-0145-02			MICROCIRCUIT,DI:QUAD 2-INP NAND BFR	01295	SN7438
A7U513	156-0645-02			MICROCIRCUIT,DI:HEX INV ST NAND GATES,SCRN	01295	SN74LS14
A7U516	156-0915-02			MICROCIRCUIT,DI:9 BIT ODD/EVEN PARITY GEN	80009	156-0915-02
A7U520	156-0651-02			MICROCIRCUIT,DI:8 BIT PRL-OUT SER SHF RGTR	01295	SN74LS164(NP3 OR
A7U523	156-0956-02			MICROCIRCUIT,DI:OCTAL BFR W/3 STATE OUT	01295	SN74LS244NP3
A7U526	156-0145-02			MICROCIRCUIT,DI:QUAD 2-INP NAND BFR	01295	SN7438
A7U530	156-0388-03			MICROCIRCUIT,DI:DUAL D FLIP-FLOP	07263	74LS74A
A7U533	156-0382-02			MICROCIRCUIT,DI:QUAD 2-INP NAND GATE	01295	SN74LS00
A7U536	156-1427-01			MICROCIRCUIT,DI:BUS CONTROLLER,SCREENED	34649	QD8288
A7U540	156-0645-02			MICROCIRCUIT,DI:HEX INV ST NAND GATES,SCRN	01295	SN74LS14
A7U543	156-0435-00			MICROCIRCUIT,DI:BUS ARBITER,BIPOLAR,SCRN	34649	8289
A7U546	156-0956-02			MICROCIRCUIT,DI:OCTAL BFR W/3 STATE OUT	01295	SN74LS244NP3
A7U550	156-1111-02			MICROCIRCUIT,DI:OCTAL BUS TRANSCEIVERS	01295	SN74LS245JP3
A7U553	156-1111-02			MICROCIRCUIT,DI:OCTAL BUS TRANSCEIVERS	01295	SN74LS245JP3
A7U556	156-0956-02			MICROCIRCUIT,DI:OCTAL BFR W/3 STATE OUT	01295	SN74LS244NP3
A7U560	156-1065-01			MICROCIRCUIT,DI:OCTAL D TYPE TRANS LATCHES	34335	AM74LS373
A7U563	156-1065-01			MICROCIRCUIT,DI:OCTAL D TYPE TRANS LATCHES	34335	AM74LS373
A7U566	156-0956-02			MICROCIRCUIT,DI:OCTAL BFR W/3 STATE OUT	01295	SN74LS244NP3
A7U570	156-1065-01			MICROCIRCUIT,DI:OCTAL D TYPE TRANS LATCHES	34335	AM74LS373
A7U573	156-0956-02			MICROCIRCUIT,DI:OCTAL BFR W/3 STATE OUT	01295	SN74LS244NP3
A7U576	156-0956-02			MICROCIRCUIT,DI:OCTAL BFR W/3 STATE OUT	01295	SN74LS244NP3
A7Y13	119-1408-00			OSC,XTAL CLOCK:16MHZ,0.01%	000IZ	X0-33B16

Component No.	Tektronix Part No.	Serial/Model No. Eff	Dscont	Name & Description	Mfr Code	Mfr Part Number
				A8 LINE TERMINATOR		
A8	011-0090-00			TERMN,LINE:		
A8	----- -----			(OPTION 45 ONLY)		
A8J1	131-3142-00			CONN,RCPT,ELEC:CKT BD,EDGE MNT,50 CONTACT		
A8R10	307-0658-00			RES NTWK,FXD,FI:14,220 OHM,14,330 OHM,2%	01121	316E221331
A8R20	307-0658-00			RES NTWK,FXD,FI:14,220 OHM,14,330 OHM,2%	01121	316E221331
				A9 LOGIC EXTENDER		
A9	670-5291-XX			CKT BOARD ASSY:LOGIC EXTENDER		
A9	----- -----			(NOT AVAILABLE,USE 067-1005-00)		
A9J611	131-1346-00			CONNECTOR,RCPT,:40/80 DOUBLE ROW	05574	000201-5440
A9J612	131-1606-01			CONNECTOR,RCPT,:W/22-44 CONTACTS	80009	131-1606-01
A9TP1-80	214-0579-00			TERM,TEST POINT:BRS CD PL	80009	214-0579-00
A9TP1-80	----- -----			(ALL TEST POINTS ARE THE SAME)		

Component No.	Tektronix Part No.	Serial/Model No. Eff	Dscont	Name & Description	Mfr Code	Mfr Part Number
				CHASSIS PARTS		
B1001	119-0026-00			FAN,AXIAL:1.500 X 4.750 INCH,WHISPER	82877	WR2A1
J3005	131-0955-00			CONN,RCPT,ELEC:BNC,FEMALE	13511	31-279

Section 16
DIAGRAMS AND SCHEMATICS

Symbols and Reference Designators

Electrical components shown on the diagrams are in the following units unless noted otherwise:

Capacitors = Values one or greater are in picofarads (pF).
Values less than one are in microfarads (μF).

Resistors = Ohms (Ω).

Graphic symbols and class designation letters are based on ANSI Standard Y32.2-1975.

Logic symbology is based on ANSI Y32.14-1973 in terms of positive logic. Logic symbols depict the logic function performed and may differ from the manufacturer's data.

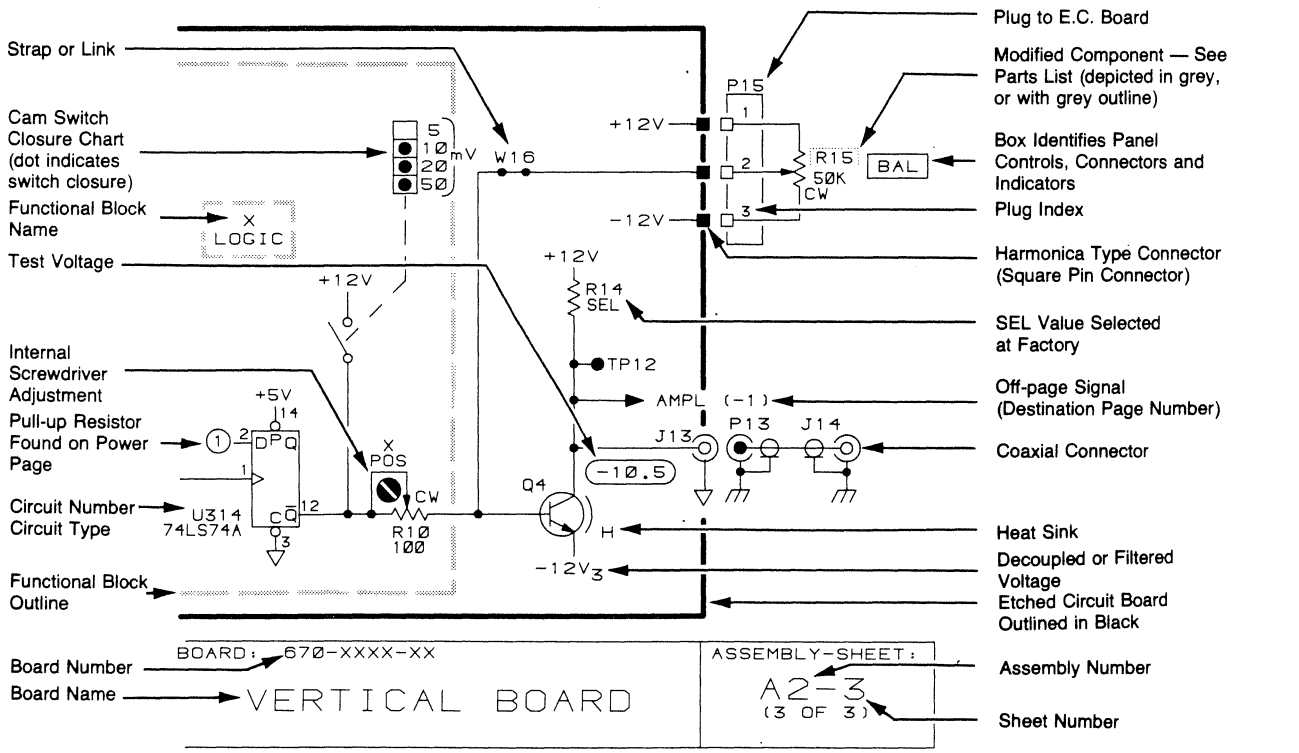
Abbreviations are based on ANSI Y1.1-1972. Other ANSI standards that are used in the preparation of diagrams by Tektronix, Inc., are:

Y14.15, 1966	Drafting Practices.
Y14.2, 1973	Line Conventions and Lettering.
Y10.5, 1968	Letter Symbols for Quantities Used in Electrical Science and Electrical Engineering.

The following prefix letters are used as reference designators to identify components or assemblies on the diagrams.

A	Assembly, separable or repairable (circuit board, etc.)
AT	Attenuator, fixed or variable
B	Motor
BT	Battery
C	Capacitor, fixed or variable
CB	Circuit breaker
CR	Diode, signal or rectifier
DL	Delay line
DS	Indicating device (lamp)
E	Spark Gap, Ferrite bead
F	Fuse
FL	Filter
H	Heat dissipating device (heat sink, heat radiator, etc.)
HR	Heater
HY	Hybrid circuit
J	Connector, stationary portion
K	Relay
L	Inductor, fixed or variable
M	Meter
P	Connector, movable portion
Q	Transistor or silicon-controlled rectifier
R	Resistor, fixed or variable
RT	Thermistor
S	Switch or contactor
T	Transformer
TC	Thermocouple
TP	Test point
U	Assembly, inseparable or non-repairable (integrated circuit, etc.)
V	Electron tube
VR	Voltage regulator (zener diode, etc.)
W	Wirestrap or cable
Y	Crystal
Z	Phase shifter

The following special symbols may appear on the diagrams:



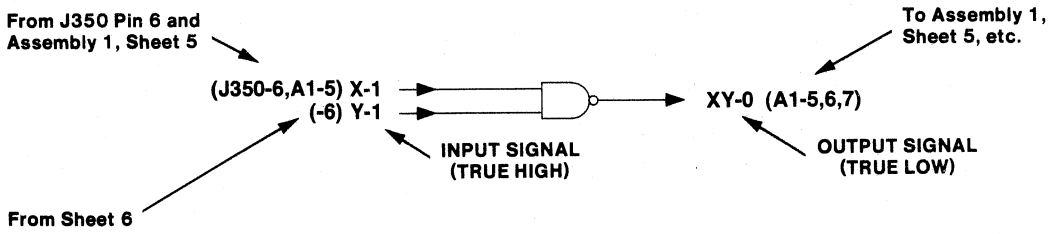
1. True High and True Low Signals

Signal names on the schematics are followed by -1 or a -0. A TRUE HIGH signal is indicated by -1, and a TRUE LOW signal is indicated by -0.

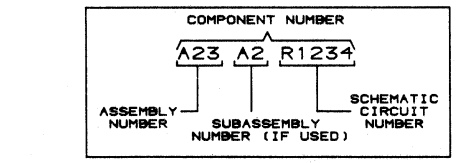
SIGNAL -1 = TRUE HIGH
SIGNAL -0 = TRUE LOW

2. Cross-References

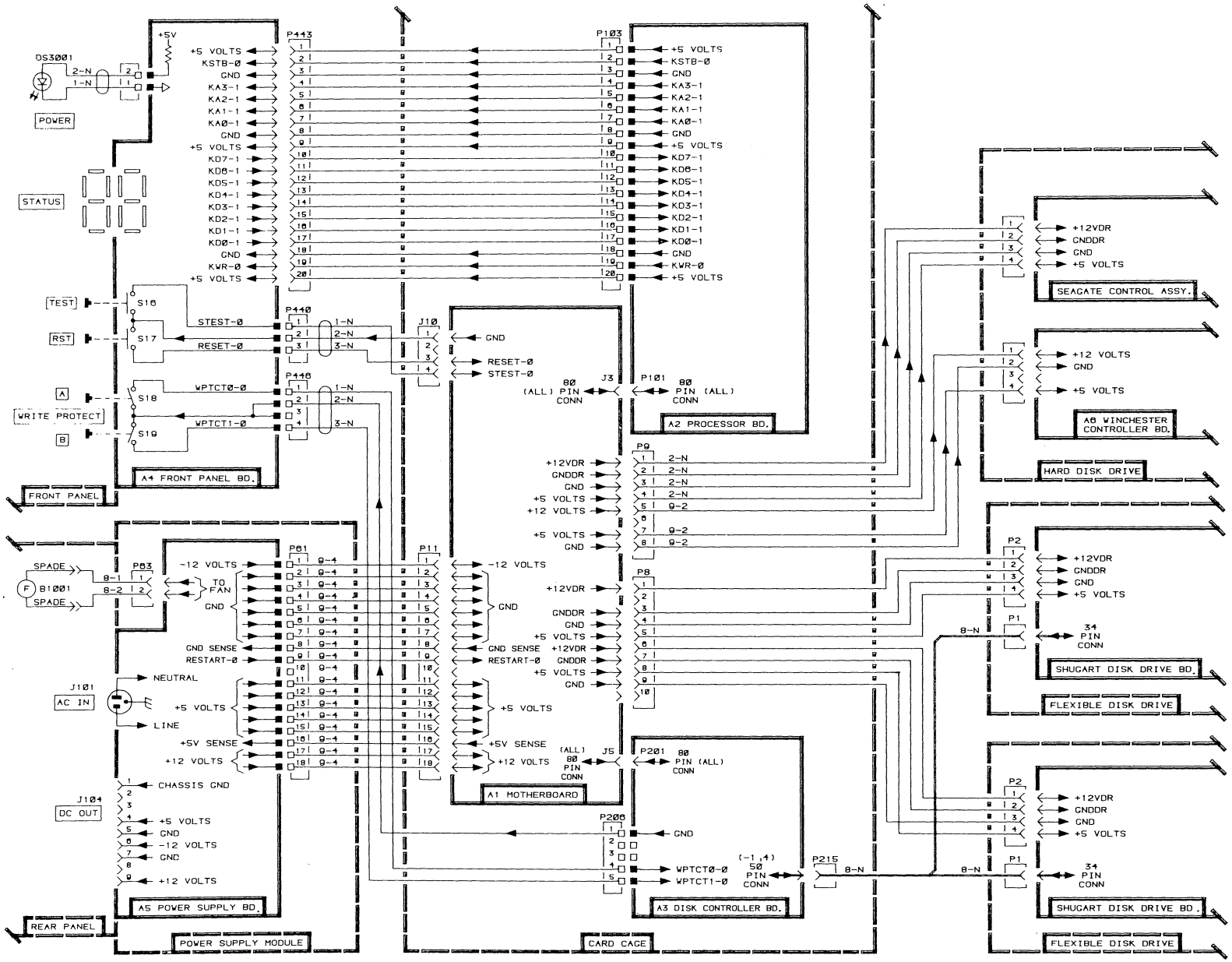
Schematic cross-references (from/to information) are included on the schematics. The "from" reference only indicates the signal "source," and the "to" reference lists all loads where the signal is used. All from/to information will be enclosed in parentheses.



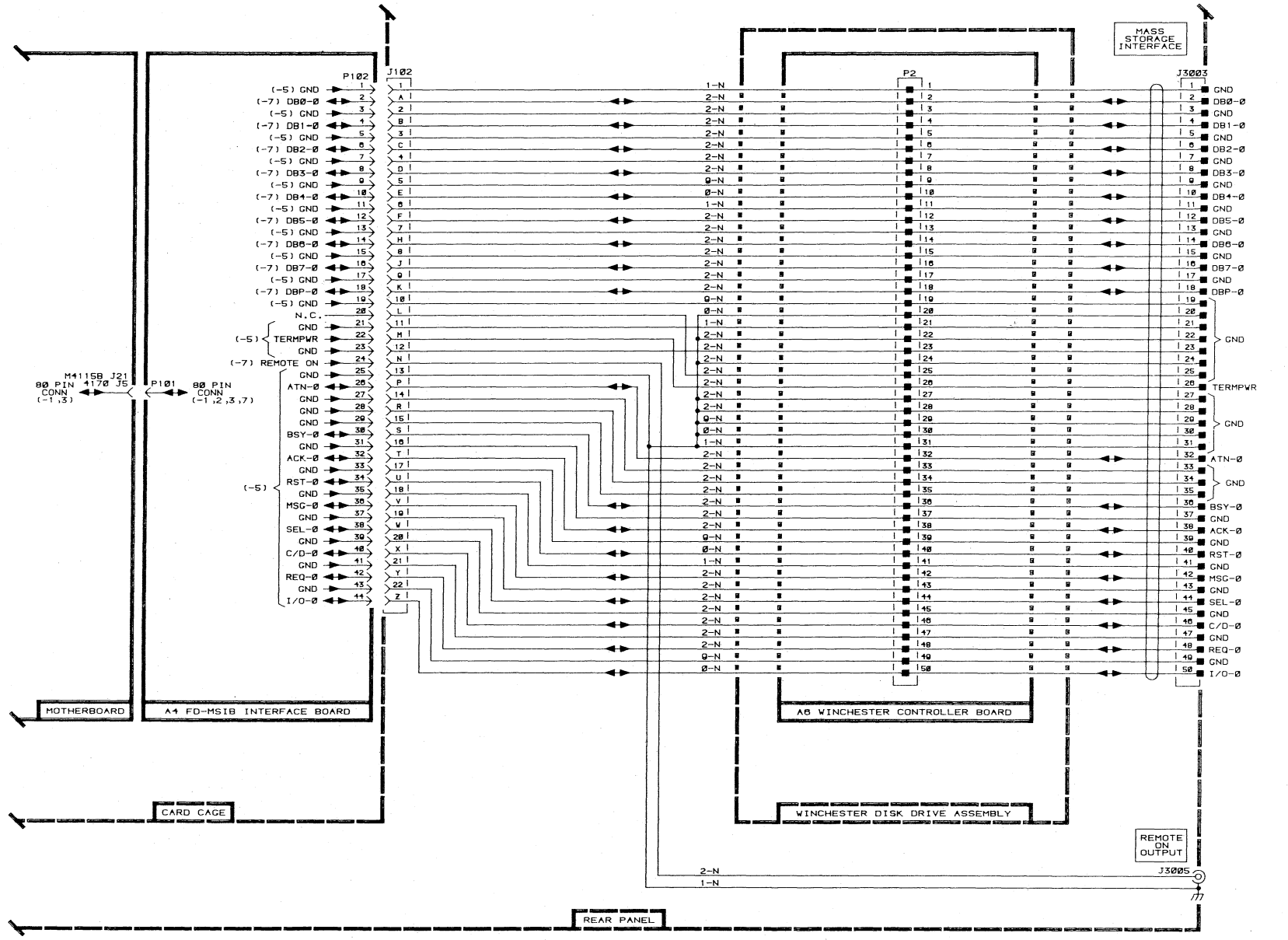
3. Component Number Example



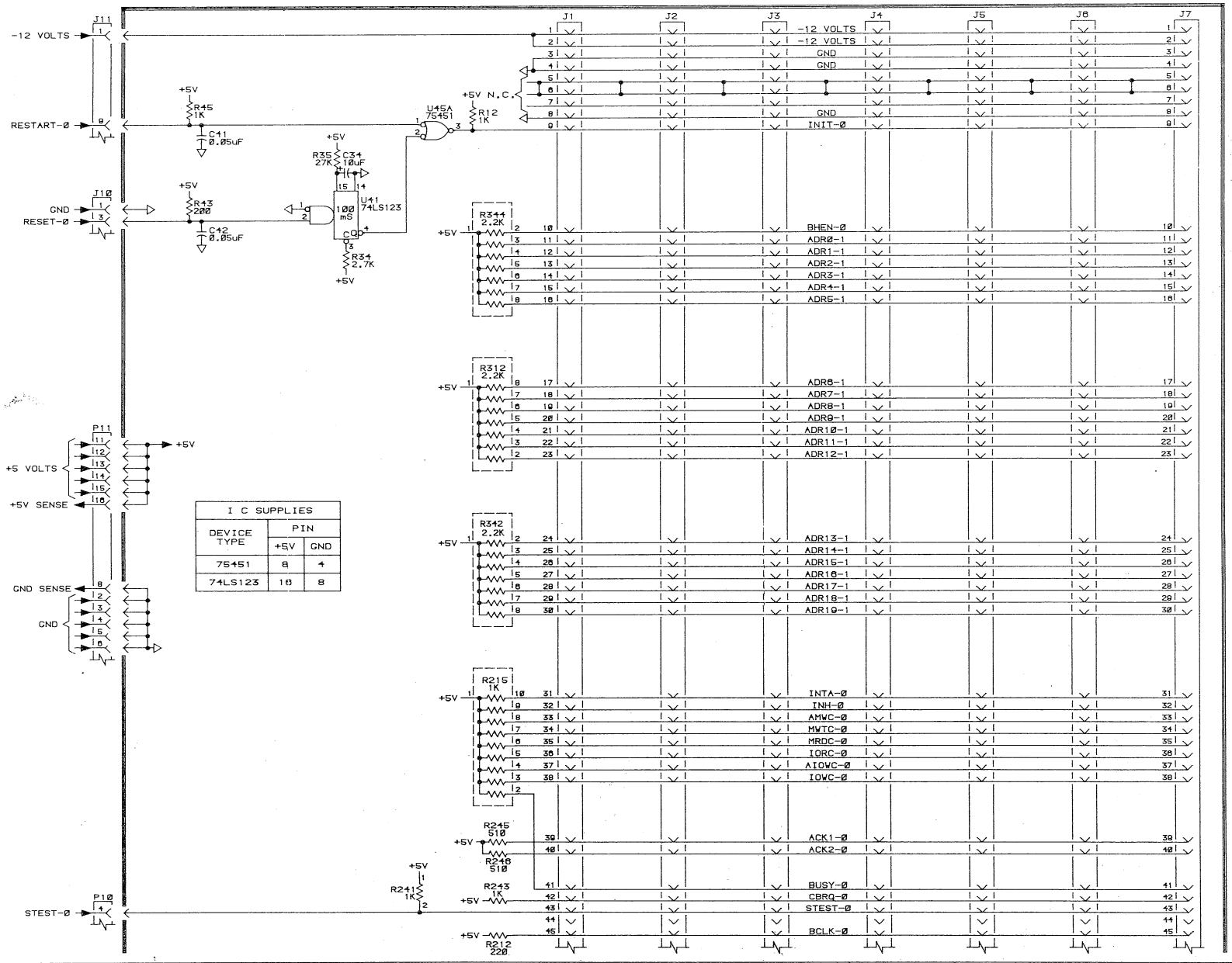
CHASSIS-MOUNTED COMPONENTS HAVE NO ASSEMBLY NUMBER PREFIX— SEE END OF REPLACEABLE ELECTRICAL PARTS LIST.



INSTRUMENT:	NOTES:	DIAGRAM:	SHEET:
4170	4885-I1	MOTHERBOARD INTERCONNECT	1 OF 1

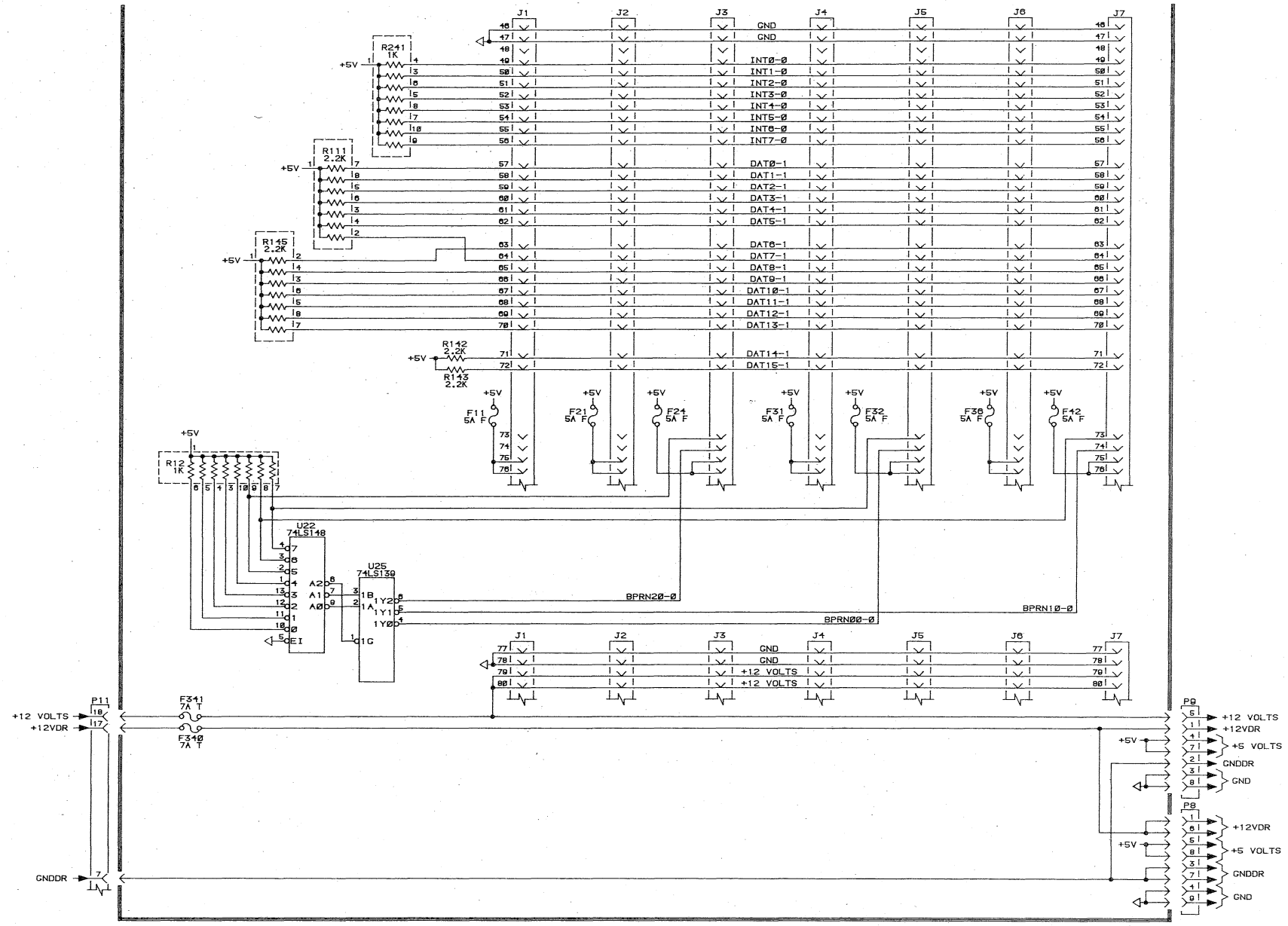


INSTRUMENT:	NOTES:	DIAGRAM:	SHEET:
4170 OPT. 45	4085-I2	FD-MSIB BOARD INTERCONNECT	1 OF 1

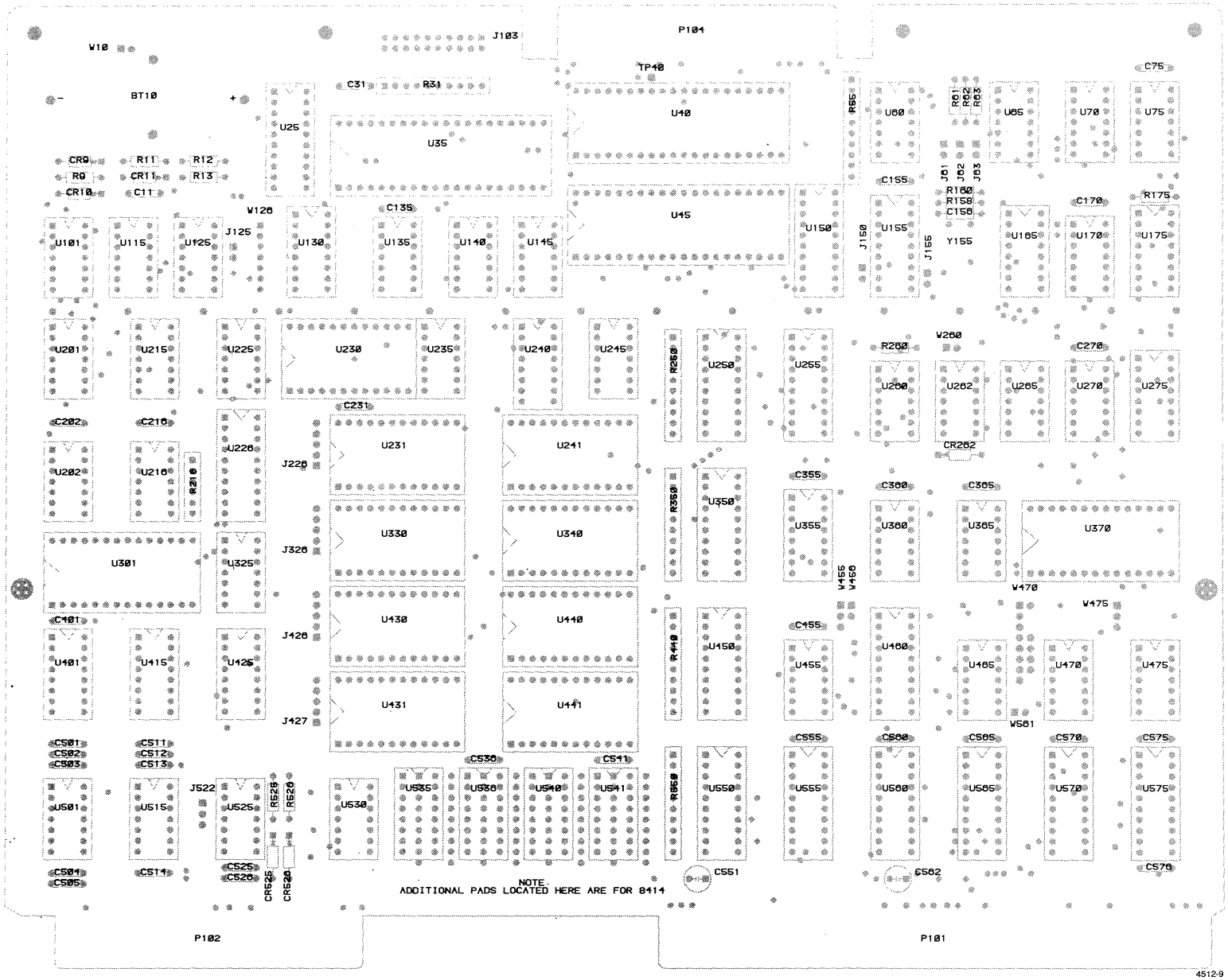


I C SUPPLIES		
DEVICE TYPE	PIN	
	+5V	GND
75451	8	4
74LS123	16	8

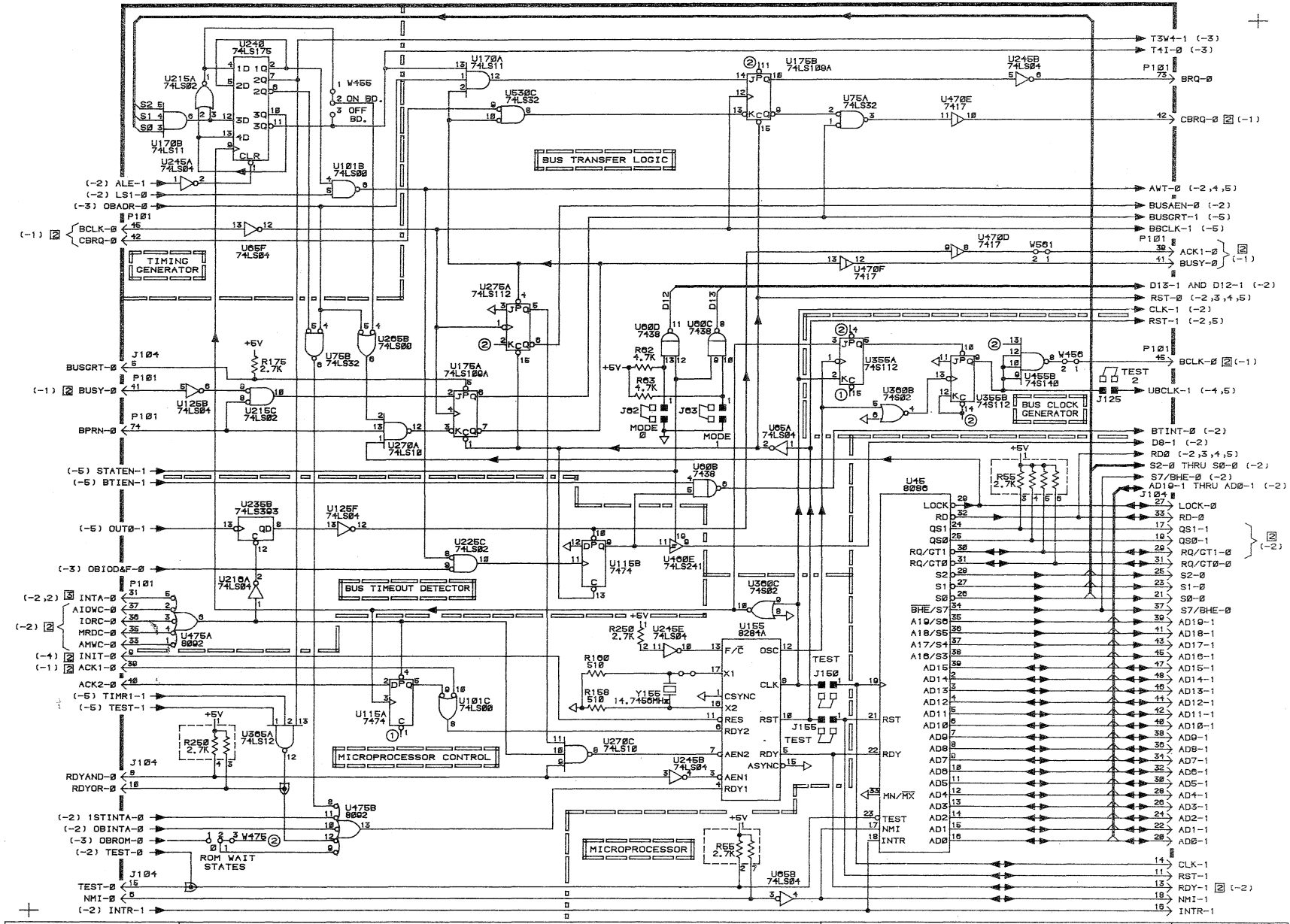
INSTRUMENT:	NOTES:	BOARD: 670-8119-00	ASSEMBLY-SHEET:
4170	4885-S1	MOTHERBOARD	A1-1 (1 OF 2)



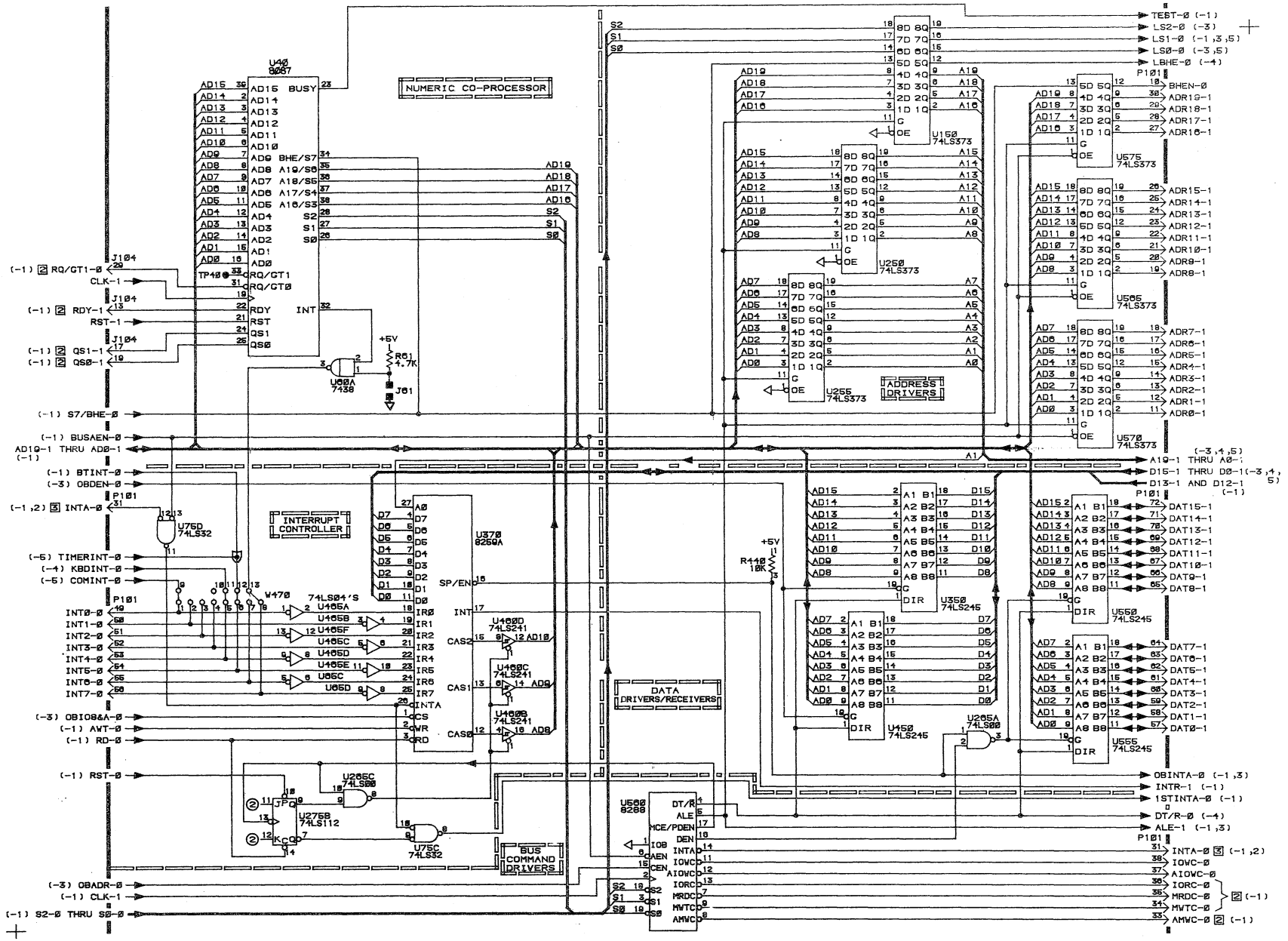
INSTRUMENT:	NOTES:	BOARD: 670-8119-00	ASSEMBLY-SHEET:
4170	4685-S2	MOTHERBOARD	A1-2 (2 OF 2)



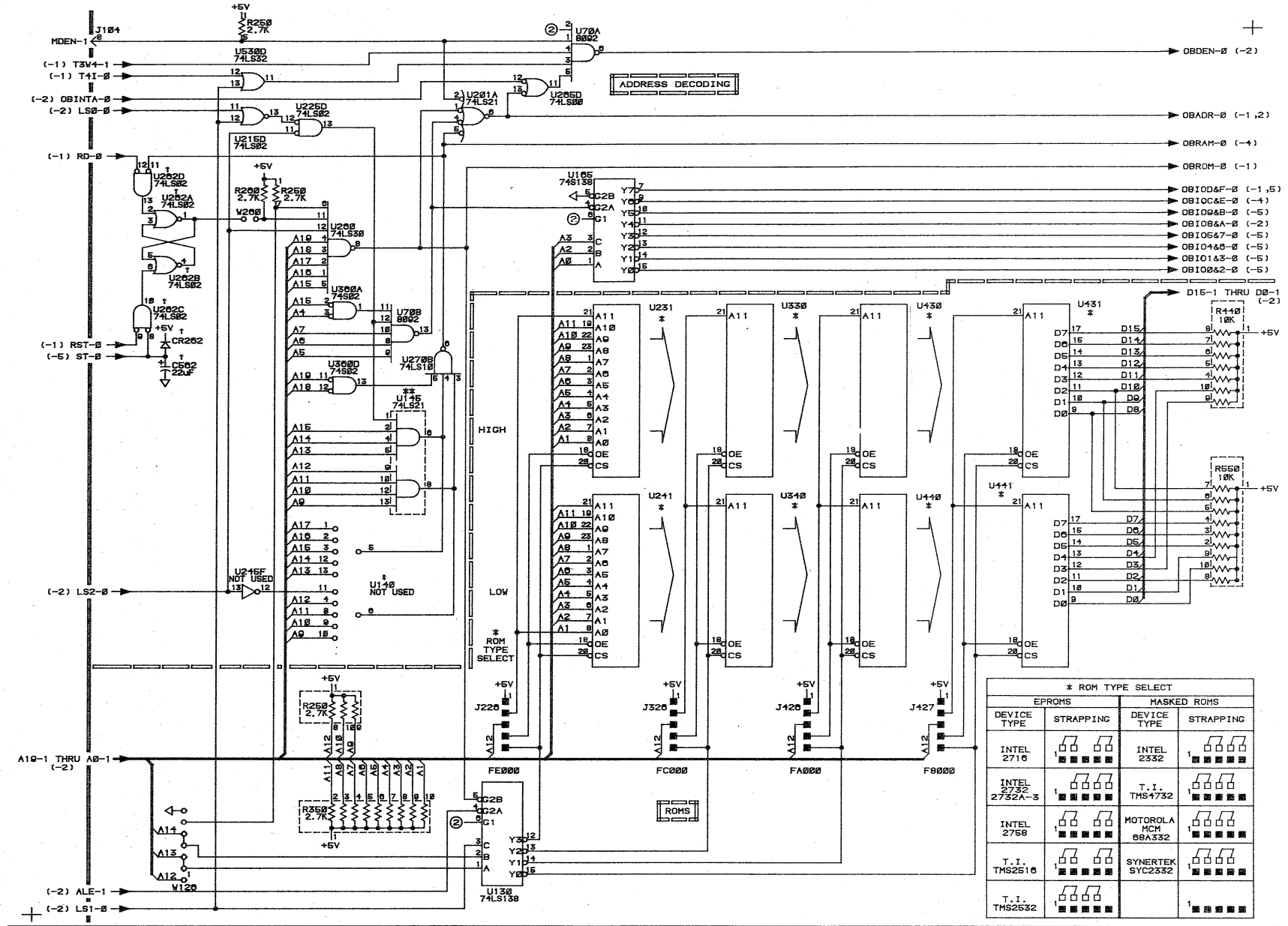
Processor (672-1140-00) Component Locations.



INSTRUMENT:	NOTES:	BOARD:	ASSEMBLY-SHEET:
4170	☐ INDICATES SHOWN MORE THAN ONCE AND WHERE (–)	672-1140-00/670-7926-00 PROCESSOR BOARD	A2-1 (1 OF 5)

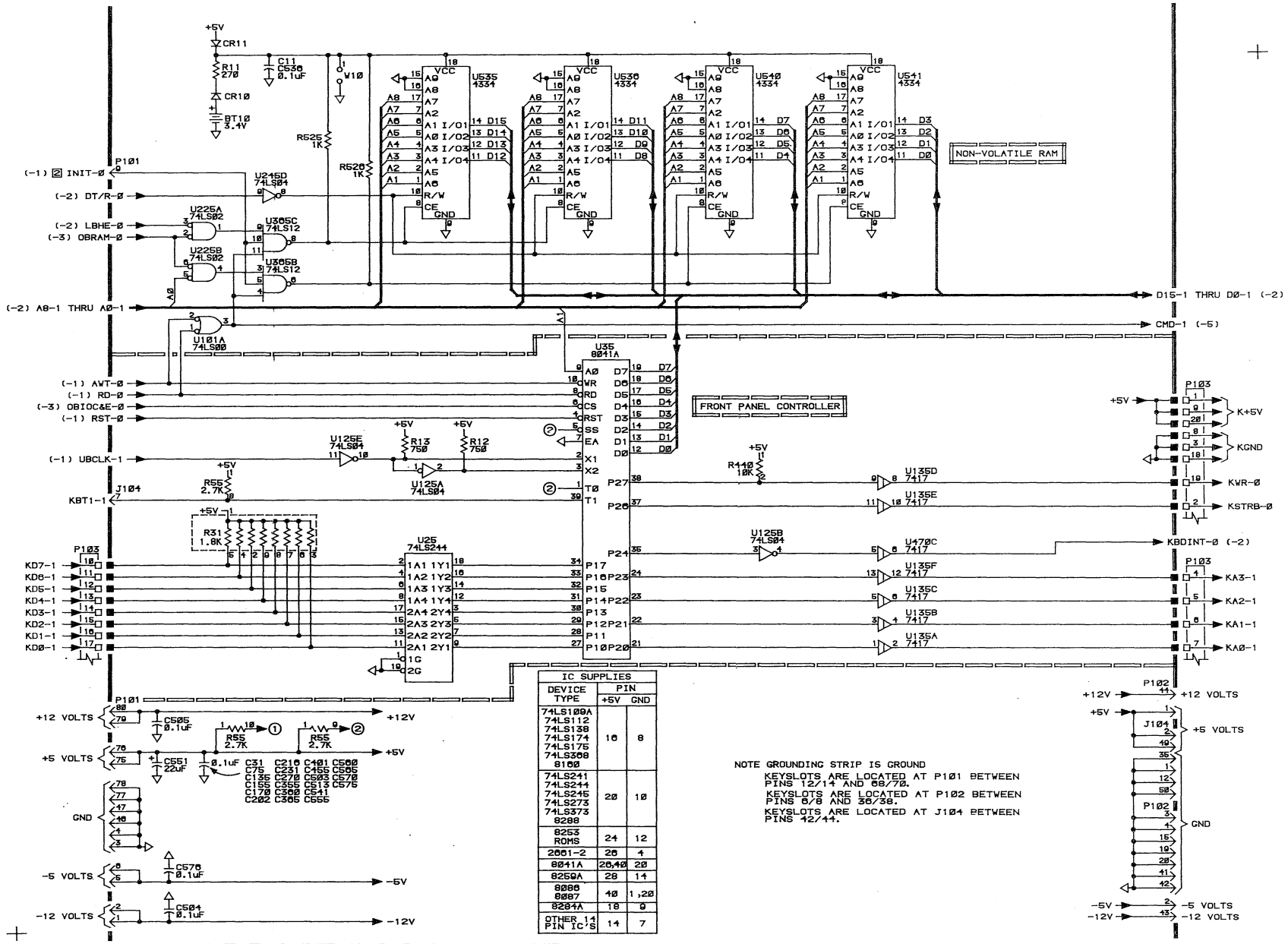


INSTRUMENT:	NOTES:		BOARD: 672-1140-00/070-7926-00	ASSEMBLY-SHEET:
4170	☐ INDICATES SHOWN MORE THAN ONCE AND WHERE (-)	4665-S4	PROCESSOR BOARD	A2-2 (2 OF 5)



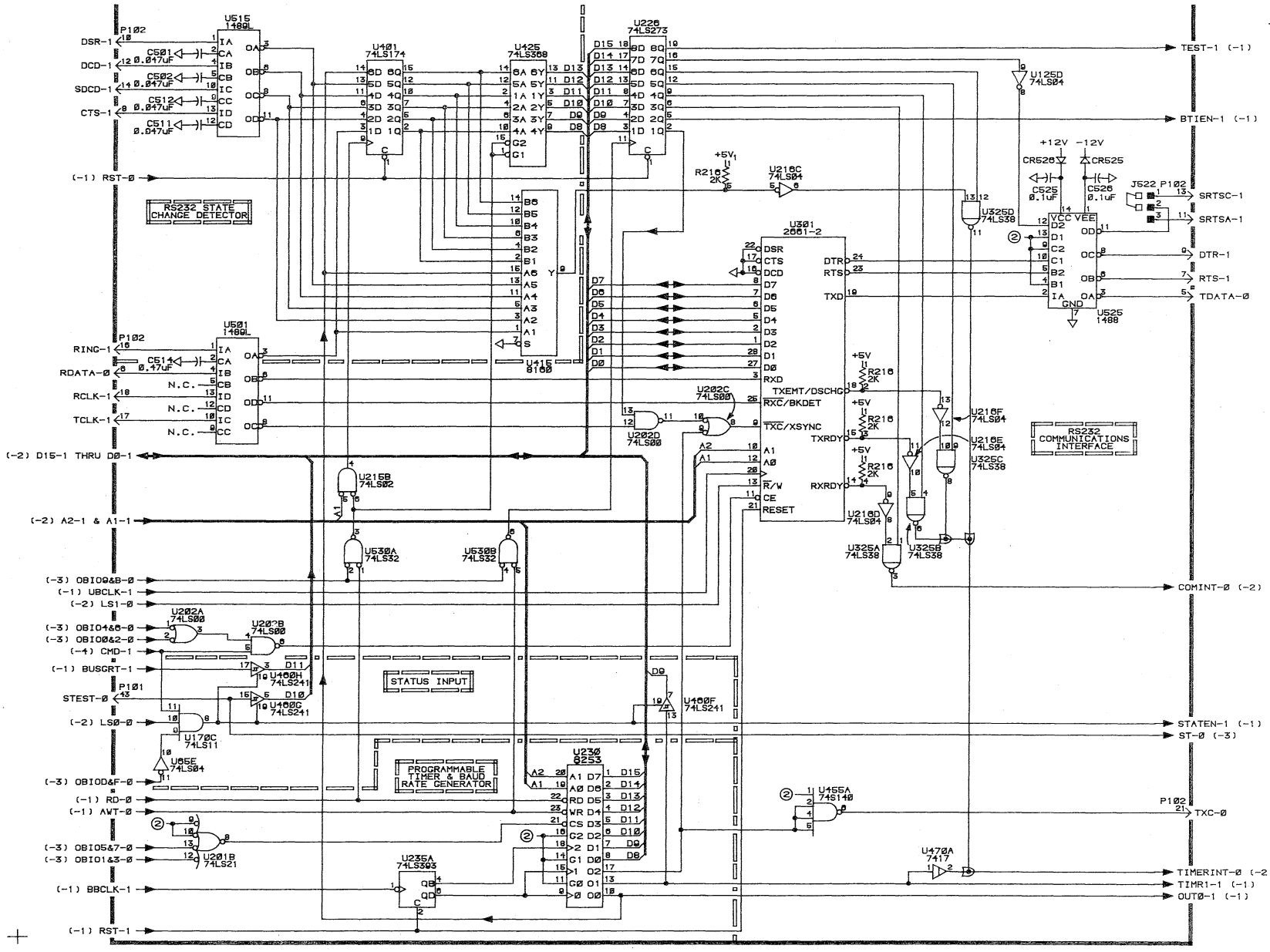
* ROM TYPE SELECT			
EPROMS		MASKED ROMS	
DEVICE TYPE	STRAPPING	DEVICE TYPE	STRAPPING
INTEL 2716		INTEL 2332	
INTEL 2732 2732A-3		T.I. TMS4732	
INTEL 2758		MOTOROLA MCM 68A332	
T.I. TMS2516		SYNERTEK SYC2332	
T.I. TMS2532			

INSTRUMENT:	NOTES: † THESE COMPONENTS ARE INSTALLED AND THE CIRCUIT IS INTENSIONALLY DISABLED. ‡ AT NO TIME WILL BOTH OF THESE IC'S BE PRESENT AT THE SAME TIME.	BOARD: 672-1140-00/670-7926-00	ASSEMBLY-SHEET:
4170	4585-S6	PROCESSOR BOARD	A2-3 (3 OF 5)

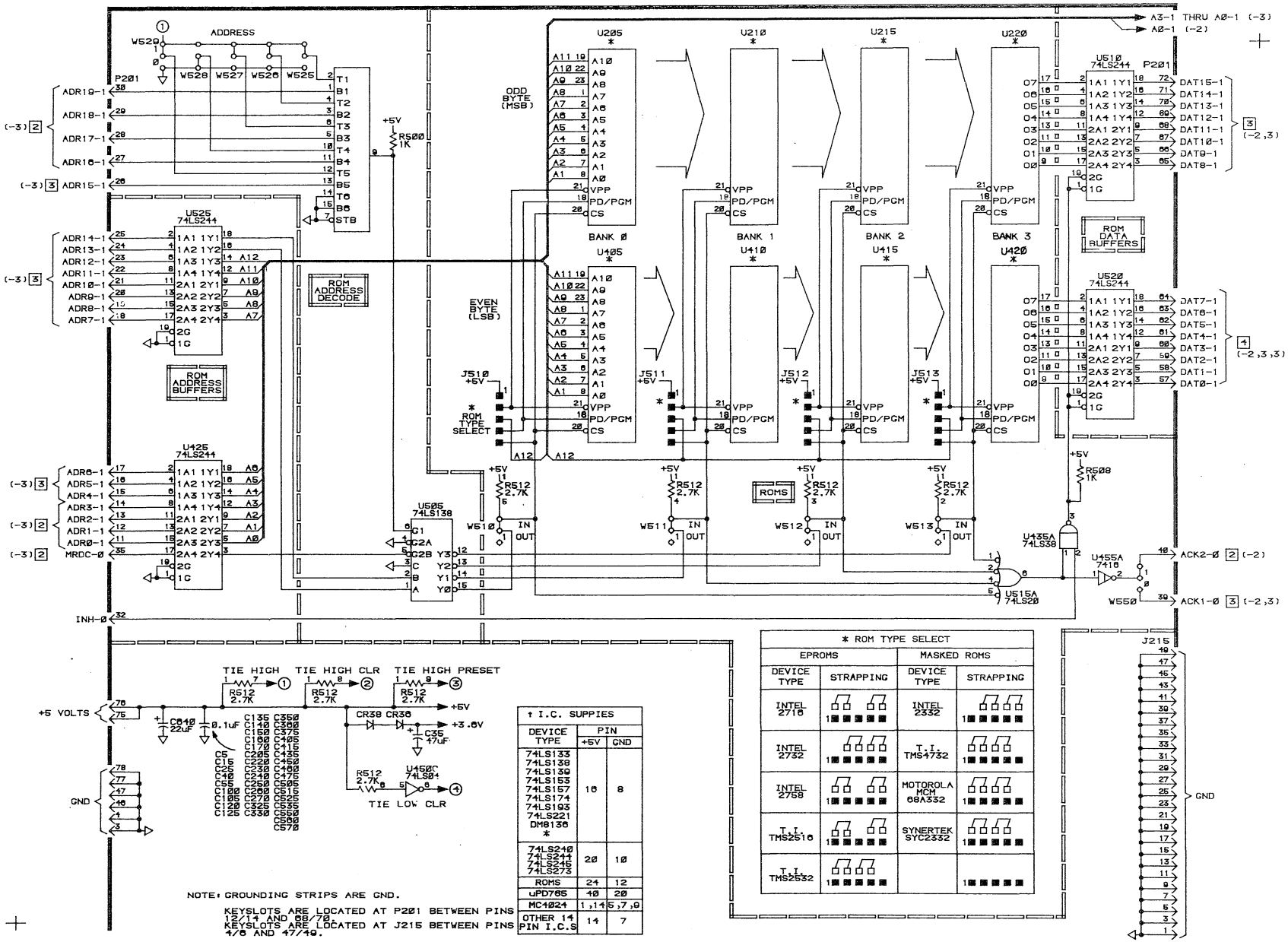


IC SUPPLIES		
DEVICE TYPE	PIN +5V	PIN GND
74LS100A 74LS112 74LS138 74LS174 74LS175 74LS308 8150	16	8
74LS241 74LS244 74LS245 74LS273 74LS373 8288	20	10
8253 ROMS	24	12
2661-2	26	4
8041A	26,40	20
8259A	28	14
8088 8087	40	1,20
8284A	18	9
OTHER 14 PIN IC'S	14	7

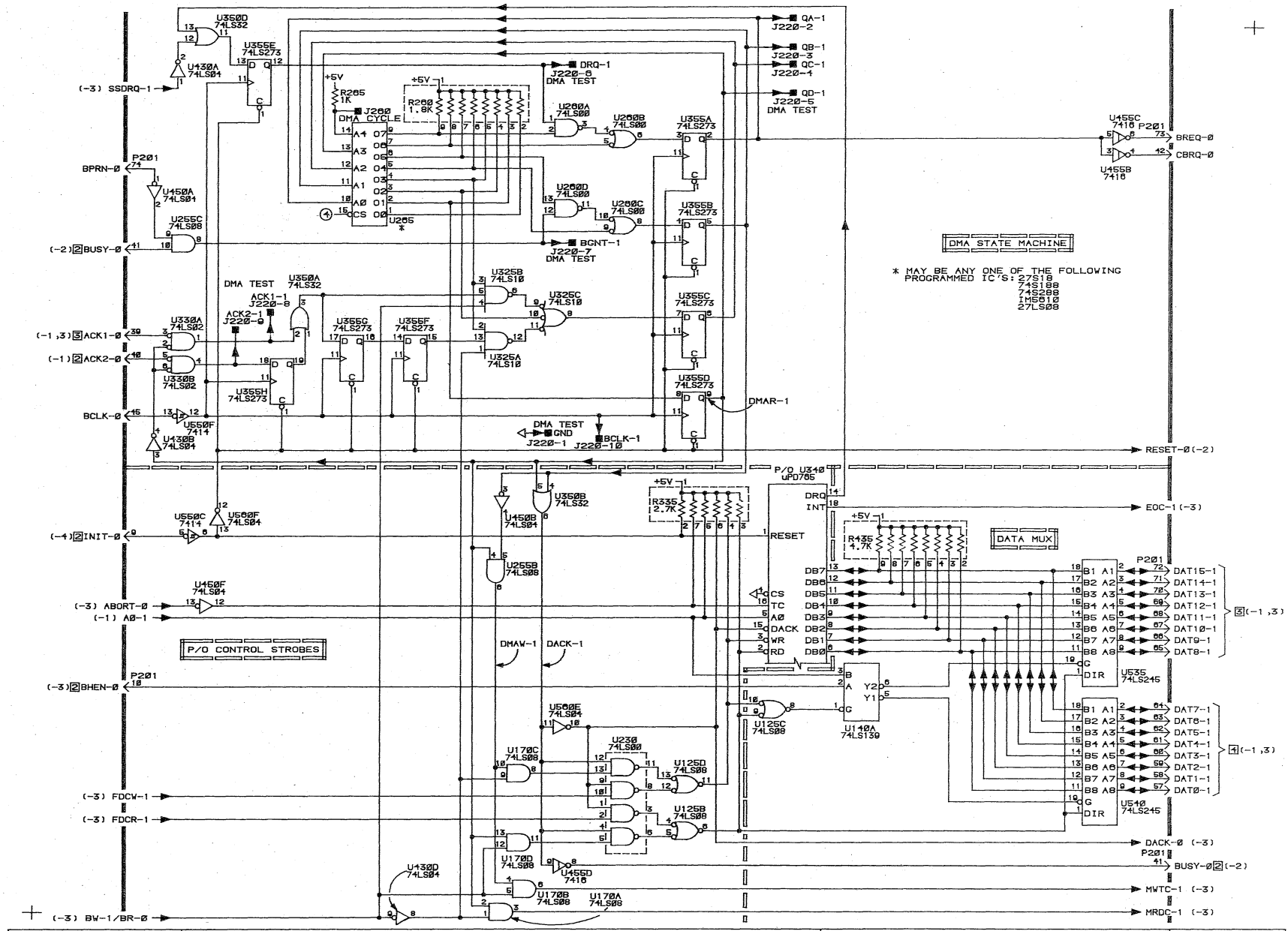
INSTRUMENT:	NOTES:		BOARD: 672-1140-00/670-7926-00	ASSEMBLY-SHEET:
4170	☐ INDICATES SHOWN MORE THAN ONCE AND WHERE (-)	4685-58	PROCESSOR BOARD	A2-4 (4 OF 5)

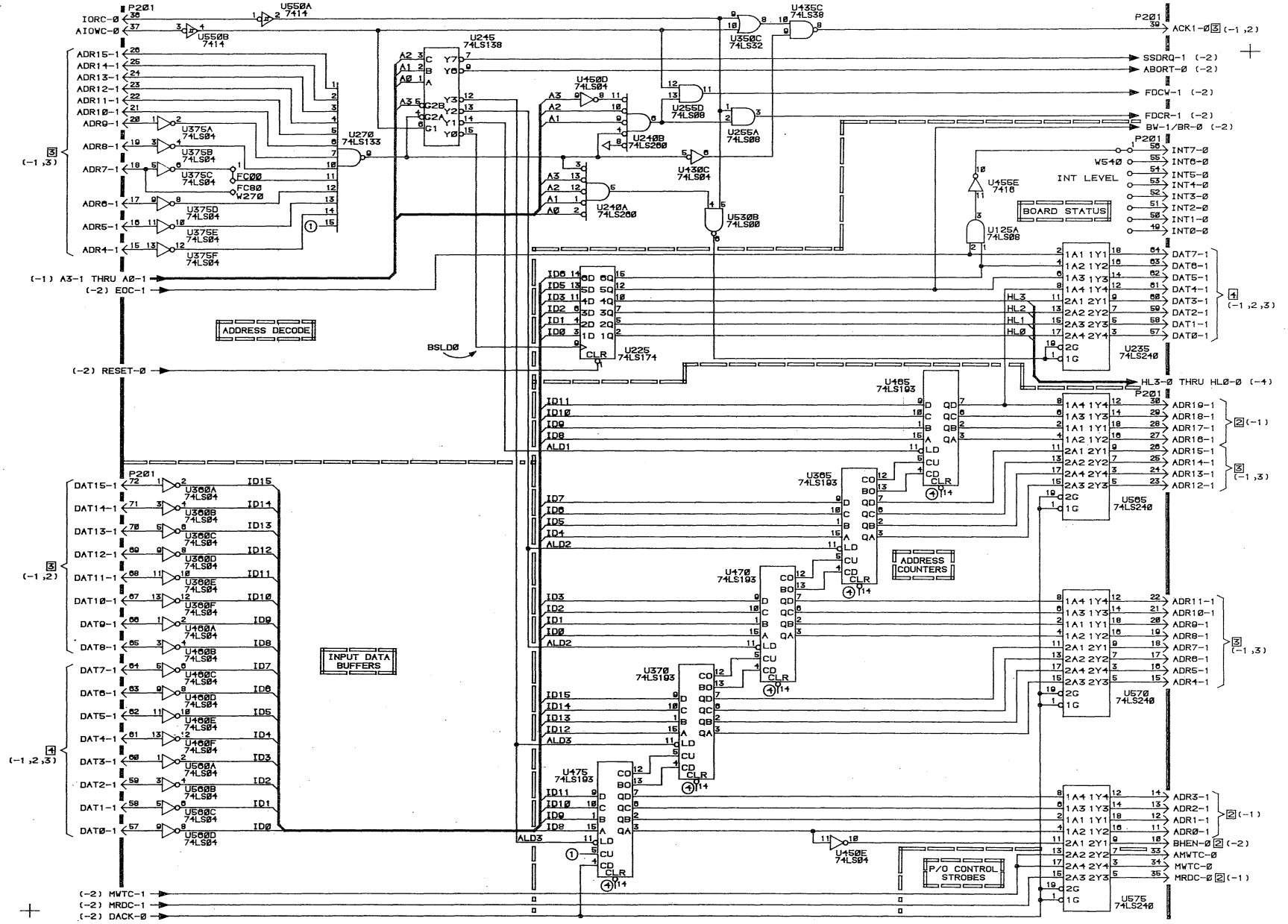


INSTRUMENT:	NOTES:	BOARD: 672-1140-00/670-7026-00	ASSEMBLY-SHEET:
4170	4685-S7	PROCESSOR BOARD	A2-5 (5 OF 5)

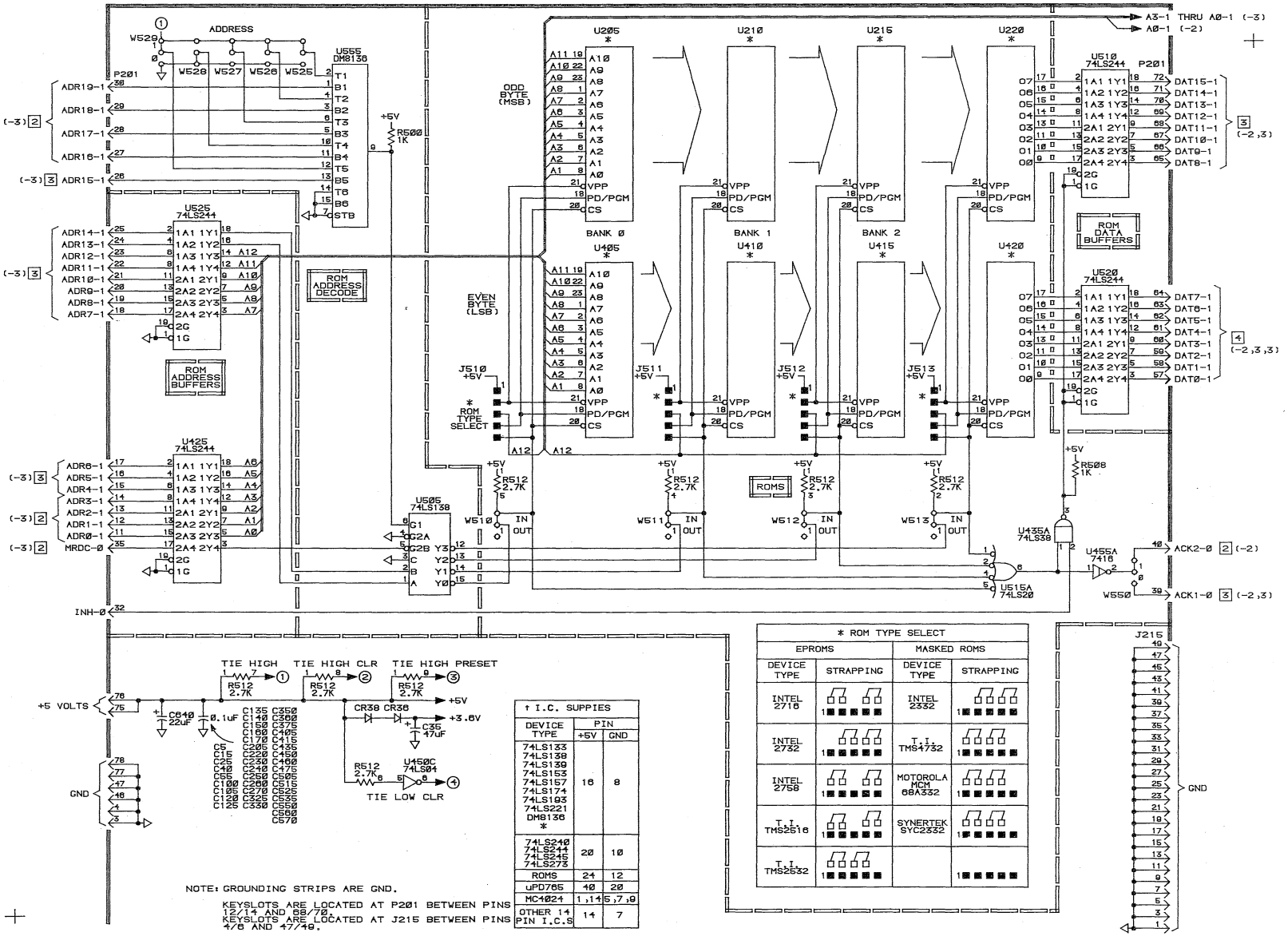


INSTRUMENT:	NOTES: † ALSO SEE SHT. 2	BOARD: 670-8148-00	ASSEMBLY-SHEET:
4170	□ INDICATES SHOWN MORE THAN ONCE AND WHERE (-)	DISK CONTROLLER BD.	A3-1 (1 OF 4)
	4685-S8		


DMA STATE MACHINE
* MAY BE ANY ONE OF THE FOLLOWING PROGRAMMED IC'S: 27S18 74S188 74S288 IM5610 27LS08
DATA MUX
P/O CONTROL STROBES
DMA TEST
DMA CYCLE
P/O U340 uPD765
(-3) SSDRQ-1
(-2)2BUSY-0
(-1,3)3ACK1-0
(-1)2ACK2-0
(-4)2INIT-0
(-3) ABORT-0
(-1) A0-1
(-3)2BHEN-0
(-3) FDCW-1
(-3) FDCR-1
(-3) BW-1/BR-0
BPRN-0
BCLK-0
DRQ-1
BGNT-1
DMAR-1
ACK1-1
ACK2-1
RESET-0(-2)
EOC-1(-3)
DACK-0 (-3)
BUSY-0 2(-2)
MWTC-1 (-3)
MRDC-1 (-3)
BREQ-0
CBRQ-0
QA-1
QB-1
QC-1
QD-1
DAT15-1
DAT14-1
DAT13-1
DAT12-1
DAT11-1
DAT10-1
DAT9-1
DAT8-1
DAT7-1
DAT6-1
DAT5-1
DAT4-1
DAT3-1
DAT2-1
DAT1-1
DAT0-1
DMAW-1
DACK-1
INSTRUMENT: 4170
NOTES: INDICATES SHOWN MORE THAN ONCE AND WHERE (-)
4685-S9
BOARD: 670-8148-00
DISK CONTROLLER BD.
ASSEMBLY-SHEET: A3-2 (2 OF 4)



INSTRUMENT:	NOTES:	BOARD: 67Ø-8148-ØØ	ASSEMBLY-SHEET:
417Ø	☐ INDICATES SHOWN MORE THAN ONCE AND WHERE (-)	DISK CONTROLLER BD.	A3-3 (3 OF 4)
	4885-S1Ø		



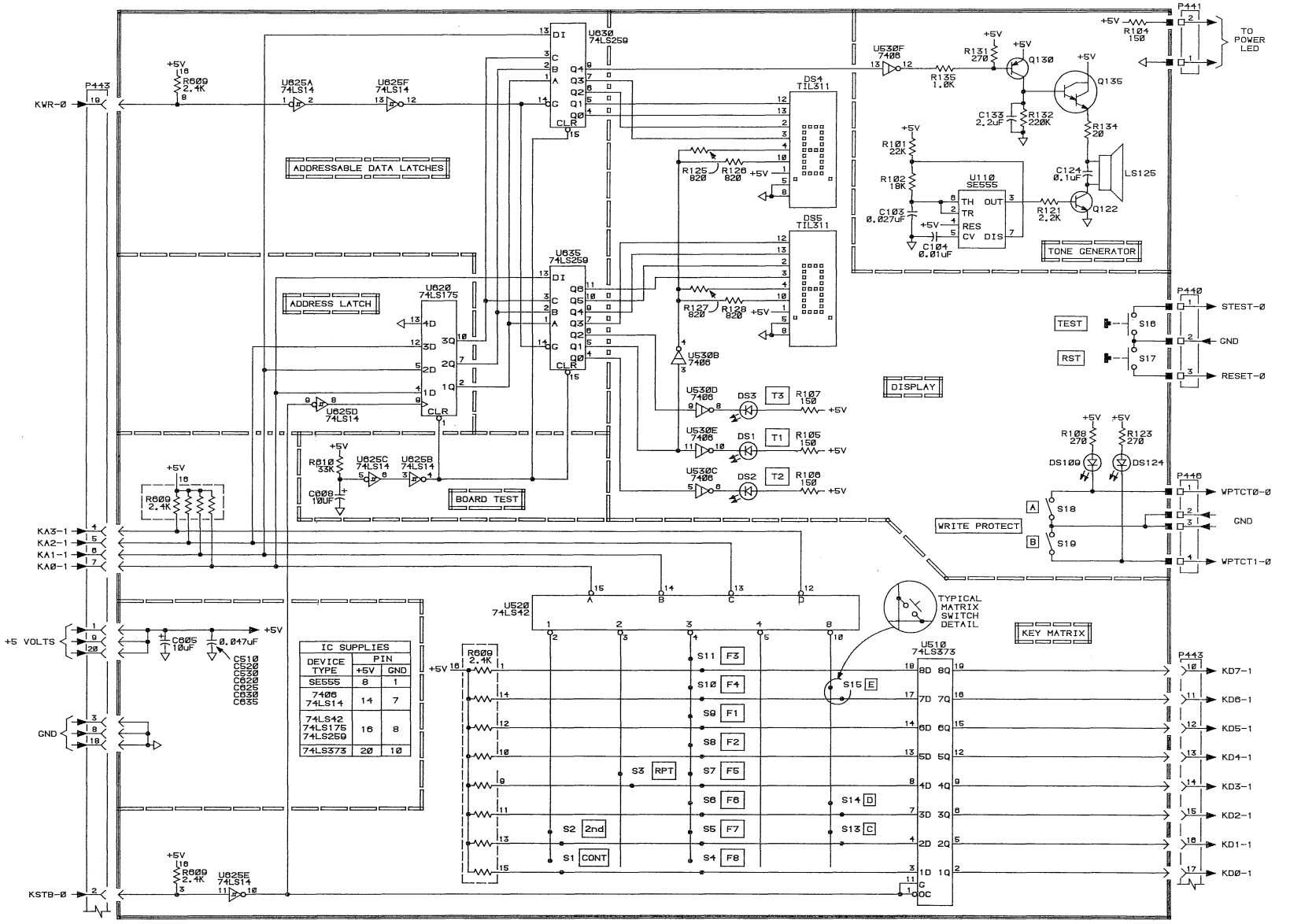
* ROM TYPE SELECT			
EPROMS		MASKED ROMS	
DEVICE TYPE	STRAPPING	DEVICE TYPE	STRAPPING
INTEL 2716		INTEL 2332	
INTEL 2732		T.I. TMS4732	
INTEL 2758		MOTOROLA MCM 68A332	
T.I. TMS2516		SYNERTEK SYC2332	
T.I. TMS2532			

† I.C. SUPPIES		
DEVICE TYPE	PIN +5V	PIN GND
74LS133, 74LS138, 74LS139, 74LS153, 74LS157, 74LS174, 74LS193, 74LS221, DM8136 *	16	8
74LS240, 74LS241, 74LS245, 74LS273	20	10
ROMS	24	12
uPD765	40	20
MC1024	1,14	5,7,9
OTHER 14 PIN I.C.S	14	7

NOTE: GROUNDING STRIPS ARE GND.

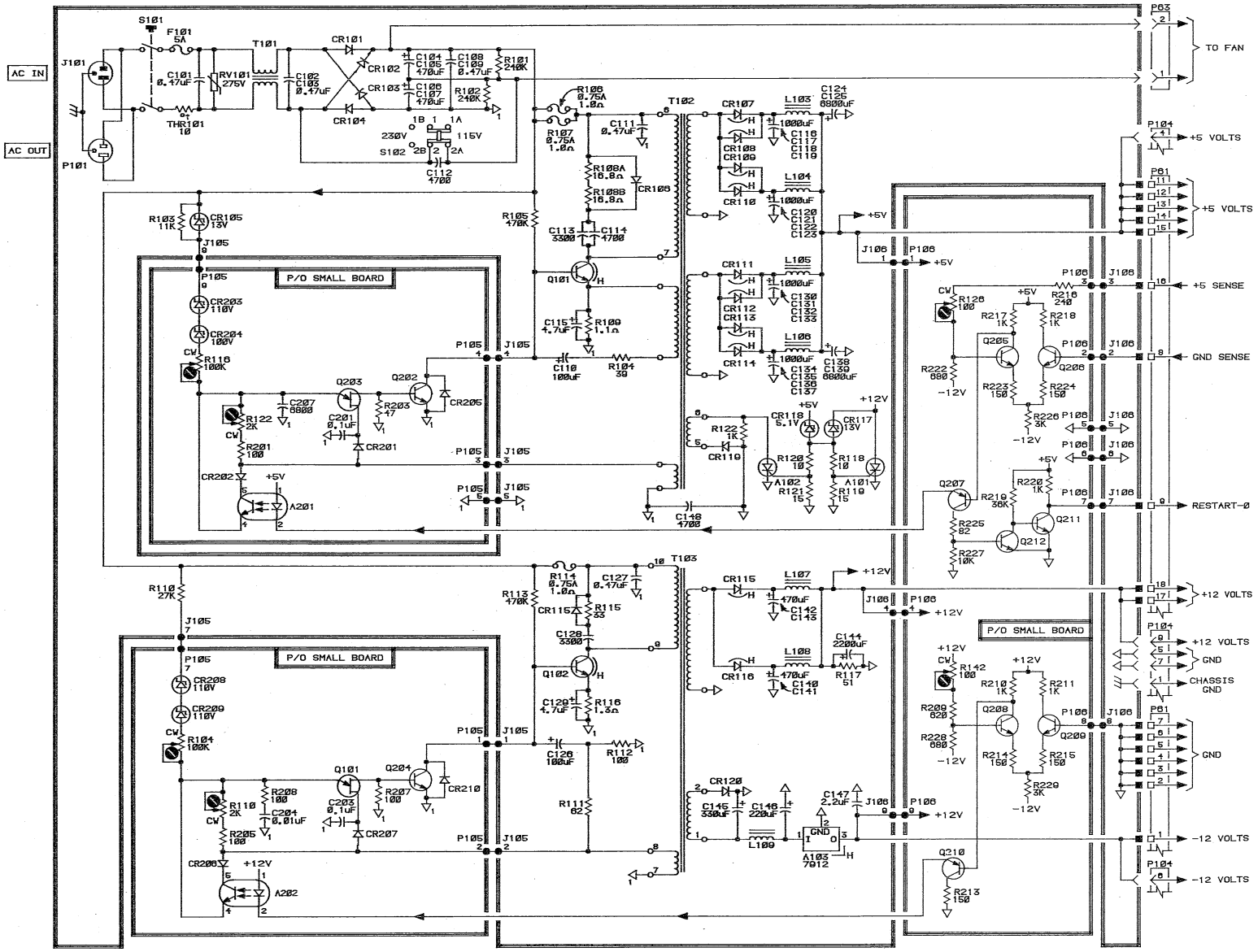
KEYSLOTS ARE LOCATED AT P201 BETWEEN PINS 12/14 AND 68/70.
KEYSLOTS ARE LOCATED AT J215 BETWEEN PINS 4/6 AND 47/48.

FIRST USE: 4170	OTHER USES:	NOTES: † ALSO SEE SHT. 2	TITLE: 670-8148-00	ASSEMBLY:
DATE: REV, JAN 1984		☐ INDICATES SHOWN MORE THAN ONCE AND WHERE (-)	DISK CONTROLLER BOARD	A3-1
CONTROL NO.: SSA025.000		TEKTRONIX, INC. © 1984	Tektronix®	SHEET: 1 OF 4

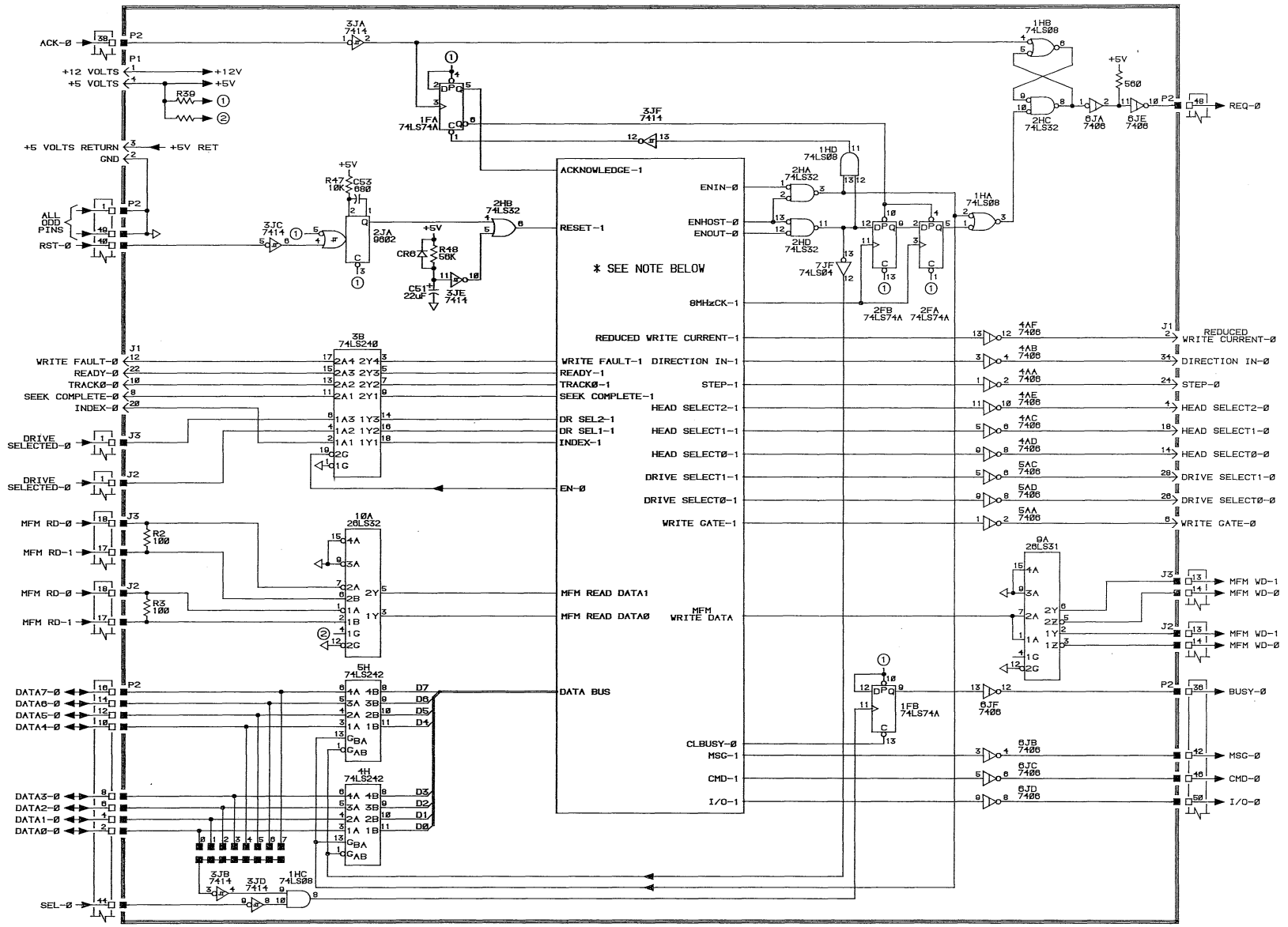


IC SUPPLIES		
DEVICE TYPE	PIN +5V	PIN GND
SE555	8	1
7406 74LS14	14	7
74LS42 74LS175 74LS259	16	8
74LS373	20	10

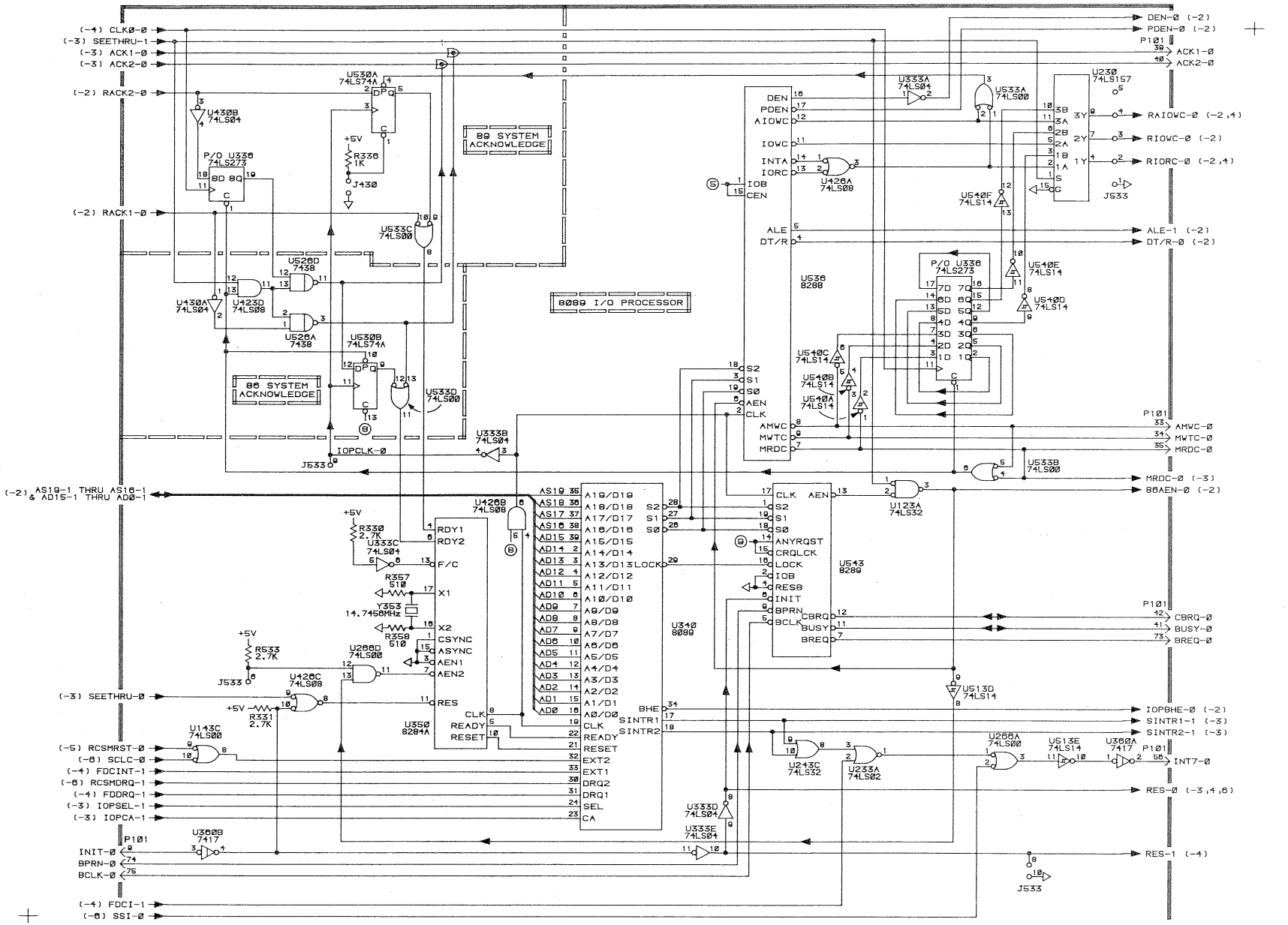
FIRST USE: 4170	OTHER USES:	NOTES:	TITLE: 670-8120-00
DATE: REV, JAN 1984		TEKTRONIX, INC. © 1984	FRONT PANEL BOARD
CONTROL NO.: SSA026.000			ASSEMBLY: A4-1
			SHEET: 1 OF 1



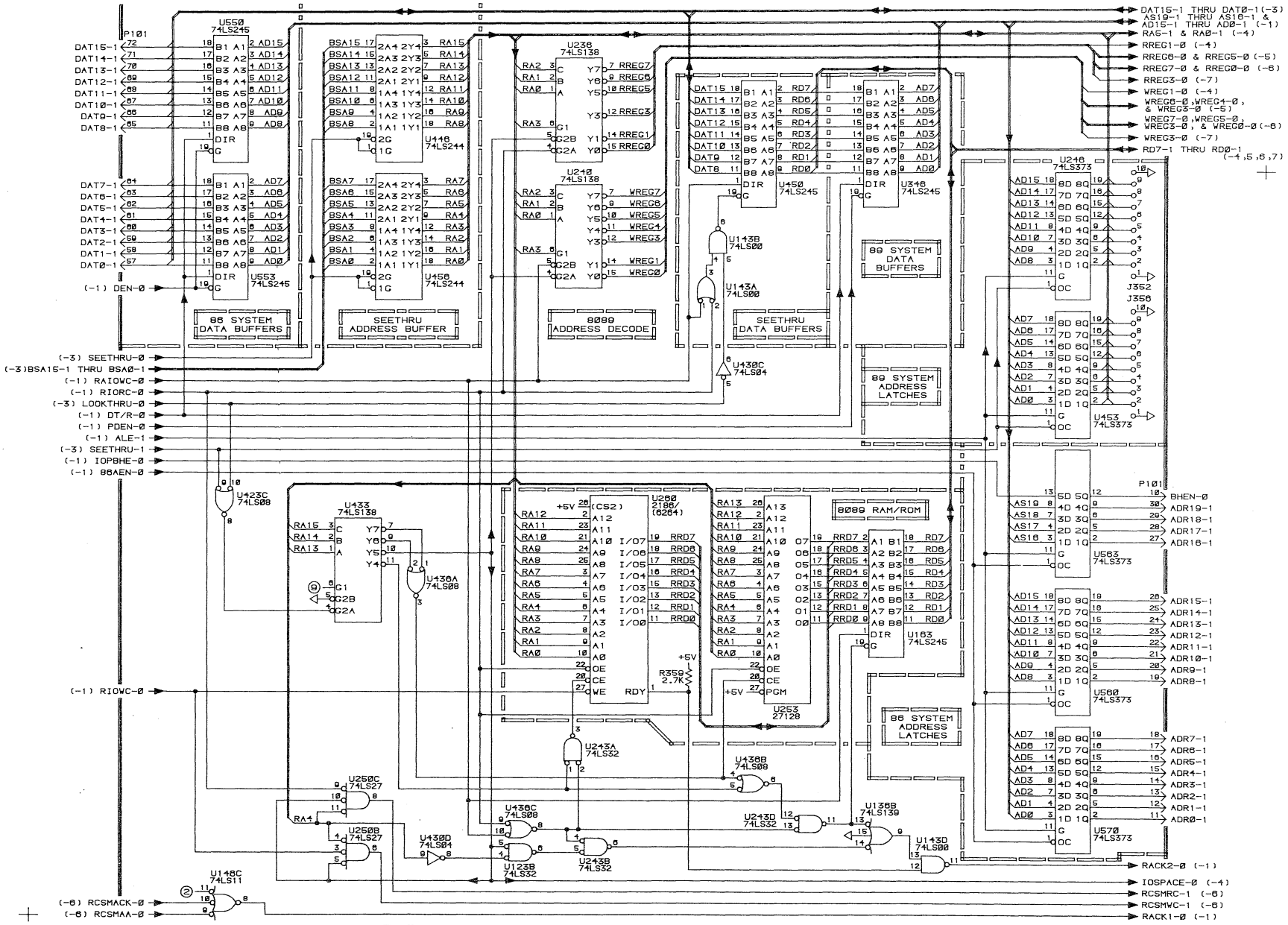
FIRST USE: 4170	OTHER USES:	NOTES:	TITLE: 119-1686-00	ASSEMBLY:
DATE: REV, JAN 1984			POWER SUPPLY BOARD	A5-1
CONTROL NO.: SSA022.000		COMPOWER CORPORATION	Tektronix®	SHEET: 1 OF 1



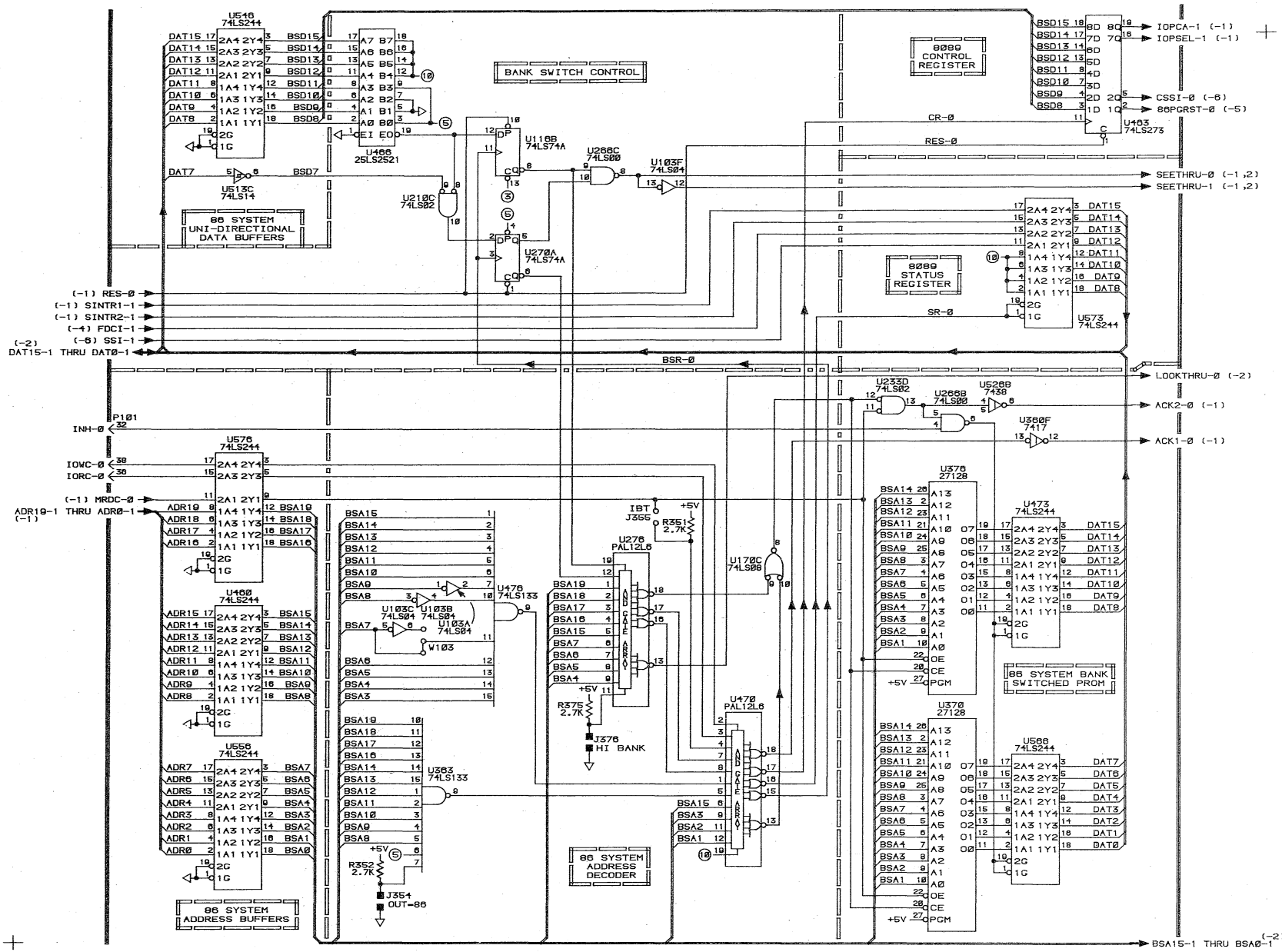
FIRST USE: 4170	OTHER USES: MASS STOR I/F	NOTES: * CIRCUITRY IN THIS BOX IS PROPRIETARY. FOR SERVICE INFORMATION CONTACT TEKTRONIX, INC., INFORMATION DISPLAY DIVISION.	TITLE: 670-8437-01	Tektronix®	ASSEMBLY: A6-1
DATE: REV, JAN 1984			WINCHESTER CONTROLLER BD.		SHEET: 1 OF 1
CONTROL NO.: SSA005.001		XEBEC SYSTEMS, INC.			



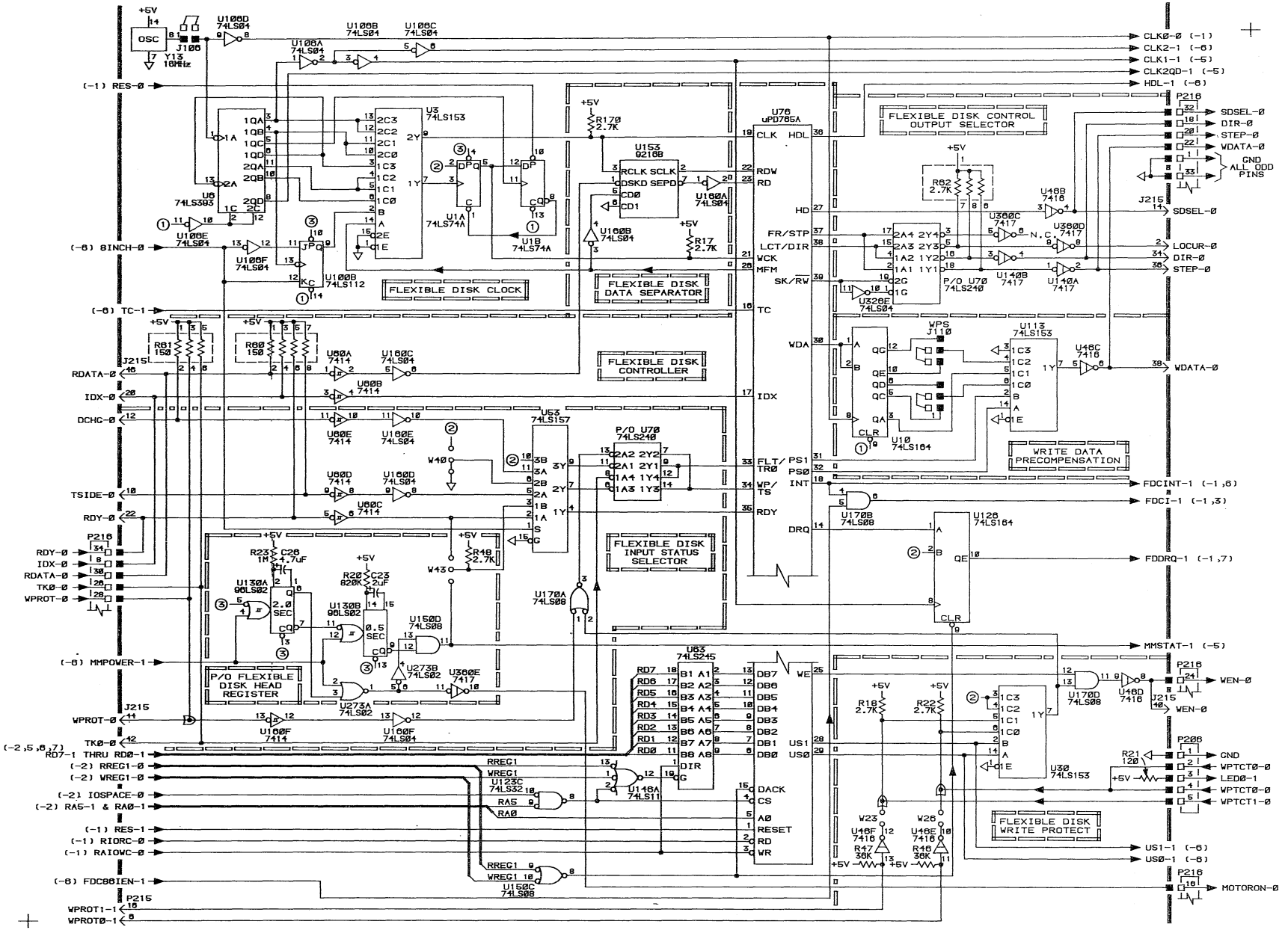
INSTRUMENT:	NOTES:	BOARD: 672-1183-00	ASSEMBLY-SHEET:
4170	4885-S15	FD-MSIB INTERFACE BOARD	A7-1 (1 OF 7)



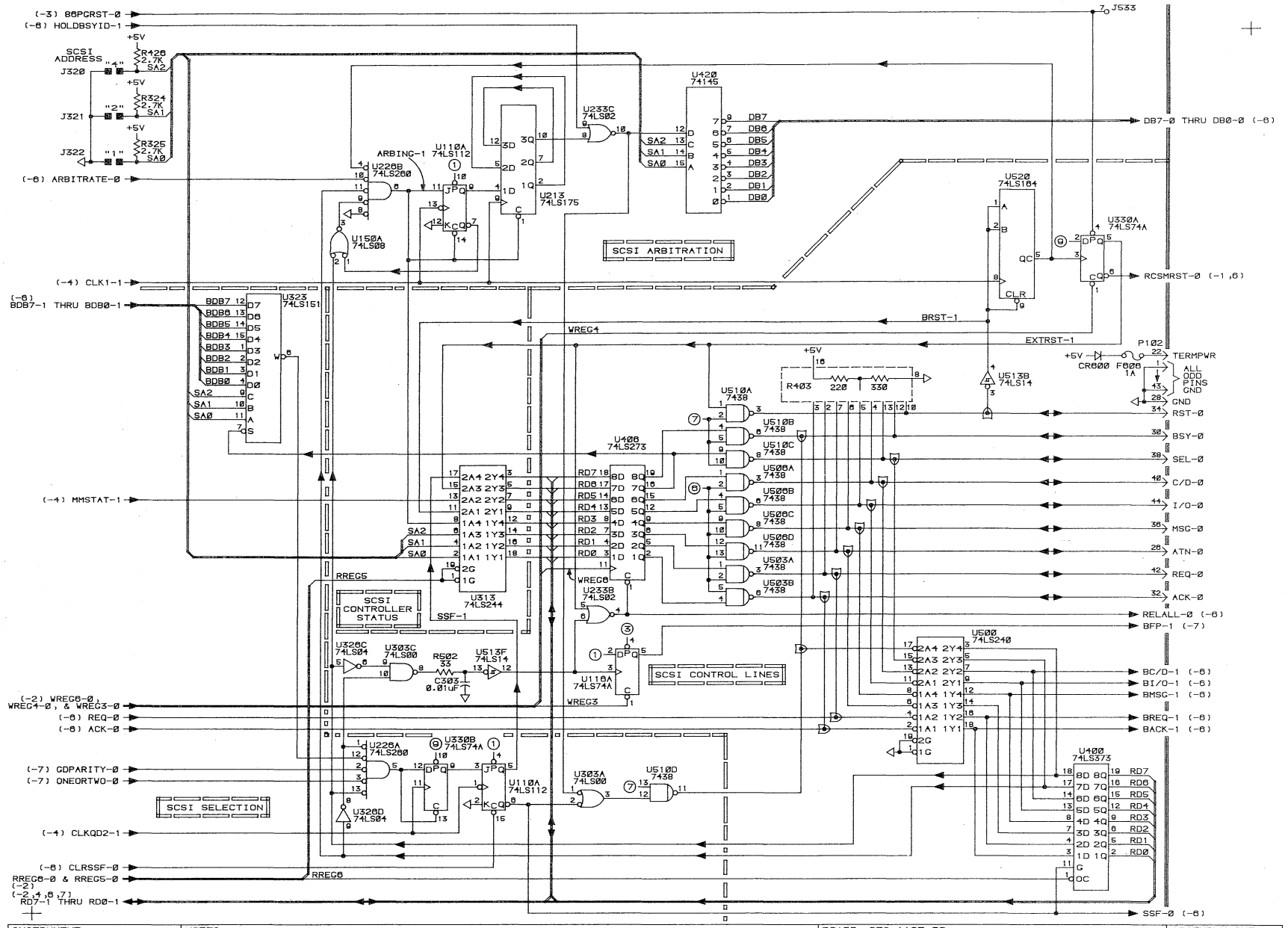
INSTRUMENT:	NOTES:	BOARD: 672-1163-00	ASSEMBLY-SHEET:
4170	4685-S16	FD-MSIB INTERFACE BOARD	A7-2 (2 OF 7)



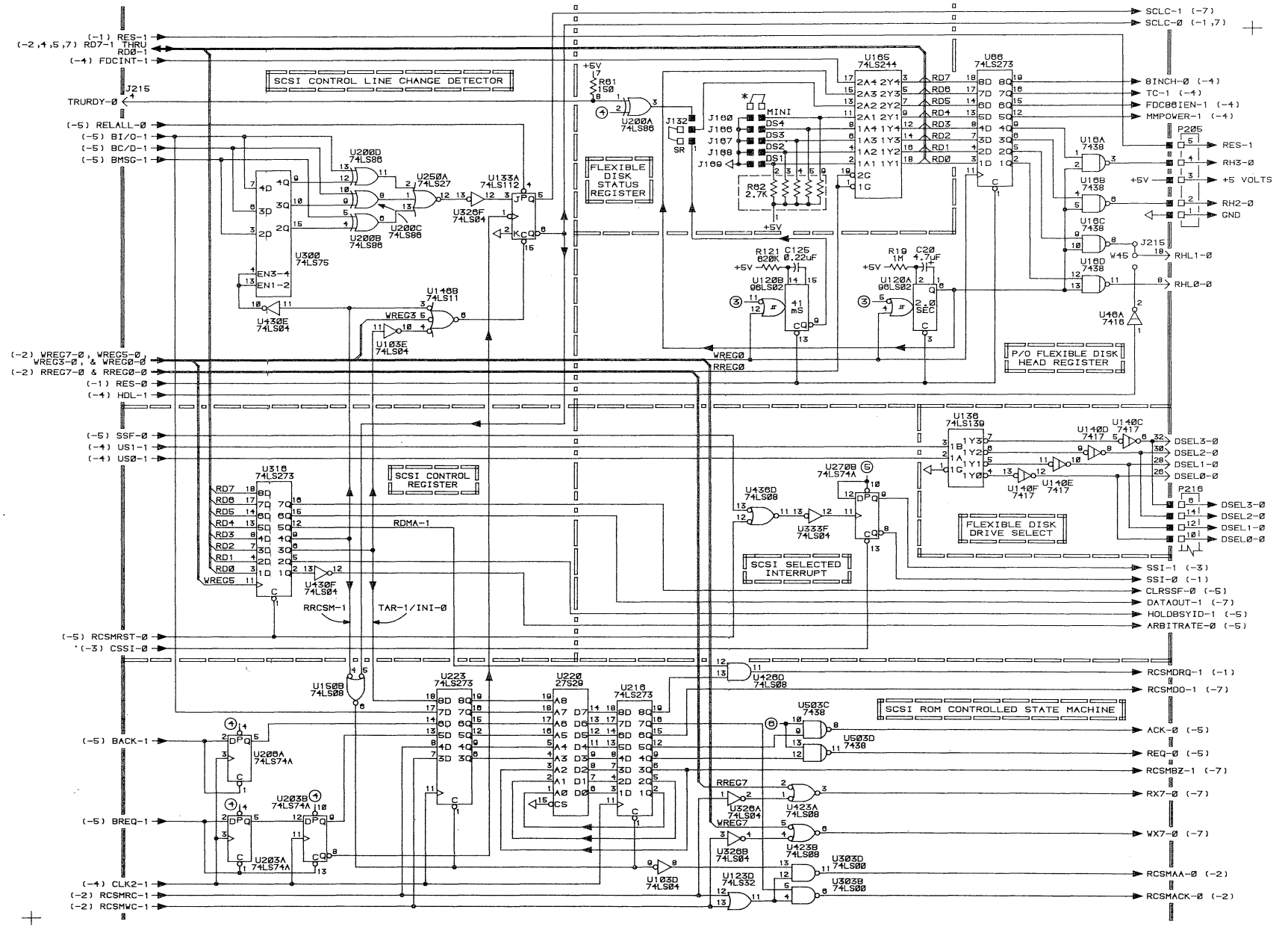
INSTRUMENT:	NOTES:	BOARD: 672-1163-00	ASSEMBLY-SHEET:
4170	4685-S17	FD-MSIB INTERFACE BOARD	A7-3 (3 OF 7)



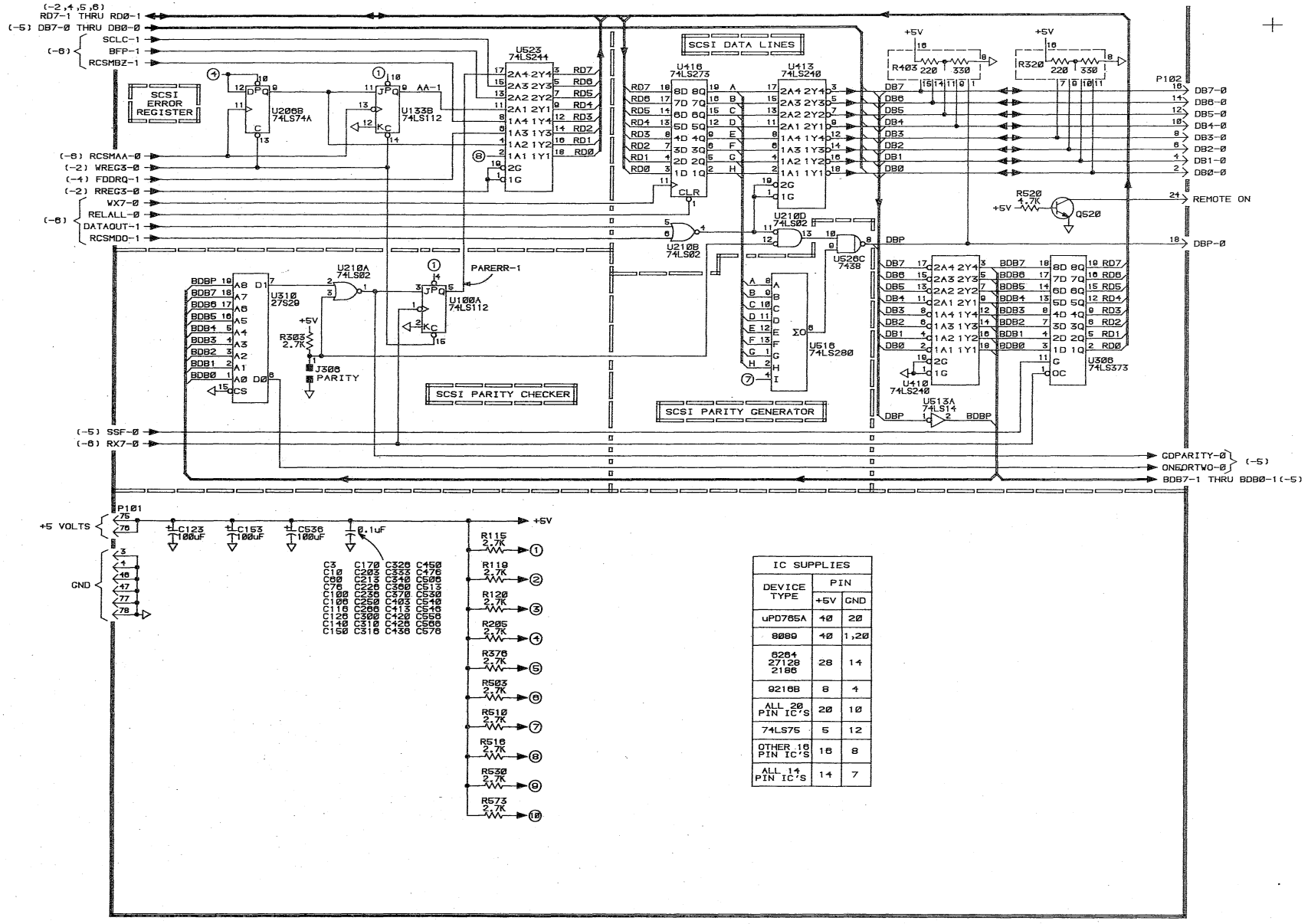
INSTRUMENT:	NOTES:	BOARD: 672-1163-00	ASSEMBLY-SHEET:
4170	4685-S18	FD-MSIB INTERFACE BOARD	A7-4 (4 OF 7)



INSTRUMENT:	NOTES:	BOARD: 672-1163-00	ASSEMBLY-SHEET:
4170	4685-S19	FD-MSIB INTERFACE BOARD	A7-5 (5 OF 7)

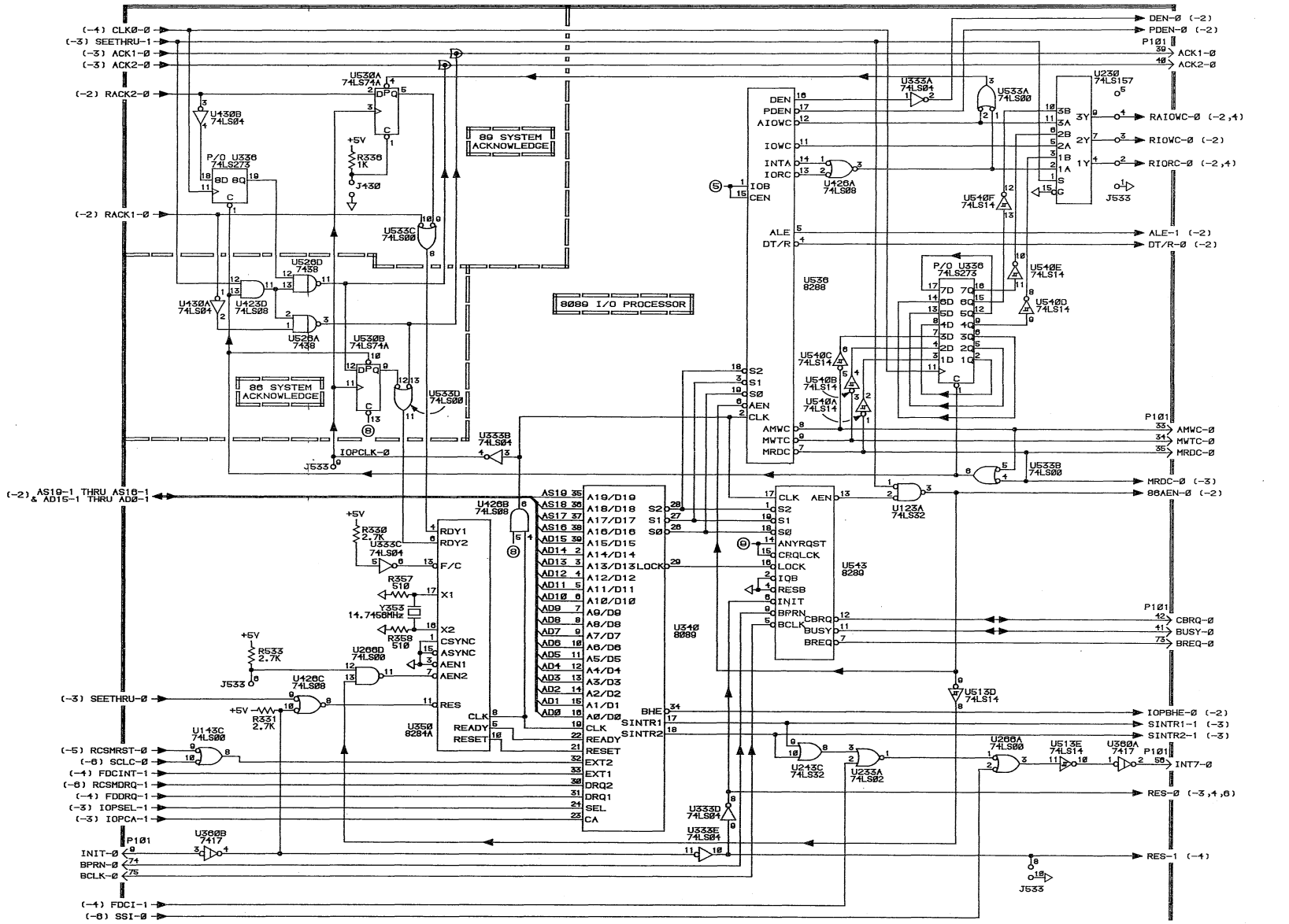


INSTRUMENT:	NOTES: * REFER TO STRAPPING TABLES.	BOARD: 672-1163-00	ASSEMBLY-SHEET:
4170	4885-S20	FD-MSIB INTERFACE BOARD	A7-6 (6 OF 7)

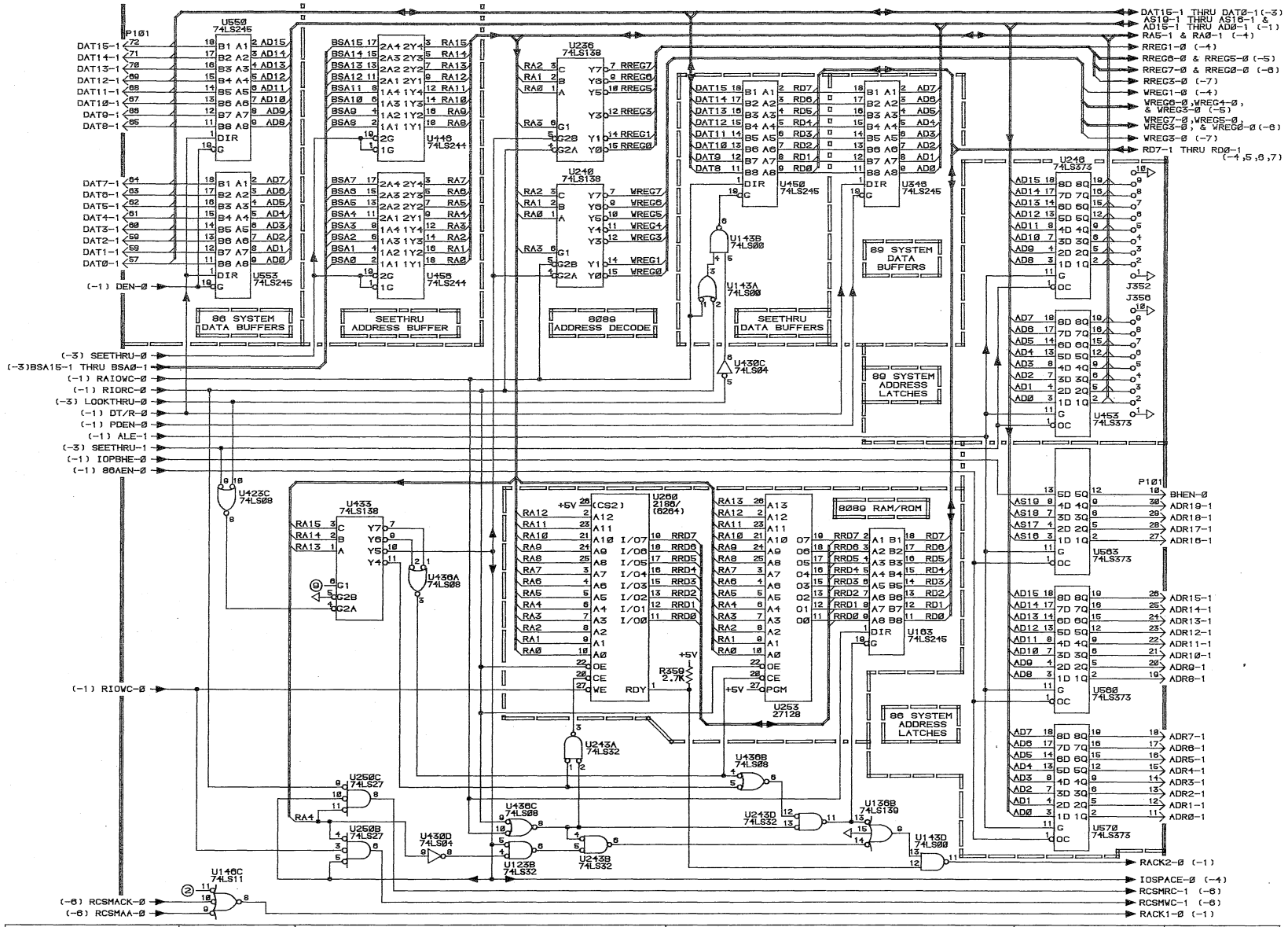


IC SUPPLIES		
DEVICE TYPE	PIN	
	+5V	GND
uPD765A	40	20
8089	40	1,20
6264 27128 2186	28	14
9216B	8	4
ALL 20 PIN IC'S	20	10
74LS75	5	12
OTHER 16 PIN IC'S	16	8
ALL 14 PIN IC'S	14	7

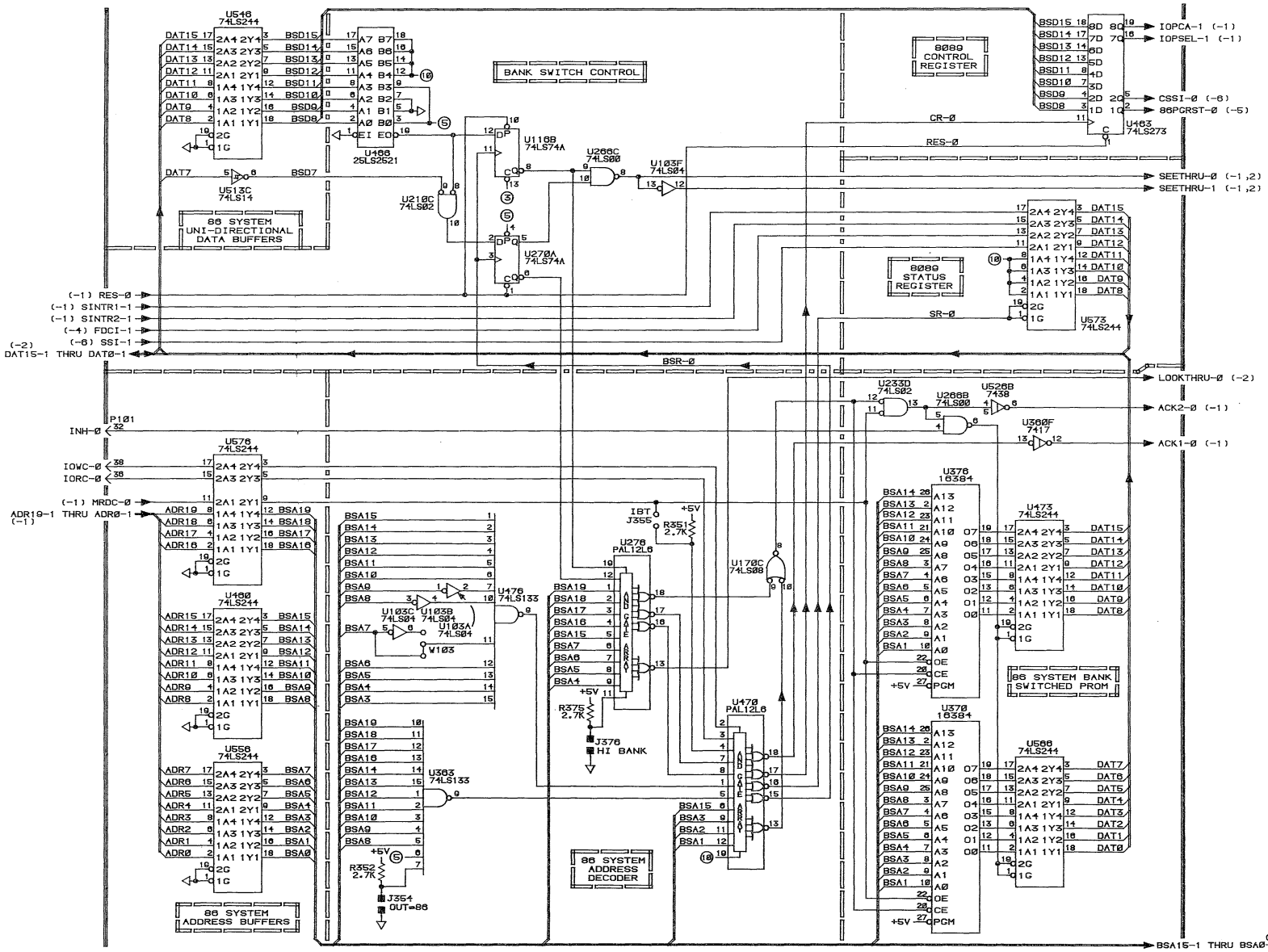
INSTRUMENT:	NOTES:	BOARD: 672-1163-00	ASSEMBLY-SHEET:
4170	4685-S21	FD-MSIB INTERFACE BOARD	A7-7 (7 OF 7)



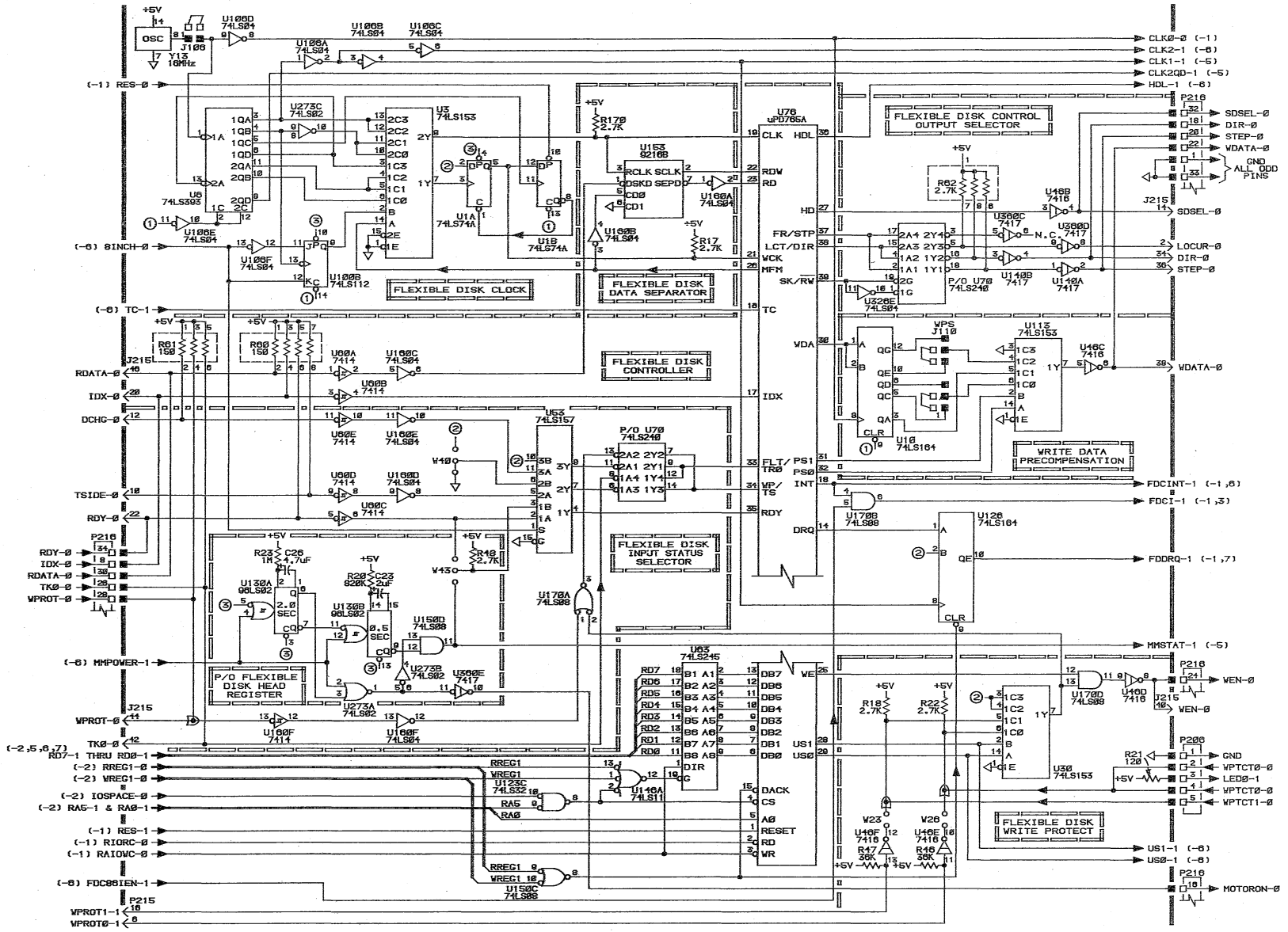
FIRST USE: 4170	OTHER USES:	NOTES:	TITLE: 672-1163-01	Tektronix®	ASSEMBLY: A7-1
DATE: JAN 1984			FD-MSIB INTERFACE BOARD		SHEET: 1 OF 7
CONTROL NO.: SSA027.001		TEKTRONIX, INC. © 1984			



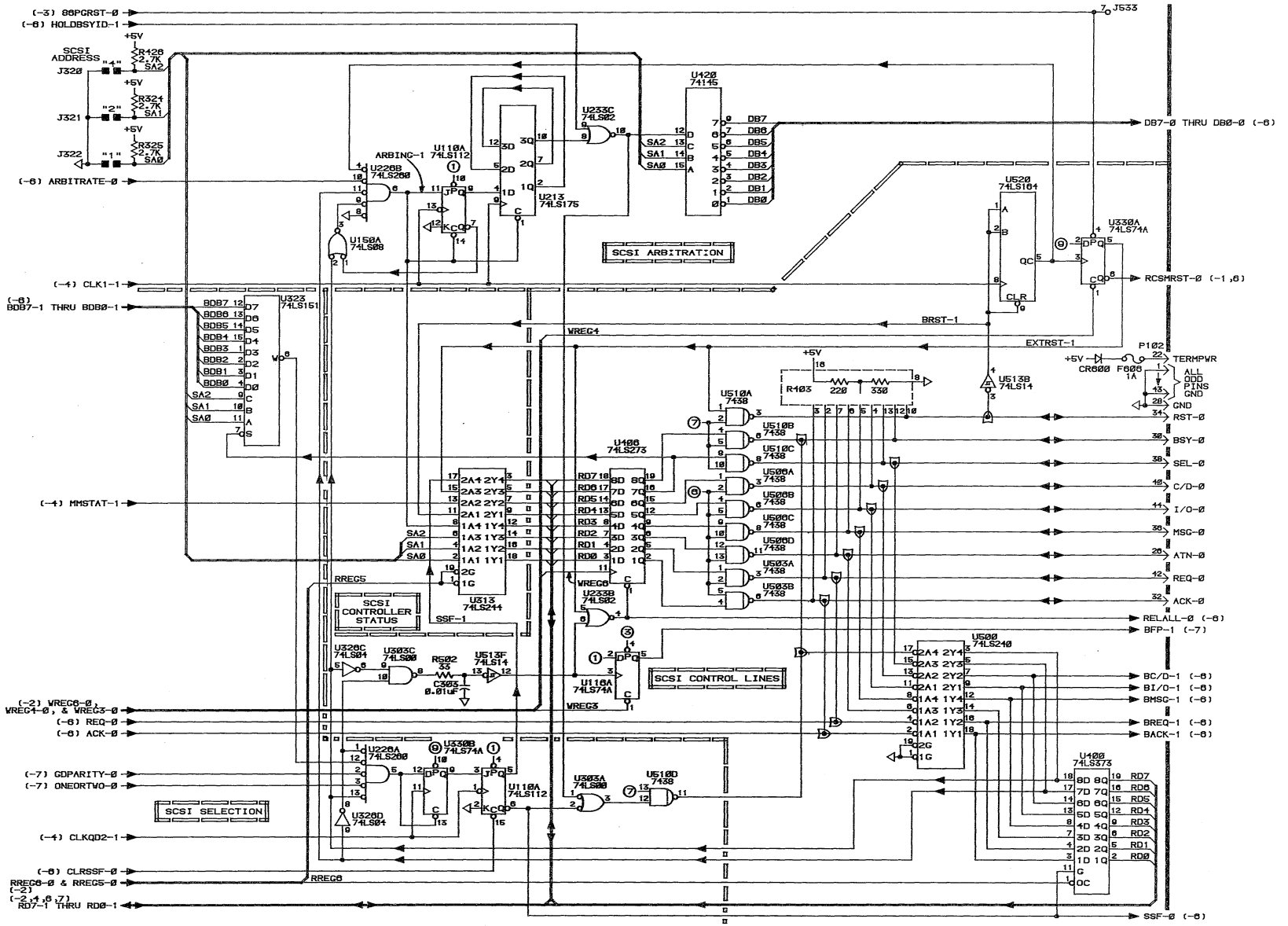
FIRST USE: 4170	OTHER USES:	NOTES:	TITLE: 672-1163-01	ASSEMBLY:
DATE: JAN 1984			FD-MSIB INTERFACE BOARD	A7-2
CONTROL NO.: SSA027.001		TEKTRONIX, INC. © 1984	Tektronix®	SHEET: 2 OF 7



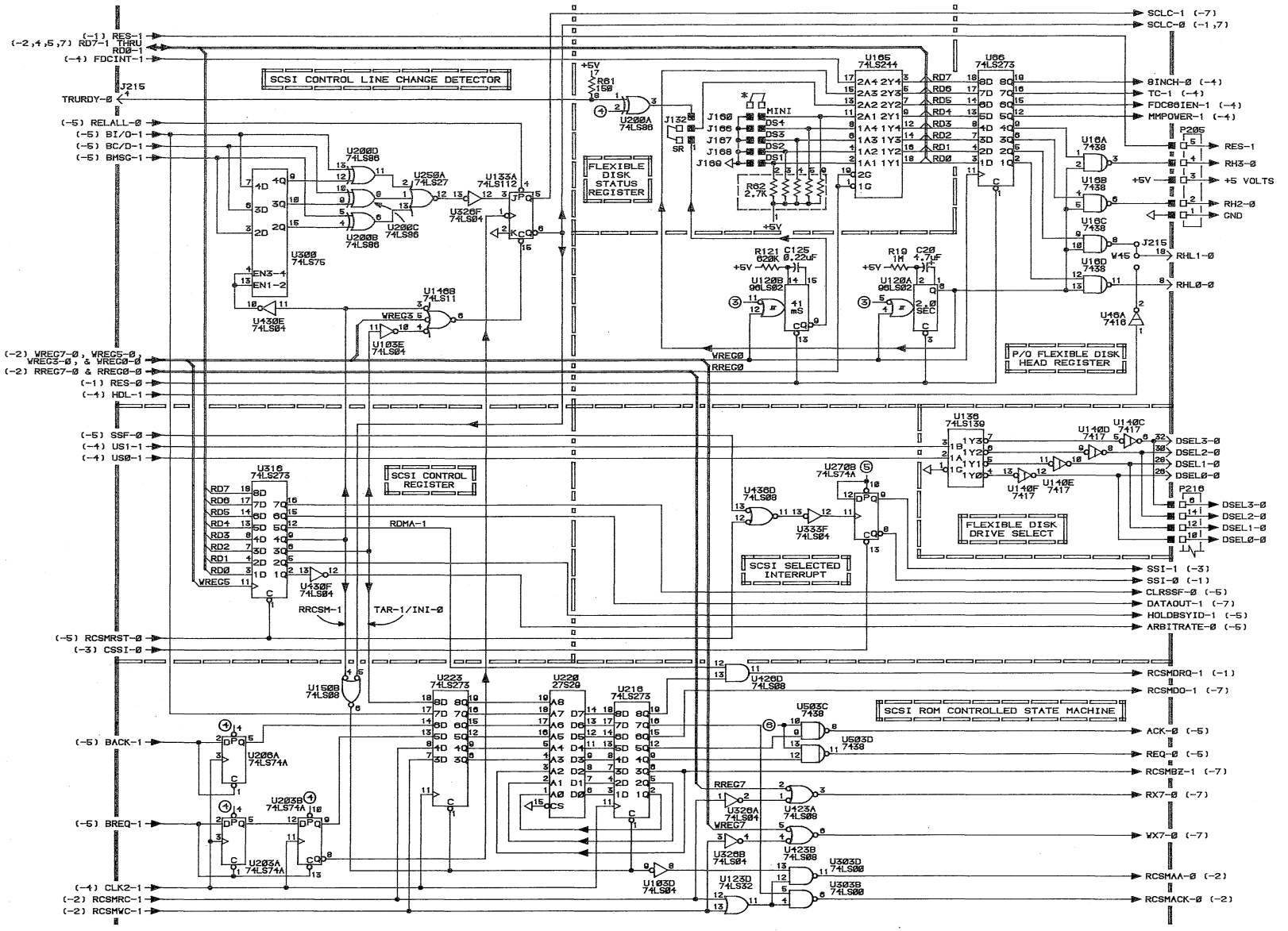
FIRST USE: 4170	OTHER USES:	NOTES:	TITLE: 672-1183-01	Tektronix®	ASSEMBLY:
DATE: JAN 1984			FD-MSIB INTERFACE BOARD		A7-3
CONTROL NO.: SSA027.001		TEKTRONIX, INC. © 1984			SHEET: 3 OF 7



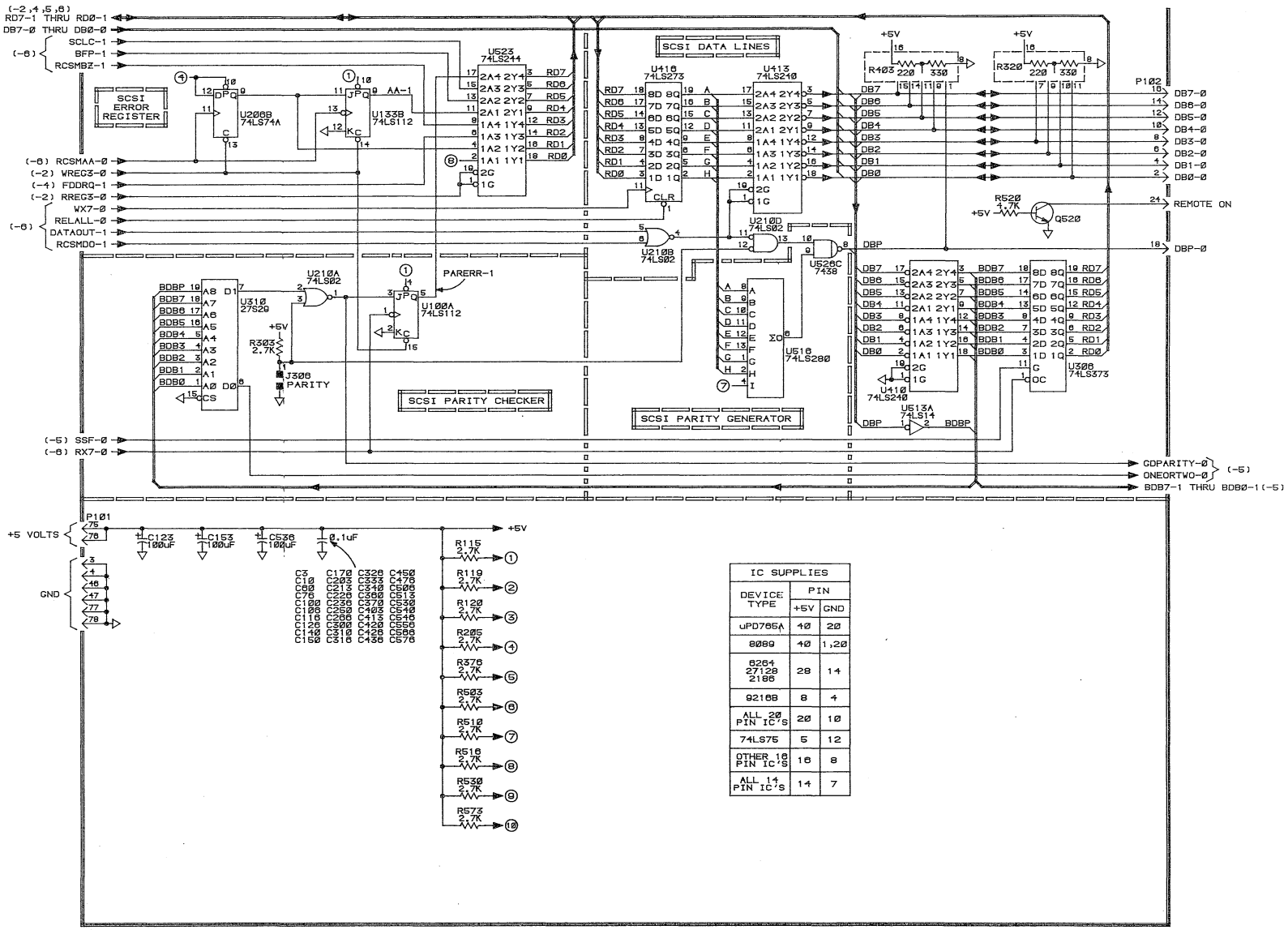
FIRST USE: 4170	OTHER USES:	NOTES:	TITLE: 672-1163-01	ASSEMBLY: A7-4
DATE: JAN 1984			FD-MSIB INTERFACE BOARD	SHEET: 4 OF 7
CONTROL NO.: SSA027.001		TEKTRONIX, INC. © 1984		



FIRST USE: 4170	OTHER USES:	NOTES:	TITLE: 672-1103-01	ASSEMBLY:
DATE: JAN 1984			FD-MSIB INTERFACE BOARD	A7-5
CONTROL NO.: SSA027.ØØ1		TEKTRONIX, INC. © 1984	Tektronix®	SHEET: 5 OF 7

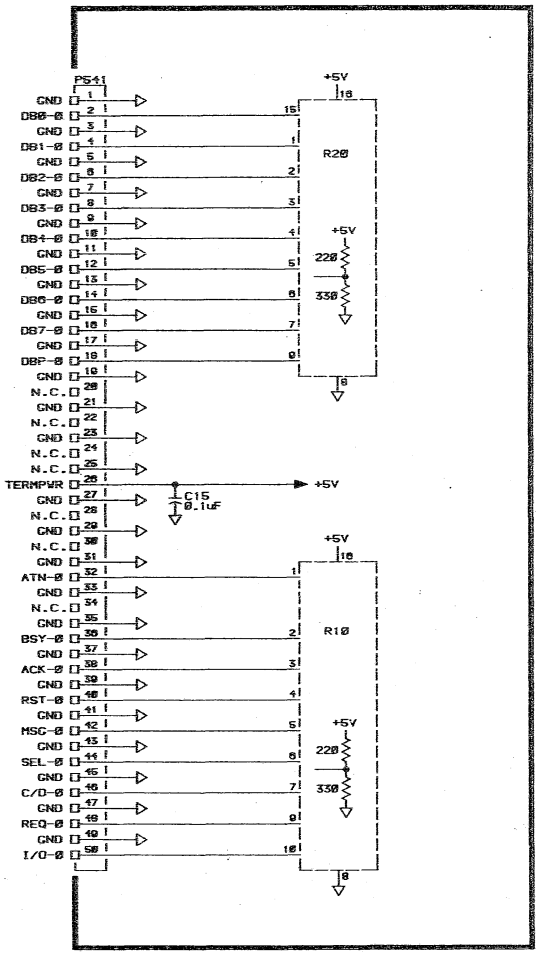


FIRST USE: 4170	OTHER USES:	NOTES: * REFER TO STRAPPING TABLES.	TITLE: 672-1163-01	Tektronix®	ASSEMBLY: A7-6
DATE: JAN 1984			FD-MSIB INTERFACE BOARD		SHEET: 6 OF 7
CONTROL NO.: SSA027.001		TEKTRONIX, INC. © 1984			

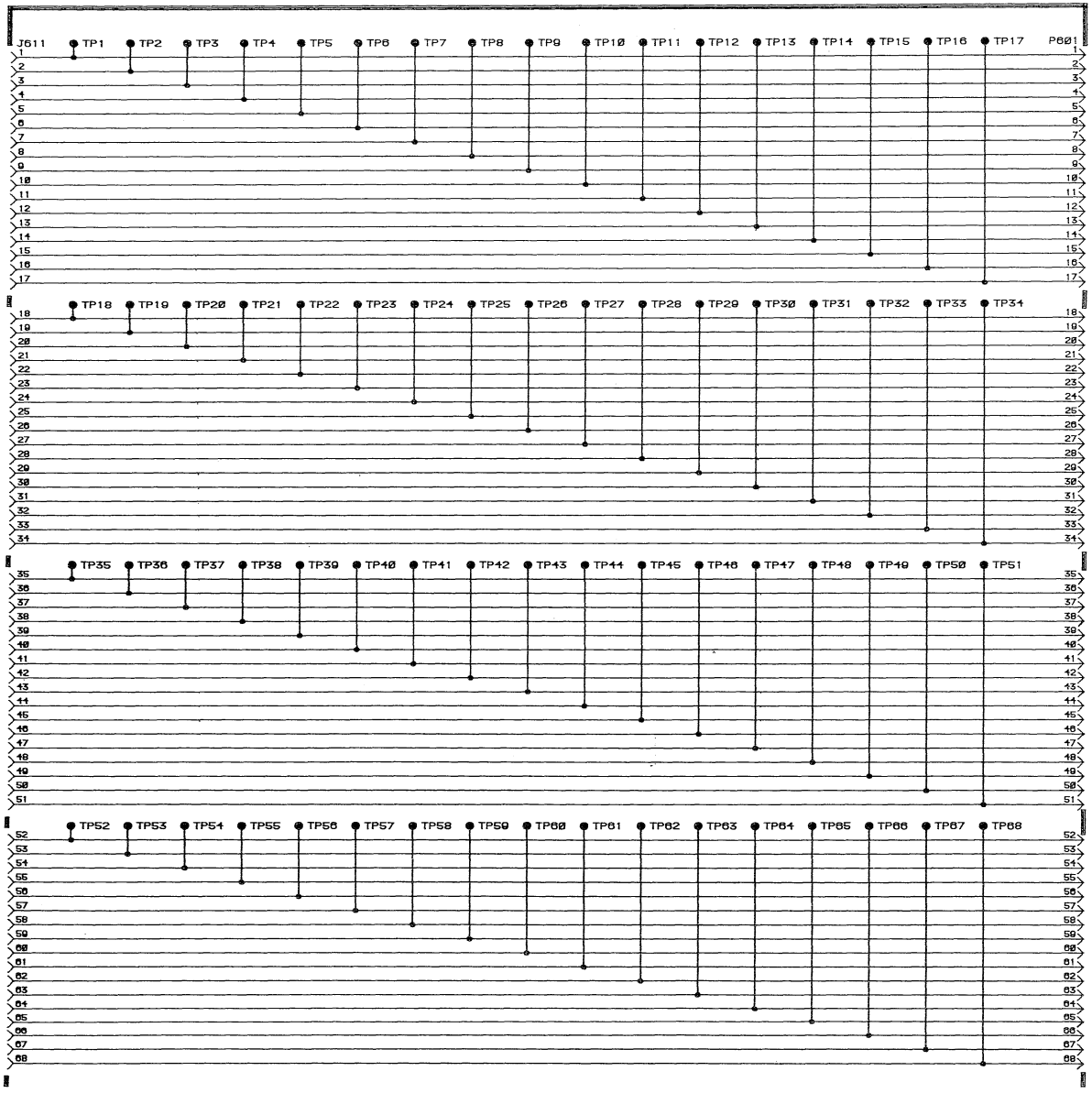


IC SUPPLIES		
DEVICE TYPE	PIN	
	+5V	GND
uPD765A	40	20
8089	40	1,20
8264 27128 2186	28	14
92188	8	4
ALL 20 PIN IC'S	20	10
74LS75	5	12
OTHER 16 PIN IC'S	16	8
ALL 14 PIN IC'S	14	7

FIRST USE: 4170	OTHER USES:	NOTES:	TITLE: 672-1183-01	ASSEMBLY:
DATE: JAN 1984			FD-MSIB INTERFACE BOARD	A7-7
CONTROL NO.: SSA027.001		TEKTRONIX, INC. © 1984	Tektronix®	SHEET: 7 OF 7

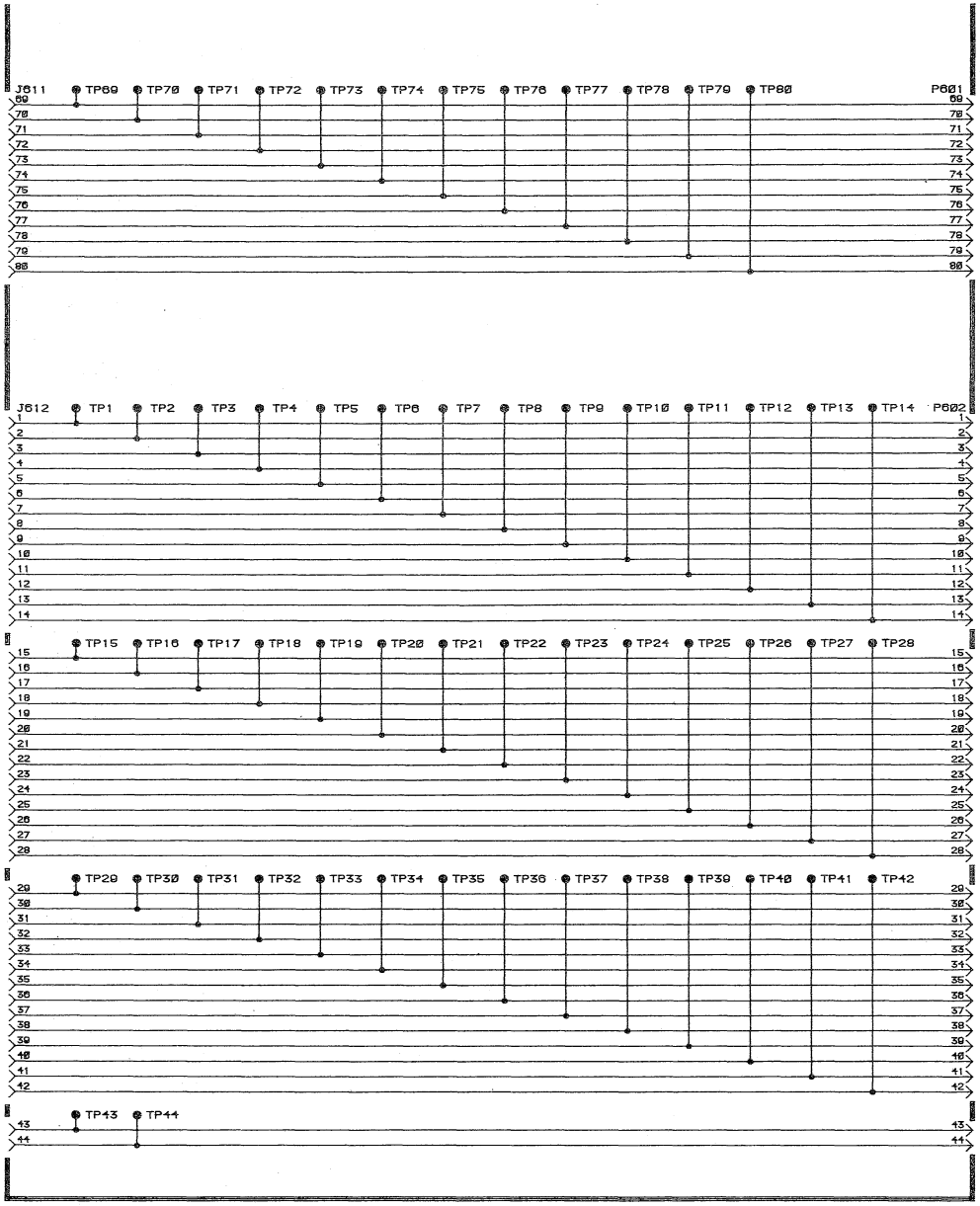


INSTRUMENT:	NOTES:	BOARD: 870-3367-00	ASSEMBLY-SHEET:
4170 OPT.45	4685-S22	MASS STORAGE TERMINATOR BOARD	A8-1 (1 OF 1)



J21 THRU J20
-12 VOLTS
-12 VOLTS
GND
GND
-5.1 VOLTS
-5.1 VOLTS
PFAIL-Ø
AGND
INIT-Ø
BHEN-Ø
ADRØ-1
ADR1-1
ADR2-1
ADR3-1
ADR4-1
ADR5-1
ADR6-1
ADR7-1
ADR8-1
ADR9-1
ADR1Ø-1
ADR11-1
ADR12-1
ADR13-1
ADR14-1
ADR15-1
ADR16-1
ADR17-1
ADR18-1
ADR19-1
INTA-Ø
INH-Ø
AMWC-Ø
MWTC-Ø
MRDC-Ø
IORC-Ø
AIOWC-Ø
IOWC-Ø
ACK1-Ø
ACK2-Ø
BUSY-Ø
CBRQ-Ø
STEST-Ø
SPARE
BCLK-Ø
GND
GND
SPARE
INTØ-Ø
INT1-Ø
INT2-Ø
INT3-Ø
INT4-Ø
INT5-Ø
INT6-Ø
INT7-Ø
DATØ-1
DAT1-1
DAT2-1
DAT3-1
DAT4-1
DAT5-1
DAT6-1
DAT7-1
DAT8-1
DAT9-1
DAT1Ø-1
DAT11-1

INSTRUMENT:	NOTES:	BOARD: 67Ø-5291-ØØ	ASSEMBLY-SHEET:
067-1005-ØØ (FOR USE WITH 4170)	4885-S23	LOGIC EXTENDER BOARD	A9-1 (1 OF 2)



* J21 THRU J29
DAT12-1
DAT13-1
DAT14-1
DAT15-1
BREQ-0
BPRN-0
+5 VOLTS
+5 VOLTS
GND
GND
+12 VOLTS
+12 VOLTS

*EXCEPT J28 THRU J29 PINS 73 AND 74 WHICH ARE N.C.

J102 (PROCESSOR)	J261 (3PPI)	J242 (TAB. CNTRL.)
+5 VOLTS	+5 VOLTS	BUTTON 1
-5 VOLTS	-5 VOLTS	PAOUT-0
GND	GND	BUTTON 2
GND	GND	PEN-0
TDATA-1	RD0-1	BUTTON 3
RDATA-1	TD0-1	MARGNOFST-1
RTS-1	CTS0-1	-12 VOLTS
CTS-1	RTS0-1	+12 VOLTS
DTR-1	DSR0-1	+12 VOLTS
DSR-1	DTR0-1	-12 VOLTS
SRTSA-1	DCD0-1	+5 VOLTS
DCD-1	SRTS0-1	DISABLE-0
SRTSC-1	N.C.	GND
SDCD-1	N.C.	GND
N.C.	N.C.	GND
RING-1	N.C.	GND
TCLK-1	GND	FIREX2-1
RCLK-1	GND	FIREY2-1
GND	RD2-1	FIREX1-1
GND	TD2-1	FIREY1-1
TXC-0	CTS2-1	N.C.
N.C.	RTS2-1	N.C.
N.C.	DSR2-1	N.C.
N.C.	DTR2-1	N.C.
N.C.	DCD2-1	N.C.
N.C.	SRTS2-1	N.C.
N.C.	N.C.	N.C.
N.C.	N.C.	N.C.
N.C.	N.C.	N.C.
N.C.	N.C.	N.C.
N.C.	GND	N.C.
N.C.	GND	N.C.
N.C.	RD1-1	N.C.
N.C.	TD1-1	N.C.
N.C.	CTS1-1	N.C.
N.C.	RTS1-1	N.C.
N.C.	DSR1-1	N.C.
N.C.	DTR1-1	N.C.
N.C.	DCD1-1	N.C.
N.C.	SRTS1-1	N.C.
N.C.	N.C.	N.C.
N.C.	N.C.	N.C.
-12 VOLTS	N.C.	N.C.
-12 VOLTS	N.C.	N.C.

INSTRUMENT:	NOTES:	BOARD: 670-5291-00	ASSEMBLY-SHEET:
067-1005-00 (FOR USE WITH 4170)	4685-S24	LOGIC EXTENDER BOARD	A9-2 (2 OF 2)

Section 17
REPLACEABLE MECHANICAL PARTS

PARTS ORDERING INFORMATION

Replacement parts are available from or through your local Tektronix, Inc. Field Office or representative.

Changes to Tektronix instruments are sometimes made to accommodate improved components as they become available, and to give you the benefit of the latest circuit improvements developed in our engineering department. It is therefore important, when ordering parts, to include the following information in your order: Part number, instrument type or number, serial number, and modification number if applicable.

If a part you have ordered has been replaced with a new or improved part, your local Tektronix, Inc. Field Office or representative will contact you concerning any change in part number.

Change information, if any, is located at the rear of this manual.

ITEM NAME

In the Parts List, an Item Name is separated from the description by a colon (:). Because of space limitations, an Item Name may sometimes appear as incomplete. For further Item Name identification, the U.S. Federal Cataloging Handbook H6-1 can be utilized where possible.

FIGURE AND INDEX NUMBERS

Items in this section are referenced by figure and index numbers to the illustrations.

INDENTATION SYSTEM

This mechanical parts list is indented to indicate item relationships. Following is an example of the indentation system used in the description column.

```
1 2 3 4 5                    Name & Description

Assembly and or Component
Attaching parts for Assembly and or Component
                - - - * - - -
   Detail Part of Assembly and or Component
   Attaching parts for Detail Part
                - - - * - - -
      Parts of Detail Part
      Attaching parts for Parts of Detail Part
                - - - * - - -
```

Attaching Parts always appear in the same indentation as the item it mounts, while the detail parts are indented to the right. Indented items are part of, and included with, the next higher indentation. The separation symbol - - - * - - - indicates the end of attaching parts.

Attaching parts must be purchased separately, unless otherwise specified.

ABBREVIATIONS

	INCH
#	NUMBER SIZE
ACTR	ACTUATOR
ADPTR	ADAPTER
ALIGN	ALIGNMENT
AL	ALUMINUM
ASSEM	ASSEMBLED
ASSY	ASSEMBLY
ATTEN	ATTENUATOR
AWG	AMERICAN WIRE GAGE
BD	BOARD
BRKT	BRACKET
BRS	BRASS
BRZ	BRONZE
BSHG	BUSHING
CAB	CABINET
CAP	CAPACITOR
CER	CERAMIC
CHAS	CHASSIS
CKT	CIRCUIT
COMP	COMPOSITION
CONN	CONNECTOR
COV	COVER
CPLG	COUPLING
CRT	CATHODE RAY TUBE
DEG	DEGREE
DWR	DRAWER
ELCTRN	ELECTRON
ELEC	ELECTRICAL
ELCTLT	ELECTROLYTIC
ELEM	ELEMENT
EPL	ELECTRICAL PARTS LIST
EQPT	EQUIPMENT
EXT	EXTERNAL
FIL	FILLISTER HEAD
FLEX	FLEXIBLE
FLH	FLAT HEAD
FLTR	FILTER
FR	FRAME or FRONT
FSTNR	FASTENER
FT	FOOT
FXD	FIXED
GSKT	GASKET
HDL	HANDLE
HEX	HEXAGON
HEX HD	HEXAGONAL HEAD
HEX SOC	HEXAGONAL SOCKET
HLCPS	HELICAL COMPRESSION
HLEXT	HELICAL EXTENSION
HV	HIGH VOLTAGE
IC	INTEGRATED CIRCUIT
ID	INSIDE DIAMETER
IDENT	IDENTIFICATION
IMPLR	IMPELLER
IN	INCH
INCAND	INCANDESCENT
INSUL	INSULATOR
INTL	INTERNAL
LPHLDR	LAMPHOLDER
MACH	MACHINE
MECH	MECHANICAL
MTG	MOUNTING
NIP	NIPPLE
NON WIRE	NOT WIRE WOUND
OBD	ORDER BY DESCRIPTION
OD	OUTSIDE DIAMETER
OVH	OVAL HEAD
PH BRZ	PHOSPHOR BRONZE
PL	PLAIN or PLATE
PLSTC	PLASTIC
PN	PART NUMBER
PNH	PAN HEAD
PWR	POWER
RCPT	RECEPTACLE
RES	RESISTOR
RGD	RIGID
RLF	RELIEF
RTNR	RETAINER
SCH	SOCKET HEAD
SCOPE	OSCILLOSCOPE
SCR	SCREW
SE	SINGLE END
SECT	SECTION
SEMICOND	SEMICONDUCTOR
SHLD	SHIELD
SHLDR	SHOULDERED
SKT	SOCKET
SL	SLIDE
SLFLKG	SELF-LOCKING
SLVG	SLEEVING
SPR	SPRING
SQ	SQUARE
SST	STAINLESS STEEL
STL	STEEL
SW	SWITCH
T	TUBE
TERM	TERMINAL
THD	THREAD
THK	THICK
TNSN	TENSION
TPG	TAPPING
TRH	TRUSS HEAD
V	VOLTAGE
VAR	VARIABLE
W	WITH
WSHR	WASHER
XFMR	TRANSFORMER
XSTR	TRANSISTOR

CROSS INDEX—MFR. CODE NUMBER TO MANUFACTURER

Mfr. Code	Manufacturer	Address	City, State, Zip
000JA	J. PHILLIP INDUSTRIES INC.	5713 NORTHWEST HIGHWAY	CHICAGO, ILL 60646
000KC	ALL METRIC	3231 FIRST AVE S.	SEATTLE, WA 98134
00779	AMP, INC.	P.O. BOX 3608	HARRISBURG, PA 17105
04919	COMPONENT MANUFACTURING SERVICE, INC.	1 COMPONENT PARK WEST	BRIDGEWATER, MA 02379
06383	PANDUIT CORPORATION	17301 RIDGELAND	TINLEY PARK, IL 60477
06915	RICHCO PLASTIC CO.	5825 N. TRIPP AVE.	CHICAGO, IL 60646
08261	SPECTRA-STRIP CORP.	7100 LAMPSON AVE.	GARDEN GROVE, CA 92642
09922	BURNDY CORPORATION	RICHARDS AVENUE	NORWALK, CT 06852
24618	TRANSCON MFG. CO.	2655 PERTH ST.	DALLAS, TX 75220
26244	UNIROYAL, LTD., INDUSTRIAL PRODUCTS DIV.	1806 NOTRE DAME STREET	E MONTREAL QUEBEC, CANADA
34785	DEK INC.	1555 HAWTHORNE LN.	W CHICAGO, IL 60185
52152	MINNESOTA MINING AND MFG CO.	INDUSTRIAL SPECIALTIES DIV. 3M CENTER	ST. PAUL, MN 55144
58361	GENERAL INSTRUMENT CORP. OPTO ELECTRONICS DIV.	3400 HILLVIEW AVE	PALO ALTO, CA 94304
70903	BELDEN CORP.	2000 S BATAVIA AVENUE	GENEVA, IL 60134
71468	ITT CANNON ELECTRIC	666 E. DYER RD.	SANTA ANA, CA 92702
77250	PHEOLL MANUFACTURING CO., DIVISION OF ALLIED PRODUCTS CORP.	5700 W. ROOSEVELT RD.	CHICAGO, IL 60650
78189	ILLINOIS TOOL WORKS, INC. SHAKEPROOF DIVISION	ST. CHARLES ROAD	ELGIN, IL 60120
80009	TEKTRONIX, INC.	P O BOX 500	BEAVERTON, OR 97077
80126	PACIFIC ELECTRICORD CO.	747 W. REDONDO BEACH,P O BOX 10	GARDENA, CA 90247
81041	HOWARD INDUSTRIES, DIVISION OF MSL INDUSTRIES, INC.	P O BOX 287	MILFORD, IL 60953
83385	CENTRAL SCREW CO.	2530 CRESCENT DR.	BROADVIEW, IL 60153
93907	TEXTRON INC. CAMCAR DIV	600 18TH AVE	ROCKFORD, IL 61101
98159	RUBBER TECK, INC.	19115 HAMILTON AVE., P O BOX 389	GARDENA, CA 90247
S3109	C/O PANEL COMPONENTS CORP.	P.O. BOX 6626	SANTA ROSA, CA 95406
T0435	LEWIS SCREW CO.	4114 SOUTH PERORIA AVE.	CHICAGO, IL 60609

Fig. & Index No.	Tektronix Part No.	Serial/Model No. Eff	Dscont	Qty	1 2 3 4 5	Name & Description	Mfr Code	Mfr Part Number
1-1	390-0434-00			1		CABINET SIDE:RIGHT		
						**********(ATTACHING PARTS)**********		
-2	212-0146-00			2		SCREW,MACHINE:8-32 X 0.625 L,FLH,100 DEG	T0435	OBD
-3	212-0010-00			2		SCREW,MACHINE:8-32 X 0.625 INCH,PNH STL	83385	OBD
						**********(END ATTACHING PARTS)**********		
-4	200-2912-00			1		COVER,KYBD ACS:	80009	200-2912-00
-5	334-5261-00			1		OVERLAY,FR PNL:POLYCARBONATE		
-6	334-0081-00			1		OVERLAY,PWR SW:MKD POWER		
-7	352-0700-00			1		HOLDER,LED:PLASTIC,2 PIECE	58361	CMP52(4-6B)
-8	334-0082-00			1		MARKER,IDENT:MKD TEKTRONIX 4170		
-9	200-2907-00			1		COVER,FRONT:		
						**********(ATTACHING PARTS)**********		
-10	213-0262-00			2		THUMBSCREW:8-32 THD X 0.45" L KNURLED	80009	213-0262-00
						**********(END ATTACHING PARTS)**********		
-11	334-0107-00			1		MARKER,IDENT:MKD 1,2,3,4,5,6,7		
-12	198-5309-00			1		WIRE SET,ELEC:		
-13	407-3203-00			1		BRACKET,CKT BD:	80009	407-3203-00
						**********(ATTACHING PARTS)**********		
-14	213-0107-00			1		SCR,TPG,THD FOR:4-40 X 0.25 INCH,FLH STL	93907	OBD
						**********(END ATTACHING PARTS)**********		
-15	----- -----			1		CKT BD ASSY:FRONT PANEL BOARD (SEE A4 REPL		
						**********(ATTACHING PARTS)**********		
-16	213-0034-00			4		SCR,TPG,THD CTG:4-40 X 0.188 INCH,PNH STL	83385	OBD
						**********(END ATTACHING PARTS)**********		
	----- -----			-		CKT BOARD ASSY INCLUDES:		
-17	136-0728-00			2		.SKT,PL-IN ELEK:MICROCKT,14 CONTACT	09922	DILB14P-108
-18	366-2031-00			17		.PUSHBUTTON:SLATE GRAY,0.45 X 0.275 X 0.17		
-19	175-8745-00			1		CA ASSY,SP,ELEC:3,26 AWG,24.0 L,RIBBON	80009	175-8745-00
-20	----- -----			1		FAN:(SEE B1001 CHASSIS PARTS REPL)		
						**********(ATTACHING PARTS)**********		
-21	210-0457-00			4		NUT,PL,ASSEM WA:6-32 X 0.312,STL CD PL	83385	OBD
-22	200-2222-00			1		GUARD,FAN:	81041	6-182-033
-23	378-0076-00			1		FILTER,MESH:4.5 SQ,AL	80009	378-0076-00
-24	211-0623-00			4		SCREW,MACHINE:6-32 X 2.25 INCHES,PNH,STL	83385	OBD
						**********(END ATTACHING PARTS)**********		
-25	386-4371-00			1		PANEL,BLANK:	80009	386-4371-00
						**********(ATTACHING PARTS)**********		
-26	211-0578-00			4		SCREW,MACHINE:6-32 X 0.438 1NCH,PNH STL	83385	OBD
						**********(END ATTACHING PARTS)**********		
-27	200-1532-06			3		COVER,INTFC CAV:ALUMINUM	80009	200-1532-06
						**********(ATTACHING PARTS)**********		
-28	211-0578-00			12		SCREW,MACHINE:6-32 X 0.438 1NCH,PNH STL	83385	OBD
						**********(END ATTACHING PARTS)**********		
-29	334-0079-00			1		OVERLAY,PANEL:REAR,LEGEND BLACK	80009	334-0079-00
-30	334-0080-00			1		OVERLAY,PANEL:REAR,SMOKE TAN		
-31	390-0423-00			1		CABINET SIDE:LEFT		

Fig. 1 CABINETIZING

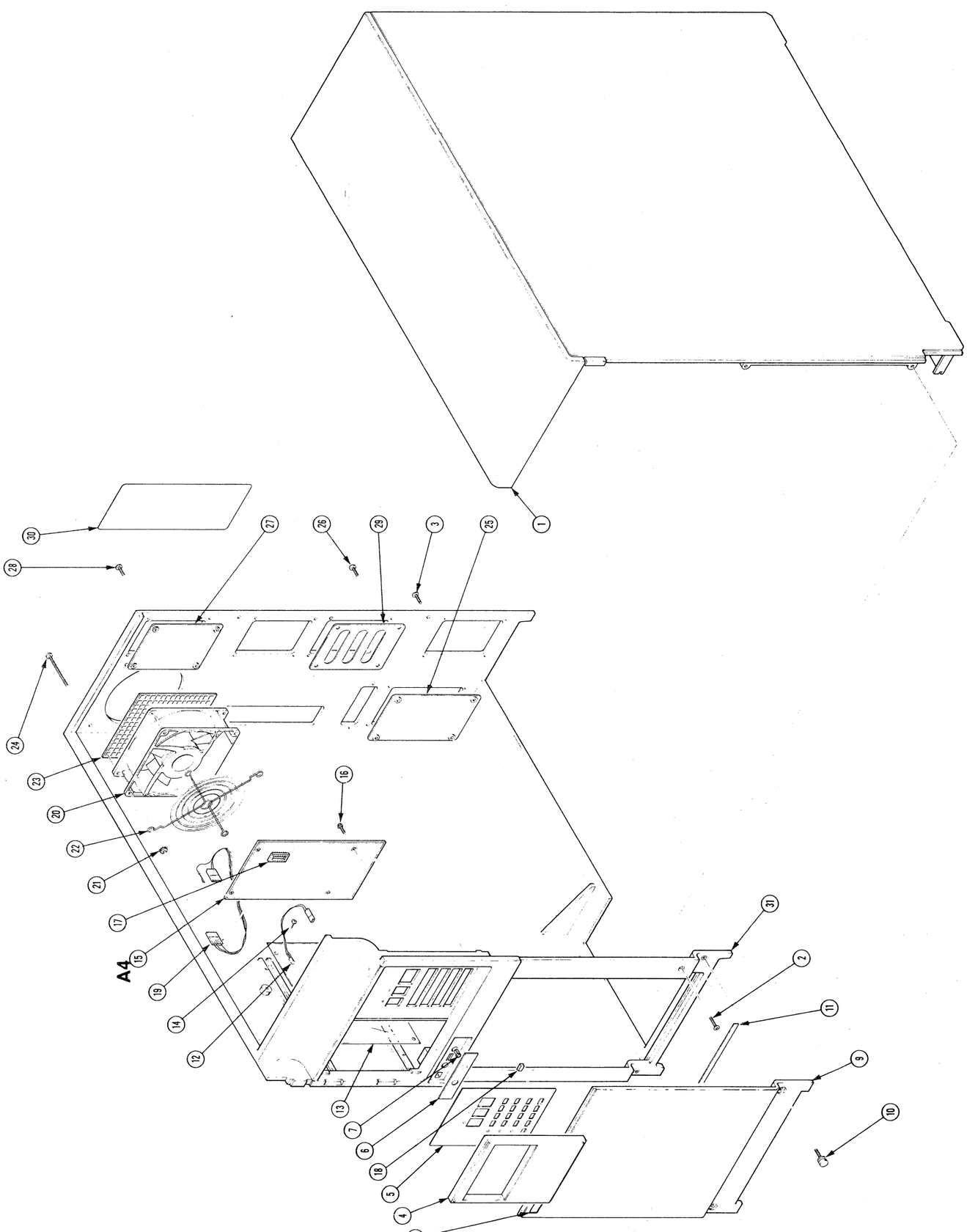


Fig. 2 DISK DRIVE
AND POWER SUPPLY

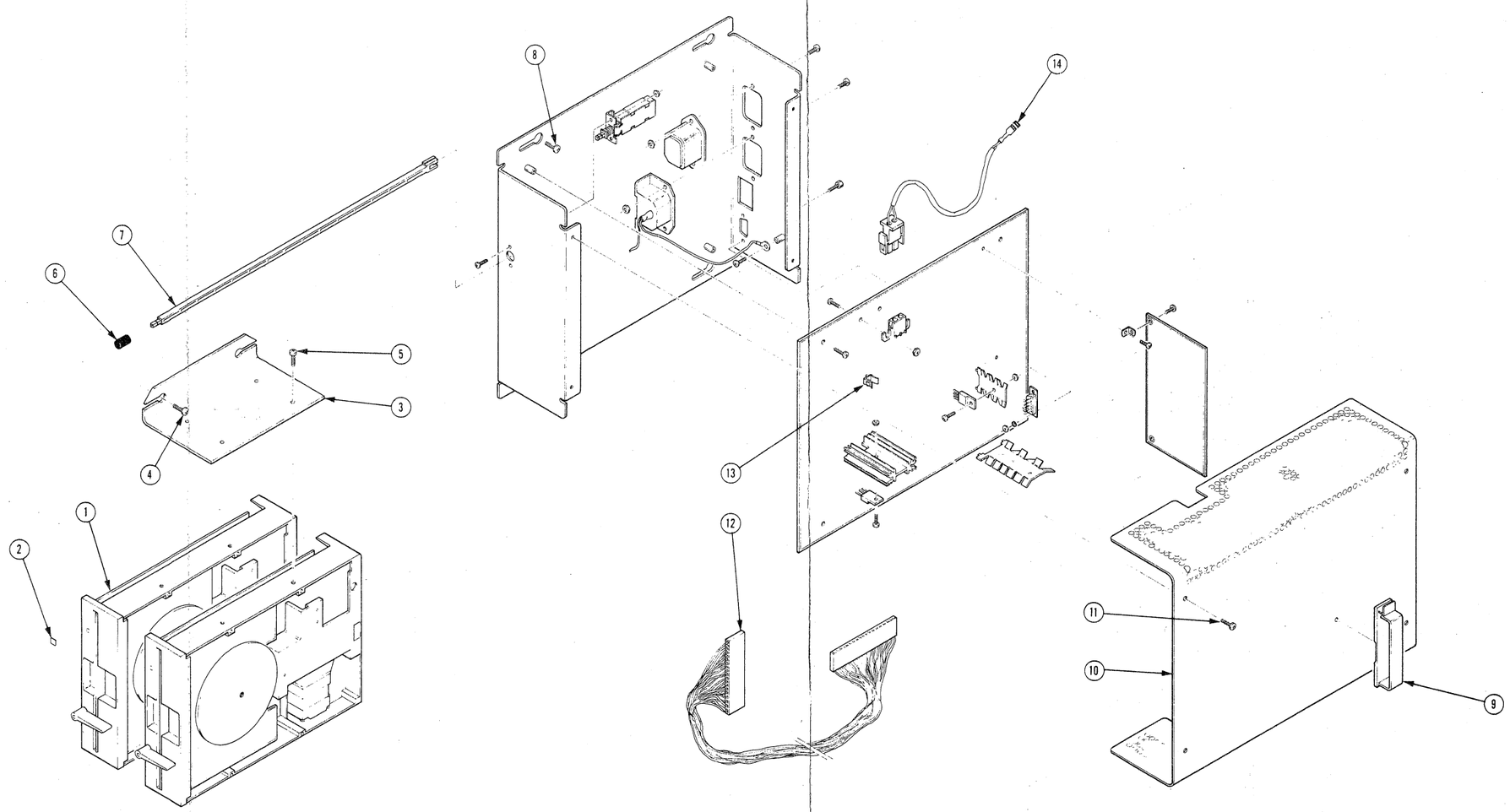


Fig. & Index No.	Tektronix Part No.	Serial/Model No. Eff	Dscont	Qty	1 2 3 4 5	Name & Description	Mfr Code	Mfr Part Number
2-1	----- -----			2		FLOPPY DISKS:(SEE OPT ACCESS FOR MANUAL NO		
-2	334-0104-00			1		MARKER,IDENT:MARKED (A)(B)(C)(D)		
-3	407-1344-00			2		BRACKET,MTG:DISK DRIVE	80009	407-1344-00
						**********(ATTACHING PARTS)**********		
-4	212-0045-00			4		SCREW,MACHINE:8-32 X 0.500 INCH,TRH,STL	83385	OBD
-5	211-0329-00			8		SCREW,MACHINE:M3 X 0.5 X 10MML,PNH	000KC	OBD
						**********(END ATTACHING PARTS)**********		
-6	366-0548-00			1		PUSHBUTTON:BLACK,0.346 DIA,1.04 H		
-7	384-0530-00			1		EXTENSION SHAFT:12.625 L X 0.187 ID,NYLON	80009	384-0530-00
	----- -----			1		POWER SUPPLY:(SEE A5 REPL)		
						**********(ATTACHING PARTS)**********		
-8	212-0045-00			4		SCREW,MACHINE:8-32 X 0.500 INCH,TRH,STL	83385	OBD
						**********(END ATTACHING PARTS)**********		
	----- -----			-		POWER SUPPLY ASSY INCLUDES:		
-9	343-0914-00			1		.CLAMP,LOOP:0.5 X 2.5,NYLON	34785	034-0500
-10	200-2904-00			1		.COVER,PWR SPLY:10.75 X 7.285 X 3.062,AL	80009	200-2904-00
						.**********(ATTACHING PARTS)**********		
-11	212-0119-00			4		.SCREW,CAP:8-32 X 0.250,BUTTON,STEEL		
						.**********(END ATTACHING PARTS)**********		
-12	198-5214-00			1		.WIRE SET,ELEC:		
-13	175-9133-00			1		CA ASSY,SP,ELEC:2,18 AWG,11.0 L		

Fig. & Index No.	Tektronix Part No.	Serial/Model No. Eff	Dscont	Qty	1 2 3 4 5	Name & Description	Mfr Code	Mfr Part Number
3-1	----- -----			1		CKT BOARD ASSY:RAM CONTROLLER & RAM ARRAY		
	----- -----			-		(SEE OPTIONAL ACCESSORIES FOR MANUAL NUMBE		
-2	----- -----			1		CKT BOARD ASSY:PROCESSOR (SEE A2 REPL)		
-3	105-0851-00			2		.EJECTOR,CKT BD:GRAY PLASTIC	80009	105-0851-00
						.**********(ATTACHING PARTS)**********		
-4	214-1337-00			2		.PIN,SPRING:0.10 OD X 0.25 INCH L,STL	80009	214-1337-00
						.**********(END ATTACHING PARTS)*********		
-5	346-0032-00			1		.STRAP,RETAINING:0.075 DIA X 4.0 L,MLD RBR	98159	2859-75-4
-6	136-0757-00			1		.SKT,PL-IN ELEK:MICROCKT,40 PIN	09922	DILB40P-108
-7	124-0388-00			1		.BUS,CONDUCTOR:CIRCUIT BOARD,16 TAB	80009	124-0388-00
-8	131-0993-00			15		.BUS,CONDUCTOR:2 WIRE BLACK	00779	850100-01
-9	136-0751-00			8		.SKT,PL-IN ELEK:MICROCKT,24 PIN	09922	DILB24P108
-10	175-8746-00			1		CA ASSY,SP,ELEC:20,28 AWG,24.0 L,RIBBON		
-11	175-3186-00			1		CA ASSY,SP,ELEC:14,26 AWG,1,22 AWG,24.0 L	80009	175-3186-00
						**********(ATTACHING PARTS)**********		
-12	211-0033-00			2		SCR,ASSEM WSHR:4-40 X 0.312 PNH,STL,CD PL	83385	OBD
						**********(END ATTACHING PARTS)*********		
	----- -----			-		CABLE INCLUDES:		
-13	214-3082-00			4		.SPRING,RTNG:CU BE,GOLD FASH OVER NP	00779	583691-3
-14	214-3114-00			2		.KEY,CONN PLZN:CIRCUIT BD CONN	00779	530687-1
-15	131-2161-01			14		.TERM,QIK DISC:24-28 AWG,PHOSPHOR	00779	583616-2
-16	204-0877-00			1		.CONN BODY,RCPT:22/44 CONTACT	00779	583891-9
-17	175-0827-00			AR		.CABLE,SP,ELEC:4,26 AWG,STRD.PVC JKT,RBN	08261	SS04267(1061)0C
-18	175-0833-00			AR		.WIRE,ELECTRICAL:10 WIRE RIBBON	08261	SS-1026-7
-19	210-0202-00			1		.TERMINAL,LUG:0.146 ID,LOCKING,BRZ TINNED	78189	2104-06-00-2520N
-20	131-1279-00			14		.CONTACT,ELEC:MALE,28-24 AWG WIRE,0.040 D	00779	205310-4
	131-1450-00			1		.CONTACT,ELEC:MALE,24-30 AWG WIRE,0.040 D	00779	205202-6
-21	131-1316-00			1		.CONNECTOR BODY,:25 MALE-CONT POSITIONS	00779	208076-1
-22	343-0549-00			1		.STRAP,TIEDOWN:0.091 W X 3.62 INCH LONG	06383	PLT1M
-23	131-0890-00			2		LOCK,CONNECTOR:4-40 X 0.312 L	71468	D 20418-2
-24	386-2903-00			1		PANEL,CONN MTG:ALUMINUM	80009	386-2903-00
						**********(ATTACHING PARTS)**********		
-25	211-0578-00			2		SCREW,MACHINE:6-32 X 0.438 1NCH,PNH STL	83385	OBD
						**********(END ATTACHING PARTS)*********		
-26	----- -----			1		CKT BOARD ASSY:3PPI		
	----- -----			-		(SEE OPTIONAL ACCESSORIES FOR MANUAL NUMBE		
-27	198-5216-00			1		WIRE SET,ELEC:		
-28	----- -----			1		CKT BOARD ASSY:MOTHER (SEE A1 REPL)		
						**********(ATTACHING PARTS)**********		
-29	211-0504-00			8		SCREW,MACHINE:6-32 X 0.25 INCH,PNH STL	83385	OBD
						**********(END ATTACHING PARTS)*********		
-30	175-8744-00			1		CA ASSY,SP,ELEC:3,22 AWG,18.0 L,RIBBON	80009	175-8744-00
-31	343-1120-00			2		CLAMP,CABLE:RBN,NYL,W/DOUBLE BACK ADHS	06915	FCR-25B-4-ADHES
-32	343-0775-00			3		CLIP,SPR TNSN:	52152	3484-1000
-33	386-2904-00			2		PANEL,CKT BD CG:UPPER & LOWER	80009	386-2904-00
						**********(ATTACHING PARTS)**********		
-34	213-0789-00			12		SCREW,TPG,TF:6-32 X 0.375,TAPTITE,PNH	93907	OBD
						**********(END ATTACHING PARTS)*********		
-35	441-0194-00			1		CHASSIS,CKT BD:ALUMINUM	80009	441-0194-00

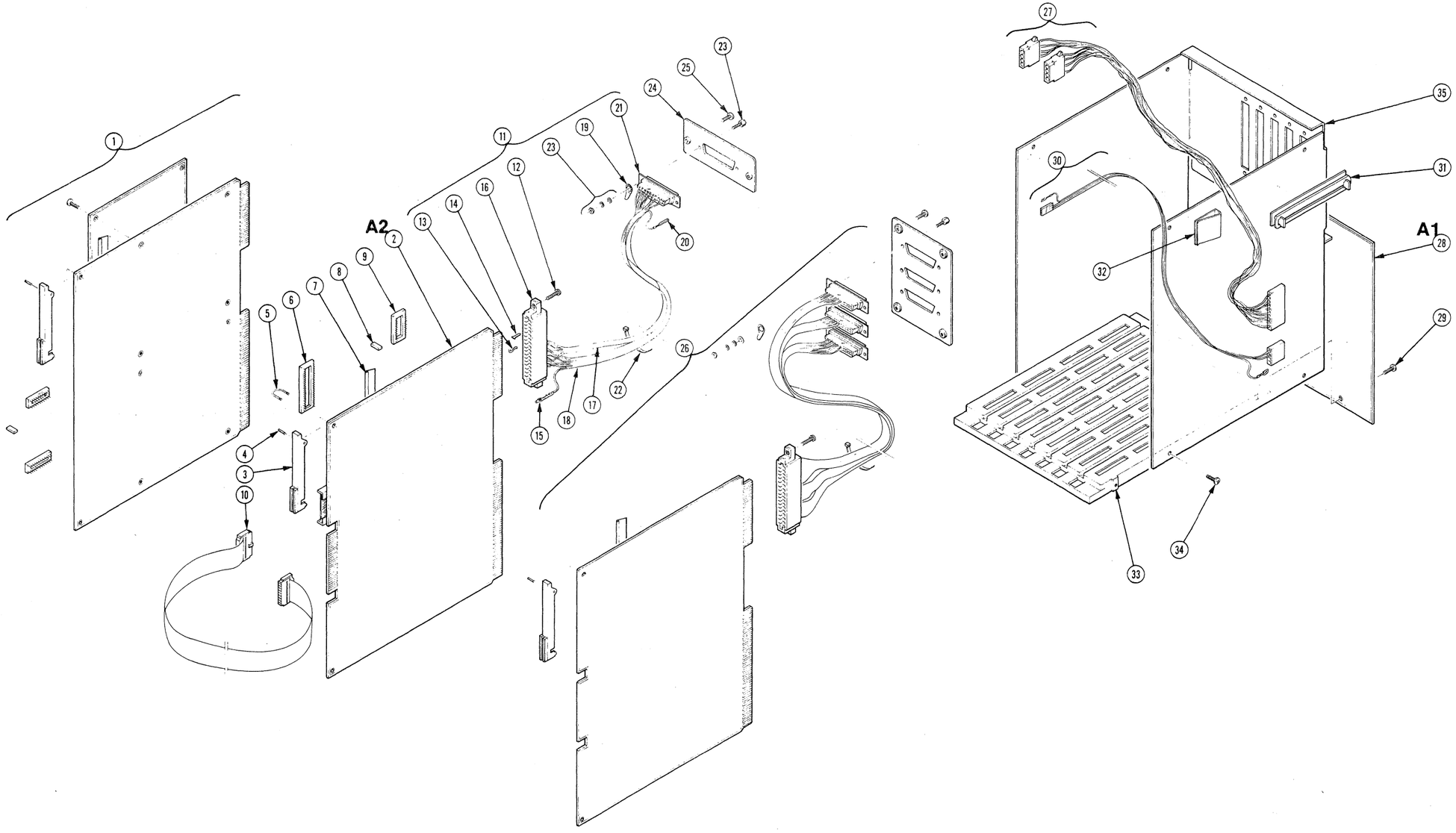
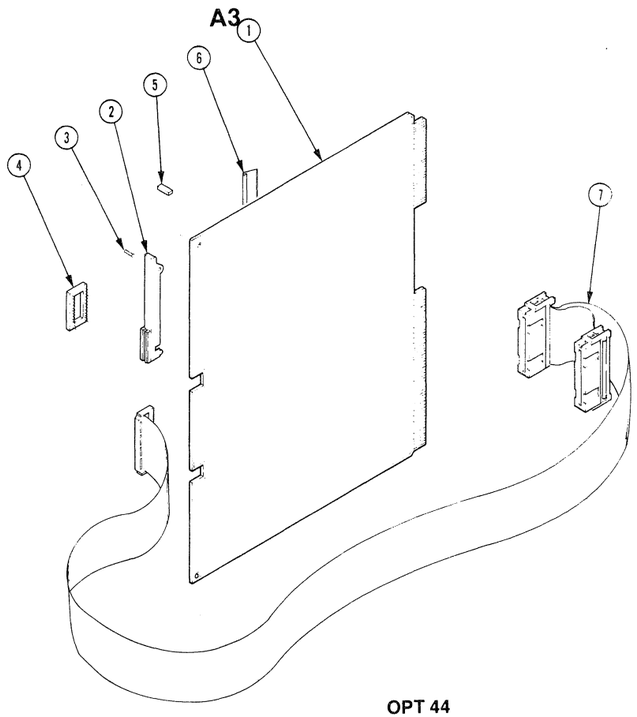


Fig. 3 CARD CAGE

Fig. 4 OPTION 3

Fig. & Index No.	Tektronix Part No.	Serial/Model No. Eff	Dscont	Qty	1 2 3 4 5	Name & Description	Mfr Code	Mfr Part Number
4-1	212-0045-00			4		SCREW,MACHINE:8-32 X 0.500 INCH,TRH,STL	83385	OBD
-2	407-3138-00			1		BRACKET,SUPPORT:REAR	80009	407-3138-00
-3	407-3137-00			1		BRACKET,SUPPORT:FRONT	80009	407-3137-00
-4	348-0008-00			6		MOUNT,RESILIENT:0.562 DIA X 0.5,(2)8-32	26244	301ABLUE
						**********(ATTACHING PARTS)**********		
-5	210-0458-00			12		NUT,PL,ASSEM WA:8-32 X 0.344 INCH,STL	83385	OBD
						**********(END ATTACHING PARTS)**********		
-6	----- -----			1		CKT BOARD ASSY:XEBEC (SEE A6 REPL)		
						**********(ATTACHING PARTS)**********		
-7	211-0198-00			4		SCREW,MACHINE:4-40 X 0.438 PNH,STL,POZ	77250	OBD
						**********(END ATTACHING PARTS)**********		
-8	198-5215-00			1		WIRE SET,ELEC:		
-9	175-9030-00			1		CA ASSY,SP,ELEC:34,28 AWG,12.5 L,RIBBON		
-10	175-9031-00			1		CA ASSY,SP,ELEC:20,28 AWG,13.0 L,RIBBON		
-11	407-1478-00			1		BRACKET,MTG:WINCHESTER	80009	407-1478-00
						**********(ATTACHING PARTS)**********		
-12	211-0504-00			4		SCREW,MACHINE:6-32 X 0.25 INCH,PNH STL	83385	OBD
						**********(END ATTACHING PARTS)**********		
-13	----- -----			1		HARD DISK DRIVE:		
	----- -----			-		(SEE OPTIONAL ACCESSORIES FOR MANUAL NUMBE		

Fig. & Index No.	Tektronix Part No.	Serial/Model No. Eff	Dscont	Qty	1 2 3 4 5	Name & Description	Mfr Code	Mfr Part Number
5-	----- -----			-		OPTION 44		
-1	----- -----			1		.CKT BOARD ASSY:DISK CONTROLLER (SEE A3 RE		
-2	105-0851-00			2		..EJECTOR,CKT BD:GRAY PLASTIC	80009	105-0851-00
						..**********(ATTACHING PARTS)**********		
-3	214-1337-00			2		..PIN,SPRING:0.10 OD X 0.25 INCH L,STL	80009	214-1337-00
						..**********(END ATTACHING PARTS)*********		
-4	136-0751-00			8		..SKT,PL-IN ELEK:MICROCKT,24 PIN	09922	DILB24P108
-5	131-0993-00			10		..BUS,CONDUCTOR:2 WIRE BLACK	00779	850100-01
-6	124-0388-00			1		..BUS,CONDUCTOR:CIRCUIT BOARD,16 TAB	80009	124-0388-00
-7	175-8789-00			1		.CA ASSY,SP,ELEC:34,28 AWG,38.5 L,RIBBON		
	----- -----			-		OPTION 45		
-8	----- -----			1		.CKT BOARD ASSY:INTERFACE (SEE A7 REPL)		
-9	105-0851-00			2		..EJECTOR,CKT BD:GRAY PLASTIC	80009	105-0851-00
						..**********(ATTACHING PARTS)**********		
-10	214-1337-00			2		..PIN,SPRING:0.10 OD X 0.25 INCH L,STL	80009	214-1337-00
						..**********(END ATTACHING PARTS)*********		
-11	136-0755-00			3		..SKT,PL-IN ELEK:MICROCIRCUIT,28 DIP	09922	DILB28P-108
-12	131-0993-00			11		..BUS,CONDUCTOR:2 WIRE BLACK	00779	850100-01
-13	136-0757-00			1		..SKT,PL-IN ELEK:MICROCKT,40 PIN	09922	DILB40P-108
-14	136-0729-00			2		..SKT,PL-IN ELEK:MICROCKT,16 CONTACT	09922	DILB16P-108T
-15	124-0388-00			1		..BUS,CONDUCTOR:CIRCUIT BOARD,16 TAB	80009	124-0388-00
-16	175-8790-00			1		.CA ASSY,SP,ELEC:34,28 AWG,33.0 L,RIBBON		
-17	175-8747-00			1		.CA ASSY,SP,ELEC:50,28 AWG,42.0 L,RIBBON		
						.**********(ATTACHING PARTS)**********		
-18	211-0033-00			2		.SCR,ASSEM WSHR:4-40 X 0.312 PNH,STL,CD PL	83385	OBD
-19	129-0471-00			2		.SPACER,POST:0.650 L,W/4-40 EXT THD ONE		
-20	210-0006-00			2		.WASHER,LOCK:#6 INTL,0.018 THK,STL CD PL	78189	1206-00-00-0541C
-21	210-0586-00			2		.NUT,PL,ASSEM WA:4-40 X 0.25,STL	83385	OBD
						.**********(END ATTACHING PARTS)*********		
-22	006-0531-00			1		.STRAP,TIEDOWN,E:BLUE PLASTIC BEADED	24618	700-3688
-23	352-0482-00			1		.HOLDER,CA,TIE:0.75 SQ,STICKY BACK,PLASTIC	06383	ABMM-A
-24	343-0549-00			1		.STRAP,TIEDOWN:0.091 W X 3.62 INCH LONG	06383	PLT1M
-25	----- -----			1		.BNC (SEE J3005 CHASSIS PARTS REPL)		
-26	386-5080-00			1		.PANEL,CONN MTG:2.98 X 4.12 ALUMINUM	80009	386-5080-00
						.**********(ATTACHING PARTS)**********		
-27	211-0578-00			4		.SCREW,MACHINE:6-32 X 0.438 1NCH,PNH STL	83385	OBD
						.**********(END ATTACHING PARTS)*********		
-28	----- -----			1		CKT BOARD ASSY: TERMINATOR (SEE A8 REPL)		



OPT 44

OPT 45

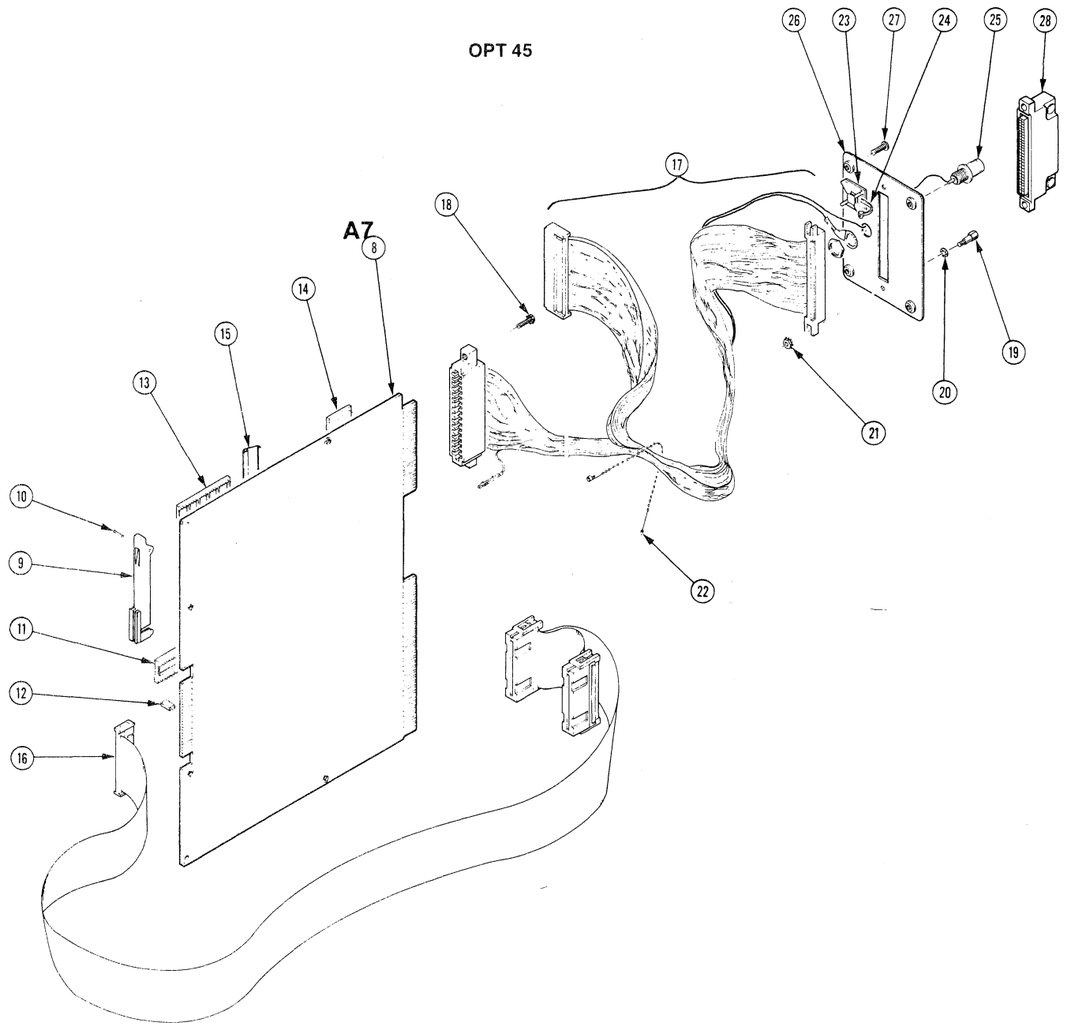


Fig. 5 OPTION 44 AND 45

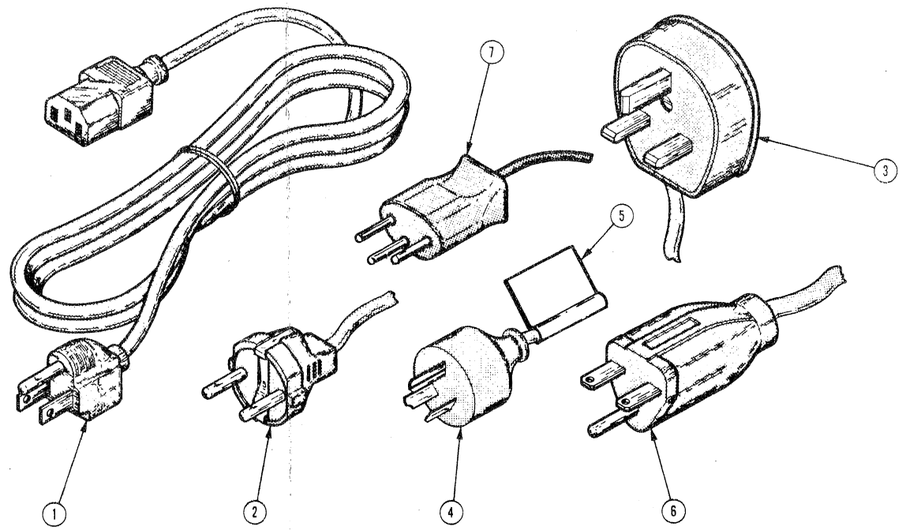


FIG. 6 STANDARD ACCESSORIES

Fig. & Index No.	Tektronix Part No.	Serial/Model No. Eff	Dscont	Qty	1 2 3 4 5	Name & Description	Mfr Code	Mfr Part Number
						STANDARD ACCESSORIES		
6-	334-0083-00					CUE CARD:		
	119-1583-00					DISK,FLOPPY:5.25 INCH,48-TPI		
	070-4482-00					MANUAL,TECH:USERS,4010C01 PLOT 10 IGL	80009	070-4482-00
	070-4609-00					MANUAL,TECH:USERS,4010C0120 PLOT 10 IGL	80009	070-4609-00
	062-6739-00					MANUAL SET,TECH:CP/M-86 DOCUMENTATION		
	062-7205-00					MANUAL,TECH:4100P01 DRI,GSX		
	062-6737-00					MANUAL,TECH:USERS,4110P01 FORTRAN-86		
	062-6736-00					MANUAL,TECH:REF GUIDE,4110P01 FORTRAN-86		
	062-6735-00					MANUAL,TECH:USERS,4110P01,IPEX 86-88		
	062-6734-00					MANUAL,TECH:REF GUIDE,4110P01,IPEX 86-88		
						STANDARD ACCESSORIES (ORDERED AS OPTION A0 THRU A5)		
-1	161-0123-00					CABLE ASSY,PWR:3,16 AWG,125V,96.01	70903	OBD
	----- -----					(OPTION A0 4170 ONLY)		
	161-0153-00					CABLE ASSY,PWR:3,26 AWG,2M L,250V,6A	80009	161-0153-00
	----- -----					(OPTION A0 THRU A5)		
-2	161-0123-01					CABLE ASSY,PWR:3,0.75MM SQ,220V,96.0 L	S3109	1100
	----- -----					(OPTION A1 EUROPEAN ONLY)		
-3	161-0123-02					CABLE ASSY,PWR:3,16 AWG,240V,96.0 L	80126	OBD
	----- -----					(OPTION A2 UNITED KINGDOM ONLY)		
-4	161-0123-03					CABLE ASSY,PWR:3,1MM SQ,240V,96.0 L	S3109	1600
	----- -----					(OPTION A3 AUSTRALIAN ONLY)		
-5	334-3995-00					MARKER,IDENT:MARKED CAUTION	80009	334-3995-00
	----- -----					(OPTION A3 AUSTRALIAN ONLY)		
-6	161-0123-04					CABLE ASSY,PWR:3,1.0MM SQ,240V,96.0 L	80126	OBD
	----- -----					(OPTION A4 NORTH AMERICAN ONLY)		
-7	161-0154-00					CABLE ASSY,PWR:3,0.75MM SQ,240V,6A,2.5M L	000JA	A25SW
	----- -----					(OPTION A5 SWISS ONLY)		
	003-0616-00					SCDR,CROSS TIP,POZIDRIVE #1,3.0 L BLADE		
	061-2880-00					MANUAL,TECH:INSTR,4170	80009	061-2880-00
	012-0911-00					CABLE,INTCON 144.0 L	04919	OBD
	067-1043-00					FIXTURE,CAL:HOST PORT LOOP BACK CONN	80009	067-1043-00
						OPTIONAL ACCESSORIES		
	119-1692-00					DISKETTE ALIGN:		
	067-1005-00					FIXTURE,CAL:EXTENDER BOARD	80009	067-1005-00
	006-5993-00					CLEANING KIT:DISK DRIVE		
	119-1583-01					DISK,FLOPPY:5.25 INCH,48-TPI	80009	119-1583-01
	----- -----					(PKG OF 10)		
	016-0764-00					STORAGE BOX:5.25 DISK		
	----- -----					(10 PER BOX)		
	070-4731-00					MANUAL,TECH:SERVICE DISK DRIVE	80009	070473100
	070-4701-00					MANUAL,TECH: SERVICE	80009	070-4701-00
	070-3815-00					MANUAL,TECH:INSTR,021-0313-00	80009	070-3815-00
	070-4497-00					MANUAL,TECH:INSTR,4113 OPT 09	80009	070-4497-00
	070-4773-00					MANUAL,TECH:INSTR,670-0073-00	80009	070-4773-00
	250-0129-00					MANUAL MATERIAL 2.0 CAP,D-RING WHITE BINDE		

Appendix A

STRAP INFORMATION

Circuit boards in the 4170 contain straps that allow flexibility in selecting operating parameters. These straps are either "jumper straps" or "cut straps".

Jumper straps are square pins (0.100 inch wide) that are connected electrically by removable wired jumpers. Jumper straps are used when operating parameters are frequently changed. In cases where an operating parameter is nearly always wired one way for the majority of applications, a cut strap is used instead of a jumper strap. Here the square pins are omitted, leaving only the circuit pad; the jumper is replaced by a circuit run between pads. To change this strap function, just cut the circuit run. In some cases it is necessary to cut the circuit run and complete the circuit path between other circuit pads. To close a normally open cut strap, connect a piece of wire between the pads or form a solder bridge. Labeling of a jumper strap takes the for of "Jxxx" and labeling of a cut strap takes the form of "Wxxx".

Actual strap positions for 4170 circuit boards is found in the installation section of this manual. This appendix describes straps for these circuit boards:

- Processor
- ECC RAM Memory
- 3PPI (both standard and optional)
- Option 09 Color Copier and Printer Interface
- Option 44 Disk Controller
- Option 45 Disk Controller

PROCESSOR BOARD STRAPPING

Table A-1 lists the Processor board straps and Figure A-1 shows the location of the Processor board straps. Figure A-2 shows the strap settings for ROM type straps J226, J326, J436, and J427.

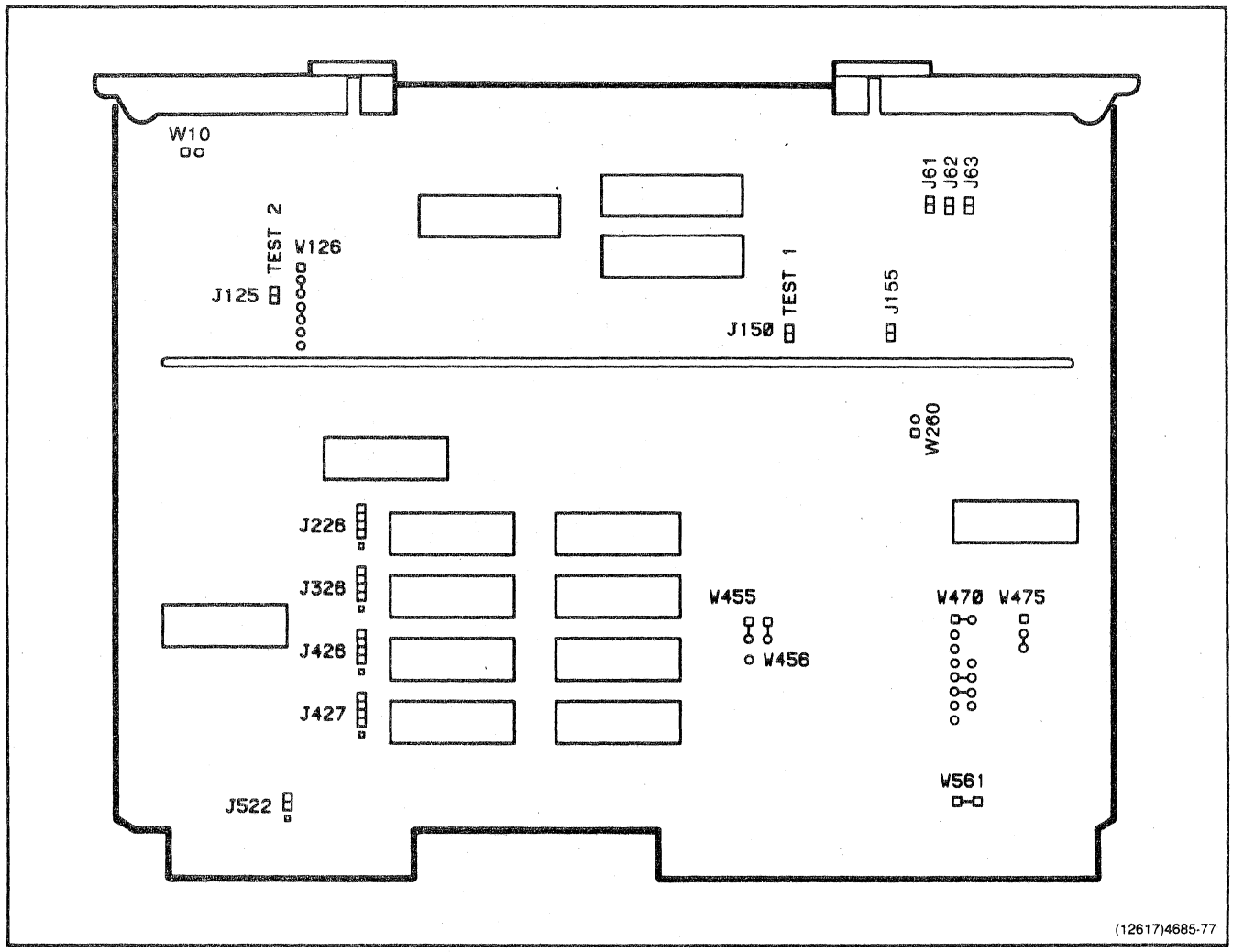


Figure A-1. Processor Board Strap Locations.

Table A-1

PROCESSOR BOARD STRAPS

Strap Label	**Definition**
ROM Logic Straps: ROM Size Strap (W126)	Selects either 16K or 32K bit ROMs, or disables all Processor board ROMs.
ROM Wait States (W475)	Normally strap for ONE wait state. If all the ROMs on the Processor board are fast enough, this strap may be changed to indicate ZERO wait states.
ROM Type Straps (J226,J326,J426, J427)	There is one set of straps for each two-ROM bank of ROMs. These straps configure the board for the pin-out of the ROM being used. See Figure A-2.
BLCK Source (W455 and W456)	Normally strapped to "on-board." If more than one Processor board is used in the system, only one should be strapped to "on-board;" all others should be strapped to "off-board."
Interrupt Level Straps: (W470)	These Straps define the interrupt priority levels of the three different interrupts. Factory strap settings are: "host port receiver interrupts" -- interrupt level 0; "keyboard interrupts" -- interrupt level 4; "timeout interrupts" and "timer interrupts" -- interrupt level 5; and "8087 FPU interrupts"-- interrupt level 7.

Table A-1 (cont)

PROCESSOR BOARD STRAPS

Strap Label	Definition
Bus Timeout Enable: (W561)	Prevents the Processor board from driving ACK1-0 on a bus timeout. Used for multi-processor board systems.
Test 1 and Test 2: (J150 and J125)	Tests 1 and 2 disable clocks on the Processor board for ATE (Automated Test Equipment) testing.
RS-232C/RS-232A: (J522)	This is normally set for RS-232C. When restrapped for RS-232A, the SRTS (Secondary Request To Send) signal is sent to Pin 11 of the 25-pin RS-232 connector rather than to Pin 19.
Test (J155)	Test disables RST-1 signal to 8086 and 8087 MPU and FPU for ATE (Automated Test Equipment) testing.
FPU Interrupt (J61)	Enables 8087 Numerical Co-processor to actuate interrupt when an exception (abnormal or error) condition occurs.
Mode 0 (J62)	Indicates to operating system through status register the 8087 FPU is present.

Table A-1 (cont)

PROCESSOR BOARD STRAPS

Strap Label	**Definition**
Mode 1 (J63)	Not used. Do not install strap.
High RAM Boot (W260)	Not used. Do not install strap.
Battery Grounding Strap (W10)	Do not use. **CAUTION** Installing this strap with power applied to the circuit will result in serious equipment damage.

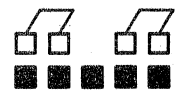
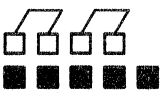
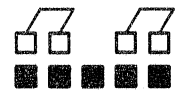
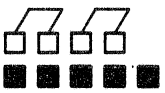
ROM TYPE SELECT					
EPROMS			MASKED ROMS		
DEVICE TYPE	SIZE	STRAPPING	DEVICE TYPE	SIZE	STRAPPING
INTEL 2716	2K		INTEL 2332	4K	
INTEL 2732A-3	4K		T.I. TMS4732	4K	
INTEL 2758	1K		MOTOROLA MCM 68A332	4K	
T.I. TMS2516	2K		SYNERTEK SYC2332	4K	
T.I. TMS2532	4K				(3812)4685-78

Figure A-2. ROM Type Strap Settings.

ECC RAM CONTROLLER BOARD STRAPPING

There are 22 different straps, divided into nine functional groups, on the ECC RAM Controller Board. Both cut straps and jumper straps are used. Figure A-3 shows the ECC RAM Controller Board strap locations and the following paragraphs describe the nine functional groups.

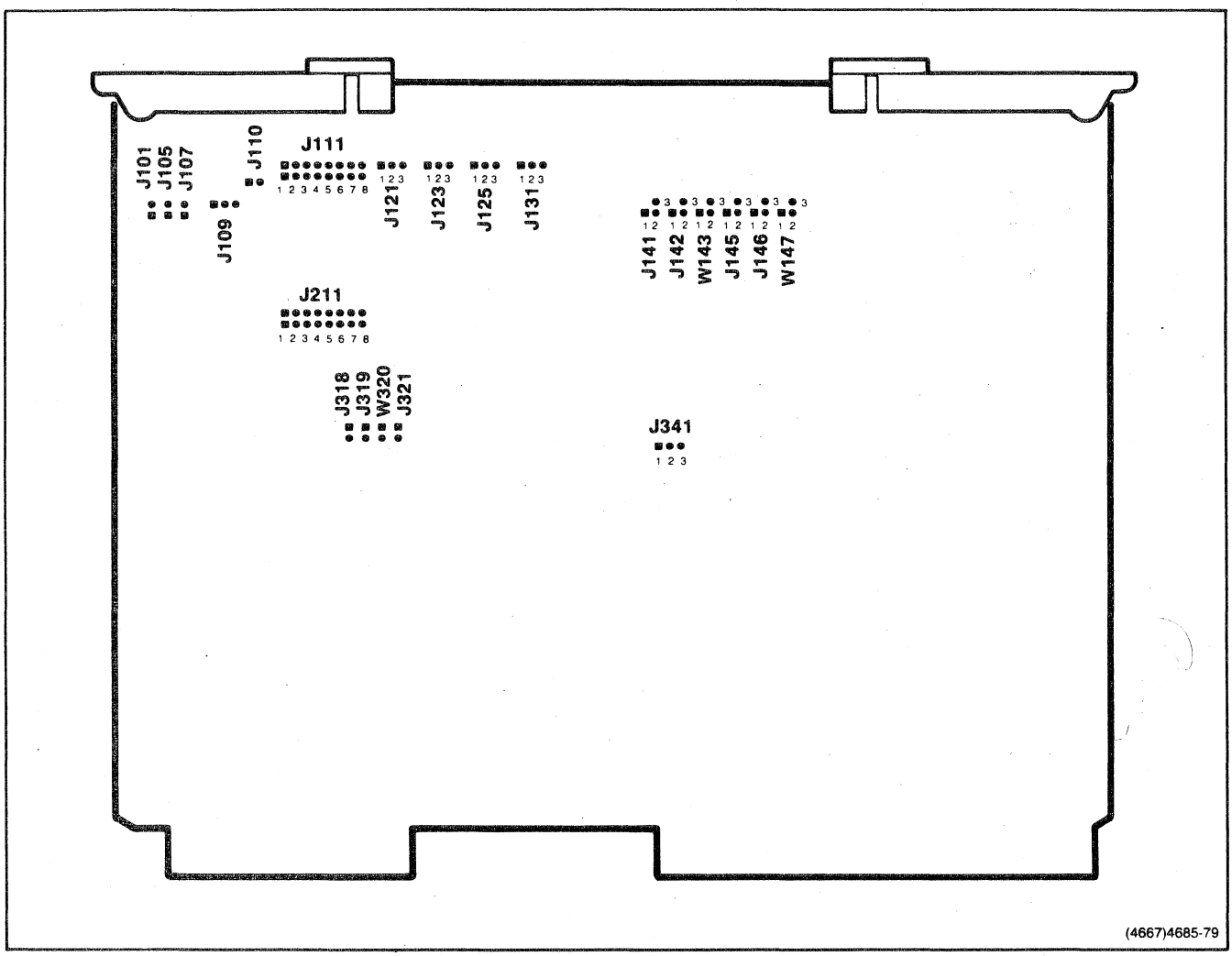


Figure A-3. ECC RAM Board Strap Locations.

I/O ADDRESS/BANK NUMBERS STRAPS

The three I/O address/bank numbers straps (J101, J105, and J107) assign bank numbers and I/O addresses to the board. The I/O address and bank numbers can not be independently assigned--assigning one automatically assigns the other. Table A-2 describes the I/O address/bank numbers strap positions. The I/O addresses in Table A-2 are hexadecimal.

Table A-2

I/O ADDRESS/BANK NUMBER STRAP POSITIONS

Bank Number	**J101**	**J105**	**J107**	**I/O Address**
0 & 1	Removed	Removed	Removed	00A0
2 & 3	Removed	Removed	Installed	00A2
4 & 5	Removed	Installed	Removed	00A4
6 & 7	Removed	Installed	Installed	00A6
8 & 9	Installed	Removed	Removed	00A8
10 & 11	Installed	Removed	Installed	00AA
12 & 13	Installed	Installed	Removed	00AC
No Banks	Installed	Installed	Installed	00AE

APPENDIX A
STRAP INFORMATION

BANKS/NO BANKS STRAP

The BANKS/NO BANKS strap is a three-pin strap that sets up to ECC Memory Board so that it recognizes or ignores bank addressing at power up. Table A-3 describes the strap positions.

Table A-3

BANK/NO BANK STRAP POSITIONS

Mode	J109
No Banks	1
Banks	3

4170 MODE

The 4170 Mode strap is a two-pin strap that configures the ECC Memory Board for operation in the 4170 Local Graphics Processing Unit or a 4110-Series Computer Display Terminal. Table A-4 describes 4170 MODE strap positions.

Table A-4

4170 MODE STRAP POSITION

Instrument Type	J110
4170	Installed
4110B-Series	Removed

PORT A ADDRESS RANGE

The Port A Address Range strap J111 is a two-pin strap that sets the address range to which port A will respond. Setting of this strap depends on the total amount of installed ECC RAM. J111 has eight legal positions; the removed position is not legal. Place the strap vertically between the two rows of pins. Square pads identify position 1 and are at the left. Sequential positions 2 thorough 8 begin at the right of the square pads. Table A-5 describes the strap positions for each increment of ECC memory.

NOTE

Each RAM Controller Board can contain a maximum of 512K bytes of memory. Strap the first RAM Controller Board for either 256K or 512K. Strap the second RAM Controller Board for the total amount of installed ECC RAM not the amount of RAM installed on the board.

Table A-5

PORT A AND PORT B ADDRESS RANGE STRAP POSITIONS

Total Amount of ECC Memory	J111/J211 for first ECC RAM Board	J111/J211 for second ECC RAM Board
256K	2	N/A
512K	3	N/A
768K	N/A	6
896K	N/A	7

PORT B ADDRESS RANGE

Port B Address Range straps perform the same function on Port B as the Port A Address Range straps perform on Port A. Refer to Table A-5 for the strap setting positions.

APPENDIX A
STRAP INFORMATION

PORT A ADDRESS SPACE

The Port A Address Space straps are three-pin straps that logically extend the one megabyte address space of port A to 16 megabytes. Table A-6 lists the address space of each strap combination.

Table A-6

PORT A ADDRESS SPACE STRAP POSITIONS

Address Space	J121	J123	J125	J131
000000-0FFFFF	3	3	3	3
100000-1FFFFF	1	3	3	3
200000-2FFFFF	3	1	3	3
300000-3FFFFF	1	1	3	3
400000-4FFFFF	3	3	1	3
500000-5FFFFF	1	3	1	3
600000-6FFFFF	3	1	1	3
700000-7FFFFF	1	1	1	3
800000-8FFFFF	3	3	3	1
900000-9FFFFF	1	3	3	1
A00000-AFFFFF	3	1	3	1
B00000-BFFFFF	1	1	3	1
C00000-CFFFFF	3	3	1	1
D00000-DFFFFF	1	3	1	1
E00000-EFFFFF	3	1	1	1
F00000-FFFFFF	1	1	1	1

BOARD SIZE

The Board Size straps identify how much RAM is on the installed RAM Array Board(s). A single RAM Array Board contains 256K bytes of RAM and two RAM Array Boards contain 512K bytes of RAM. Table A-7 lists the possible strap positions for the two memory sizes.

Table A-7

BOARD SIZE STRAP POSITIONS

Memory Size	J141	J142	J145	J146
256K	3	1	1	3
512K	3	3	1	1

8207 CONFIGURATION

There are four straps in this functional group that initialize the 8207 RAM controller. Strap J318 is normally removed and establishes the refresh period. Strap J319 is normally installed and controls the initialization mode of the 8207. Cut strap W320 completes a circuit run and programs the 8207 for slow RAM cycles. Strap J321 is normally removed and establishes the port priority arbitration scheme. Table A-8 lists the normal positions of the 8207 configuration straps.

Table A-8

8207 CONFIGURATION STRAP POSITIONS

Strap Name	Position	
J318	Removed	
J319	Not Installed	(Do not initialize with zeros)
W320	SHORTED	
J321	Removed	

APPENDIX A
STRAP INFORMATION

CLOCK SOURCE

This straps selects the source of clock for the 8207 RAM controller. Clock source selection depends on the main processor servicing the ECC Memory Board. Table A-9 list the possible selections.

Table A-9

CLOCK SOURCE STRAP POSITION

Processor Board	**J341**
Future	1
8086/8087	3

STANDARD AND OPTION 10 3PPI BOARD STRAPPING

The standard and Option 10 3PPI boards use both jumper and cut straps. Strapping on this board sets the board I/O address and sets up the ROM circuitry to operate with a specific ROM type. Figure A-4 shows the location of the 3PPI board straps.

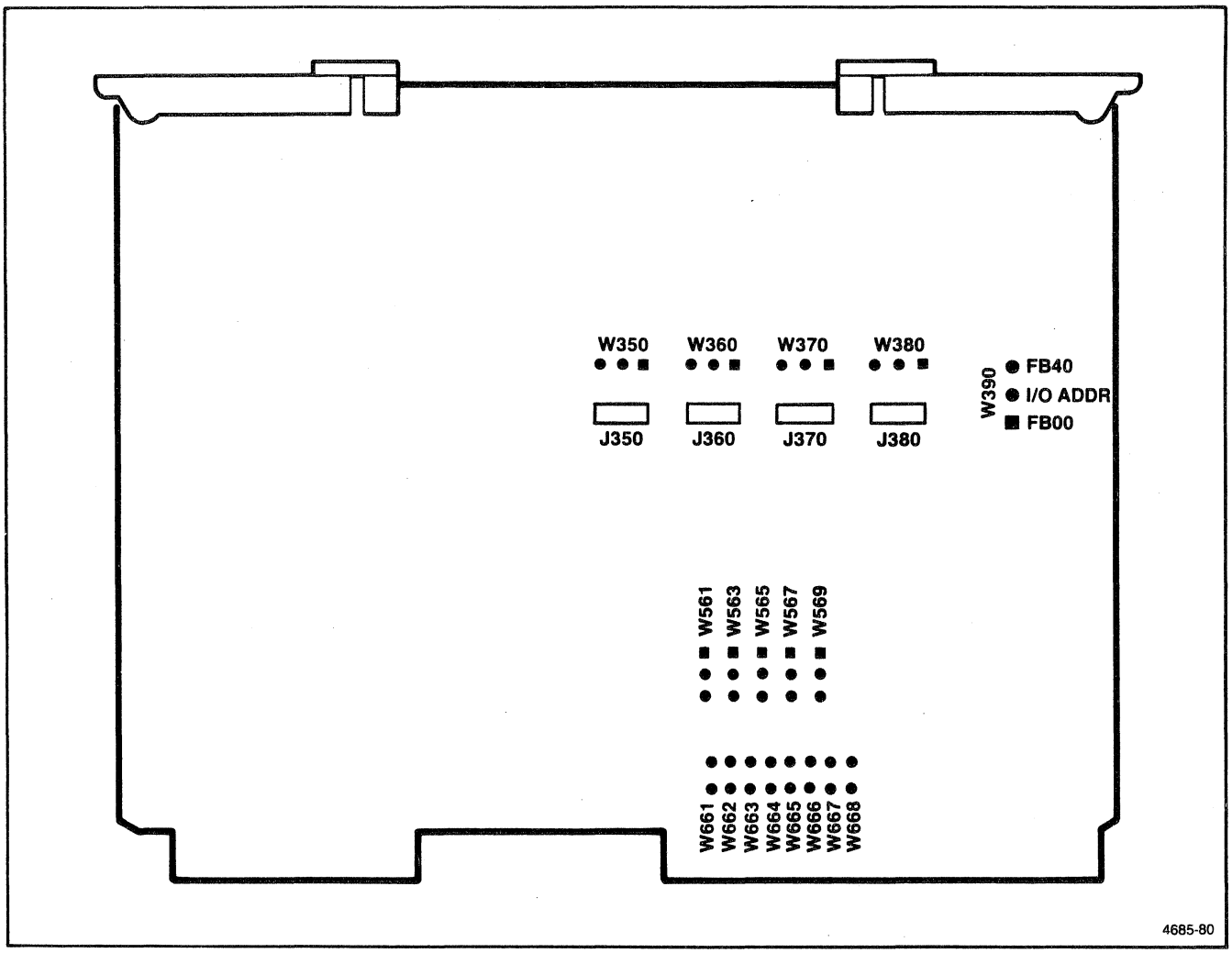


Figure A-4. 3PPI Strap Locations.

APPENDIX A
STRAP INFORMATION

I/O ADDRESS STRAP

Strap W390 selects the 3PPI board I/O address. Table A-10 describes its positions.

Table A-10

I/O ADDRESS STRAP POSITIONS

Strap	Function	Cut Strap Position
W390 (I/O ADDR)	Selects I/O address	Standard Board: FB00 Opt. 10 Board: FB40

ROM STRAPS

Both 3PPI boards have provisions for using on-board ROM; however, on-board ROM is not used in the 4170. The following is a list of strap associated with on-board ROM. The position of these straps is meaningless in the 4170.

Strap Number	Strap Function
J350,J360, J370,J380	ROM type select
W350,W360 W370,W380	ROM bank enable

OPTION 09 COLOR COPIER AND PRINTER INTERFACE BOARD STRAPPING

Straps on the Option 09 circuit board are associated with the on-board ROM which is not used in the 4170. The position of these straps does not effect 4170 operation.

OPTION 44 DISK CONTROLLER BOARD STRAPPING

The Option 44 Disk Controller board has various straps that allow increased flexibility in selecting operating parameters. The following paragraphs describe the straps associated with the Option 44 Disk Controller board. Figure A-5 shows the location of Option 44 Disk Controller board straps.

All references to up, down, right or left listed under "Cut Strap Position" indicates the direction of the normal strap connection as shown on the schematic diagrams. Optional operating modes require cutting the circuit run and jumpering across the opposite strap point.

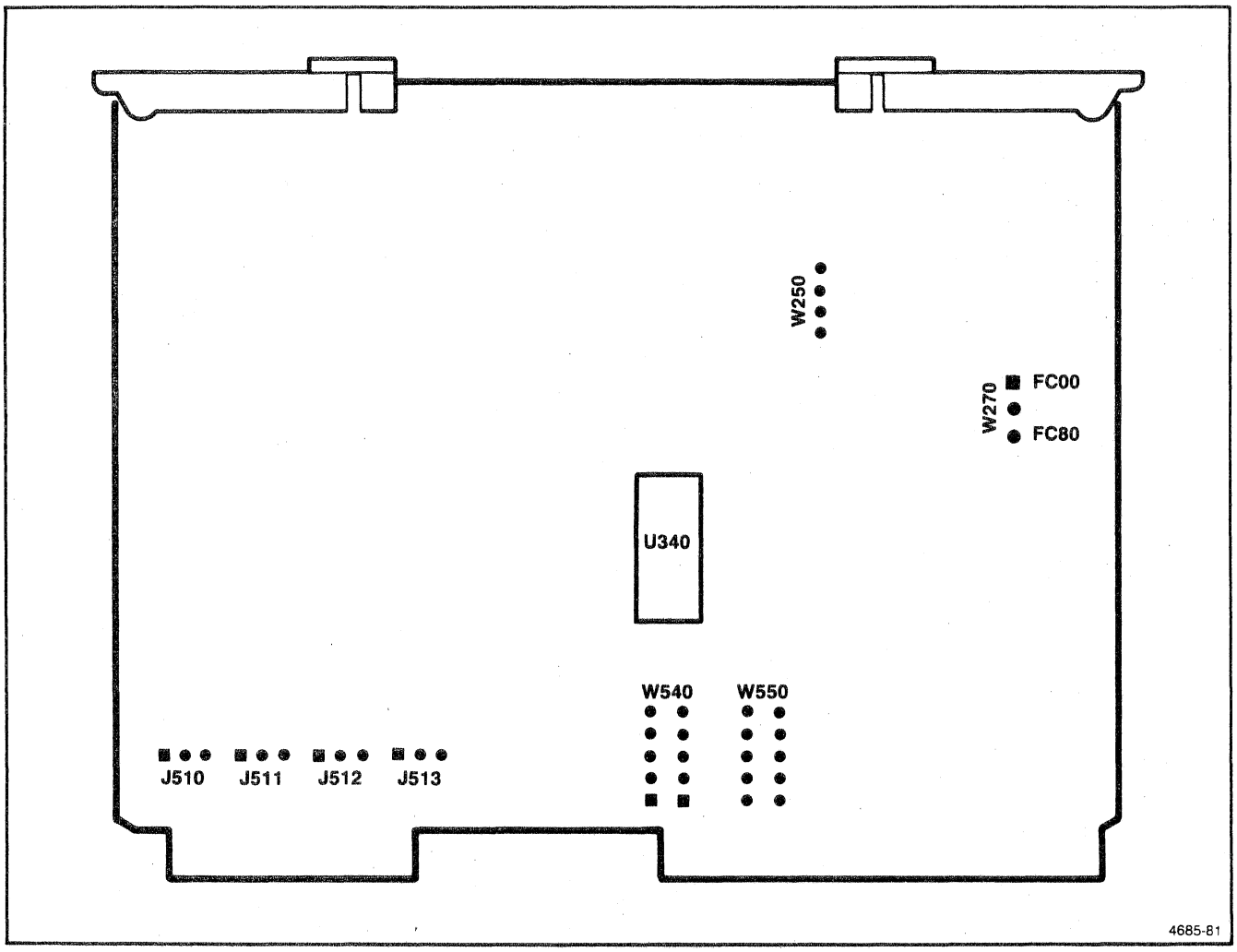


Figure A-5. Option 44 Disk Controller Board Strap Locations.

ROM STRAPS

The Option 44 Disk Controller board has mounting and circuitry for on-board ROMs. In the 4170 these ROM slots are not used and the position of straps associated with on-board ROM does not matter. The following is a list of the straps associated with the on-board ROM:

Strap Number	Strap Function
W525 through W529	Sets ROM base address
W510 through W513	ROM type select
J510 through J513	ROM type select
W550	ROM wait state select

INTERRUPT STRAPS

The Option 44 Disk Controller generates an interrupt after the execution phase of an flexible disk controller command. The interrupt level may be selected by strap W540. Settings of W540 are summarized in Table A-11. A cut strap selects the factory setting (Pin 1 selecting INT 7).

Table A-11

INTERRUPT LEVEL SELECT (W206) STRAP SETTINGS

W540 Pin Number	Interrupt Level Selected
1	INT7 (factory setting)
2	INT6
3	INT5
4	INT4
5	INT3
6	INT2
7	INT1
8	INT0

BASE ADDRESS STRAP

Strap W270 selects the base address for all the Option 44 Disk Controller board I/O registers as shown in Table A-12.

Table A-12

I/O BASE ADDRESS

Strap	Function	Cut Strap Position
W270	I/O base address (up=X'FC00', down=X'FC80')	Up

HEAD LOAD CONTROL STRAP

Strap W47 may be changed to allow the Disk Controller chip, instead of the main processor, to control the head load. Table A-13 describes its settings.

Table A-13

HEAD LOAD CONTROL

Strap	Function	Cut Strap Position
W47	Head Load (up=processor, down=FDC).	Up

WRITE-PROTECT STRAPS

Cut straps W45 and W46 can be changed to alter write-protect operation. Table A-14 describes W46 and W47 settings.

Table A-14

WRITE-PROTECT STRAP SETTINGS

Strap	Function	Cut Strap Position
W46	DRIVE A (in=installed, out=not installed)	In
W45	DRIVE B (in=installed, out=not installed)	In

WRITE PRECOMPENSATION STRAP

Strap W250 sets the precompensation time.

MASTER CLOCKS STRAP J225

Master clocks strap J225 connects the clock oscillator output to the frequency divider. This strap is a test/troubleshooting strap and is normally installed.

READ RECOVERY STRAP J10

Read recovery strap J10 completes a feedback path in the Read Recovery circuit. This strap is a test/troubleshooting strap and is normally installed.

OPTION 45 DISK CONTROLLER BOARD STRAPPING

The Option 45 Disk Controller board, like the Option 44 Disk Controller board has various straps on it that allow increased flexibility in selecting operating parameters. Table A-16 describes its straps. Figure A-6 illustrates the location of the straps.

For outboard 4926 operations the hard disk controler board in the 4926 must be strapped to a value other than 0. The inboard hard disk controller (when present) is strapped to 0 and is assigned to logical device E. When the outboard 4926 is strapped to 1, devices C and D are assigned. (If a 4926, Option 25 is attached, only two devices will be recognized by the 4170.)

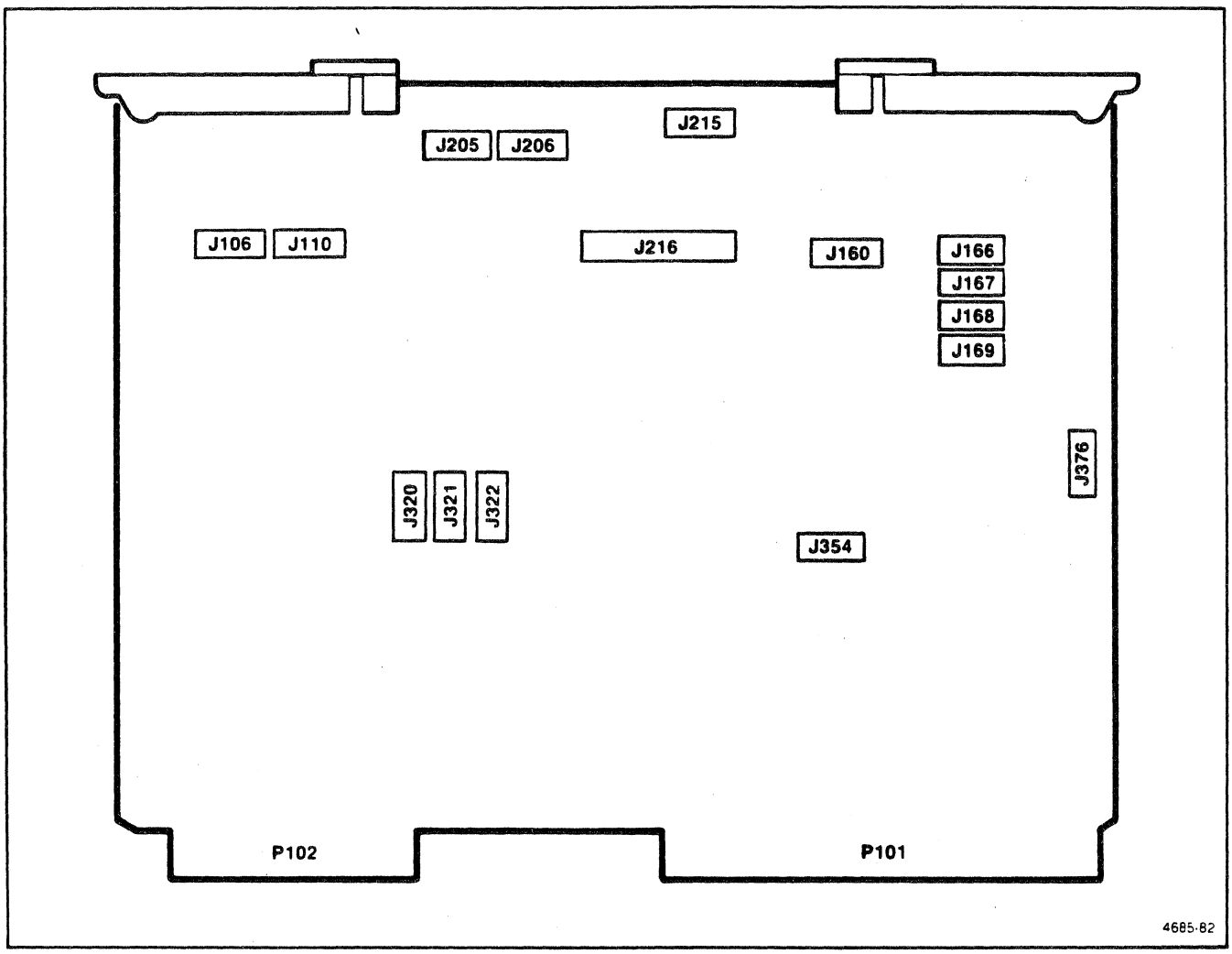


Figure A-6. Option 45 Disk Controller Board Strap Locations.

Table A-16

OPTION 45 DISK CONTROLLER STRAPS

JUMPER	FUNCTION
MSIB Device Address	A binary encoded number, 0 (0) through 111 (7) on J320, J321, and J322. If the jumper is on, the number = 0; if it is off, the number = 1. (0) 000 J320 on J321 on J322 on (1) 001 J320 on J321 on J322 off (2) 010 J320 on J321 off J322 on (3) 011 J320 on J321 off J322 off (4) 100 J320 off J321 on J322 on (5) 101 J320 off J321 on J322 off (6) 110 J320 off J321 off J322 on (7) 111 J320 off J321 off J322 off Factory default is (7) 111 (all jumpers off).
16 MHz Clock	J106. The clock is enabled when the jumper is installed. Factory default is enabled.
Drive Size	J160 - Install the jumper if the small (5 1/4") flexible disks are used (4170). Factory default is NOT installed on the 4110B Series terminals.
Drive Select	J166, J167, J168, J169. Install a jumper to match the drive selects specified on the internal disk drives attached to the board. J169 is DS1; J168 is DS2; J167 is DS3; J166 is DS4. Install J169 for a single drive system; install J168 and J169 for a two drive system.

Table A-16 (cont)

OPTION 45 DISK CONTROLLER STRAPS

JUMPER	FUNCTION
Precompensation	J110. Connect pin 2 to pin 3 and pin 5 to pin 6 for 8" flexible disks. Connect pin 1 to pin 2 and pin 4 to pin 5 for 5 1/4" flexible disks. The factory default is set for the flexible disks that are in the terminal the board is used in.
CPU Select	J354. Always removed.
High Bank	J376. If the board is in a 4110 Series terminal, remove the jumper to set memory bank space to 08000 -- OFFFF. If the board is in a 4170, install the jumper to set the memory bank space to E8000 -- EFFFF.
Settled Ready	J132. Always installed between Pins 1 and 2.
Parity	J306. If all MSIB devices use parity, install the jumper. (Either all devices must use parity, or none may use it.) The factory default is jumper not installed (no parity).
Write Protect (cut strap)	W23, W26. These cut straps are open for 4170 but not cut for the 4110 Series.