HP Vectra 486/33T Hardware Technical Reference Manual

An HP EISA Personal Computer Product

HP Part No. 5960-0748
Printed in USA March, 1991

Edition 1

HP Computer Museum
www.hpmuseum.net

For research and education purposes only.

Printing History

January 1991 Printed in USA

FCC Statement

Federal Communications Commission (FCC) Radio Frequency Interference Statement (USA Only)

Warning: This equipment has been certified to comply with the limits for a Class B computing device, pursuant to Subpart J of Part 15 of FCC Rules. Only peripherals (computer input/output devices, terminals, printers, etc.) certified to comply with the Class B limits may be attached to this computer. Operation with non-certified peripherals is likely to result in interference with radio and television reception.

More About Radio and Television Interference

Because the HP Vectra personal computer generates and uses radio frequency energy, it may cause interference with radio and television reception in a residential installation.

Hewlett-Packard's system certification tests were conducted with HP-supported peripheral devices and HP shielded cables, such as those you received with your system.

Warning: Cables used with this computer must be properly shielded to comply with the requirements of the FCC.

The Vectra personal computer meets the requirements for a Class B computing device in accordance with the specifications in Subpart J of Part 15 of FCC Rules. These rules are designed to provide reasonable protection against such interference in a residential installation.

Hewlett-Packard provides instructions for using this computer in manuals covering setup, connection of peripheral devices, operation, service, and technical reference.

Installing and using the computer in strict accordance with Hewlett-Packard's instructions will minimize the chances that your Vectra personal computer will cause radio or television interference. However, Hewlett-Packard does not guarantee that the computer will not interfere with radio and television reception.

If you think your computer is causing interference, turn it off and see if the radio or television reception improves. If the reception improves, your computer is causing the problem.

To correct interference, take one or more of the following interference remedies, as needed:

- Relocate the radio or television antenna.
- Move the computer away from the radio or television.
- Plug the computer into a different electrical outlet, so that the computer and the radio or television are on separate electrical circuits.
- Make sure that all your peripheral devices are also certified Class B by the FCC.
- Make sure you use only shielded cables to connect peripheral devices to your computer.
- Consult your computer dealer, Hewlett-Packard, or an experienced radio/television technician for other suggestions.

Electrical Safety

Warning

For your safety, the power cords supplied with this product have grounded plugs. The power cords should be used with properly grounded (3-hole) wall outlets to avoid electrical shock. You can also use multiple-outlet strips that have their own circuit breakers.

Contents

Figures

Tables

1

Introduction

This chapter explains for whom this manual is intended, gives an overview of the manual's content and organization, explains the conventions used, and gives references to other documentation for the Hewlett-Packard Vectra 486/33T, an Extended Industry-Standard Architecture (EISA) personal computer.

Manual Audience, Contents, and Organization

This technical reference manual is for

1. original equipment manufacturers, independent hardware vendors, and independent software vendors in their hardware design efforts, and
2. system managers and support personnel who need information on the HP Vectra 486/33T.

This manual covers only the HP Vectra 486/33T PC. For technical information on other Hewlett-Packard computers, refer to the appropriate reference manuals.

In this manual, each chapter lists references for more information on the topics covered in that chapter.

Conventions Used in this Manual

Hexadecimal Notation

Hexadecimal numbers are given in a series of up to eight digits, followed by a lower case letter h; for example: 0FFFFFFFh.

Signal Names

To indicate an active-low signal (using negative-true logic), the convention used here is to give the signal name, followed by an asterisk (*). For example, NOWS* indicates that the No Wait State signal is active low.

To indicate a range of bus signals, the convention used is to give the bus signal name, followed by angle brackets enclosing a range of numbers. For example, D<7:0> indicates the Data bus's low 8 bits.

To indicate a slot-specific signal, the convention used is to give the signal name, followed by a lower case "x". For example, MREQx* indicates an EISA Bus Master's Request for bus access, using a slot-specific (and active low) signal.

2

Product Description

Introduction

This chapter gives an overview of the Hewlett-Packard Vectra 486/33T personal computer. Included in this chapter are general descriptions of the system's physical and electrical characteristics, a summary of the system's configurations, and a listing of the system's specifications and dimensions.

For a more complete description of supported product configurations, refer to the *HP Vectra 486/33T Service Manual.*

Features

The HP Vectra 486/33T PC, a deskside Extended Industry-Standard Architecture (EISA) computer, has the standard features below.

- Microprocessor: Intel 80486
- Cache memory on 80486 chip: 8 Kbytes
- 128 Kbytes External Program and Data Cache
- System dynamic RAM (4 MB of DRAM standard, expandable to a maximum of 64 MB)
- Extended Industry-Standard Architecture (EISA)
- Address and data buses: 32-bit
- Backplane I/O connector slots for up to 8 extended industry-standard architecture (EISA) expansion boards (32-bit) or 8 industry-standard architecture (ISA) expansion boards (8 or 16-bit)
- Programmable interrupts for interrupt sharing by multiple input devices
- Direct-memory access (DMA) data transfer rate of up to 33 Mbytes/second
- Mass storage devices supported: up to 6 (4 flexible and 2 hard disk)
- Flexible disk controller for up to 4 flexible disk drives
- Flexible disk drive: 5.25-inch, 1.2 MB (Megabyte)
- Flexible disk drives (optional): 3.5-inch, 1.44-MB and 5.25-inch, 360 KB (Kilobyte)
- Hard Disk Drive Controller (HP D1664A): ESDI controller
- Hard disk drives (optional): 84 MB, 108 MB, 152 MB, 170 MB, 330 MB, 440 MB, 670 MB or 1,000 MB
- Tape drive (optional): 120 MB
- Keyboard: industry-standard, 101-key, mini-DIN connector
- Mouse: HP Mouse with mini-DIN interface
- One Parallel port (25-pin) and two serial ports (9-pin)
- Power supply: 264-watts (360 watts peak)

Product Description

The HP Vectra 486/33T PC is based on the Intel 80486 microprocessor and the 32-bit Extended Industry-Standard Architecture (EISA) I/O bus. The following is a block diagram of the components of the system.

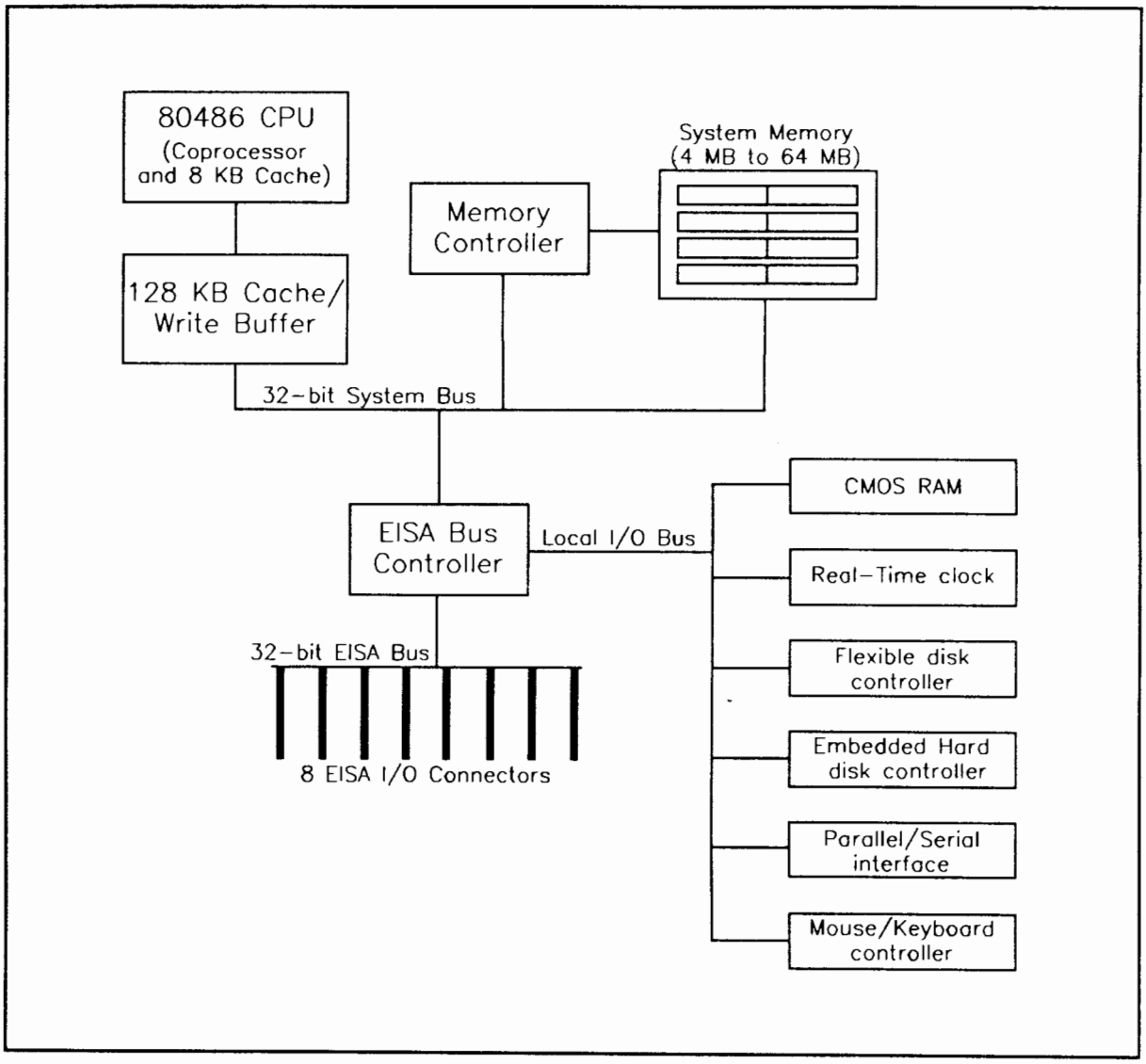


Figure 2-1. System Block Diagram

SPU Hardware: Printed Circuit Boards

System Board

The System Board (also called backplane, or mother board) provides the interconnection point for all other boards used in the System Processing Unit (SPU). Its I/O support circuitry provides eight 32-bit I/O connector slots for either ISA or EISA expansion boards.

Note

Expansion slots 3 and 5 do not support EISA expansion boards that contain bus masters.

Processor Board The Processor Board is mounted to the System Board with a proprietary 420-pin slot. The Processor Board contains the Intel 80486 microprocessor and support circuitry for all the system's buses, and support for all the system's subsystems. The Processor Board also contains system ROM.

Memory Board

The Memory Board is mounted to the System Board with a proprietary 360-pin slot (slot A, or 2nd from bottom slot as seen from the rear of the computer). The Memory Board provides the system with 4 MB of dynamic RAM (standard) in the form of memory modules and supports up to 32 MB if single-density memory modules are used, or 64 MB if double-density memory modules are used.

Mouse/Keyboard/Serial Port Board

The Mouse/Keyboard/Serial Port Board is mounted to the System Board via a proprietary 50-pin slot (slot B, or topmost slot as seen from the rear of the computer). The Mouse/Keyboard/Serial Port Board provides a mini-DIN socket for the keyboard, a mini-DIN socket for an optional mouse, and one industry standard serial port (9-pin).

There is a two-position Password Switch on the component side of the board. The password is set using the system's configuration utility, EASY CONFIG. In the event that a user has set up a password and then forgotten it, the switch can be set to the ON position to disable the password. This switch is shipped in the OFF (password enabled) position.

Parallel/Serial Port Board

The Parallel/Serial Port Board is mounted to the System Board via a proprietary 50-pin slot (slot C, second slot from top as seen from the rear of the computer). The Parallel/Serial Port Board provides an industry standard 9-pin serial port and 25-pin parallel port.

Power Supply

The power supply has 6 mass storage power connectors and a power connector to the System Board. It provides 264 watts of continuous power (360 watts peak).

Mass Storage

Six half-height mass storage shelves are available. Except for the Model 1-3 or 1-5, which do not come with a hard disk drive, the HP Vectra 486/33T comes standard with one flexible and one hard disk drive unit.

Flexible Disk Drive Controller

All HP Vectra 486/33T PC models come standard with a flexible disk drive controller which can support up to four flexible type disk drives (including an internal tape drive).

Flexible Disk Drives

All Vectra 486/33T PC models come standard with a 1.2 MB flexible disk drive (5.25-inch media) or a 1.44 MB flexible disk drive (3.5-inch media). All models can support up to four flexible disk drive units from the following:

- 360 KB flexible disk drive (5.25-inch)
- 1.2 MB flexible disk drive (5.25-inch)
- 1.44 MB flexible disk drive (3.5-inch)

Hard Disk Drive Controller

Except for Model 1-3 and 1-5, the HP Vectra 486/33T comes standard with the HP D1664A ESDI Hard Disk Controller Board. The controller board supports up to two ESDI hard disk drives, in the combinations shown below, for a maximum of up to 1000 MB of storage.

- two 152 MB hard disks
- one 152 MB and one 108 MB hard disk
- one 152 MB and one 330 MB hard disk
- one 152 MB and one 670 MB hard disk (*see note*)
- two 330 MB hard disks
- one 330 MB and one 152 MB hard disk
- one 330 MB and one 108 MB hard disk
- one 330 MB and one 670 MB hard disk (*see note*)
- two 670 MB hard disks (*see note*)

Note

The controller board has a drive splitting feature for drives above 528 MB. This allows operating systems that use the computer's ROM BIOS, such as MS-DOS® and OS/2®, to access the entire drive by splitting it into two logical drives (528 MB and 142 MB). With this feature enabled, you may not add a second hard disk drive. With this feature disabled, you may only access up to 528 MB of any single hard disk.

Since UNIX® does not use the computer's ROM BIOS, it does not have this limitation and therefore does not require the drive splitting feature.

Although the Model 1-3 or 1-5 does not have a hard disk controller board, it does have a connector that can support up to two hard disks with embedded controllers.

SCSI Hard Disk Drives

The HP Vectra 486/33T can also be equipped with an optional SCSI hard disk subsystem. This subsystem is currently available with three hard disk sizes, 440 MB, 670 MB, and 1,000 MB. Host Bus Adapters (controllers) are available in both EISA and ISA configurations, and can daisy-chain up to 8 drives from a single controller. Complete information on the SCSI subsystem is provided in the Accessories Technical Reference Manual.

Tape Backup Drives

For hard disk drive backup, all models support an internal 120 MB tape drive, with an average data transfer rate of 500 Kbit/second. The tape drive is attached via a cable attached to one of the flexible disk cable connectors on the System Board, and a power cable.

Display Adapters

The high resolution Video Graphics Display (monochrome or color) and HP Video Graphics Array Plus Board are optional.

Keyboard

The HP Vectra 486/33T supports the "HP Vectra Enhanced Keyboard," a detachable, 101-key, industry-standard keyboard with separate numeric and cursor control keypads and an industry-standard mini-DIN cable connector. Industry-standard international keyboards (with 102 keys) are also available.

Note

The HP Vectra 486/33T does not support the "Vectra PC/HIL Keyboard," nor any HP-HIL accessory.

System Configuration

The EASY CONFIG configuration program provided with each HP Vectra 486/33T helps simplify installation and basic configuration of user installable expansion boards. Through the use of configuration (CFG) files, supplied by EISA and ISA expansion board manufacturers, the computer is provided with the information necessary for conflict free operation. The configuration utility is used to customize System and expansion board settings for such features as memory remapping, serial and parallel port I/O address, display modes, hard disk address and type, and other specific operations without having to set switches.

For ISA expansion boards without configuration files, features may be set via hardware switches on the expansion board. Configuration files may be created as explained in the *Dealer Configuration File Creation Guide* included with each HP Vectra 486/33T PC.

Compatibility

In general, the EISA system is fully compatible with ISA systems, expansion boards, and software. The EISA processor board automatically translates for 16-bit ISA bus masters, 8-bit or 16-bit memory and I/O slaves, and 8-bit or 16-bit DMA devices. The EISA bus system allows EISA expansion boards to communicate with devices that are ISA-compatible, and allows ISA 8-bit and 16-bit expansion boards to be installed in EISA slots.

The HP Vectra 486/33T is also compatible with most existing accessories and applications supported by the HP Vectra RS (except HP-HIL accessories).

HP Vectra 486/33T Models

All models of the HP Vectra 486/33T contain the following standard features:

- 33-MHz 80486 SPU
- 4 MBytes DRAM
- A parallel port
- 2 serial ports
- A mini-DIN Mouse port
- A mini-DIN Keyboard and port
- A flexible disk drive controller (supports up to 4 drives)
- A 1.25-inch, 1.2 MB flexible disk drive
- EISA configuration utility (named EASY CONFIG)

Specifications and Dimensions

This section lists the HP Vectra 486/33T's system and environmental specifications, and its dimensions. (For the system's power supply specifications, see the "Power Supply" chapter.)

HP Vectra 486/33T System Specifications

Regulatory Compliance

1. Datacomm

Belgium (RTT) *
Germany (FTZ) *
Sweden and the Nordic Network *
United Kingdom *

Ergonomics

Complies with German Standard ZH1/618

3. Radio Frequency Interference

Complies with FCC Class B (U.S.A.)
Manufacturer Declaration FTZ 1046/84 Level B (Germany)
VCCI approval, Class 2 (Japan)

4. Safety Approvals

CSA certified (Canada)
IEC 65/380/435/950 (International) compliance
NEMKO (Norway)
SASO (Saudi Arabia) *
TUV certified (Germany)
UL listed (United States)

(*) pending

HP Vectra 486/33T Environmental Specifications

Temperature Specifications

Operating temperature: +5° to +40° C (+41° to +104° F)
Non-Operating temperature: -40° C to +70° C (-40° F to 158° F)

Humidity Specifications

Operating humidity: 15% to 80% relative humidity over operating temperature range.

Altitude Specifications*

Operating altitude: 0 to 4,600 meters (0 to 15,000 feet)
Non-Operating altitude: 0 to 15,250 meters (0 to 50,000 feet)

* If an ESDI Hard Disk Controller is installed, the altitude specification changes to the following: Operating altitude: 0 to 3050 meters (0 to 10,000 feet)
Operating temperature: +5° to 40° C (32 to 104 F)

HP Vectra 486/33T Dimensions

System Processing Unit Dimensions:

Height:	60 centimeters (24 inches)
Width:	21 centimeters (8.3 inches)
Base Width:	35.5 centimeters (14 inches)
Depth:	51.5 centimeters (20.31 inches)
Weight:	27 kilograms (60 pounds) *

* Weight includes floor-mount base, but excludes monitor and keyboard.

Keyboard Dimensions

Height:	3.4 centimeters (1.4 inches)
Width:	46.8 centimeters (18.4 inches)
Depth:	19.8 centimeters (7.8 inches)
Weight:	1.9 kilograms (4.2 pounds)

Cable Lengths

Keyboard mini-DIN cable:	3 meters (9.9 feet)
Mouse mini-DIN cable:	2.5 meters (8.25 feet)

3

EISA System and Bus Overview

Introduction

This chapter gives a brief overview of the Extended Industry Standard Architecture (EISA), including the EISA bus, memory and interrupt operations, EISA expansion boards, software, and EISA bus masters. Where appropriate, the differences between the EISA and the Industry Standard Architecture (ISA) are illustrated.

The HP Vectra 486/33T EISA personal computer conforms to the design specifications detailed in the *EISA Specification* document published by:

BCPR Services, Inc.
A Delaware Corporation
1400 L Street, N.W.
Washington, D.C. 20005-3502

For more specific and detailed information on design, operation, and timing of the components of the EISA system, refer to the *EISA Specification*.

EISA Features

The Extended Industry-Standard Architecture (EISA) uses 32-bit architecture to extend the existing industry-standard 8-bit and 16-bit architecture, while maintaining full compatibility with it. The following are some of the features introduced with the EISA architecture:

- Synchronous data transfer protocol allowing for high-speed burst transfers as well as normal single transfers
- 8-, 16- or 32-bit data transfers for the CPU, bus masters, and direct-memory access (DMA) devices
- 33-Mbyte/second maximum (burst) data transfer rate for bus masters and DMA devices
- Interrupts are programmable for either edge or level triggering
- Support for intelligent bus master peripheral controllers
- 32-bit memory addresses for CPU, bus masters, and DMA devices
- Bus cycles automatically translated between EISA and ISA masters and slaves
- System and expansion boards automatically configured

EISA Bus Overview

The EISA bus extends the ISA (Industry Standard Architecture) bus to include support for 32-bit address paths and data transfers, in addition to the 8-bit or 16-bit address paths and data transfers that the ISA bus supports. The EISA bus is fully compatible with ISA software, and 8-bit and 16-bit ISA expansion boards can be installed in EISA I/O connector slots.

In contrast to the ISA bus, the EISA bus can transfer data faster, is more flexible, and supports multiple bus masters and multiple microprocessors. The EISA bus gets peak performance from both the bus masters and the microprocessors by synchronizing their bus cycles to BCLK, a common clock signal which synchronizes events on the EISA bus with the system's 33 MHz clock. This same synchronization allows the EISA bus to create a synchronous data transfer protocol for executing burst cycles (a type of zero-wait-state data transfer). The burst cycles provide a continuous sequence of read or write cycles between a bus master or DMA device, and system or EISA (but not ISA) memory, and they transfer up to 33 Mbytes of data per second.

As discussed in the *EISA Specification*, the EISA bus system allows communication between EISA boards and ISA compatible devices by generating EISA data and control signals. The control lines used in ISA were kept, and additional control lines have been implemented. This allows the system to copy data to the appropriate byte lanes and to translate the control signals as necessary.

The system handles the automatic translations for

- 16-bit ISA bus master transactions
- 8- or 16-bit memory and I/O slave transactions
- DMA device transactions
- transactions between 16- and 32-bit EISA devices

In addition, the following ISA-compatible buses have been implemented:

- LA2:LA31, the latchable address bus, used in conjunction with byte enables BE0:3, extended from ISA's LA17:LA23
- SA0:SA19, the system address bus, carried over from ISA
- SD0:SD31, the system data bus, also carried over from the ISA's SD0:SD15, but expanded to SD0:SD31

EISA Bus Interface Circuitry

The EISA bus interface circuitry is composed of these components:

- EISA bus
- Intel 82357 Integrated System Peripheral (ISP)
- Intel 82358 EISA bus controller (EBC)
- Host bus (the 80486 microprocessor bus)

The EISA bus and the host (CPU) bus interface through the 82358 EISA Bus Controller (EBC). The EBC also interfaces with the 82357 Integrated System Peripheral (ISP) in order to run the appropriate cycles requested by the ISP (DMA and REFRESH).

In addition to providing an integrated synchronous system reset, the EBC performs the following operations:

- translation of:
  - host bus cycles to EISA/ISA bus cycles
  - EISA/ISA master bus cycles to host bus cycles
- generation of:
  - EISA signals for ISA masters
  - ISA signals for EISA masters
- support of:
  - 8-, 16-, or 32-bit data transfers between EISA, ISA, and host buses
  - 8-, 16-, or 32-bit DMA transfers between buses
  - host and EISA/ISA refresh cycles

EISA Bus Signal Descriptions

The following figure gives the pinout for the EISA bus I/O connector. In the figure, the letters A (indicating pin 1) through D represent ISA pins, the letters E through H represent EISA pins. Table 3-1 lists the bus signals on the ISA part of the EISA bus and describes them. The EISA bus accepts these ISA signals and has 55 additional pins assigned to EISA-specific signals, described in Table 3-2.

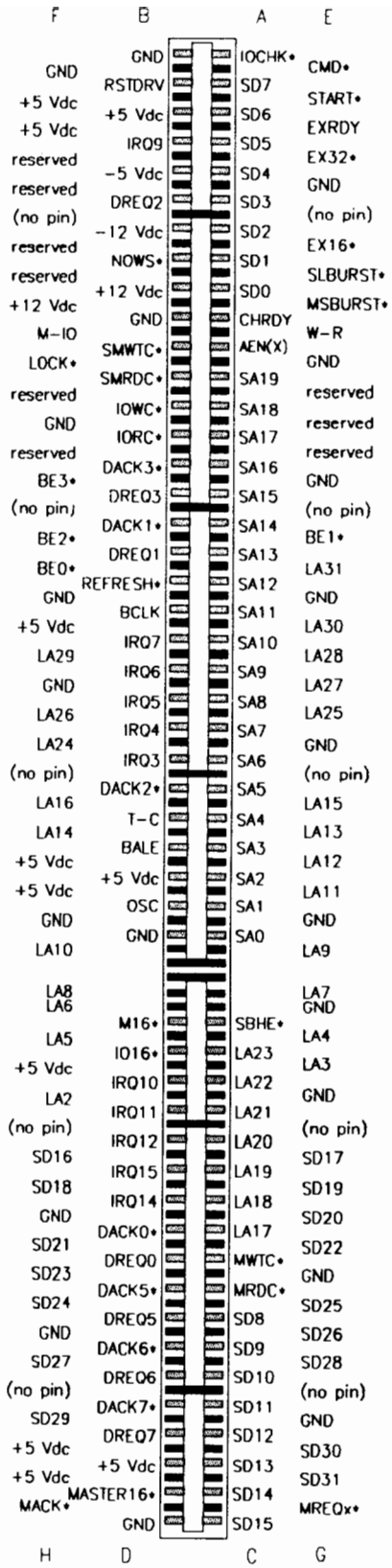


Figure 3-1. EISA Bus I/O Connector

The following table lists and describes the ISA signals of EISA bus. Note that an asterisk (*) indicates an active low signal.

Table 3-1. Signal Descriptions for ISA Part of EISA Bus

Bus Signal	Bus Signal Description
AEN(x)	**Address Enable** Allows I/O slaves to accept addresses and commands. The "x" refers to the slot number (1 = I/O slot 1; 2 = I/O slot 2 ... etc.).
BALE	**Bus Address Latch Enable** Indicates a valid address is present on LA <31:2>. (If LA <31:2> must be latched, it must be latched on the trailing edge of BALE.)
BCLK	**Bus Clock** Synchronizes events on the EISA bus.
CHRDY	**Channel Ready** May be negated by ISA memory or an I/O slave to get additional wait states.
DACK <7:5>* DACK <3:0>*	**DMA Acknowledge** Indicate that a particular direct-memory access (DMA) channel has been granted the bus.
DREQ <7:5> DREQ <3:0>	**DMA Request** Used to request direct-memory access (DMA) service from a DMA subsystem, or for a 16-bit ISA bus master to request access to the system bus.
IO16*	**Input/Output 16-bit** Asserted by 16-bit I/O slaves to indicate a 16-bit data size.
IOCHK*	**Input/Output Check** Asserted by an EISA expansion board or an ISA expansion board when a serious error occurs.
IORC*	**Input/Output Read Command** When simultaneously asserted with the AEN(x) signal, an ISA I/O slave latches data from the data bus.
IOWC*	**Input/Output Write Command** When simultaneously asserted with the AEN(x) signal, an ISA I/O slave latches data from the data bus.
IRQ <15:14> IRQ <12:9> IRQ <7:3>	**Interrupt Request** Used to interrupt the 80486 microprocessor to request service through prioritized interrupt lines.
LA <23:17>	**Latchable Address** LA17:LA23 are part of the 32-bit latchable address bus.
M16*	**Memory 16 (16-bit)** Asserted by 16-bit ISA memory slaves to indicate 16-bit memory.
MASTER16*	**Master 16-bit** Asserted by a 16-bit bus master to indicate a 16-bit data size. MASTER16* can occur only if a 16-bit bus master has control of the bus.
MRDC*	**Memory Read Command** A read command to an ISA memory slave to drive data onto the data bus.
MWTC*	**Memory Write Command** A write command to an ISA memory slave to latch data from the data bus.

Table 3-1. Signal Descriptions for ISA Part of EISA Bus (continued)

Bus Signal	Bus Signal Description
NOWS*	**No Wait State** Asserted by an ISA memory slave after the slave's address and command has been decoded. Indicates that additional clock signals are not required.
OSC	**Oscillator** A clock signal of 14.31818 MHz, used for timing applications.
REFRESH*	**Refresh** Indicates that the current bus cycle is to be used for the refresh of dynamic random access memory.
RSTDRV	**Reset Drive** Causes a hardware reset of ISA/EISA devices. This will occur during a power-on reset, bus time-out, or by software command.
SA <0:19>	**System Address** Lower 20 bits of 32-bit system address that is available on the EISA bus.
SBHE*	**System Bus High Enable** Indicates that EISA expansion boards are enabled to transfer data on the high half of the system data SD <0:15> bus.
SD <0:15>	**System Data** 16 data lines on the ISA system data bus.
SMRDC*	**System Memory Read Command** A read command to an ISA memory slave to drive data onto the data bus.
SMWTC*	**System Memory Write Command** A write command to an ISA memory slave to latch data from the data bus.
T-C	**Terminal Count** The Terminal Count signal has two programmable modes. 1. In the input mode, a DMA slave can stop a DMA transfer. 2. In the (default) output mode, a DMA controller asserts T-C when a word count reaches terminal count.

* Indicates active low signal.

The following table lists and describes the EISA signals of the EISA bus. Note that an asterisk (*) indicates an active low signal.

Table 3-2. Signal Descriptions for EISA Part of EISA Bus

Bus Signal	Bus Signal Description
BE <3:0>*	**Byte Enable** Specify the bytes within an addressed double-word.
CMD*	**Command** Asserted when the START* signal is negated, and CMD* continues until the end of the cycle.
EX16*	**EISA 16-bit Device** Asserted by an EISA memory or I/O slave to indicate that the memory or I/O slave supports 16-bit data transfers.
EX32*	**EISA 32-bit Device** Asserted by an EISA memory or I/O slave to indicate that the memory or I/O slave supports 32-bit data transfers.
EXRDY	**EISA Ready** Negated by an EISA memory or I/O slave in order to request that wait states be added.
LA <16:2> LA <31:24>	**Latchable Address** Part of the latchable address bus.
LOCK*	**Lock** When asserted by the 80486 microprocessor or the bus master, it locks the bus, preventing interrupts and granting exclusive use of the bus to either the 80486 or the bus master.
M-IO	**Memory - Input/Output** Asserted by either the 80486 microprocessor or the EISA bus master and indicates either a memory or an I/O access.
MACKx*	**Master Acknowledge** Slot-specific signal asserted by the system board to grant bus access to an EISA bus master. The "x" refers to slot (refer to schematics in Appendix A).
MREQx*	**Master Request** Slot-specific signal used by EISA bus masters to request bus access. The "x" refers to the slot number.
MSBURST*	**Master Burst** Asserted by either the 80486 microprocessor or the bus master to indicate to a slave that the master can provide burst cycles.
RSVD <1:9>	**Reserved** Reserved. (No connections.)
SD <31:16>	**System Data** Most significant 16 data lines on the EISA system data bus.
SLBURST*	**Slave Burst** Asserted by a slave to indicate that it can support burst cycles.
START*	**Start** Provides timing control at the start of a cycle. START* is asserted after LA <31:2> and M-IO become valid, and START* is negated one BCLK later.
W-R	**Write - Read** Identifies a cycle as a write cycle or a read cycle. The W-R signal becomes valid after the START* signal is asserted and before the CMD* signal is asserted.

EISA/ISA Devices

The EISA bus supports both EISA and ISA devices and can readily distinguish between master and slave devices. ISA devices include both memory and I/O which provide data transfer cycle types of both 8- and 16-bits. Most cycles can be shortened via a memory or I/O slave by asserting NOWS*, or extended by negating CHRDY. If NOWS* is negated and CHRDY is asserted, ISA devices generate a default 6 BCLK (for 8-bit devices), or 3 BCLK (for 16-bit devices) memory or I/O cycle.

EISA devices include

- Memory slaves
- I/O slaves
- Bus masters
- Burst bus masters
- Downshift burst bus masters
- DMA devices
- The EISA system board

Memory Slaves

Memory slaves use either the ISA control signals (ISA memory slaves) or the EISA control signals (EISA memory slaves) to interface to the bus. While not initiating bus cycles themselves, these slaves respond to control signals presented by the CPU or a bus master. EISA memory slaves support either 16- or 32-bit data transfers using standard memory cycles (refer to the *EISA Specification* for details). Memory slaves that also support 16- or 32-bit burst cycles assert SLBURST* to indicate when ready for a burst cycle, and sample MSBURST to determine if the master or system will run a burst cycle. Memory slaves may also use compressed cycles to improve data transfer rates with the system. Compressed cycles cannot be executed by bus masters.

I/O Slaves

As with memory slaves, I/O slaves interface to the bus via either ISA control signals (ISA I/O slaves) or extended EISA control signals (EISA I/O slaves). These slaves only respond to control signals presented by the CPU or a bus master. EISA I/O slaves support standard (refer to the *EISA Specification*), burst, and compressed I/O cycles. Compressed cycles cannot be executed by bus masters. I/O slaves can extend the cycle timing by negating then asserting EXRDY on BCLK edges.

EISA devices (such as bus masters) can be designed to respond as an 8-bit as well as a 16- or 32-bit I/O slave.

Bus Masters

Bus masters (16- or EISA 32-bit) perform bus operations independently of the CPU using the ISA control signals (ISA 16-bit bus masters) or the EISA control signals (EISA 16- or 32-bit bus masters). Bus masters are able to request control of the bus. Once in control of the bus, bus masters can perform all I/O and memory cycles (except compressed cycles), accessing both the system and other EISA and ISA devices on the bus.

The EISA bus architecture provides a high-speed data channel (using EISA burst cycles, up to 33 Megabytes/second maximum) for intelligent peripherals. Such peripherals (such as disk controllers, LAN cards, and video graphics controllers) utilize a bus master that provides

"intelligence" via a dedicated local I/O processor and memory which can perform memory access and data transfers tasks that the host CPU would, otherwise, have to perform.

This bus architecture uses a rotational arbitration scheme that maximizes bus sharing among multiple EISA bus masters, DRAM refresh, DMA devices and CPU functions. This arbitration scheme can be used to pre-empt an active bus master or DMA device. It can also reset the system if a device does not release the bus upon pre-emption.

EISA bus masters are synchronous with BCLK, and drive LA <31:2>, BE <3:0>*, M-IO, W-R, SD <31:0>, START*, MREQx*, and MSBURST* from BCLK edges. A 32-bit master monitors EX32*, EXRDY, and MACKx* (a 16-bit master monitors EX16 and EX32 and treats them the same).

Bus arbitration for EISA/ISA bus masters, the microprocessor, DMA channels, and refresh is handled by Intel's 82357 ISP (Integrated System Peripheral) controller chip. In addition, the ISP incorporates:

- a high performance 7-channel programmable DMA controller,
- two 8-channel interrupt controllers,
- five programmable 16-bit counter/timers,
- logic for generation and control of non-maskable interrupts,

which, when used in conjunction with the Intel 82358 EBC (explained earlier in this chapter), provide system functions for applications that are EISA specific.

Burst and Downshift Burst Bus Masters

An EISA 16- or 32-bit bus master may execute burst cycles which allow high speed (up to 33 megabyte/second) read or write cycles across the bus to the system, or other masters or slaves which support burst cycles. A burst cannot be a mixture of read and write cycles. Burst transfers must be split up into two or more separate transfers if the transfer crosses the 1024-byte page boundary.

An EISA 32-bit bus master which can shift to a 16-bit burst master is called a downshift burst master. A downshift burst master must drive MASTER16 on each START* for the system to tell the difference between it and a 32- or 16-bit master.

DMA (Burst and Non-Burst) Devices

DMA devices are typically located on an expansion board plugged into the bus. DMA devices initiate DMA transfers under the control of the system DMA controller. The DMA controller, using several signals to determine the type and status of the device transfer, and provides address and control signals to the device. The EISA system provides 32-bit address support for both ISA and EISA DMA devices. EISA system DMA channels can be programmed for 8-, 16-, or 32-bit EISA/ISA DMA data transfers using non-burst cycles. In addition, EISA DMA devices can be programmed for 32-bit 33 MB/sec data transfers using burst cycles.

In non-burst mode transfers from memory, the system asserts a DMA channel's DACK<x>*. It asserts IOWC* and holds it until it until the system is presented with valid DMA read (I/O write) cycle data. Memory slave requirements for wait states are filled while IOWC* is asserted. The DMA device may not add wait states of its own, but must conform to the system's DMA controller. For a memory write (I/O read), a DMA device's DACK<x>* and IORC* signal go active until the data is latched by EISA memory, or the system (if 8- or 16-bit disassembly is required by the memory).

Normal demand and block modes do not release the bus between cycles. DREQ<x> line is monitored to determine when to release the bus and stop cycling for demand mode. The entire block is transferred from one DREQ<x> for block mode.

In a burst mode transfer, only demand or block mode is legal. The system performs cycle translation of a burst DMA transfer to memory that does not support burst.

Refer to the *EISA Specification* for more specific detail on DMA Device transfer modes.

The System Interface

The architecture of the EISA system provides for the following:

- Main memory access
  Memory (or I/O) access need not reflect on the EISA bus.
- Back-to-back I/O delay
  Back-to-back ISA I/O accesses caused by CPU cycles are automatically delayed a minimum of 2 1/2 BCLK (300 nsec).
- Slot specific I/O
  Reserved I/O spaces at 0z000h-0z0FFh, 0z400h-0z4FFh, 0z800h-0zFFh, and 0zC00h-0zCFFh for slot specific I/O slaves on EISA and ISA accessory boards.
- I/O addressing
  Reserved I/O addresses 0400h-04FFh for EISA system board peripherals (present and future). I/O addresses 0800-08FFh and 0C00-0CFFh may be used for manufacturer specific system board peripherals.

Bus Arbitration

EISA's centralized bus arbitration scheme allows bus sharing among the CPU, refresh controller, DMA controller and bus masters. A central arbitration controller (on the ISP chip) takes requests from devices asserting bus requests, and when the bus becomes available the system asserts a bus grant signal and the device gains control of the bus.

Other devices may pre-empt an active device in control of the bus by asserting MREQx* (used by 16- or 32-bit EISA masters), DREQ<x> (used by 8-, 16-, or 32-bit DMA slaves, or 16-bit ISA masters), or a refresh request. The system indicates to the active device to release the bus, however, the active device is allowed to hold the bus for a maximum of 8 microseconds before release to the system. When the bus is released, arbitration between contending devices is performed and the system asserts the appropriate bus grant to the winning device.

A multilevel rotating priority is used by the system. Requesting devices are granted bus access based on their position in the rotation.

The top (three-way) priority scheme for devices which can use the bus is as follows:

1. DMA (to remain compatible with ISA devices that require short bus latency)
2. Refresh controller
3. device (CPU or bus master)

These devices gain control of the bus, at worst, once every three arbitration cycles. Devices not requesting use of the bus are skipped. DMA priorities can be modified to either a fixed (default) or rotating priority by programming the controller. The following figure illustrates the process.

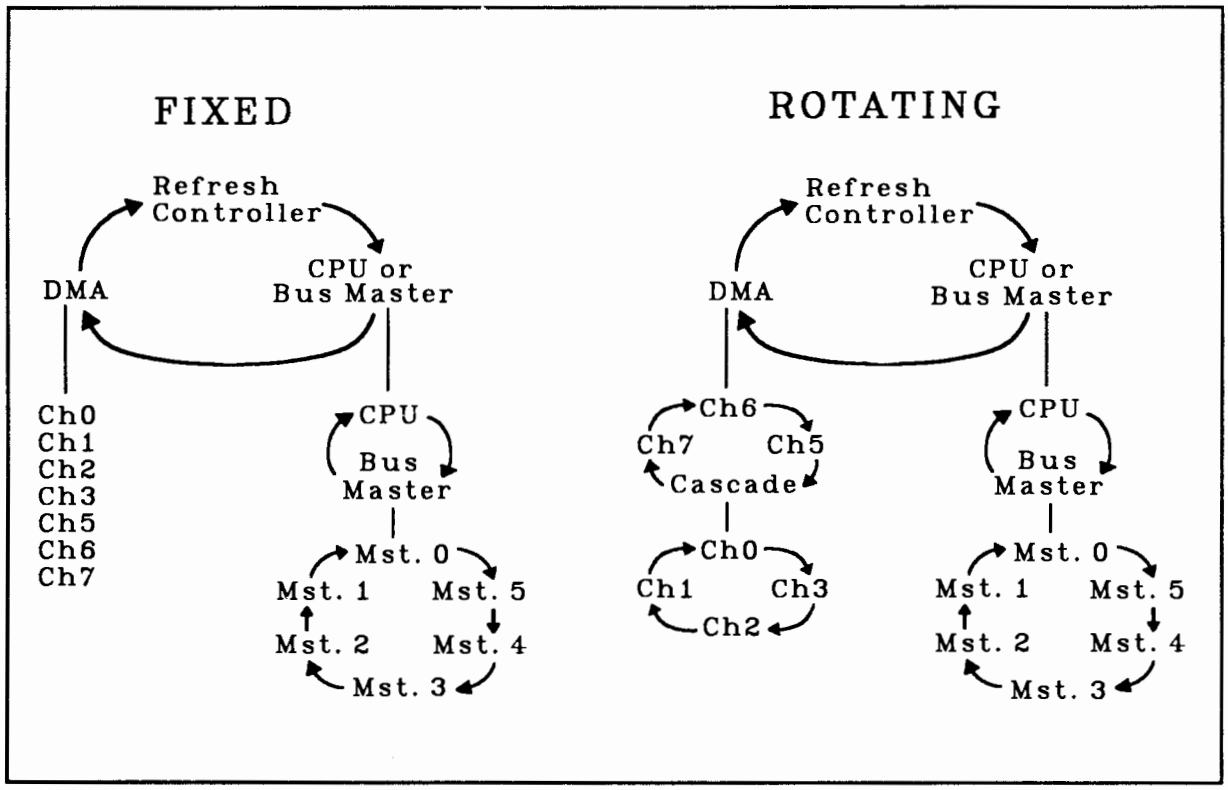


Figure 3-2. DMA Priority Arbitration Sequences

Refer to the *EISA Specification* for more details about device bus grant priorities and latencies.

DMA Operations

In an ISA or EISA direct-memory access (DMA) operation, an I/O device interfaces with a DMA controller to bypass the central processing unit and directly access the computer's memory for data transfers with it. During ISA and EISA direct-memory access transfers, the system board asserts a DMA acknowledge signal (DACKx*) and the appropriate I/O command signal (an I/O read or an I/O write). For a DMA write operation, the system board negates the I/O read command signal when the system board latches the data. For a DMA read operation, the system board negates the I/O write command signal when the data lines are valid. However, the I/O command signal remains asserted during memory wait states or data translation.

EISA systems are fully compatible with ISA DMA operations, while also offering the DMA enhancements shown in the following table.

Table 3-3. Contrast of ISA and EISA DMA Operations

Type of DMA Operation	ISA DMA Operation	EISA DMA Operation (EISA DMA Enhancements Italicized)
Address support for DMA transfers	24-bit address support, all of which can be used for DMA (16 MB)	32-bit address support, all of which can be used for DMA (4 GB)
Type of data transfer	8-bit and 16-bit	8-, 16-, and 32-bit
Bus arbitration	8-bit or 16-bit	8-, 16-, or 32-bit
DMA transfer rate	Up to 4 Mbytes/second	Up to 33 Mbytes/second

DMA Controller

In the EISA system, direct-memory access is provided by the ISP (Integrated Systems Peripheral) chip—part of the Intel EISA chip set. DMA circuitry on the chip provides the functionality of two 8237 DMA controllers plus the EISA enhancements (previous table).

The EISA DMA controller consists of seven independently programmable and ISA-compatible DMA channels, channels 0 to 3 and channels 5 to 7. The registers of channel 4 of the EISA DMA controller are accessible, but this channel is dedicated to cascade mode to maintain ISA compatibility and should **not** be reprogrammed.

The EISA DMA controller operates in either the Master Condition during DMA and refresh or in the Slave Condition (when its registers are being addressed). The DMA controller is in the Master Condition when (1) it generates cycle control for DMA data transfers, or when (2) a 16-bit ISA master uses a DMA channel for bus requests, (3) during refresh. The DMA controller is in the Slave Condition when it monitors the bus and decodes read or write I/O cycles that the main CPU and bus masters use for programming or reading the DMA registers. (Only when it is in the Slave Condition will the DMA controller accept read or write accesses to its registers.)

The EISA DMA controller's memory addressing circuitry supports the full 32-bit addresses for DMA devices, and all DMA channels support an extended addressing mode. In this mode, the EISA address registers count sequentially like a 32-bit up/down counter. As a result, DMA devices can sequentially address a 32-bit address range without programmatically incrementing the address extension registers each time a DMA address crosses a 64-Kbyte boundary (as in the industry-standard architecture).

When a DMA transfer takes place, the DMA device transfers data directly to the memory slave unless data size translations are required. DMA read and write operations to memory use the normal memory interface.

The EISA DMA controller includes a set of Stop Registers for a ring buffer data structure which reserves a fixed portion of memory on doubleword boundaries for use in a DMA channel. The Stop Registers prevent the DMA from over-writing data not yet read by the central processing unit.

DMA Controller Registers

The following table lists and describes the DMA controller registers. Information on programming the individual channels follows. (For more information on the registers and for complete bit-by-bit programming information, see the *EISA Specification* and the *Intel 82357 Integrated System Peripheral Data Sheet*.)

Table 3-4. EISA DMA Controller Register Descriptions

Register Name	Register Description
Base Page, Base Address, and Base Word Count	These write only registers used to store the original values of a channel's associate Current Address Register. Used in autoinitialization.
Chain Buffer Expiration Control	A read only register that reflects the expiration of a chain buffer.
Channel Interrupt Status	A read only register that indicates a pending interrupt (IRQ13) caused by the DMA controller.
Command	A write only register that 1. Initializes DMA request and DMA acknowledge logic levels to active high or low. 2. Initializes channel group arbitration priority scheme to "fixed" or "rotating."
Current Address	A read/writer register that stores intermediate value of DMA channel's current address during DMA transfers.
Current Word Count	A read/write register that stores intermediate value of DMA channel's word count during DMA transfers.
DMA Memory Low Page	A read/write register that contains the eight second most significant bits of the 32-bit address.
DMA Memory High Page	A read/write register that contains the eight most significant bits of the 32-bit address.
Extended Mode	A write only register used to program the DMA device data size and timing mode.
Mask	Disables a DMA service request.
Mode	A write only register that 1. For each DMA channel, four modes can be programmed: Single—performs one DMA transfer per arbitration cycle Block—performs a block DMA transfer per arbitration cycle Demand—performs a group of DMA transfers for each arbitration cycle Cascade—cascades two DMA controllers together. 2. Bit 4 can be programmed for auto-initialization, which automatically loads the Current Page, Current Address, and Current Word Count Registers from the Base Page, Base Address, and Base Word count registers each time the DMA controller reaches terminal count or an external End-of-Page is received.
Request	A write only register that initiates a DMA service request.
Set Chaining Mode	A write only register that enables or disables DMA buffer chaining, which transfers data from an I/O device to several different memory areas, within one transfer operation.

Table 3-4. EISA DMA Controller Register Descriptions (continued)

Register Name	Register Description
Set Chaining Mode Status	A read only register that determines if Chaining Mode is enabled or disabled.
Status	A read only register that stores DMA channel status information that may be read by the central processing unit.
Stop	Used to support a common data structure for data communication (the ring buffer) which reserves a portion of memory on doubleword boundaries for a DMA channel.

DMA Channels

Unlike ISA DMA channels, which could be programmed only for 8-bit or 16-bit data transfers between a DMA device and memory, EISA DMA channels 0-3 and 5-7 can be programmed for an 8-, 16-, or 32-bit data transfer between a DMA device and memory. (As with ISA, EISA DMA channel 4 is used for cascading the two controllers and may not be reprogrammed for any other purpose.)

DMA channels are request lines for I/O devices that need servicing. The on-chip DMA circuitry of the ISP provides for 32-bit addresses for DMA devices, with each channel supporting:

- a 16-bit ISA-compatible current address register (containing the least significant bits of the 32-bit address)
- an 8-bit ISA-compatible page register for address lines LA <23:16>, the low page register (containing the second most significant bits)
- an 8-bit EISA page register for address lines LA <31:24>, the high page register (containing the most significant bits)

For compatibility, the High Page register of a channel can be set to zero. With the High Page register set to zero, the DMA will operate in an "address compatible mode" used in ISA computers and addresses will not be incremented or decremented across page boundaries (either 64 KB for 8-bit channels or 128 KB for 16-bit channels).

The High Page register of a DMA channel can be also be programmed so that the channel will operate in "extended address mode" to increment or decrement the entire 32-bit address.

EISA DMA Data Transfer Types and DMA Cycles

Any EISA DMA channel can be programmed for 8-, 16-, or 32-bit data transfers between a DMA device and memory. (In contrast, ISA DMA channels can be programmed for only 8- or 16-bit data transfers.) EISA DMA channels are programmed using one of four DMA cycle types. These cycle types are listed and described in the following table and text.

Table 3-5. DMA Data Transfer Rates and DMA Cycles for EISA Products

Type of DMA Cycle	Type of Data Transfer Possible	Type of DMA Devices Compatible with DMA Cycle Type/ Data Transfer Type	Peak Data Transfer Rate (Mbytes/second)
ISA-Compatible	8-bit	All ISA	1.0
(Default DMA Cycle)	16-bit	All ISA	2.0
EISA Type "A"	8-bit	Most ISA	1.3
DMA Cycle	16-bit	Most ISA	2.6
	32-bit	EISA Only	5.3
EISA Type "B"	8-bit	Some ISA	2.0
DMA Cycle	16-bit	Some ISA	4.0
	32-bit	EISA Only	8.0
EISA Type "C"	16-bit	EISA Only	16.5
DMA Cycle	32-bit	EISA Only	33.0
(Burst Cycle)			

ISA-Compatible (or Default) Cycle

To ensure compatibility with ISA products, the EISA system provides a default ISA-compatible mode, which has ISA timing and functions. The ISA-compatible DMA cycle supports 8-, or 16-bit DMA devices that are transferring data to 8-, 16-, or 32-bit memory. This cycle, which is the same for all types of memory, does not require any hardware devices to be modified, nor does it require fast EISA memory. However, because of EISA's faster bus arbitration, the EISA product's default cycle for ISA-compatible DMA devices can transfer data at a faster rate than an ISA product can.

In ISA-Compatible DMA cycles, one transfer cycle is executed per 8 BCLK period. Two BCLK periods are added for each wait state.

DMA: Type A Cycle

Type A DMA cycles can be used with 8-, 16-, or 32-bit DMA devices. While normally this cycle type does not require any ISA-compatible hardware devices to be modified, it does require fast, EISA memory. By programming the EISA DMA controller for the Type A cycle (rather than for the default ISA-compatible cycle), most 8-bit or 16-bit ISA-compatible DMA devices can transfer data to memory 1.3 times faster. This performance improvement is made possible by reducing the time required for the memory read or write operation and the duration of the I/O read command or I/O write command.

For Type A DMA cycles, transfers which do not require data size translations (or "conversions") execute one cycle every 6 BCLK periods.

DMA: Type B Cycle

Type B DMA cycles can be used with 8-, 16-, or 32-bit DMA devices. While this cycle type may not require an ISA-compatible hardware device to be modified, it does require fast, EISA memory. By programming the EISA DMA controller for the Type B cycle (rather than for the default ISA-compatible cycle), most 8-bit or 16-bit ISA-compatible DMA devices can transfer data twice as fast as the default ISA-compatible transfer.

This performance improvement is made possible by reducing:

- the time required for the memory read or write operation
- the data setup time for I/O write sequences
- the read access time for I/O read sequences

For Type B DMA cycles, transfers are executed once every 4 BCLK periods (if no data size translations are required).

DMA: Type C (Burst) Cycle

Type C DMA cycles, the highest performance DMA cycle, are available only to DMA devices specifically designed to use them. The Type C burst DMA cycles can be used to make 16- or 32-bit transfers between EISA memory and DMA devices. These are burst cycles and execute one transfer every 1 BCLK period.

Interrupt Operations

During an ISA or EISA interrupt operation:

1. A hardware device sends an interrupt signal to an interrupt controller unit.
2. The interrupt controller unit:
   a. issues a hardware interrupt request to the system's main microprocessor,
   b. resolves priority on pending interrupts, and
   c. provides an 8-bit interrupt vector to the system's main microprocessor.
3. The main microprocessor uses the interrupt vector as an index to determine which interrupt service routine (ISR) should be executed.

To ensure that EISA systems are fully compatible with ISA interrupt operations, interrupts for ISA devices can be configured as "edge-triggered." In comparison, interrupts for EISA devices can be configured as "level-triggered," which enables the sharing of a single system interrupt between multiple devices, such as serial ports.

Interrupt Controller

In the EISA system, hardware interrupts are provided by the ISP (Integrated System Peripheral) chip - part of the Intel EISA chip set. Interrupt control circuitry on the chip is the system-wide interrupt manager and has the functionality of two cascaded 8-input 8259 interrupt controllers plus the EISA enhancement of level triggering. (Unlike an ISA interrupt controller, an EISA interrupt controller must never be programmed to the 8259's 8080 mode.)

The interrupt controller unit consists of 15 programmable and ISA-compatible interrupt request signals, IRQ <7:0> on the EISA master interrupt controller (INT-1), and IRQ <15:8> on the EISA slave interrupt controller (INT-2). The interrupt request output signal from INT-2, INT, is internally connected to INT-1's IRQ2. As a result, IRQ2 is not used as an interrupt, but is used to cascade interrupts from INT-2.

The interrupt controller defaults to the fixed priority mode, with IRQ0 having the highest priority and IRQ7 having the lowest priority, but the priorities can be programmed differently.

Refer to the I/O Address Map in the PROCESSOR chapter for a breakdown of the interrupt controller register assignments. For additional information regarding system interrupts, refer to the *HP Vectra 486/33T BIOS Technical Reference Manual.*

Interrupt Controller Registers

The table below lists and describes the interrupt controller's initialization and control registers. (For more information on the registers and for complete bit-by-bit programming information, refer to the *EISA Specification* and the *Intel 82357 Integrated System Peripheral Data Sheet.*)

Table 3-6. EISA Interrupt Controller Register Descriptions

Register Name	Register Description
Edge/Level Control	Provides a bit for each interrupt that programs the interrupt to either: 1. edge sensitive (the default), fully compatible with ISA expansion, or 2. level sensitive, which may be used in shared interrupt configurations
In-service	Indicates which interrupts are being serviced.
Initialization Command Word 1	Programs the interrupt controllers for the cascade mode.
Initialization Command Word 2	Initializes the interrupt controller with the 5-most significant bits of the interrupt vector address.
Initialization Command Word 3	When programmed to 02h for INT-2, an interrupt request on IRQ2 causes INT-1 to enable INT-2 to present the interrupt vector address during the second interrupt acknowledge cycle.
Initialization Command Word 4	Programs the system for: 1. type of nested mode 2. type of end-of-interrupt 3. type of microprocessor
Interrupt Mask	Masks a particular interrupt by setting the appropriate bit in this register.
Interrupt Request	Contains the status of each interrupt. Set bits indicate pending interrupts. Clear bits indicate interrupts that have not requested service.
Operation Control Word 2	Programs a specified interrupt for: 1. a non-specific or specific End of Interrupt (EOIs), used to clear an interrupt's in-service bit. 2. type of automatic rotation 3. type of specific rotation 4. no operation
Operation Control Word 3	1. Selects interrupt mask mode type. 2. Enables, disables poll mode command. 3. Selects for "read interrupt request register," or "read in-service" command.

Interrupts are defined in the associated BIOS Technical Reference Manual, Appendix A.

EISA Memory

In contrast to the ISA architecture's memory system, the EISA architecture's memory system:

- supports a greater memory capacity (which results in a changed memory map)
- supports 32-bit memory data transfer cycles

EISA Memory Capacity

An ISA product's total memory capacity is limited by its system architecture to 16 Mbytes. In contrast, an EISA product's total memory capacity is limited only by that product's memory packaging constraints, and as much memory may be added as needed for applications. (See the "Product Description" and "Memory" chapters for more information on the type and the quantity of memory used.)

An EISA product's main central processing unit, bus masters, and DMA devices can access the microprocessor's entire 4 gigabytes of memory space via the EISA 32-bit address path. (In the lower 16 Mbytes of DRAM, ISA memory cards may be used without modification.)

EISA Memory Data Transfer Operations

One of the main differences between ISA architecture and EISA architecture is that ISA supports 8-bit and 16-bit memory data transfer cycles, while EISA supports 16-bit and 32-bit memory data transfer cycles. Both ISA and EISA systems support memory slaves, but ISA memory slaves are 8 bit or 16 bit, while EISA memory slaves are 16 bit or 32 bit. The EISA system supports the following CPU, memory and I/O cycles for transferring data:

Standard Cycles

Standard EISA cycles are used to transfer data between the microprocessor and EISA memory or I/O slaves at a rate of one transfer per 2 BCLK periods (for zero wait states). Each additional BCLK period adds one wait state.

Compressed Cycles

Compressed cycles are used by the microprocessor to transfer data to or from fast EISA memory or I/O slaves. Compressed cycles are similar to standard memory cycles, but improve the data transfer rate. The compressed sequence has a data transfer rate of one transfer for each 1.5 BCLK cycles.

Burst Cycles

EISA memory supports multiple timing operations, used in burst cycles that provide zero-wait-state transfers to or from EISA memory. The burst sequence is defined such that it provides a continuous 1 BCLK sequence of all reads or writes. The W-R signal line does not change during a burst.

EISA memory slaves that support burst cycles also support the standard memory cycles. Burst memory slaves can transfer data each BCLK after the initial cycle, (16 bits or 32 bits, depending on whether the memory slave is 16 bits or 32 bits).

A burst sequence can not cross a 1-Kbyte (1,024-byte) address boundary. Instead, if a memory transfer to DRAM crosses a 1 Kbyte boundary, the system or master terminates

the original burst sequence by negating the MSBURST* signal. Then the system or master continues the burst sequence by restarting it within the next 1-Kbyte address block.

EISA Memory Refresh Operations

For the EISA system, DRAM refresh request signals are generated by two sources:

1. **Interval Timer 1, Counter 1**
   This is an interval counter/timer located on the ISP chip. This counter is programmed to request a system DRAM refresh about every 15 microseconds. (Do not reprogram this counter.)
2. **16-bit ISA bus masters**
   The 16-bit ISA bus masters assert the REFRESH* signal when they are in control of the bus for more than 15 microseconds.

In contrast to the ISA system's 16-bit ISA bus master, the EISA system's 32-bit EISA bus master does not need to supply a refresh cycle for the DRAM since the refresh controller can pre-empt the bus master and perform the necessary refresh cycles. Also, in contrast to ISA systems, EISA systems allow refresh to be held off for a maximum of 75 microseconds, which prevents the following devices from causing refresh loss when they do not release the bus:

- 16-bit ISA bus masters
- ISA compatible block DMA devices
- ISA demand mode DMA devices

Each time a refresh request is not serviced within the normal 15-microsecond interval, the EISA system's refresh controller:

- increments an "incomplete refresh counter," which counts up to four such refresh requests,
- executes one refresh cycle when it gains control of the bus, and
- decrements the pending refresh count.

However, if more refresh requests are queued up, the refresh controller immediately requests the bus again, without waiting the normal 15-microsecond interval. (Refer to the *EISA Specification* and the *Intel 82357 Integrated System Peripheral Data Sheet* for memory refresh timing diagrams.)

Real-Time Clock/Static CMOS RAM

The real-time clock and static CMOS RAM chip consists of

- 64 bytes of memory:
- 10 time-of-day clock bytes for real-time clocking, calendar, and alarm functions,
- 4 bytes for control and status lines, and
- 50 bytes of low-power, high-speed static CMOS RAM. (See Table 3-8 for the real-time clock/static CMOS RAM memory map.)

The 10 bytes for the RTC time-of-day clock provide the time of day, and count seconds, minutes, hours, days, and years. They also provide calendar functions, keeping track of the day of the week, the month, the date, and providing automatic leap year compensation. In addition, these bytes provide programmable interrupt requests (IRQ8) at either a fixed interval (for example, every second), or when a certain time has arrived (for an alarm clock mode).

The 4 bytes for control and status lines in Registers A, B, C, and D control the RTC/static CMOS RAM operation and monitor its status.

The 50 bytes of static CMOS RAM provide non-volatile memory to store ISA system configurations and calibration parameters when power is removed from the system. All the 50 static CMOS RAM bytes are either currently used or reserved, with no user-definable bytes. However, applications may read these bytes to determine system configuration. (See the byte definitions of Table 3-8.) The static CMOS RAM bytes are not affected by the RTC and are accessible during the update cycle.

Real-Time Clock/Static CMOS RAM Operation

When the system power is on, the RTC gets its clocking from the 32.768-kHz crystal oscillator on the System Board and gets its power from the system's power supply. During power-on reset, the system software reads the current time and date from the RTC and converts it into the appropriate number of system clock "ticks." From this point on, system time is kept via system clock ticks generated by Interval Timer No. 1's Counter 0. (See section "Counters and Timers," for more information on this counter.)

The RTC/static CMOS RAM's 64 locations are accessed by the 80486 microprocessor writing an address (or "index") ranging from 00h to 3Fh to I/O address 70h. (Bit 7 is the non-maskable interrupt bit.) At this point, the 80486 microprocessor can access the desired location by reading from or writing to I/O address 71h.

When the system power is off, the RTC has a battery for backup power to the RTC and to the 32.768-kHz crystal oscillator for the RTC's clocking.

As a result of the battery backup, the RTC keeps track of the time and date and maintains the static CMOS RAM. Disconnection or failure of the battery requires that the RTC must be reprogrammed for the correct time and the HP Vectra 486/33T must be reconfigured. The battery used is a 6-volt lithium battery pack and has an average life of 2.3 years.

For a complete listing of the contents of the CMOS RAM memory map refer to the associated BIOS Technnical Reference Manual.

Counters and Timers

The system's counters and timers are provided by the Intel 82357 Integrated System Peripheral (ISP) on the System Board. The 82357 has the equivalent of two separate 8254 interval timers, each with three 16-bit counter/timers. (However, counter 1 of the second interval timer is not implemented for EISA systems.)

The minimum time resolvable for the interval timers is approximately 838 nanoseconds (1 input clock cycle), and the maximum time resolvable is 55 milliseconds (65,536 clock cycles).

All counter/timers have the same control logic, which decodes information written to them. The control logic also provides the controls to load, read, and configure the counter/timers, all of which are dedicated to system functions. Hex address 43 holds the mode and command information for interval timer 1's counters, and hex address 4B holds the mode and command information for interval timer 2's counters.

Table 3-9 summarizes the operation of the counter/timers, each of which operates independently to provide the following functions:

- Interval timer 1's counter 0 provides a system timer interrupt (for time of day, diskette time-out, and other system timing functions).
- Interval timer 1's counter 1 provides DRAM refresh requests.
- Interval timer 1's counter 2 provides the speaker tone.
- Interval timer 2's counter 0 provides a fail-safe timer that prevents system lockup by providing a non-maskable interrupt to the 80486.
- Interval timer 2's counter 1 is not implemented for EISA systems.
- Interval timer 2's counter 2 provides CPU speed control for additional timing functions.

Programming Counter/Timers

The counter/timers are programmable by I/O accesses, the addresses for which are given in Table 3-9.

Interval timer 1's counter 2 (42h) drives the speaker. This counter has a gate input, bit 0 of the industry-standard Port B (61h), which is under program control. Writing a 0 to this bit inhibits counting action, turning the speaker off. (Because bit 1 of 61h is ANDed with this counter's output, writing a 0 to bit 1 also turns the speaker off.)

The BIOS programs each of the interval timers to a default state. All of the control registers are accessible but should **not** be reprogrammed (except for the speaker-tone counter) dues to the critical nature of the timers in the system.

Interval timer 2's counter 2 must be programmed if it is to slow down the CPU. By writing a value of 92 hex to I/O hex address 4B, interval timer 2's counter 2 is programmed for a one-shot period, which is triggered by the DRAM refresh request signal from interval timer 1's counter 1. As a result, the CPU stops for the period programmed for the one-shot.

Table 3-7. Counter/Timer Summary

	Interval Timer 1 Counter 0	Interval Timer 1 Counter 1	Interval Timer 1 Counter 2	Interval Timer 2 Counter 0	Interval Timer 2 Counter 1	Interval Timer 2 Counter 2
Function	System timer interrupt	DRAM refresh request	Speaker tone	Fail-safe timer	Not implemented	CPU speed control
Clocking	1.193 MHz (Y1/12)	1.193 MHz (Y1/12)	1.193 MHz (Y1/12)	0.298 MHz (Y1/48)	Not implemented	8 MHz (BCLK)
When Does Gate Input Activate Counter?	Gated input always on	Gated input always on	When port 61 bit 0 is active	Gated input always on	Not implemented	When REFRESH* is active
Hex I/O Address	40	41 (Do not reprogram)	42	48	Not implemented	4A
Signal out of 82357	INTR (connects to IRQ0)	REFRESH*	SPKR	NMI	Not implemented	SLOWH*

* Indicates a signal is active low.

I/O Address Map Differences

This section discusses the following general differences between the ISA and the EISA I/O address maps:

- the extended I/O address space available for EISA system boards (peripheral boards that work as an integral part of the system, but are not plugged into an expansion slot, such as the main memory board, and system interface board) and expansion boards (such as video boards, network boards, etc.),
- address ranges that are **aliases**, of ISA expansion board I/O space (100h-3FFh)
- **slot-specific** address ranges.

The following tables lists the general differences between ISA and EISA I/O address maps.

Table 3-8. General Differences between ISA and EISA I/O Address Maps

ISA I/O Address Map Characteristics	EISA I/O Address Map Characteristics
ISA I/O address map goes up to 3FFh.	EISA I/O address map goes up to FFFFh
An ISA system board has 256 bytes of I/O space available.	An EISA system board has 768 additional bytes of I/O space available, for a total of 1,024 bytes.
Addressing for ISA expansion boards is not slot specific.	An EISA expansion board, and the slot for it, has 1,024 bytes of slot-specific I/O address space, in addition to the I/O address space that the ISA I/O address map allocates to expansion boards.
An ISA expansion board is limited to an address in the ISA I/O range of 100h-3FFh.	An EISA expansion board can use any address or any alias address in the ISA I/O range of 100h-3FFh, except to ensure ISA compatibility, as long as it does not conflict with other installed ISA boards.

Extended I/O Address Space for EISA System Boards

Listed in the table below, EISA systems reserve the addresses 400h through CFFh of extended I/O address space for EISA system boards. The addresses 000h through 3FFh can be used for EISA or ISA system boards. A detailed version of the I/O Port Map is shown in the associated BIOS Technical Reference Manual.

Table 3-9. I/O Address Space for EISA System and Expansion Boards

Hex I/O Address Range	I/O Address Reserved For:
0000 - 00FF	EISA/ISA system board
0100 - 03FF[1]	ISA expansion boards[1]
0400 - 04FF	Current and future EISA system board peripherals.
0500 - 07FF[2]	Alias of 0100h - 03FFh[2]
0800 - 08FF	Manufacturer-specific system board peripherals.
0900 - 0BFF[2]	Alias of 0100h - 03FFh[2]
0C00 - 0CFF	Manufacturer-specific system board peripherals.
0D00 - 0FFF[2]	Alias of 0100h - 03FFh[2]
01000 - 010FF	Expansion slot 1
01100 - 013FF	Alias of 0100h - 03FFh
01400 - 014FF	Expansion slot 1
01500 - 017FF	Alias of 0100h - 03FFh
01800 - 018FF	Expansion slot 1
01900 - 01BFF	Alias of 0100h - 03FFh
01C00 - 01CFF	Expansion slot 1
01D00 - 01FFF	Alias of 0100h - 03FFh
0z000 - 0z0FF[3]	Expansion slot-z[3]
0z100 - 0z3FF	Alias of 0100h - 03FFh
0z400 - 0z4FF	Expansion slot-z
0z500 - 0z7FF	Alias of 0100h - 03FFh
0z800 - 0z8FF	Expansion slot-z
0z900 - 0zBFF	Alias of 0100h - 03FFh
0zC00 - 0zCFF[4]	Expansion slot-z[4]
0zD00 - 0zFFF	Alias of 0100h - 03FFh
⋮	⋮

Notes:

1. An EISA expansion board can also use this address range, but it must assure that its addresses do not conflict with other ISA expansion boards.
2. These address ranges are an alias of ISA expansion board I/O addresses, and cannot be used by an EISA *system* board.
3. The "z" refers to the most significant digit in the address, and represents a slot number. Valid values range from 1 to 15.
4. In this slot-specific address range, the addresses 0zC80h through 0zC83h are reserved for product ID, and 0zC84h is reserved for control bits of EISA expansion boards.

Aliases

As listed in Table 3-11, the I/O address ranges that are aliases of normal ISA expansion board I/O space are 100h through 3FFh. Alias addresses may be used only by EISA expansion boards that can assure no conflict occurs between the alias address and ISA expansion board I/O addresses (100h-3FFh).

Note

The configuration utility provided with the HP Vectra 486/33T does not identify conflicting use of ISA alias addresses.

Slot-Specific I/O Address Ranges

Table 3-9 also indicates slot-specific address ranges. (See the *EISA Specification* and the *HP Vectra 486/33T BIOS Technical Reference Manual* for more information on slot-specific ranges.) The system board decodes the EISA slot-specific I/O address ranges and all I/O for system board devices to generate slot-specific signals. Unless otherwise specified (as in 0zC80h through 0zC83h, and 0zC84h), all slot-specific addresses can be used by EISA expansion boards for configuration registers and general-purpose I/O. The slot-specific ranges do not conflict with any ISA expansion board.

Configuration of The EISA System

In an EISA system, a method for automatic configuration is provided (for both EISA system and expansion boards) to keep the system's resources conflict-free. Manufacturers, such as Hewlett-Packard, provide configuration (CFG) files for EISA system and expansion boards. These files contain initialization and configuration information which the system configuration software (EASY CONFIG) utilizes and stores in non-volatile CMOS memory. Thus, when the system is powered-up, ROM initialization routines and device drivers are able to use this information to configure expansion boards for correct operation.

For more information about the EASY CONFIG utility, refer to the documentation that comes with the HP Vectra 486/33T PC.

The following types of boards can be supported for automatic configuration:

- **EISA system boards**
  These can either be system peripheral boards that use slot-specific I/O (like an EISA expansion board would) but are integrated on the system board (not plugged into a bus connector), or system peripheral boards that use the ISA I/O address space.
- **EISA expansion boards**
  These boards are plugged in to an expansion slot (also called a bus connector) and use the 1024 byte, slot-specific I/O address space.
- **ISA expansion boards**
  These boards are plugged into an expansion slot, but can only be automatically configured if a configuration file is supplied with the board.

- **Software Drivers**
  These drivers are typically supplied with expansion boards, require system resources (such as LIM drivers for memory boards), and are treated by the system as "virtual devices."

For more information on what configuration files are, and how to create them, refer to the references listed at the end of this chapter.

EISA Expansion Board Design Considerations

An expansion board built specifically for an EISA system must be built to the EISA connector's electrical and mechanical specifications. The EISA expansion board will work only with an EISA connector and system; it will not be compatible with an ISA connector and system. However, 8-bit and 16-bit ISA expansion boards are compatible with the EISA connector and system.

Refer to the *EISA Specification* document for details on board and connector dimensions.

ISA/EISA Expansion Boards Differences

As explained below, ISA expansion boards do not have to be modified to adhere to EISA specifications in order to be used in an EISA system. EISA expansion boards, on the other have, must adhere to EISA specifications in order to be used in an EISA system, and they can not be used in ISA systems.

ISA/EISA Expansion Board Electrical Considerations

The following table shows the power specification for each slot in which an ISA or EISA expansion board can be placed, assuming a maximum of 8 bus slots. For more information on the capabilities of the power supply, refer to the "POWER SUPPLY" chapter.

Table 3-10. Electrical Specifications for EISA/ISA Expansion Boards

Power Supply Voltage	Maximum Slot Current	Guaranteed Slot Current
+5 Volts ± 5%	4.5 Amps	2.0 Amps
−5 Volts ± 10%	0.2 Amps	—
+12 Volts ± 5%	1.5 Amps	—
−12 Volts ± 10%	0.3 Amps	—

Note

1. For ISA expansion boards, current on any pin cannot exceed 1.5 Amps.
2. For EISA expansion boards, current on any pin cannot exceed 0.5 Amps.
3. Because of the large number of ground pins on the EISA I/O connector, EMI characteristics for EISA expansion boards are consistent with present ISA expansion board EMI standards.

ISA/EISA Expansion Board Mechanical Considerations

An ISA expansion board used in an EISA system has the same mechanical specifications as an ISA expansion board used in an ISA system. The connector has positive stops which prevent an ISA expansion board from going in as deep as EISA expansion boards can. As a result, an ISA expansion board can be inserted only so far as the upper-level (ISA) contacts, and can not connect with the lower-level (EISA) contacts.

In contrast with ISA expansion boards, as shown in Figure 3-3 and 3-4, an EISA expansion board is longer and is notched, which allows it to be inserted so that it connects with both the upper-level ISA contacts and the lower-level EISA contacts.

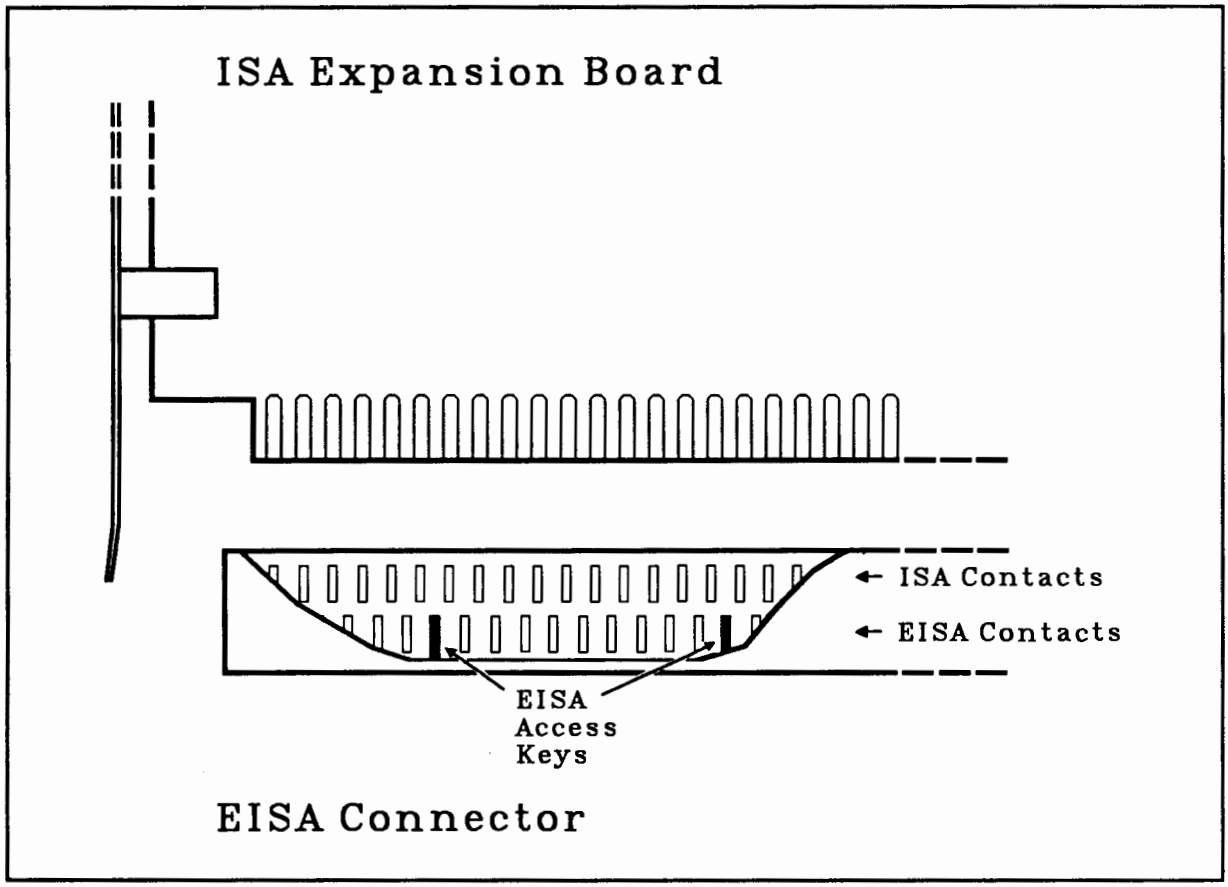


Figure 3-3. ISA Expansion Board in EISA Connector

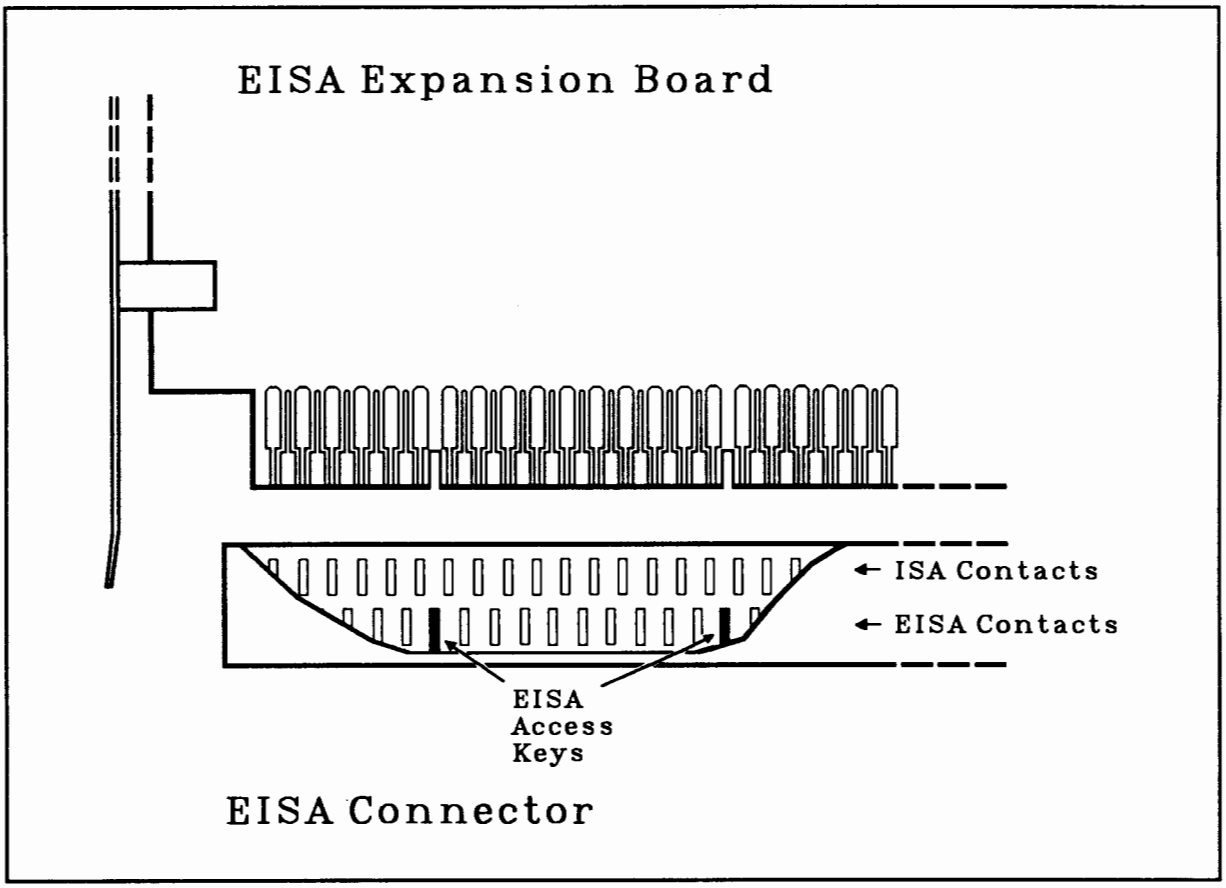


Figure 3-4. EISA Expansion Board in EISA Connector

ISA/EISA Expansion Board Insertion Force

The following table gives the different insertion forces required for the two types of expansion boards. The EISA I/O connector contacts must be able to maintain a contact force of 75 grams (0.167 pounds) for the rated life expectancy of the connector.

Table 3-11.
Force Needed to Insert Expansion Boards into EISA I/O Connector

Type of Expansion Board	Force Required In Kilograms	Force Required In Pounds
ISA	12.6 max.	28 max.
EISA	15.75 max.	35 max.

4

Processor

Introduction

This chapter discusses the system clocks, and describes the main processor's features, applications, operation, and interface with the buses. In addition, the HP-specific EISA I/O address map is given.

System Clocks

The system clocks generate timing signals for the components listed in the following table. (For more information on the clocks, refer to the components for which the system clocks provide signals and to the schematics in Appendix A.)

Table 4-1. System Clocks for the HP Vectra 486/33T

Board on which Clock Appears	Clock Speed	Clock Function
Processor Board	66 MHz	The 66-MHz clock is divided down externally to 33 MHz. The resulting 33 MHz signal provides the fundamental timing and the internal operating frequency for the system's 80486 micro- processor. All external timing parameters are specified with respect to this clock's rising edge.
Keyboard/Mouse/ Serial Board	12 MHz	Provides timing to the 8042 keyboard/mouse controller.
Serial/Parallel Board	1.8432 MHz	Provides timing to the serial controller.
System Interface Board	32.768 kHz	Crystal oscillator which provides timing to the MC146818A real-time clock/CMOS RAM chip.
System Interface Board	9.6 MHz	Provides timing to the flexible disk drive controllers.
System Interface Board	16 MHz	Provides timing to the flexible disk drive controllers.
System Interface Board	14.31818 MHz	Provides timing to the 82357 Integrated System Peripheral.
Main Memory Board	1.16 MHz	Crystal oscillator which provides timing to the Memory Controller.

Microprocessor Features and Applications

Microprocessor Features

The HP Vectra 486/33T uses as its central processing unit the Intel 33-MHz 80486 microprocessor. The 80486 provides all the features provided by an 80386 microprocessor while offering two to three times the 80386's performance. In addition, the 80486 provides new features on-chip: cache memory, a floating point coprocessor, and page memory management. The 80486 microprocessor can support up to 4 gigabytes of memory space, the same as the 80386.

Here is a summary of what the 80486 microprocessor provides when it is used with the HP Vectra 486/33T.

- cache memory on chip: 8-Kbytes, 4-way set associative with write-through
- comprehensive instruction set, fully compatible with that of the 8088, 8086, 80286, and 80386 microprocessors
- data types supported: 8-, 16-, and 32-bit
- direct-memory access (DMA) support capabilities
- floating point coprocessor on chip, compatible with 80387 coprocessor
- interrupt support capabilities
- I/O locations supported: up to 64K
- memory addressing: 4 gigabytes of physical memory and 64 terabytes of virtual memory
- multi-tasking support
- multiple processor support
- operating address modes: real, protected, and virtual 8086
- page memory management on chip
- registers: 32-bits for both address and data registers
- self-test built in
- separate address and data paths

Microprocessor On-Chip Coprocessor

An important feature of the 80486 microprocessor is the system's on-chip coprocessor. Also known as a "floating point unit" (FPU), the coprocessor is a numeric processing unit that can significantly improve the performance of those applications which have been designed to use the functions of a coprocessor.

Here is a summary of some of the types of applications which support the use of a coprocessor.

- computer-aided design (CAD)
- computer-aided engineering (CAE)
- computer-aided manufacturing (CAM)
- other applications for science and engineering
- spreadsheets

Coprocessor Operation

The 80486's on-chip coprocessor extends the 80486 microprocessor's instruction set by providing hardware for binary-coded decimal data and the high-precision integer functions and floating-point calculations otherwise performed by software. The 80486's coprocessor uses the same clocking as the 80486, and it operates exactly the same as the Intel 80387 coprocessor.

Software written for the 80387 coprocessor can be used with the 80486's coprocessor without any changes.

Coprocessor Compatibility

When unmasked floating point errors or exceptions occur, compatibility with the error-reporting scheme used in DOS-based systems is ensured by two 80486 pins:

- IGNNE*—an active-low Ignore Numeric Error pin, input to the 80486
- FERR*—an active-low Floating Point Error pin, output from the 80486, and similar to the 80387 coprocessor's ERROR* pin.

When an unmasked floating point error occurs, these pins are used as follows to ensure coprocessor compatibility:

1. The 80486 asserts the FERR* signal, indicating a floating point error has occurred.
2. As a result of the FERR* signal being driven active, it generates interrupt request IRQ13 to the Intel 82357's Integrated System Peripheral's 8259 interrupt controller unit.
3. If there are exceptions to (i.e., errors with) the coprocessor's functioning, the 80486 operation is interrupted by a non-maskable interrupt (02 hex).
4. The IGNNE* pin is high at the time of this interrupt and the microprocessor freezes (disallows execution of subsequent) floating point instructions until the interrupt handler is invoked.
5. By driving the IGNNE* pin low, (when clearing the interrupt request), the interrupt handler is able to execute floating point instructions within the interrupt handler before the error condition is cleared.
6. The IRQ13 then performs an end-of-interrupt (EOI) function.
7. The FERR*'s error condition is reset by a 0 being written to I/O address F0h.

NOTE: The FERR* signal is not floated during bus holds.

Microprocessor On-Chip Cache

An additional feature of the 80486 microprocessor is an on-chip 8 Kbyte cache for both code and data which supports a continuous request rate of one per clock. The caching of both code and data provides for better hit rates and a faster cache. The cache has the ability to listen to the bus to prevent the microprocessor from using stale cache memory in multiprocessor systems. The cache design is such that it is transparent to software and keeps the contents of the cache coherent with that in the main memory. Operation of the cache is controlled by two bits (CD and NWT) in the CR0 registers; CD disables the cache and NWT controls memory write throughs. The differing operation modes of these bits are as follows:

CD	NWT	Cache Operating Modes
1	1	Disable cache fills, write through and invalidates.
1	0	Disable cache fills. Enable write through and invalidates.
0	1	Invalid configuration.
0	0	Enable cache fills, write through and invalidates.

Any part of memory can be cached, and caching is always active. When the cache is enabled, a cache miss will initiate a cache fill. When the cache is disabled, a cache miss will not initiate a cache fill. Disabling the cache causes the cache to be flushed.

Microprocessor Operation

The following sections describe the 80486 microprocessor's operation: its clocking, compatibility with hardware and software, and operating modes.

Microprocessor Clocking

In contrast to the 2X clock used by the 80386 microprocessor, the 80486 uses a 1X clock. For this 1X clock, a 66 MHz signal from OSC1 is divided by a clock generator into a 33 MHz signal which is then buffered before being sent to the 80486. Some industry-standard operations require the 80486 to run at slower speeds.

This is achieved by programming the external hardware as follows:

1. The Intel 82357's Interval Timer 2, Counter 2 is programmed for a one-shot mode by writing a hex value of 92 to I/O hex address 4B.
2. When a refresh request signal is generated by Interval Timer 1's, Counter 1, it triggers the Interval Timer 2, Counter 2.
3. The Interval Timer 2, Counter 2's output signal (SLOWH*) stops the 80486 for the one-shot period pre-programmed into Interval Timer 2, Counter 2.
4. As a result of the 80486 being stopped for a programmed interval of time, it slows from 33 MHz to the chosen speed.

Microprocessor Hardware and Software Compatibility

The 80486 microprocessor maintains both hardware and software compatibility, as described below. (For information on the 80486 on-chip coprocessor's compatibility, see section "Microprocessor On-Chip Coprocessor.")

Hardware Compatibility

The 80486 microprocessor has all the hardware features of the 80386 microprocessor plus additional features: an on-chip memory management unit completely compatible with the 80386, an on-chip 80387 coprocessor, and an on-chip cache memory unit. The 80486 is completely compatible with the 80386 microprocessor and the 8088/8086 and 80286 microprocessors. It relies on the 82358 bus controller to dynamically size data to fit the 8 and 16 bit cards.

Software Compatibility

The 80486 microprocessor maintains software compatibility, in that its object code is compatible with the 80386, and it can run code developed for systems based on the 8088/8086 and 80286 microprocessors. The 80486 runs nearly all industry-standard application software and operating systems, including Microsoft MS-DOS 3.3, MS-DOS 4.0, OS/2, and Windows/386. In addition, the 80486 runs new application software and operating systems written for it (including UNIX V.3/386 and Xenix 2.3.2).

Caution To maintain the 80486's software compatibility:

- observe which 80486 register bits are reserved by Intel.
- avoid software dependence upon the state of undefined 80486 register bits.
- follow the compatibility guidelines given in the Intel *i486 Microprocessor* data sheet.

Microprocessor Operating Modes

The 80486 microprocessor offers real-address, protected, and virtual 8086 operating modes. In the real-address operating mode, the 80486 operates as an 8086 microprocessor and runs nearly all industry-standard applications. The protected and virtual 8086 operating modes allow the 80486 to use its full capabilities, while these modes remain transparent to those applications which do not take advantage of them.

Real-Address Operating Mode

The 80486 microprocessor's real-address operating mode allows the 80486 to operate as a very fast 8086 microprocessor. In this mode, the 80486 operates with the same base architecture as the 8086, and with the same 16-bit default operand size as the 8086.

The prime purpose of the 80486's real-address operating mode is to set up the 80486 for the protected mode operation, but in this mode, the 80486 does run microprocessor software written for the 8086, 80286, and the 80386.

The 80486's real-address operating mode is compatible with operating systems written for the 8086 microprocessor. With a few differences (as described in Intel data books), when the 80486 is in the real-address operating mode, its object code is compatible with the real-address object code of the 8088, 8086, 80286, and 80386 microprocessors. Furthermore, in the real-address operating mode, the 80486's addressing mechanism, 1-Mbyte memory size, and interrupt handling are identical to, and compatible with, those for the 80286 and the 80386 when those microprocessors are in the real-address mode.

The following are highlights of the 80486 microprocessor's real-address mode operation:

- The 80486 microprocessor's real-address operating mode is entered when the system processing unit is powered up or a system reset occurs.
- Although the real-address operating mode addresses just up to 1 Mbyte of system memory, 32-bit operands are allowed if override prefixes are used, and if the 32-bit effective addresses have a hex value less than 0000FFFF.
- With a few exceptions detailed in Intel data books, all of the 80486's instructions are available in the real-address mode.
- The real-address mode does not support memory protection features.
- Paged memory management is not allowed in the real-address mode; paging is allowed only in the protected mode and the virtual 8086 mode.
- Two memory areas are reserved in the real-address mode: system initialization and the interrupt table area.

- To minimize the physical memory needed for a program, memory segments may be overlapped in the real-address mode.

Protected Operating Mode

The 80486 microprocessor's protected operating mode allows the 80486 to operate at its fullest capacity. In this mode, as in the real-address mode, the 80486 operates with the same base architecture (registers, instructions, addressing modes) as the 8086. However, in contrast with the real-address mode, when the 80486 is in the protected mode, it can use 32-bit operands, memory space is available above the real-mode's 1-Mbyte cutoff, and sophisticated memory management techniques can be used.

The purpose of the protected operating mode is to support multi-tasking operating systems, to support memory protection, and to provide access to the 80486's memory management. In addition, the protected operating mode runs 8086, 80286, and 80386 microprocessor software and extends memory addressing beyond 1 Mbyte to support those programs that use memory above 1 Mbyte.

The 80486's protected operating mode is compatible with operating systems written for the 8086 microprocessor. When the 80486 is in the protected mode, its object code is compatible with the protected mode object code of the 80286 and 80386 microprocessors. However, when the 80486 is in the protected mode, its addressing mechanism is different compared to the real-address mode. Furthermore, its object code is not compatible with the real-address object code of the 80286 or 80386, and its object code is not compatible with the 8088 or 8086.

The following are highlights of the 80486 microprocessor's protected mode operation:

- The 80486 microprocessor's protected operating mode is entered by:
  - first initializing the global and interrupt descriptor tables and Control Register 0 (CR0) with appropriate values,
  - loading the current code segment (CS) register and flushing the instruction decode queue,
  - loading data segment registers with initial selector values.
- Alternatively, the protected operating mode can be entered by using the 80486's built-in task switch to load all registers (especially recommended if the operating system being used allows multi-tasking).
- The 80486's protected operating mode is exited by resetting Control Register 0 to its original value.
- All of the 80486's instructions that are available in the real-address mode are also available when the 80486 is in the protected mode.
- The 80486's protected operating mode allows for either segmented or paged memory management.
  - Segmented memory management provides memory protection by encapsulating memory regions which have common attributes into variable-length segments. As a result, data and code are easily moved and global resources are efficiently shared.
  - Paged memory management (used with Unix/386) provides a means to manage large program segments by dividing them into multiple uniform pages of 4 Kbytes; at any one point in time, only a few pages from each active task need be in memory. (Paging operates beneath segmentation and is transparent to it.)

- In the protected mode, the 80486 has four levels of protection which support multi-tasking operating systems' need to isolate and protect user programs from the operating system and from each other.

Virtual 8086 Operating Mode

The 80486 microprocessor's virtual 8086 operating mode is a type of protected operating mode that allows the 80486 to operate multiple 8086 sessions of 1 Mbyte each. In this mode, as in the real-address mode, the 80486 operates with the same base architecture (registers, instructions, addressing modes) as the 8086. However, in contrast with the real-address mode, when the 80486 is in the virtual 8086 mode, it uses the 80486's protection mechanism to support multi-tasking with 16-bit operands.

The purpose of the virtual 8086 operating mode is to support multi-tasking and multi-user situations (multiple-DOS sessions, Windows/386) to support memory protection, and to provide access to the 80486's page memory management.

When the 80486 is in the virtual 8086 mode, the 80486's software and operating systems are compatible with those written for the 8088 and 8086 microprocessors.

The following are highlights of the 80486 microprocessor's virtual 8086 mode operation:

- The 80486 microprocessor's virtual 8086 address operating mode is entered in one of two ways:
  - switch to a task with a 80486 task state segment (TSS) that has a 1 in the virtual mode (VM) bit in the flags register (EFLAGS), or
  - execute a 32-bit interrupt return (IRET) instruction at privilege level 0, where the stack has a 1 in the VM bit in the EFLAGS flags register.
- The 80486's virtual 8086 mode is exited (and the 80486's protected mode entered) only when the 80486 receives an interrupt or an exception.
- When the 80486 is interrupted while it is in the virtual 8086 mode:
  - The 80486 determines if the interrupt comes from a protected mode application or from a virtual mode application by detecting the EFLAGS' VM bit.
  - If the interrupt comes from a virtual mode application, the program execution passes to the interrupt routine at level 0, and the VM bit in EFLAGS is cleared.
  - The 80486 handles the interrupt and returns control to the 8086 program.
- All virtual 8086 mode programs execute at privilege level 3, the level of least privilege.
- On a per task basis, the 80486 allows the operating system to specify which programs use the 8086-style addressing mechanism and which use the protected mode style of addressing mechanism. By enabling the paging hardware, the 1-Mbyte address space of the virtual 8086 mode's task can be mapped anywhere in the 80486's address space.
- Paging hardware must be enabled to run multiple virtual 8086 mode tasks, or as mentioned above, to move the 1-Mbyte address space of a virtual 8086 mode's task within the 80486's address space.

Microprocessor Interface Circuitry

The following sections explain how the 80486 microprocessor interfaces with the HP Vectra 486/33T's buses.

Microprocessor/Host Bus Interface

The 80486 microprocessor's interface with the host bus (through the write buffer and the 128 KB cache) connects the 80486 to the rest of the system. The 80486/host bus interface is through the bus interface pins listed in Table 4-3.

Table 4-2. Bus Interface Pins

Type of Bus Interface Pin	Name of Bus Interface Pin	Pin Name Explanation
Address Bus	HA31 - HA2	Address pins 31 through 2
Data Bus	HD31 - HD0	Data pins 31 through 0
Byte Enables	HBE3* - HBE0*	Byte enable
Control Pin	HADS0*	Address status
Control Pin	HD/C*	Data/control
Control Pin	HLOCK*	Bus cycle lock indicator
Control Pin	HM/IO*	Memory/input-output
Control Pin	HRDYI*	Non-burst ready
Control Pin	HW/R*	Write/read

* Indicates a pin's signal is active low.

Microprocessor/Host Bus Interface Operation

The host address bus, which is separate from the host data bus, generates 32-bit addresses as follows: the 30 host address bus signals (HA31 to HA2) indicate a 4-byte location, from which the host address bus's four byte-enable signals (HBE3* to HBE0*) select the bytes to be enabled. The 32 host data bus signals (HD31 to HD0) transfer either 8, 16, or 32 bits of data.

Host Bus/EISA Bus Interface

The 80486 microprocessor interfaces with the EISA bus through the Intel 82358 bus controller. The EISA bus is a combination of the system bus (the S bus) and the latchable address bus (the LA bus).

Microprocessor/Peripheral Bus Interface

The 80486 interfaces with the peripheral bus (the X bus) through the 82358 bus controller and the system bus (the S bus).

Microprocessor/Host Bus Interface

The 80486 interfaces with the memory data bus through the memory controller circuitry on the Main Memory Board.

5

Memory

Introduction

The main memory for the HP Vectra 486/33T supports 33 MHz CPU operation and 32-bit burst data access on the EISA bus. Main memory is located on a separate board mounted to the System Board. The main memory consists of system RAM and the memory controller residing on the Memory Board. A block diagram of the Memory Board is shown below:

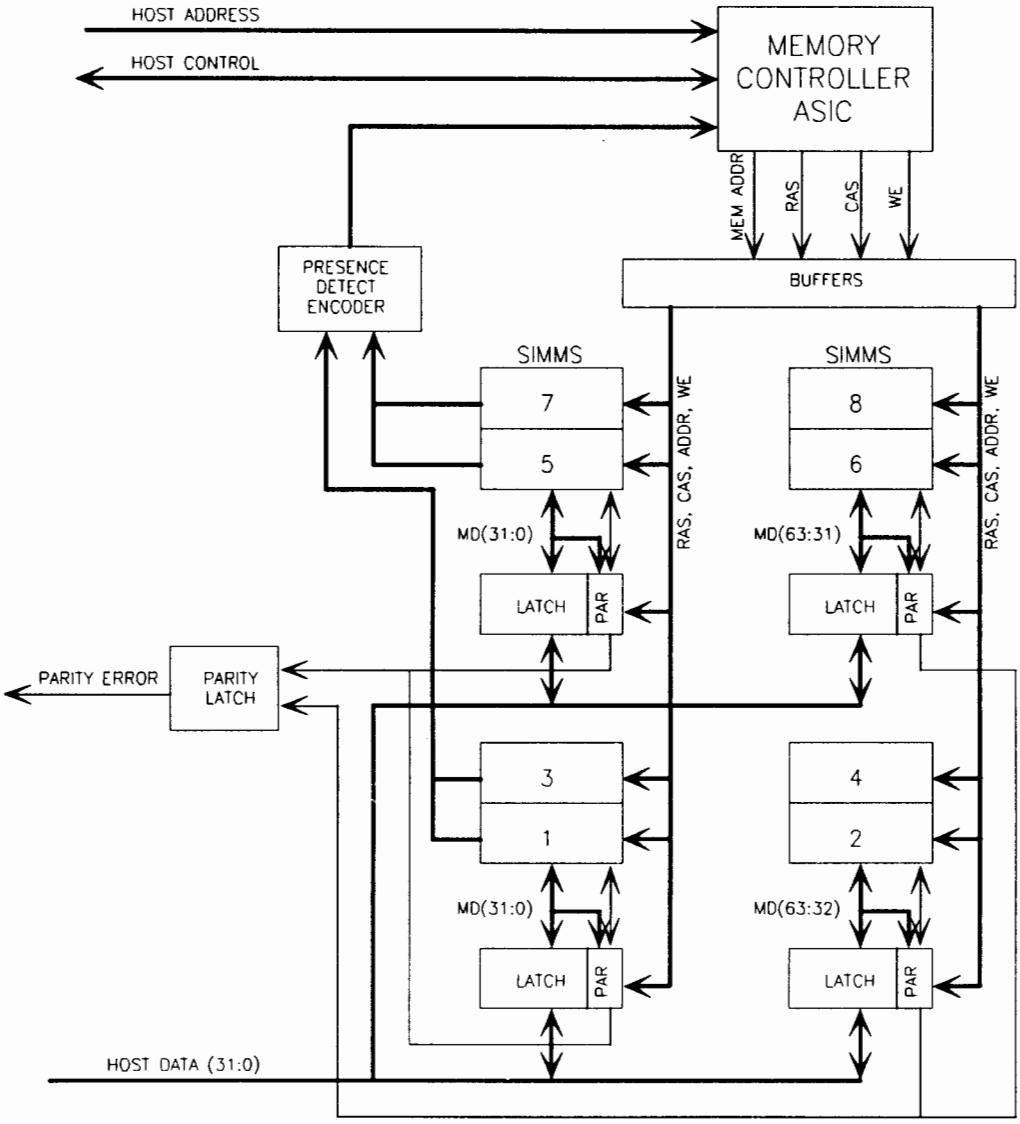


Figure 5-1. Memory Board Block Diagram

Main Memory (DRAM)

The main memory (also known as system memory, system RAM, or RAM) consists of dynamic RAM (DRAM) chips mounted on memory modules. The types of memory modules supported are:

Single-Density memory modules:	1 MB (256 KB x 36 bits) - (1 MB chips) 4 MB (1024 KB x 36 bits) - (4 MB chips)
or	
Double-Density memory modules:	2 MB (512 KB x 36 bits) - (1 MB chips) 8 MB (2048 KB x 36 bits) - (4 MB chips)

Note

Certain combinations of *either* all single-density or all double-density supported memory modules may be used (see Tables 5-1 and 5-2), however; single and double-density memory modules may not be mixed on the memory board.

DRAM is organized into socket pairs; up to 4 socket pairs supported as shown in the figure below.

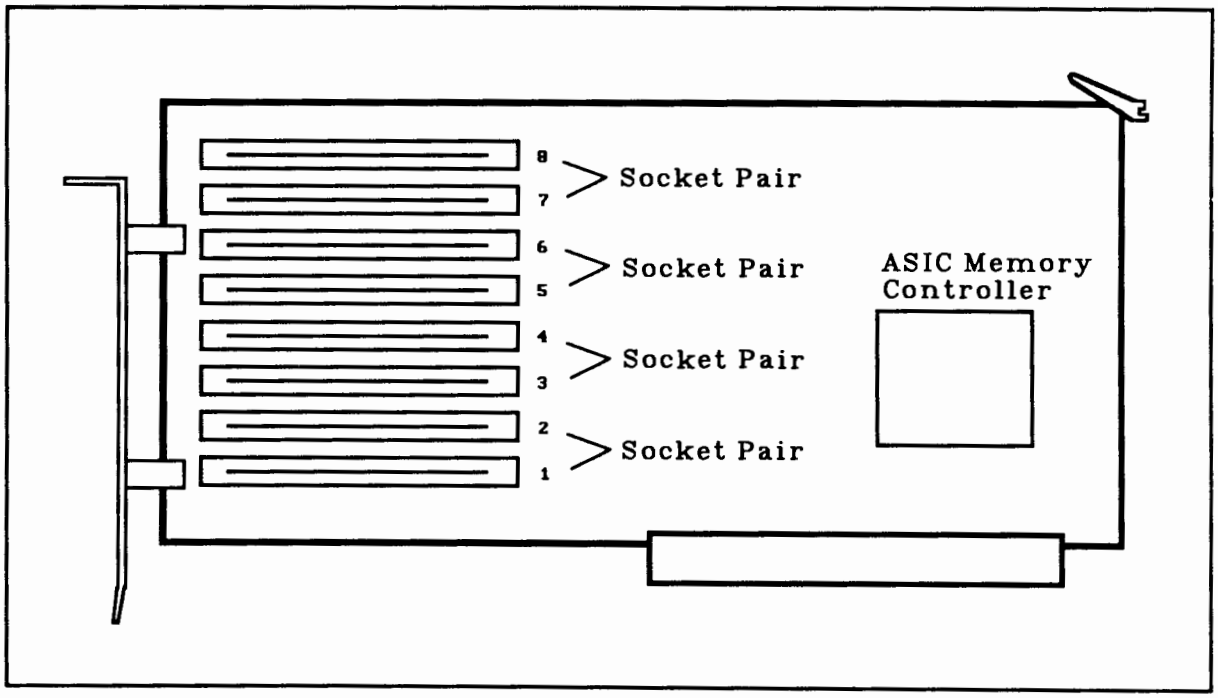


Figure 5-2. Memory Board Layout

Base RAM

The 640 KB of base RAM can be configured for 640 KB or 512 KB by using the computer's configuration software. This may be necessary for some applications and accessory boards to operate properly; refer to the manual that came with the board.

Memory Upgrades

The HP Vectra 486/33T PC comes standard with 4 MB (megabytes) of main memory. Memory is expandable in even increments up to either 32 MB or 64 MB, depending upon the type of memory modules you use:

For single-density memory modules, the following combinations are supported:

1 MB and 4 MB Single-Density Memory Configuration

For total memory to equal	Put these memory modules	In these module slots
2 MB	Two 1 MB modules	1, 2
4 MB	Four 1 MB modules	1, 2, 3, 4
6 MB	Six 1 MB modules	1, 2, 3, 4, 5, 6
8 MB	Eight 1 MB modules *or* two 4 MB modules	1, 2, 3, 4, 5, 6, 7, 8 1, 2
10 MB	Two 4 MB modules *and* two 1 MB modules	1, 2 3, 4
12 MB	Two 4 MB modules *and* four 1 MB modules	1, 2 3, 4, 5, 6
14 MB	Two 4 MB modules *and* six 1 MB modules	1, 2 3, 4, 5, 6, 7, 8
16 MB	Four 4 MB modules	1, 2, 3, 4
18 MB	Four 4 MB modules *and* two 1 MB modules	1, 2, 3, 4 5, 6
20 MB	Four 4 MB modules *and* four 1 MB modules	1, 2, 3, 4 5, 6, 7, 8
24 MB	Six 4 MB modules	1, 2, 3, 4, 5, 6
26 MB	Six 4 MB modules *and* two 1 MB modules	1, 2, 3, 4, 5, 6 7, 8
32 MB	Eight 4 MB modules	1, 2, 3, 4, 5, 6, 7, 8

For double-density memory modules, the following combinations are supported:

2 MB and 8 MB Double-Density Memory Configuration

For total memory to equal	Put these memory modules	In these module slots
4 MB	Two 2 MB modules	1, 2
8 MB	Four 2 MB modules	1, 2, 3, 4
12 MB	Six 2 MB modules	1, 2, 3, 4, 5, 6
16 MB	Eight 2 MB modules *or* two 8 MB modules	1, 2, 3, 4, 5, 6, 7, 8 1, 2
20 MB	Two 8 MB modules *and* two 2 MB modules	1, 2 3, 4
24 MB	Two 8 MB modules *and* four 2 MB modules	1, 2 3, 4, 5, 6
28 MB	Two 8 MB modules *and* six 2 MB modules	1, 2 3, 4, 5, 6, 7, 8
32 MB	Four 8 MB modules	1, 2, 3, 4
36 MB	Four 8 MB modules *and* two 2 MB modules	1, 2, 3, 4 5, 6
40 MB	Four 8 MB modules *and* four 2 MB modules	1, 2, 3, 4 5, 6, 7, 8
48 MB	Six 8 MB modules	1, 2, 3, 4, 5, 6
52 MB	Six 8 MB modules *and* two 2 MB modules	1, 2, 3, 4, 5, 6 7, 8
64 MB	Eight 8 MB modules	1, 2, 3, 4, 5, 6, 7, 8

Automatic RAM Hardware Configuration

The memory board uses four presence-detect bits per socket pair (16 in all) for automatic RAM hardware configuration. 8 bits of encoded information which represent the amount of memory installed is provided through 2 I/O ports (FCA2h and FCA3h) for use within the system.

Memory Module Type Detection

The I/O port that detects the type of memory module installed in each bank is described later in this chapter under "SIMM Type".

Main Memory Architecture and Operations

The HP Vectra 486/33T uses a paged main memory architecture with a custom memory controller. DRAM is organized into rows and columns. One memory strobe (the RAS, or row address strobe) latches the row address of the page; a second memory strobe (the CAS, or column address strobe) latches the column address.

On the memory board itself, DRAM is physically organized into socketed pairs (4 pairs total). Each socketed pair can be either 1 bank (single-density memory modules) or 2 banks (double-density memory modules), for a maximum of up to 4 or 8 banks respectively. Upgrading must always be done in pairs of sockets.

The Memory Controller

The HP Memory Controller is a standard cell ASIC chip compatible with the Intel 82350 EISA chip set and is at the heart of the Memory Board. It supports the 80486 at 33 MHz in addition to the following:

- 32-bit CPU access
- 80486 1 wait burst read
- Shadow RAM
- EISA/ISA DMA and bursting EISA bus masters
- Control for memory parity generation/check
- Up to 8 banks of DRAM, from 4 to 64 Megabytes total memory
- RAS-only refresh

The Memory Controller provides RAS, CAS, WE and memory address (MA) for the DRAM array. Configuration registers allow programmable Shadow RAM and memory remapping.

Memory Controller Operation

The HP Memory Controller (1) controls all main memory accesses from the 80486 microprocessor, DMA (direct-memory accesses), bus masters, and refreshes and (2) organizes the main memory DRAM into paged DRAM to decrease the average number of wait states in an access. Clocking for the memory controller comes from the signal CLK, which originates from the Processor Board and is synchronous to the 80486 CLK.

This paged memory can be accessed as 32-bit dwords (double words). To access memory, the Memory Controller issues a row address and then a column address to the accessed module. The row address indicates the page, while the column address and the eight (two sets) Latched Byte Enable (LBE) signals indicate which of the bytes on that page (row) are to be accessed. The row and column addresses are multiplexed over MA0-MA9 from the memory controller and are latched into a module by the appropriate Row Address Strobe (RAS) and Column Address Strobe (CAS) signals. To achieve 8-bit, 16-bit, or 32-bit operations, the CAS signal is ANDed with the appropriate Latch Byte Enable (LBE<7:0>) signal from the Memory Controller. If the next access is within the current page (row), a new row address does not need to be issued, just a new column address and CAS signal, resulting in fewer wait states.

In direct-memory accesses (DMA), 8-bit, 16-bit, or 32-bit operations are supported. The address on the local address bus comes from the System Board's 82357 Integrated System Peripheral via the peripheral address bus and the address buffers. The memory controller decodes the address as it would an access from the 80486 microprocessor.

A refresh access is initiated when the memory controller receives a REFRESH* input from the 82357 (ISP).

Memory Bus

The memory data bus, which via the data buffer interfaces to the host data bus (HD), is 72 bits wide (64 data bits, with 8 parity bits). The memory address bus (MA), is a 10-bit bus (MA0 to MA9) from the memory controller that accesses a particular DRAM location. (For more information on how memory accesses are performed, refer to the section in this chapter, "Memory Controller.")

Refresh

A refresh cycle begins with the 82357 Integrated System Peripheral (ISP) Controller's counter 1 timer 1 generating a refresh request (REFRESH*) to the system arbiter every 15 microseconds. If a refresh request is not serviced within 15 microseconds a counter keeps track of the unfulfilled requests (up to four are allowed) to make sure that a refresh cycle occurs within 75 microseconds of the first request.

DRAM refresh cycles require 3 BCLKs as follows:

1. At the first BCLK, RAS0 and RAS7 are activated simultaneously.
2. At 1 and 1/2 BCLKs, RAS1 and RAS6 are activated.
3. At 2 BCLKs, RAS2 and RAS5 are activated.
4. At 2 and 1/2 BCLKs, RAS3 and RAS4 are activated.
5. By 3 and 1/2 BCLKs, all RAS lines are deactivated.

To perform a refresh, the ISP asserts a signal to the EBC to get control of the bus. The EBC generates a hold request (HOLD) to the CPU, which subsequently generates a hold acknowledge (HLDA) and releases control of the bus to the EBC. During the refresh, MRDC*, CMD* and host bus control signals are generated by the EBC to refresh system memory via the memory controller.

The REFRESH* signal also goes to the backplane I/O and indicates to I/O accessory cards that a refresh operation is taking place. In addition, an ISA master can issue a REFRESH* signal to initiate a refresh cycle independent of the Integrated System Peripheral Controller's periodic refresh requests.

* indicates an active low signal.

Access Time

The main memory uses DRAM with a RAS access time of 80 nanoseconds and a cycle time of approximately 160 nanoseconds. (Hewlett-Packard assumes no responsibility for the use of memory modules not supplied by HP-approved vendors.)

Parity

The memory system generates the signal PARITY* for reads. Parity generation/check is done via 74AS286s, PALS, and latches under the control of the memory controller. PARITY* goes to the ISP. If there is a parity error, the ISP will cause an NMI. The same parity generation/check is used for 80486 and backplane accesses to memory.

Self-tests

Upon power-up, the i486 goes through a self-test of main memory. For a listing of BIOS error codes and their descriptions, refer to the *HP Vectra 486/33T Computer Configuration and Service Manual*.

Memory Map

Table 5-1 lists the memory map. As with industry-standard products, the first 1 MB of main memory is reserved for industry-standard conventional memory. The base RAM (either the first 512 or 640 KB of main memory, depending on how RAM is configured with setup software) is used for applications programs. The system reserves main memory from hex address 0A0000 to 0FFFFF, for video RAM, ROM on I/O accessory cards, option ROM, and BIOS ROM. Main memory from hex address 0100000 to 3FFFFF is reserved for 32-bit dynamic RAM expansion, using the sockets for the single in-line memory modules. Option ROMs are also mapped at hex address FFFE0000 to FFFEFFFF, and BIOS ROMs are also mapped at hex address FFFF0000 to FFFFFFFF.

Using configuration information in CMOS RAM, the power-up routine:

- identifies non-cacheable memory address ranges and supplies them to the hardware that disables the memory cache during accesses to non-cacheable addresses
- identifies RAM and ROM address ranges and configures the hardware that disables memory writes during accesses to the ROM addresses

Table 5-1. HP Vectra 486/33T Memory Map

Hex Address Range	Description of Memory Area	Cacheable or Noncacheable Area
0000 0000-0009 FFFF	Base RAM, 32-bit dynamic RAM) is either the first 512 KBytes or 640 KBytes of main memory depending on how RAM is configured with system setup software. Base RAM is reserved for industry-standard memory, and used for MS-DOS applications.	Cacheable
000A 0000-000B FFFF	Reserves 128 KB of main memory for video RAM.	Non-cacheable
000C 0000-000D FFFF	Reserves 128 KB of ROM or RAM on I/O expansion boards. Includes video BIOS.	Cacheable
000E 0000-000E FFFF	Reserves 64 KB of main memory for option ROM.	Cacheable
000F 0000-000F FFFF	Reserves 64 KB of main memory for BIOS ROM.	Cacheable
0010 0000-03FF FFFF	63 MBytes of extended memory space for 32-bit dynamic RAM which uses the SIMM sockets.	Cacheable
0400 0000-3FFF FFFF	1 GByte - 64 MBytes of address space.	Cacheable
4000 0000-BFFF FFFF	Expansion memory or memory-mapped I/O	Non-cacheable
C000 0000-C1FF FFFF	Weitek coprocessor (if installed)	Non-cacheable
C200 0000-FFFD FFFF	Expansion memory or memory-mapped	Non-cacheable
FFFE 0000-FFFE FFFF	64 KBytes of option ROM, upon power-up these bytes are automatically mapped here from 0E0000h to 0EFFFFh.	Non-cacheable
FFFF 0000-FFFF FFFF	64KB of BIOS ROM. Upon power-up these bytes are automatically mapped here from 0F0000 to 0FFFFF.	Non-cacheable

Shadow RAM

The HP Vectra 486/33T supports the use of Shadow RAM which allows ROM code to be copied into 32-bit system RAM at the same physical address for high-speed execution. Code from ROM is transferred to system RAM by the BIOS by performing the following steps:

1. System RAM located at the same physical address as the ROM code is enabled (this region is disabled upon RESET) and the write protect bit is turned off.
2. Data is read from ROM and written back to the same location in RAM. During the read cycle, ROMCS is activated and no DRAM memory cycles are performed. During the subsequent write cycle, ROMCS is disabled and a RAM memory cycle is executed to update the corresponding memory location.

Enabling host shadow for a region of memory means that CPU reads in that regions come from RAM, and that region is write protected. Disabling host shadow for a region means that writes to that region go to RAM (if not write protected) and reads are from ROM. Enabling bus master shadowing has the same results for masters other than the CPU.

Shadow RAM can be enabled programmatically (using either the EASY CONFIG utility, or directly programming I/O port FCA0 or FCA1) in blocks of 64 KB starting from 640 KB to 1024 KB. The regions supported for shadowing are indicated below.

Table 5-2. Blocks Available for Shadowing

Hex Address Range	Memory Block	Type of Memory	Shadowing Supported
0A0000-0AFFFF	640 KB - 704 KB	video RAM	NO
0B0000-0BFFFF	704 KB - 768 KB	video RAM	NO
0C0000-0CFFFF	768 KB - 832 KB	video ROM	YES *
0D0000-0DFFFF	832 KB - 896 KB	optional	NO
0E0000-0EFFFF	896 KB - 960 KB	option ROM	YES *
0F0000-0FFFFF	960 KB - 1024 KB	BIOS ROM	YES *

* See notes following Table 5-6.

The following table is a description of the Host Bus Shadowing port.

Table 5-3. Host Bus Shadowing Port (FCA0h) Description

Bit	Data	Description	Type
0	1	Enable RAM at 0A0000-0AFFFFh	R/W
	0	Disable	
1	1	Enable RAM at 0B0000-0BFFFFh	R/W
	0	Disable	
2	1	Enable host shadow 0C0000-0CFFFFh	R/W
	0	Disable	
3	1	Enable RAM at 0D0000-0DFFFFh	R/W
	0	Disable	
4	1	Enable host shadow 0E0000-0EFFFFh	R/W
	0	Disable	
5	1	Enable host shadow 0F0000-0FFFFFh	R/W
	0	Disable	
6	1	Enable backplane segment 0E0000h	R/W
	0	Disable	
7	1	512 - 640 KB enable	R/W
	0	Disable	

Notes:

1. Shadowing cannot be enabled on RAM which is remapped (see Memory Remapping).
2. The power-on value for bits 0-6 is '0'; bit 7 is '1'.

3. No shadowing (write protected RAM) is possible at segments 0A000h, 0B000h, and 0D000h. The RAM is either completely disabled or enabled for Read/Write. Remapping is still possible, but the appropriate segments (A/B, C/D) must first be disabled.

The following table is a description of the Bus Master Shadowing port.

Table 5-4. Bus Master Shadowing Port (FCA1h) Description

Bit	Data	Description	Type
0	1	Disable 0F0000-0FFFFFh	R/W
	0	Enable	
1	1	Disable 0E0000-0EFFFFh	R/W
	0	Enable	
2	1	Disable 0C0000-0CFFFFh	R/W
	0	Enable	
3	1	Disable 0C0000-0CFFFFh	R/W
	0	Enable	
4	1	Enable bus master shadow 0E0000-0EFFFFh	R/W
	0	Disable	
5	1	Enable bus master shadow 0F0000-0FFFFFh	R/W
	0	Disable	
6		Reserved	
7	1	ODD parity enable	R/W
	0	EVEN parity enable	

Notes:

1. When shadowing is disabled, READs come from ROM and WRITEs go to RAM. When ROM is shadowed, READs come from RAM and WRITEs are ignored. For example, when shadowing is enabled, RAM is write protected, therefore, 'write protect without shadowing' effectively disables the segment controlled by that bit. This avoids contention with expansion boards that may have on-board RAM.

   There are two bits per segment that should be taken into account when shadowing. For example, 2 bits that control segment 0F000h are interpreted as follows:

```
Port     Port
FCA0     FCA1
Bit 5    Bit 0    Description
-------------------------------------------------------------
  0        0        Read from ROM (0F000h) write to RAM (0F000h)
  0        1        Read from ROM, ignore writes
  1        0        Read from shadow RAM (0F000h), ignore writes
  1        1        Both BIOS ROMs and 0F000h shadow RAM are OFF
```

2. The power-on value for bits 0-5 is '0'. Bits 1 and 3 should be set high early on in POST to avoid contention with expansion boards that may have RAM in segments 0C000h and 0E000h.

3. Bit 7 is used during extended RAM soft error testing at the manufacturer, the power-on for this bit is '0'.

Memory Remapping

The memory region between 640 KB and 896KB (typically called the *reserved memory area*) can be remapped to reside on megabyte boundaries at the top of memory. Either the 128 KB (0A0000-0BFFFFh) or 256 KB (0A0000-0DFFFFh) regions of memory can be selectively remapped to the top of installed memory via software: either using the EASY CONFIG utility, or by writing the correct data directly to the I/O ports of the Memory Board (see table below). The top of memory can be programmed to accommodate memory expansion cards installed in the I/O backplane.

The following table is a description of the Memory Remap Enable port.

Table 5-5. Memory Remap Enable Port (FCA2h) Description

Bit	Data	Description	Type
0	1	Enable remap 0A0000-0BFFFFh to top of memory	R/W
	0	Disable	
1	1	Enable remap 0A0000-0DFFFFh to top of memory	R/W
	0	Disable	
2-3		Reserved	
4	1	Socket pair 0 populated	R
	0	Not populated	
5	1	Socket pair 1 populated	R
	0	Not populated	
6	1	Socket pair 2 populated	R
	0	Not populated	
7	1	Socket pair 3 populated	R
	0	Not populated	

Notes:

1. Memory which is remapped cannot be written to in its 'old' address. For example, if bit 0 it set, a write to address 0A0000h will not write any data into system memory; it will go to the backplane instead.
2. To remap segments 0A000h and 0B000h, bits 0 and 1 of port FCA0h must first be set to '0'. To remap segments 0C000h and 0D000h, bit 3 of port FCA0h must be set to '0' and bit 3 of port FCA1h must be set to '1'.
3. If bit 1 of this port is set to '1', bit 0 is a don't care (i.e., will remap 256 KB).
4. Power-on value for bits 0 and 1 is '0'.

SIMM Type

The memory can be configured with four types of memory. They are:

- 1MB Single Density (256K X 4) DRAMS
- 2MB Double Density (256K X 4) DRAMS
- 4MB Single Density (1M X 4) DRAMS
- 8MB Double Density (1M X 4) DRAMS

Single and double density SIMMs cannot be mixed.

The write of FCA3h (top of memory) is done by the BIOS and the result is used by the memory controller to handle memory mapping of the reserved areas.

Table 5-6. SIMM Type Port (FCA3h) Description (Read)

Bit	Data	Description	Type
0	1	Double density in socket 0	R
	0	Single density in socket 0	
1	1	4/8MB parts in socket 0	R
	0	1/2MB parts in socket 0	
2	1	Double density in socket 1	R
	0	Single density in socket 1	
3	1	4/8MB parts in socket 1	R
	0	1/2MB parts in socket 1	
4	1	Double density in socket 2	R
	0	Single density in socket 2	
5	1	4/8MB parts in socket 2	R
	0	1/2MB parts in socket 2	
6	1	Double density in socket 3	R
	0	Single density in socket 3	
7	1	4/8MB parts in socket 3	R
	0	1/2MB parts in socket 3	

Table 5-7. SIMM Type Port (FCA3h) Description (Write)

Bits (HA 25:20)	Memory Size (MBytes)	Bits (HA 25:20)	Memory Size (MBytes)
000010	2	011000	24
000100	4	011010	26
000110	6	011100	28
001000	8	100000	32
001010	10	100100	36
001100	12	101000	40
001110	14	110000	48
010000	16	110100	52
010010	18	000000	64
010100	20		

Notes:

1. Memory modules cannot be mixed and matched: all have to be either single- or double-density.
2. The Write is done by the BIOS and the result is used by the memory controller to handle memory mapping of the reserved area.
3. If double-density SIMMs are installed in bank 0, bits 0, 2, 4, and 6 of FCA3h will be "1". You must therefore read FCA2h if you wish to determine the amount of memory installed.

CPU Cycle and C800

Table 5-8. CPU Cycle and C800 (FCA4h) Description

Bit	Data	Description	Type
0	1	Delay write hits 1 CLK at 33 MHz	R/W
	0*	No delay	
1	1	Add 1 wait state to 1st, 3rd dw read	R/W
	0*	No wait state	
2	1*	Enable Write Buffers	R/W
	0	Disable Write Buffers	
3	1*	Enable Write Data Hold at 33 MHz	R/W
	0	Disable Write Data Hold	
4	1*	2 CLKs per each dw read	R/W
	0	No CLKs per each dw read	
5	1*	CPU speed = 33 MHz	R/W
	0	CPU speed = 25 MHz	
6	1	Reserved	
7	1	Disable shadowing in C800h region	R/W
	0*	Enable shadowing in C800h region	

* = default

Cache Memory

The system incorporates two forms of cache memory. The first is the cache as incorporated on the 80486 CPU. The second type is additional cache memory provided to supplement the CPU's internal cache.

80486 CPU Cache

The 80486's incorporates an on-chip cache that can store data most recently used from any cacheable memory to the 80486's on-chip cache of 8 KB of static RAM. This cache of code and data allows for faster access by the 80486. If the data in cacheable memory is changed as a result of a DMA or bus master transfer from an I/O device, the data in the cache is invalidated so that it will not be used.

The following table describes the cache enable port 0C03h. This port can be enabled programmatically (using the DOS EXMODE command, or directly programming the I/O port 0C03h).

Table 5-9. Cache Memory Enable Port (0C03h) Description

Bit	Data	Description	Type
0	1	Enable internal caching	R/W
	0	Disable	
1	1	Enable write to cache enable RAM	R/W
	0	Disable	
2	1	Enable external cache	R/W
	0	Disable	
3	1	Weitek coprocessor present	R
	0	Not present	
4-7		Reserved	

Notes:

1. When bit 1 is enabled, memory writes no longer go to memory. Instead, all bits except bit 0 are lost. Bit 0 goes to a *n*K x 1 SRAM, where the address in the SRAM is dependent on the address space being written to. Each bit enables or disables caching on a 16 KB boundary.
2. The power-on value for bits 0, 1 and 2 is '0'.

External Cache

The external cache system in the Hewlett-Packard Vectra 486/33T provides an additional 128K cache to supplement the internal cache of the 80486 CPU. As shown in the following figure, the internal cache in the CPU has a typical cache read hit rate of approximately 90%. The additional external cache provides an additional hit rate of about 80% of the remaining 10% missed by the internal cache of the CPU.

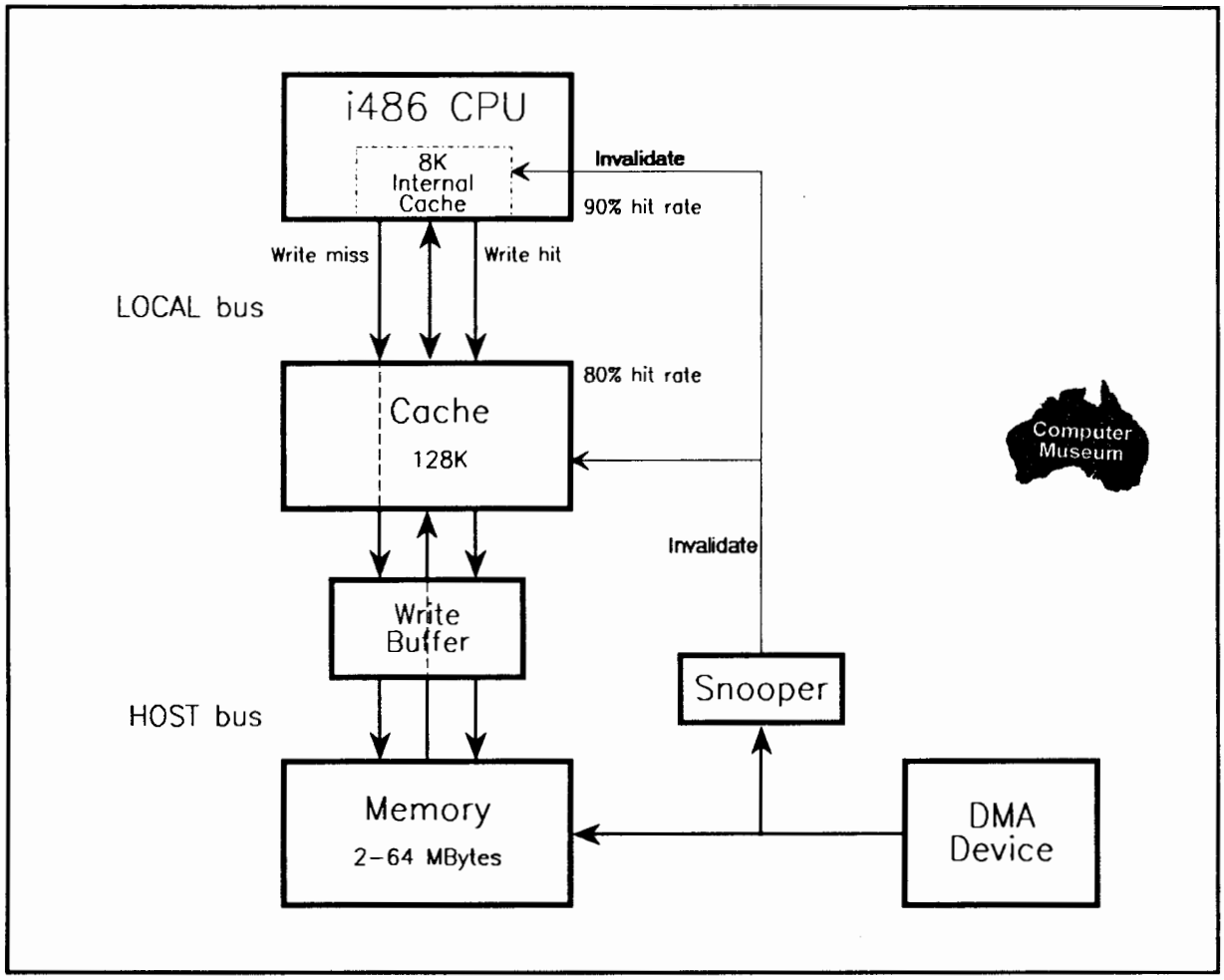


Figure 5-3. Cache System Block Diagram

When the CPU goes to an address to read data, it first searches the 8K internal cache. If it doesn't find the data there, it looks in the external 128K cache. If it fails to find it there, it then goes to memory and retrieves the data. The path of the data is through the 128K cache where an image of the data is stored, as well as through the 8K internal cache, where another copy of the data is stored. Since what is brought back is a block of data, and not just a single instruction, it is likely that subsequent data will also be brought back, and the CPU will find this in either it's internal cache or the external 128K cache. Since the cache memory is high-speed RAM, there are no wait-states involved and data access is faster than if the CPU had to access dynamic RAM.

When the CPU writes the data back to data, if the address it needs to write is not imaged in the cache, the data is passed on through the cache to the write buffer which then passes the data on to memory. If the data is imaged in the cache, it is written there and then also written by the write buffer to the memory.

A DMA device, such as a hard disk controller, may write data to memory, thus making the data in memory and the data in the cache different. A snooper circuit detects these writes to memory and if the address locations stored in the cache and the addresses altered by the DMA device are the same, the data in the cache is invalidated. This makes the CPU bypass the cache and go to memory the next time this address is accessed.

Non-cacheable areas and operations

Using the cache enable (KEN*) input on the 80486 tells the cache controller whether the current 80486 address is cacheable or not. This KEN* signal is driven by external logic.

Certain operations are non-cacheable, including the following:

I/O operations
Interrupt Acknowledge Cycles
Halt/Shutdown
Locked cycles
All memory addresses above 1 GB

Read-Only Memory (ROM)

For all models, the Processor Board provides sockets for two 32 KB option ROM chips and one 64 KB ROM chip (containing the BIOS firmware). All data coming from the BIOS ROMs or option ROMs go over the 16-bit (backplane) data bus.

6

Keyboard and Mouse

Introduction

This chapter discusses the keyboard and mouse components and circuitry.

Mouse/Keyboard/Serial Port Board

Two 50-pin input/output device connector boards provide an interface between the System Board and connectors for input/output devices. The Mouse/Keyboard/Serial Port Board has connectors for a keyboard, mouse, and serial device; the 8742 keyboard/mouse controller also resides on this board.

Figure 6-1 shows the pinouts for the Keyboard/Mouse/Serial board, and Table 6-1 gives its signal definitions (with the direction of the inputs and outputs in reference to the 8742 Keyboard Controller).

Note

For a description of the serial portion of the Mouse/Keyboard/Serial Port Board, refer to chapter 8.

Password Disable Switch

There is a password disable switch on the Mouse/Keyboard/Serial Port Board that allows a user that has set a password with the system's configuration utility EASY CONFIG, and has later forgotten it, to bypass the power-on password prompt. To disable a password, set the password bypass switch (on the board closest to the power supply) to the ON position as shown in figure 6-1. Disabling the password erases it from the computer's non-volatile CMOS memory. A new password can then be set using the EASY CONFIG utility.

For more about the password feature refer to your computer's setting up manual.

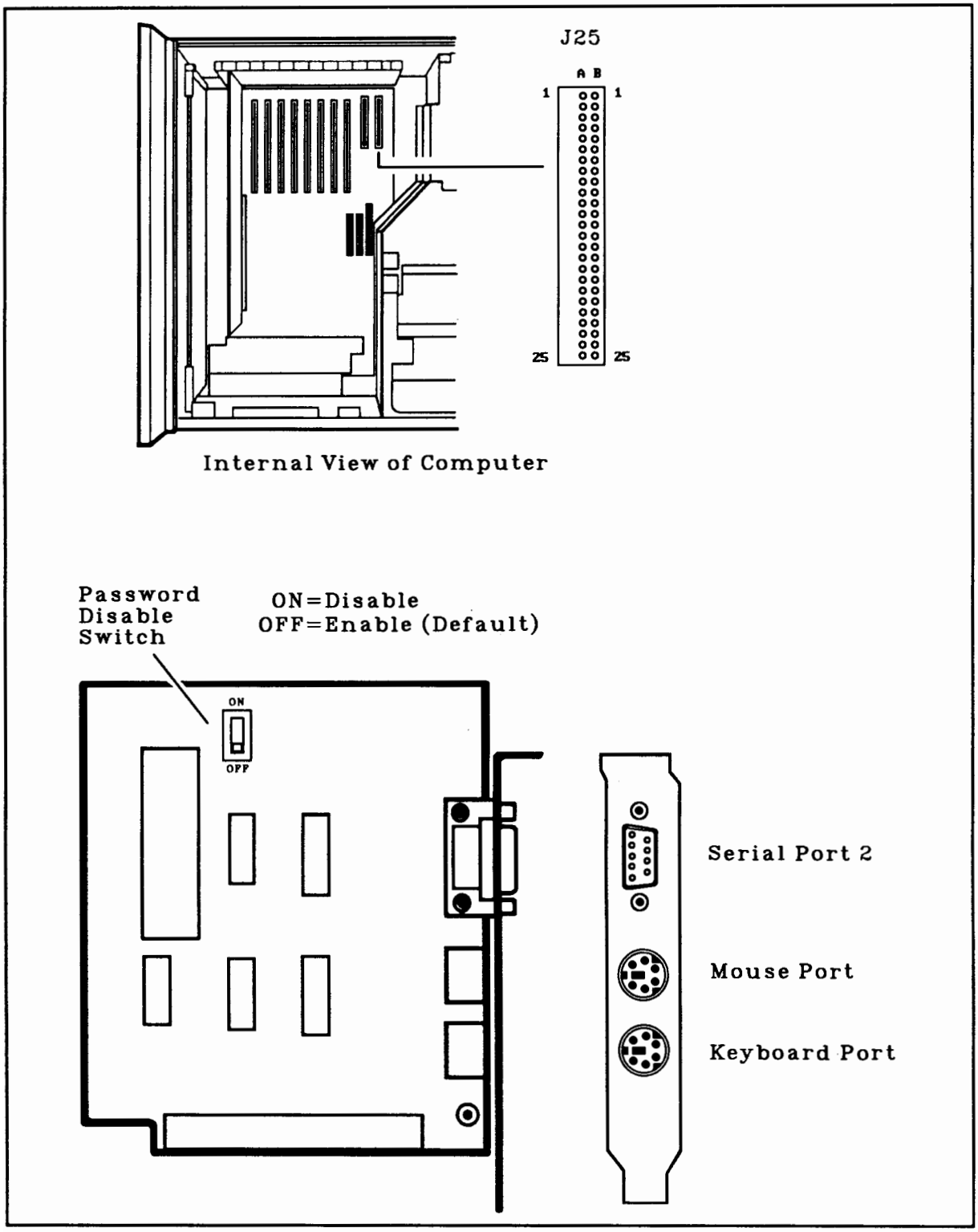


Figure 6-1. Mouse/Keyboard/Serial Port Board Connectors

Table 6-1. Mouse/Keyboard/Serial Port Board Signal Definitions[1]

Connector Pin	Direction	Signal Name	Signal Definition
A12	—	-12V	-12 Volts
B13	—	+12V	+12 Volts
A3	I	CTS1*	Clear to Send No. 1
A2	I	DSR1*	Data Set Ready No. 1
B2	O	DTR1*	Data Terminal Ready No. 1
A1, 6, 8, 14, 19, 22, 25	—	GND	Ground
B1, 5, 19, 22, 25	—	GND	Ground
A11	I	IORD*	I/O Read
A15	I/O	KBCLICK	Keyboard Click
A13	I/O	KBINH	Keyboard Inhibit
B15	I/O	KYA20	Keyboard A20
B18	I/O	KYBD_IRQ	Keyboard Interrupt Request
A18	I	KYCS*	Keyboard Chip Select
B14	I/O	KYRST*	Keyboard Reset
B11	I	KYWR*	Keyboard Write
B17	I/O	MOUSE_IRQ	Mouse Interrupt Request
A9	I	PWROK	Power OK
B6	I	RI1*	Ring Indicator No. 1
A5	I	RLSD1*	Received Line Signal Detect No. 1
B8	I	RTS1*	Request to Send No. 1
A7	I	RXD1*	Receive Data No. 1
B4	I	TXD1	Transmit Data No. 1
A4, 10, 16	—	VCC	+5 Volts
B3, 7, 16	—	VCC	+5 Volts
A17	I	XA <2>	X-Bus Address Line 2
A24	I/O	XD <0>	X-Bus Data Line 0
B24	I/O	XD <1>	X-Bus Data Line 1
A23	I/O	XD <2>	X-Bus Data Line 2
B23	I/O	XD <3>	X-Bus Data Line 3
A21	I/O	XD <4>	X-Bus Data Line 4
B21	I/O	XD <5>	X-Bus Data Line 5
A20	I/O	XD <6>	X-Bus Data Line 6
B20	I/O	XD <7>	X-Bus Data Line 7
B9, 10, 12	—	—	Not connected

Notes:

1. The direction of the signal inputs (I) and outputs (O) is in reference to the 8742 keyboard controller. An asterisk (*) indicates a signal is active low.

Keyboard and Mouse Interfaces

Below is a list of hardware components that make up the system interface circuit for the keyboard, mouse, and other input devices.

- HP Enhanced Vectra Keyboard (an industry-standard keyboard)
- HP Mini-DIN mouse (optional; Microsoft-compatible)
- Mini-DIN keyboard connector (on the Mouse/Keyboard/Serial Port Board)
- Mini-DIN mouse connector (on the Mouse/Keyboard/Serial Port Board)
- Intel 8742 keyboard and mouse controller (on the Mouse/Keyboard/Serial Port Board)
- 80486 microprocessor (on the System Board)
- System fan and keylock assembly power, and LED/speaker (on the System Board)

Figure 6-2 shows the software communication protocol for the system the keyboard.

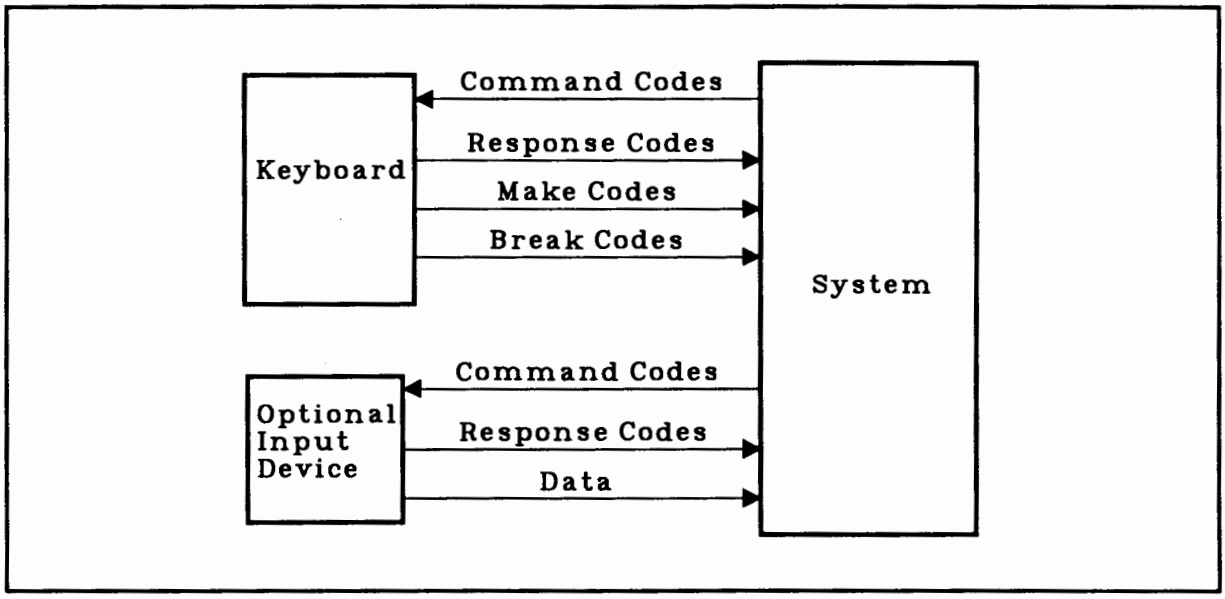


Figure 6-2. Keyboard/Input Device Communication Protocol (Software Interface)

The communications link between the system and the keyboard or other input devices is synchronous, serial, and bi-directional. Software communications consist of data (command and response codes) and clock information (make and break codes, for the keyboard only). Data are communicated by the system, the keyboard, the mouse, or other input devices, but only one of these items may control the data line at any time. The clock line is controlled by the 8742, the mouse, and the keyboard's firmware.

When a keyboard key is pressed or an input device is operated, the keyboard or input device respond by producing serial data ("response codes"). The keyboard, unlike other input devices also sends "make" and "break" codes in response to key strokes. These data are sent to the system in a serial data stream which has an industry-standard format. The data are latched into the system's 8742 controller, which instructs the 80486 microprocessor to read the data. In return, the system may send serial data ("command codes") to the keyboard or to the input device.

See the *HP Vectra 486/33T BIOS Technical Reference Manual* for a detailed description of the keyboard and mouse software interfaces, a more complete listing of scancodes, alternate scancodes, and the commands sent between (1) the keyboard or mouse and (2) the system.

Keyboard

The **keyboard** for the HP Vectra 486/33T is the industry-standard AT II keyboard, and has the same layout as the "HP Enhanced Vectra Keyboard" used for the HP Vectra RS Personal Computer. This keyboard includes:

- 101 keys (international versions have 102 keys)
- twelve function keys
- separate cursor control and numeric keypads
- mode keys (Print Screen, Scroll Lock, Pause, etc.)
- mode indicator LEDs (Num Lock, Caps Lock, Scroll Lock)

Note

The HP Vectra 486/33T also supports industry-standard international keyboards (with 102 keys), which are available through Hewlett-Packard dealers. The HP Vectra 486/33T does not support the "Vectra PC/HIL Keyboard," used by early versions of the HP Vectra PC.

Mouse

The **HP Mini-DIN Mouse** is a Microsoft-compatible mouse with a 2.5 meter (8.25 feet) cable. The mouse is an auxiliary input device that uses receives and sends information over the same serial data and clock lines used by the keyboard and the 8742 controller. When a mouse button is pressed, or the mouse is moved, it sends a code to the 8742. These codes are classified as *response* and *data* codes.

Keyboard and Mouse Connectors

Figure 6-3 shows the pin numbers for the keyboard cable's end that connects to the keyboard. To attach the keyboard and mouse to the system, the Mouse/Keyboard/Serial Port Board has one connector for the keyboard (J1) and one connector for the mouse (J2). Both connectors are the same type: a six-pin, industry-standard, mini-DIN connector as shown in Figure 6-4. (See Figure 6-1 to locate J1 and J2.)

The mini-DIN connector interface allows industry-standard keyboards and mice to be plugged into the system. The following gives the pin numbers for the System Processing Unit's (SPU's) connector for the keyboard cable, the corresponding keyboard connector signal name, a description of these signals, and the corresponding pin numbers for the keyboard cable's connector plug to the keyboard.

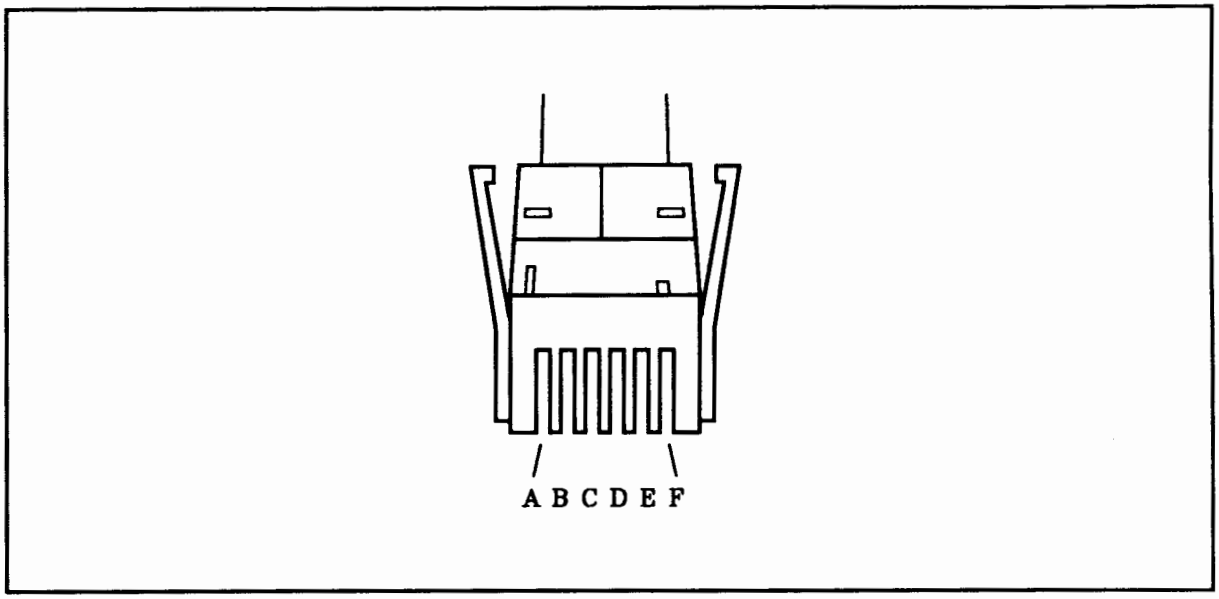


Figure 6-3. Keyboard Cable Connector to Keyboard

Table 6-2. Keyboard Cable Connector to Keyboard Signals

Cable Connector Pins	Signal Name	Signal Description
B	DATA	Data Signal
F	Not Used	Not Used
C	GND	Ground
E	PWR	Power (+ 5 Vdc)
D	CLK	Clock Signal
A	Not Used	Not Used

The following figure and table gives the pin numbers for the SPU's connector for the mouse cable, the corresponding mouse connector signal name, and a description of these signals.

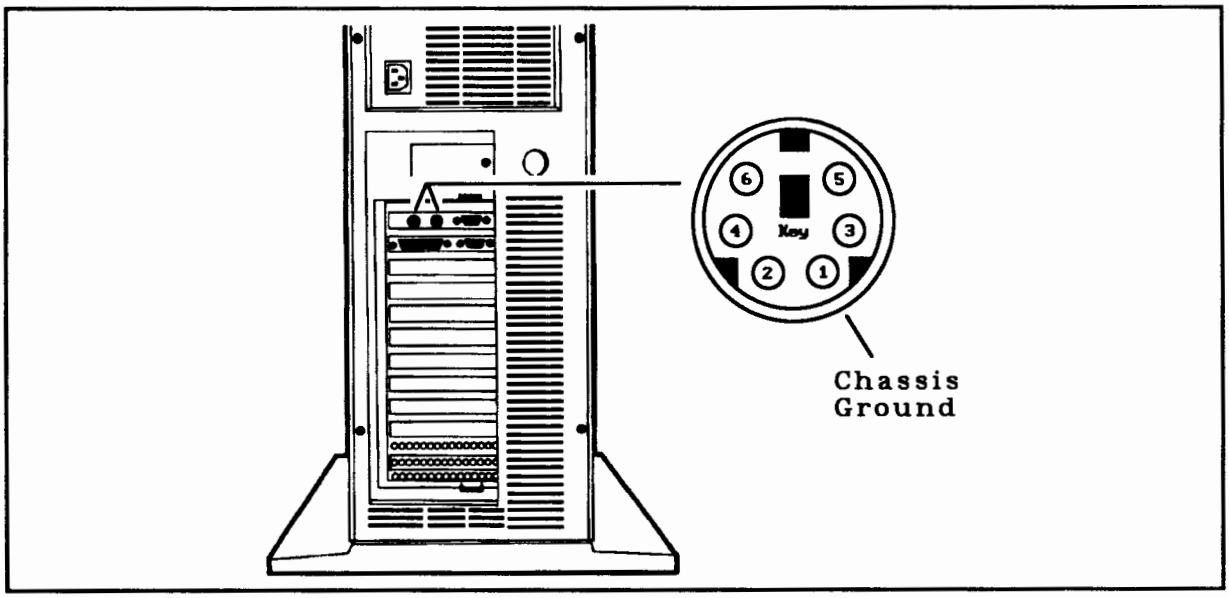


Figure 6-4. Keyboard and Mouse Mini-DIN Connectors

Table 6-3. Keyboard and Mouse Mini-DIN Connector Signals

Keyboard and Mouse Connector Pins *	Signal Name	Signal Description
1	DATA	Data Signal
2	Not Used	Not Used
3	GND	Ground
4	PWR	Power (+ 5 Vdc)
5	CLK	Clock Signal
6	Not Used	Not Used

* On Mouse/Keyboard/Serial Port Board.

8742 Keyboard/Mouse Controller

Figure 6-5 gives a functional block diagram of the interface circuit for the keyboard and mouse controller, an Intel 8742 microprocessor on Mouse/Keyboard/Serial Port Board. The 8742 is programmed to interface with the keyboard, the mouse (or other input devices), the computer's speaker, and the 80486 microprocessor.

For the keyboard, the 8742 uses hex scancodes to scan the keyboard interface for keys that have been pressed or released. Data from the keyboard are latched into the 8742, which instructs the 80486 microprocessor to read the data.

For the mouse, the 8742 uses hex scancodes to scan the mouse interface for mouse buttons that have been pressed or released. Similar to the 8742's keyboard operation, data from the mouse are latched into the 8742, which instructs the 80486 to read the data.

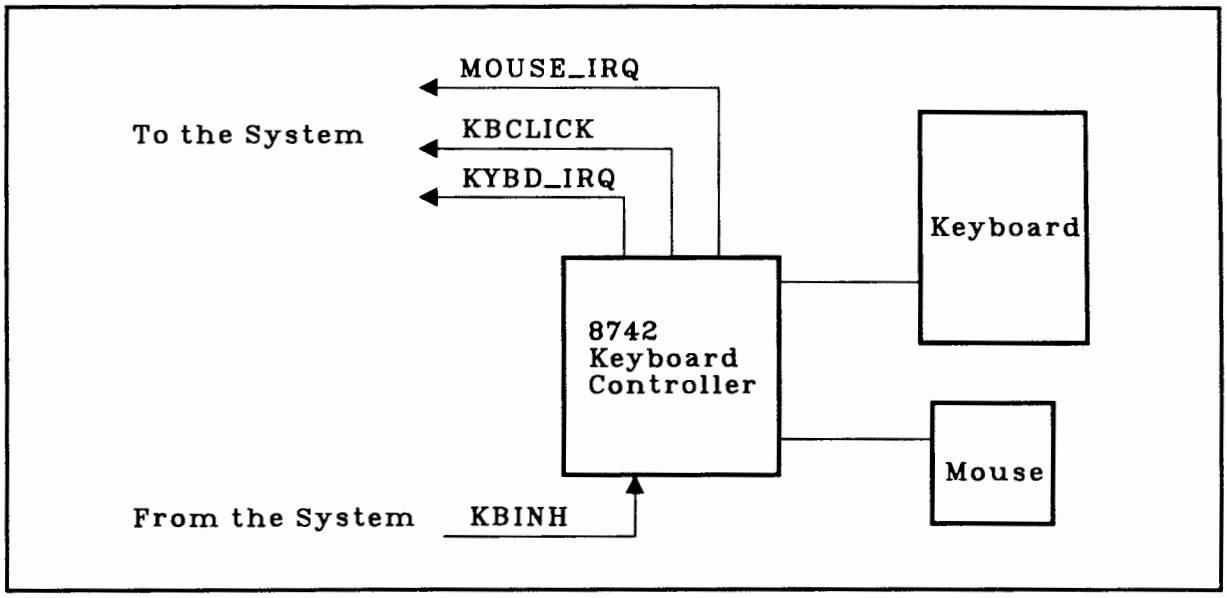


Figure 6-5. Block Diagram of Interface Circuit for Keyboard/Mouse Controller

8742 Keyboard/Mouse Controller Operation

The direct logic path from the 8742 microprocessor controller to the 80486 microprocessor consists of two I/O register and two interrupt lines. The first 80486 I/O register, at I/O address 60h, is used for keyboard/mouse commands and data, and for 8742 data. The second 80486 I/O register, at I/O address 64h, is used to write commands to, and read the status of, the 8742 controller. The 8742's output interrupt lines, either KYBD_IRQ or MOUSE_IRQ (enabled by the appropriate command byte), are set to interrupt the 80486 when the 8742 places data in port 60h.

The system is reset in DOS by pressing the Ctrl, Alt, and Del together, which causes the BIOS to jump to the reset address. The system is reset in an operating system such as OS/2 when the Ctrl, Alt, and Del combination tells OS/2 to send a reset command to the 8742. There is a "snooper circuit" which intercepts and processes this reset command and resets the CPU. The "snooper circuit" is designed to decrease the time spent waiting for the 8742 to signal command acceptance for such commands as system reset and state change of the microprocessor's A20GATE signal.

The 8742's KYA20 signal can be used to extend the 80486 microprocessor's real-address operating mode, which is otherwise limited to 1 Mbyte. When the 8742's KYA20 signal is programmed for a logical 1 (high), it is intercepted by the "snooper circuit" which processes the command and enables the 80486 microprocessor's address line A20GATE, extending the 80486's real-address operating mode.

For the keyboard, the 8742 microprocessor controller's clock and data signals enable it to send commands to, or receive status inputs from, the keyboard. The keyboard, in turn, sends its own clock and data signals when a key is pressed.

Each time the user presses or releases a key, the keyboard sends a corresponding 11-bit scancode to the 8742, in the format shown in the following table. The 8742 writes the scancode to its output buffer, causing a keyboard interrupt (if enabled) of the 80486 (IRQ1). The 80486 then reads the scancode from I/O address 60h.

Table 6-4. Keyboard Scancode Data Transmission Format

Bit	Definition
1	Start bit (always 1)
2	Data bit 0 (least-significant bit)
3	Data bit 1
4	Data bit 2
5	Data bit 3
6	Data bit 4
7	Data bit 5
8	Data bit 6
9	Data bit 7 (most-significant bit)
10	Parity bit * (0 or 1)
11	Stop bit (always 1)

* The eight data bits, plus the parity bit, always equal an odd number.

For the mouse, the 8742 microprocessor controller's clock and data signals enable it to send commands to, or receive status inputs from, the mouse. The mouse, in turn, sends its own clock and serial data signals when its button is pressed or the mouse is moved.

Each time the user presses or releases the mouse button, the mouse sends a corresponding 11-bit code to the 8742, in the format shown in the following table. By writing the code to its output buffer, the 8742 sends the code to the 80486 microprocessor, which causes a mouse interrupt of the 80486 (IRQ12) if enabled. The 80486 then reads the code from I/O address 60h.

Table 6-5. Mouse Data Transmission Format

Bit	Definition
1	Start bit (always 1)
2	Data bit 0 (least-significant bit)
3	Data bit 1
4	Data bit 2
5	Data bit 3
6	Data bit 4
7	Data bit 5
8	Data bit 6
9	Data bit 7 (most-significant bit)
10	Parity bit * (0 or 1)
11	Stop bit (always 1)

* The eight data bits, plus the parity bit, always equal an odd number.

8742 Interface with the Speaker

Each time the 8742 microprocessor controller receives a keyboard character from the keyboard, the 8742 sends a keyboard click signal (KBCLICK) to the computer's speaker. The keyboard click's volume control is implemented in the BIOS via a configuration option. The 8742 is given the value of the volume of the click at boot-up (the pulse length of the click depends on this value). The click is turned off and on by giving this value a zero. The keyboard beep function is also implemented in the BIOS.

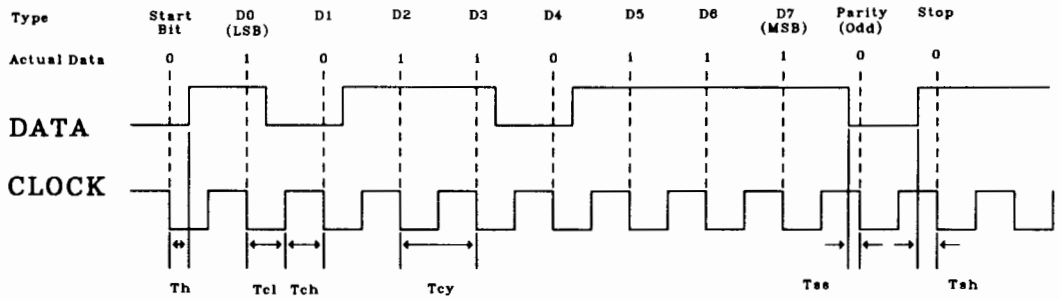
8742 Interface with Keyboard Inhibit Switch

The 8742 microprocessor controller monitors the keyboard inhibit (KBINH) input line, which comes from the keyboard inhibit keylock switch. When the user turns this SPU front-panel switch to the lock position, the KBINH signal is set to 0. In this case, data from the keyboard are still returned to the 8742, but not sent to the 80486 microprocessor.

Keyboard and Mouse Timing Diagrams

Figures 6-6 and 6-7 give timing diagrams for (1) the timing of the keyboard's data and clock transmissions and (2) the timing for the mouse's data and clock transmissions.

Tch = clock high (25us min., 45us max.)
Tcl = clock low (25us min., 35us max.)
Tcy = cycle time (60us min., 80us max.)
Th = data hold (0us min., 25us max.)
Tsh = stop bit hold (15us min., 25us max.)
Tss = stop bit setup (8us min., 20us max.)

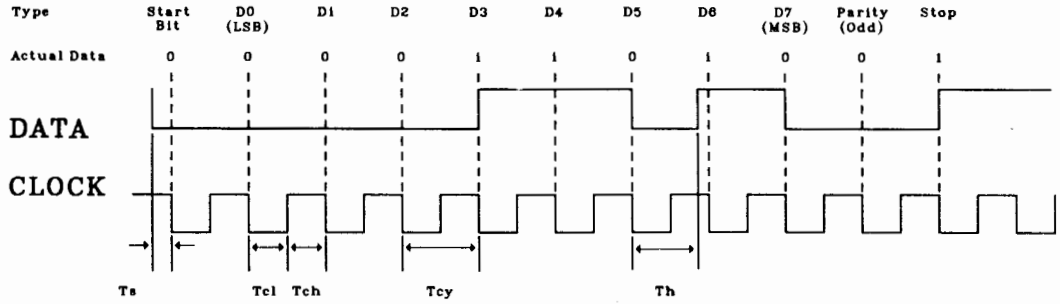


Notes:

- Code being sent in diagram is EDh.
- DATA is held low by the system as RTC signal.
- DATA is held low by the keyboard as Stop bit.
- CLOCK is held low by the system to inhibit keyboard.

Figure 6-6. Timing Diagram for System Signals to the Keyboard

Tch = clock high (30us min., 45us max.)
Tcl = clock low (30us min., 35us max.)
Tcy = cycle time (60us min., 80us max.)
Th = data hold (45 us min., 62us max.)
Ts = data setup (8us min., 14us max.)



Notes:

- Code being sent in diagram is 56h.
- CLOCK is held low by the system to inhibit keyboard.

Figure 6-7. Timing Diagram for Keyboard Signals to the System

7

Disk Drives

Introduction

This chapter discusses the flexible disk and hard disk subsystems. These flexible disk subsystem is designed for compatibility with industry standard disk formats. The System Board comes with an interface for embedded-AT hard disk drives (hard disk drives that have on-board controller subsystems). A separately mounted ESDI (Enhanced Small Disk Interface) hard disk controller subsystem board is also available. Both solutions are logically similar to the industry standard ST-506 interface.

Flexible Disk Controller (FDC) Subsystem

The WD37C65 Flexible Disk Controller (FDC) generates the signals which control the electromechanical parts of the drive, interprets the status signals received from the drive, and converts between parallel and serial data formats. The flexible disk controller subsystem supports a maximum of four internal flexible disk drives, or three flexible disk drives and one tape drive discussed in this chapter.

The FDC chip transfers data to and from the system memory through DMA (Channel 2). Commands to the FDC chip and status information from it are transferred by programmed I/O. Command termination and error conditions are signaled through interrupts (IRQ6). The FDC supports 250 kHz, 300 kHz, or 500 kHz data rates. The data rates are selected by the digital control port.

Flexible Disk Controller Configuration

The computer's configuration utility (EASY CONFIG) selects the location of the interface registers in the system I/O map. The I/O address can be either primary addressing, 3F0h through 3F7h (operational state-default) or secondary addressing, 370h through 377h.

System to FDC Interface

The FDC is designed for compatibility with the industry standard disk formats, both single and double density. It is designed around compatibility with the NEC uPD765A flexible disk controller.

The flexible disk controller subsystem interfaces with the microprocessor through six registers. There are three write registers and three read registers. Table 7-1 lists the registers and their primary and secondary I/O addresses.

Table 7-1. FDC Accessible Registers

Register	R/W	Primary/Secondary
Media Identification Register	R/W	3F1h (only)
Digital Output Register	W	3F2h/372h
FDC Status Register	R	3F4h/374h
FDC Data Register	R/W	3F5h/375h
Digital Control Register	W	3F7h/377h
Digital Input Register	R	3F7h/377h

Digital Output Register (Write only)

This register controls the disk drive spindle motors and selects the desired disk drive.

Table 7-2. Digital Output Register Definitions

Bit	Data	Definition
0	0	Drive select, Drive A (Drive 0)
	1	Drive select, Drive B (Drive 1)
1	—	Reserved
2	0	FDC reset, active low
3	1	Interrupt and DMA enable, active high
4	1	Drive 0 motor enable, active high turns on motor 5.25-inch drive
5	1	Drive 1 motor enable, active high turns on motor 5.25-inch drive
6	—	Reserved
7	0	Flexible disk type ID, 3.5-inch drive
	1	Flexible disk type ID, 5.25-inch drive

FDC Status Register (Read only)

This is an 8-bit main status register. This register contains the status information to facilitate the transfer of data between the host and the FDC, and may be accessed at any time.

Table 7-3. FDC Status Register Definitions

Bit	Data	Definition
0	1	Disk drive A Busy (DBA) in seek mode.
1	1	Disk drive B busy (DBB) in seek mode.
2-3	—	Reserved
4	1	Disk controller is busy (CB) read/write being executed.
5	1	Disk controller is in non-DMA mode (NDM).
6	1	Data Input/Output (DIO). Data transfer from the data register of the FDC to the host.
	0	Data transfer from the host to the data register of the FDC.
7	1	Request for master (RQM) indicates data register is ready to send or receive data to or from the host.
	0	Data register is not ready for data transfer to and from host.

FDC Data Register (Read/write)

This is a bi-directional register. 8-bit data is written to (write function), or read from (read function), the FDC as commands via programmed I/O and write data during disk transfers via DMA.

This register (which actually consists of several registers in a stack with only one register presented to the data bus at a time) stores data, command, parameters and flexible disk drive status information. Data bytes are read out of, or written into, the data register in order to program or obtain the results after a particular command.

Digital Control Register (Write only)

The two LSBs of the digital control register select the desired data rate. Clock switchover is "deglitched", allowing continuous operation after changing the data rate.

Table 7-4. Digital Control Register Definitions

Bit	Definition
0, 1	Least significant bit in rate select code of PLL Bit Data Rate 1 0 (kHz) --- 0 0 500 0 1 300 1 0 250 1 1 Unused
2-7	Reserved

Digital Input Register (Read only)

This register is shared with the hard disk controller—bits 0-6 are used in the hard disk controller subsystem. The flexible disk controller uses the following information:

Table 7-5. Digital Input Register Definitions

Bit	Data	Definition
0-6		Used by the hard disk subsystem
7	1	State of the disk change line. Becomes active when the drive door is opened and at power-on. It will remain active until reset when a step pulse is issued. It is active high.

Operations and Commands

The FDC subsystem is capable of performing 15 different commands. Each command is initiated by a multibyte transfer from the host. The results after execution of the command may also be a multibyte transfer back to the host. The commands consists of three phases: Command Phase, Execution Phase, and the Result Phase.

- Command Phase - The FDC receives all information required to perform a particular operation from the host
- Execution Phase - The FDC performs the operation is was instructed to do.
- Result Phase - After completion of the operation, status and other housekeeping information are made available to the host.

The commands and required parameter and their results are in the following tables. Most commands require nine command bytes and return seven bytes during the result phase. The "W" to the left of each byte indicates a command phase byte to be written. An "R" indicates a result byte to be read by host.

The bytes of data which are sent to the FDC to form the command phase and are read out of the FDC in the result phase must occur in the order shown in the following command tables. This is the command code must be sent first and the other bytes sent in the prescribed sequence. No foreshortening of the command or result phases is allowed. After the last byte of data in the command phase is sent to the FDC, the execution phase automatically starts. In a similar fashion, when the last byte of data is read out in the result phase, the command is automatically ended and the FDC is ready for a new command.

It is important to note that during the result phase all bytes shown in the following tables must be read. The READ DATA command, for example, has seven bytes of data in the result phase. All seven bytes must be read in order to successfully complete the READ DATA command. The FDC chip will not accept a new command until all seven bytes have been read. Other commands may require fewer bytes to be read during the result phase.

The FDC commands are listed on the following pages. Definitions for the command symbols are listed in the table immediately following the commands.

1. READ DATA

PHASE	R/W	D7	D6	D5	D4	D3	D2	D1	D0	REMARK
COMMAND	W	MT	MF	SK	0	0	1	1	0	Command Code
	W	X	X	X	X	X	HS	US1	US0	
	W	<---------- C ---------->								Sector ID info. prior
	W	<---------- H ---------->								to command execution.
	W	<---------- R ---------->								4 bytes are compared
	W	<---------- N ---------->								against header on disk
	W	<---------- EOT ---------->								
	W	<---------- GPL ---------->								
	W	<---------- DTL---------->								
EXECUTION										Data transfer between FDD (Flexible Disk Drive) and host.
RESULTS	R	<---------- ST0 ---------->								Status information
	R	<---------- ST1 ---------->								after the command
	R	<---------- ST2 ---------->								execution.
	R	<---------- C ---------->								Sector ID information
	R	<---------- H ---------->								after the command
	R	<---------- R ---------->								execution.
	R	<---------- N ---------->								

2. READ DELETED DATA

PHASE	R/W	D7	D6	D5	D4	D3	D2	D1	D0	REMARK
COMMAND	W	MT	MF	SK	0	1	1	0	0	Command Code
	W	X	X	X	X	X	HS	US1	US0	
	W	<---------- C ---------->								Sector ID info. prior
	W	<---------- H ---------->								to command execution.
	W	<---------- R ---------->								4 bytes are compared
	W	<---------- N ---------->								against header on disk
	W	<---------- EOT ---------->								
	W	<---------- GPL ---------->								
	W	<---------- DTL---------->								
EXECUTION										Data transfer between FDD and host.
RESULTS	R	<---------- ST0 ---------->								Status information
	R	<---------- ST1 ---------->								after the command
	R	<---------- ST2 ---------->								execution.
	R	<---------- C ---------->								Sector ID information
	R	<---------- H ---------->								after the command
	R	<---------- R ---------->								execution.
	R	<---------- N ---------->								

3. WRITE DATA

PHASE	R/W	D7	D6	D5	D4	D3	D2	D1	D0	REMARK
COMMAND	W	MT	MF	0	0	0	1	0	1	Command Code
	W	X	X	X	X	X	HS	US1	US0	
	W	<---------- C ---------->								Sector ID info. prior
	W	<---------- H ---------->								to command execution.
	W	<---------- R ---------->								4 bytes are compared
	W	<---------- N ---------->								against header on disk
	W	<---------- EOT ---------->								
	W	<---------- GPL ---------->								
	W	<---------- DTL---------->								
EXECUTION										Data transfer between FDD and host.
RESULTS	R	<---------- ST0 ---------->								Status information
	R	<---------- ST1 ---------->								after the command
	R	<---------- ST2 ---------->								execution.
	R	<---------- C ---------->								Sector ID information
	R	<---------- H ---------->								after the command
	R	<---------- R ---------->								execution.
	R	<---------- N ---------->								

4. WRITE DELETED DATA

PHASE	R/W	D7	D6	D5	D4	D3	D2	D1	D0	REMARK
COMMAND	W	MT	MF	0	0	1	0	0	1	Command Code
	W	X	X	X	X	X	HS	US1	US0	
	W	<---------- C ---------->								Sector ID info. prior
	W	<---------- H ---------->								to command execution.
	W	<---------- R ---------->								4 bytes are compared
	W	<---------- N ---------->								against header on disk
	W	<---------- EOT ---------->								
	W	<---------- GPL ---------->								
	W	<---------- DTL---------->								
EXECUTION										Data transfer between FDD and host.
RESULTS	R	<---------- ST0 ---------->								Status information
	R	<---------- ST1 ---------->								after the command
	R	<---------- ST2 ---------->								execution.
	R	<---------- C ---------->								Sector ID information
	R	<---------- H ---------->								after the command
	R	<---------- R ---------->								execution.
	R	<---------- N ---------->								

5. READ A TRACK

PHASE	R/W	D7	D6	D5	D4	D3	D2	D1	D0	REMARK
COMMAND	W	0	MF	SK	0	0	0	1	0	Command Code
	W	X	X	X	X	X	HS	US1	US0	
	W	<---------- C ---------->								Sector ID info. prior
	W	<---------- H ---------->								to command execution.
	W	<---------- R ---------->								
	W	<---------- N ---------->								
	W	<---------- EOT ---------->								
	W	<---------- GPL ---------->								
	W	<---------- DTL---------->								
EXECUTION										Data transfer bet. FDD and host. FDD reads all data fields from index hole to EOT
RESULTS	R	<---------- ST0 ---------->								Status information
	R	<---------- ST1 ---------->								after the command
	R	<---------- ST2 ---------->								execution.
	R	<---------- C ---------->								Sector ID information
	R	<---------- H ---------->								after the command
	R	<---------- R ---------->								execution.
	R	<---------- N ---------->								

6. READ ID

PHASE	R/W	D7	D6	D5	D4	D3	D2	D1	D0	REMARK
COMMAND	W	0	MF	0	0	1	0	1	0	Command Code
	W	X	X	X	X	X	HS	US1	US0	
EXECUTION										The 1st correct ID info. on the cylinder is stored in data reg.
RESULTS	R	<---------- ST0 ---------->								Status information
	R	<---------- ST1 ---------->								after the command
	R	<---------- ST2 ---------->								execution.
	R	<---------- C ---------->								Sector ID information
	R	<---------- H ---------->								read during execution
	R	<---------- R ---------->								phase from disk.
	R	<---------- N ---------->								

7. FORMAT A TRACK

PHASE	R/W	D7	D6	D5	D4	D3	D2	D1	D0	REMARK
COMMAND	W	0	MF	0	0	1	1	0	1	Command Code
	W	X	X	X	X	X	HS	US1	US0	
	W	<---------- N ---------->								Bytes/Sector
	W	<---------- SC ---------->								Sectors/Track
	W	<---------- GPL ---------->								Gap 3
	W	<---------- D ---------->								Filler Byte
EXECUTION										FDC formats an entire track.
RESULTS	R	<---------- ST0 ---------->								Status information
	R	<---------- ST1 ---------->								after the command
	R	<---------- ST2 ---------->								execution.
	R	<---------- C ---------->								In this case, the ID
	R	<---------- H ---------->								info has no meaning.
	R	<---------- R ---------->								
	R	<---------- N ---------->								

8. SCAN EQUAL

PHASE	R/W	D7	D6	D5	D4	D3	D2	D1	D0	REMARK
COMMAND	W	MT	MF	SK	1	0	0	0	1	Command Code
	W	X	X	X	X	X	HS	US1	US0	
	W	<---------- C ---------->								Sector ID info. prior
	W	<---------- H ---------->								to command execution.
	W	<---------- R ---------->								
	W	<---------- N ---------->								
	W	<---------- EOT ---------->								
	W	<---------- GPL ---------->								
	W	<---------- STP---------->								
EXECUTION										Data compared between FDD and host.
RESULTS	R	<---------- ST0 ---------->								Status information
	R	<---------- ST1 ---------->								after the command
	R	<---------- ST2 ---------->								execution.
	R	<---------- C ---------->								Sector ID information
	R	<---------- H ---------->								after the command
	R	<---------- R ---------->								execution.
	R	<---------- N ---------->								

9. SCAN LOW OR EQUAL

PHASE	R/W	D7	D6	D5	D4	D3	D2	D1	D0	REMARK
COMMAND	W	MT	MF	SK	1	1	0	0	1	Command Code
	W	X	X	X	X	X	HS	US1	US0	
	W	<---------- C ---------->								Sector ID info. prior
	W	<---------- H ---------->								to command execution.
	W	<---------- R ---------->								
	W	<---------- N ---------->								
	W	<---------- EOT ---------->								
	W	<---------- GPL ---------->								
	W	<---------- STP---------->								
EXECUTION										Data compared between FDD and host.
RESULTS	R	<---------- ST0 ---------->								Status information
	R	<---------- ST1 ---------->								after the command
	R	<---------- ST2 ---------->								execution.
	R	<---------- C ---------->								Sector ID information
	R	<---------- H ---------->								after the command
	R	<---------- R ---------->								execution.
	R	<---------- N ---------->								

10. SCAN HIGH OR EQUAL

PHASE	R/W	D7	D6	D5	D4	D3	D2	D1	D0	REMARK
COMMAND	W	MT	MF	SK	1	1	1	0	1	Command Code
	W	X	X	X	X	X	HS	US1	US0	
	W	<---------- C ---------->								Sector ID info. prior
	W	<---------- H ---------->								to command execution.
	W	<---------- R ---------->								
	W	<---------- N ---------->								
	W	<---------- EOT ---------->								
	W	<---------- GPL ---------->								
	W	<---------- STP---------->								
EXECUTION										Data compared between FDD and host.
RESULTS	R	<---------- ST0 ---------->								Status information
	R	<---------- ST1 ---------->								after the command
	R	<---------- ST2 ---------->								execution.
	R	<---------- C ---------->								Sector ID information
	R	<---------- H ---------->								after the command
	R	<---------- R ---------->								execution.
	R	<---------- N ---------->								

11. RECALIBRATE

PHASE	R/W	D7	D6	D5	D4	D3	D2	D1	D0	REMARK
COMMAND	W	0	0	0	0	0	1	1	1	Command Code
	W	X	X	X	X	X	0	US1	US0	
EXECUTION										Head retrack to Track 0.

12. SENSE INTERRUPT STATUS

PHASE	R/W	D7	D6	D5	D4	D3	D2	D1	D0	REMARK
COMMAND	W	0	0	0	0	1	0	0	0	Command Code
RESULTS	R	<--------- STO ---------->								Status info. about the FDC at the end of seek operation.
	R	<--------- PCN ---------->								

13. SPECIFY

PHASE	R/W	D7	D6	D5	D4	D3	D2	D1	D0	REMARK
COMMAND	W	0	0	0	0	0	0	1	1	Command Code
	W	<--- SRT ---->				<--- HUT ---->				
	W	<--------- HLT ---------->							ND	

14. SENSE DRIVE STATUS

PHASE	R/W	D7	D6	D5	D4	D3	D2	D1	D0	REMARK
COMMAND	W	0	0	0	0	0	1	0	0	Command Code
	W	X	X	X	X	X	HS	US1	US0	
RESULTS	R	<--------- ST3 ---------->								Status info about the FDC.

15. SEEK

PHASE	R/W	D7	D6	D5	D4	D3	D2	D1	D0	REMARK
COMMAND	W	0	0	0	0	1	1	1	1	Command Code
	W	X	X	X	X	X	HS	US1	US0	
	W	<--------- NCN ---------->								
EXECUTION										Head is positioned over proper cylinder on the disk.

Table 7-6. Command Symbol Descriptions

Symbol	Description
C	Cylinder number; the current/selected cylinder (track) numbers, 0 through 255 of the medium.
D	Data pattern which is going to be written into a sector.
D7-D0	8 bit data bus, D7 is the MSB and D0 is the LSB.
DTL	Data Length; when N is defined as 00, DTL is which users are going to read out or write into the sector.
EOT	End of Track (the final sector number on a cylinder). During read or write operations, FDC will stop data transfer after a sector number equal to EOT.
GPL	Length of Gap 3. During the FORMAT Command it determines the size of Gap 3.
H	Head number 0 or 1, as specified in the ID field.
HLT	Head Load Time of drive (2 to 254ms in 2ms increments).
HS	Head Select, either 0 or 1.
HUT	Head Unload Time after a Read or Write operations has occurred (16 to 240ms in 16ms increments).
MF	If MF is low ("0") FM mode is selected, and if it is high, MFM mode is selected.
MT	If MT is high ("1"), a multitrack operation is performed. If MT=1 after finishing Read/Write operation on disk side 0, FDC will automatically start searching for sector 1 on disk side 1.
N	Number of data bytes written in a sector.
NCN	New Cylinder Number which is going to be reached as a result of the SEEK operation. Desired position of head.
ND	It is the Non-DMA mode operation.
PCN	The cylinder number at he completion of the SENSE INTERRUPT STATUS command. Position of head at present time.
R	The sector number which will be read or written.
R/W	Read or Write signal.
SC	The number of sectors per cylinder.
SK	Skip deleted data address mark.
SRT	Stepping rate of the drive (1 to 16ms in 1ms increments). It applies to all drives. In 2's complement format, i.e. F(Hex)=1ms, E(Hex)=2ms, etc.
STP	During the SCAN operation, if STP=1, the data in contiguous sectors is compared byte by byte with data send from the host via DMA operation, if STP=2, then alternate sectors are read and compared.

Table 7-6. Command Symbol Descriptions (continued)

Symbol	Description
US0-US1	Selected drive; binary encoded, 1 of 4.
ST0-ST3	The four registers which store the STATUS information after a command has been executed. This information is available during the result phase after command execution. These register should not be confused with the main status register These registers may be read only after a command has been executed and contains information relevant to that particular command. (See next page.)

Note that the FDC contains five status registers. The main status register may be read by the host at any time. The other four status registers (ST0, ST1, ST2 and ST3) are available only during the result phase and may be read only after completing a command. The particular command that has been executed determines how many of the status registers will be read.

Tables 7-7 through 7-10 are the bit definitions for the 4 command status registers (ST0-ST3).

Table 7-7. Status Information Register ST0

Bit	Name	Definition
0-1	US0, US1	UNIT SELECT 0, UNIT SELECT 1 - Binary encoded to indicate the drive's unit number at interrupt. Bit 1 Bit 0 Drive Selected 0 0 A 0 1 B 1 0 Unused 1 1 Unused
2	HS	HEAD SELECT - Indicate the state of the head at interrupt.
3	NR	NOT READY - This bit is set, when the disk drive is in the not ready state and a READ or WRITE command is issued.
4	EC	EQUIPMENT CHECK - This bit is set, if a fault signal is received from the flexible disk drive or if the track 0 signal fails to occur after 255 step pulses (Recalibrate command).
5	SE	SEEK END - This bit is set when the FDC completes the SEEK command.
6-7	IC	INTERRUPT CODE - Encoded in binary for the interrupt code: Bit 7 Bit 6 Definition 0 0 Normal termination of command. Command was completed and properly executed. 0 1 Abnormal termination of command. Execution of command was started but was not successfully completed. 1 0 Invalid command issued. Command which was issued was never started. 1 1 Abnormal termination because during command execution the ready signal from disk drive changed state.

Table 7-8. Status Information Register ST1

Bit	Name	Definition
0	MA	MISSING ADDRESS MARK - This bit is set if: ■ the FDC cannot detect the address mark after encountering the index hole twice. or, ■ the FDC cannot detect the data address mark or deleted data address mark. Also at the same time status register bit 2 will be set to 1.
1	NW	NOT WRITABLE - During execution of Write Data, Write Deleted Data or Format A Cylinder command, if the FDC detects a write protect signal from the flexible disk drive, this bit is set to 1.
2	ND	NO DATA - This bit is set, if: ■ The FDC cannot find the sector specified in the internal data register, during execution of Read Data, Write Deleted Data or Scan command. ■ The FDC cannot read the ID field without an error, during execution of Read ID command. ■ The starting sector cannot be found during execution of the Read A Cylinder command.
3	—	Not used, this bit is always set to 0
4	OR	OVERRUN - This bit is set to 1 if the FDC is not serviced by the host system during the data transfers within a certain time interval.
5	DE	DATA ERROR - This bit is set when the FDC detects a CRC error in either the ID field or the data field.
6	—	Not used, this bit is always set to 0.
7	EN	END OF CYLINDER - This bit is set to 1 when the FDC tries to access a sector beyond the final sector of a cylinder.

Table 7-9. Status Information Register ST2

Bit	Name	Definition
0	MD	MISSING ADDRESS MARK IN DATA FIELD - This bit is set if the FDC cannot find a Data Address mark or Deleted Data Address Mark when data is read from the medium.
1	BC	BAD CYLINDER - This bit is related to the ND bit. This bit is set when the contents of C on the medium is different from that stored in the internal data register and the contents of C is FFh.
2	SN	SCAN NOT SATISFIED - This bit is set to 1 if the FDC cannot find a sector on the cylinder which meets the condition during execution of the SCAN command.
3	SH	SCAN EQUAL HIT - This bit is set to 1 if the condition of "equal" is satisfied, during execution of the Scan command.
4	WC	WRONG CYLINDER - This bit is related to the ND bit. This bit is set to 1 when the contents of C on the medium are deferent from that stored in the internal data register.
5	DD	DATA ERROR IN DATA FIELD - This bit is set to 1 if FDC detects a CRC error in the data field.
6	CM	CONTROL MARK - This bit is set to 1 if the FDC encounters a sector which contain a Deleted Data Address Mark during execution of the Read Data or Scan command.
7	—	Not used, this bit is always set to 0.

Table 7-10. Status Information Register ST3

Bit	Name	Definition
0	US0	UNIT SELECT 0 - This bit is used to indicate the status of the Unit Select 0 signal to the flexible disk drive.
1	US1	UNIT SELECT 1 - This bit is used to indicate the status of the Unit Select 1 signal to the flexible disk drive.
2	HD	HEAD ADDRESS/SELECT - This bit is used to indicate the status of the Side Select signal of the flexible disk drive.
3	WP	WRITE PROTECTED - This bit is used to indicate the status of the WRITE PROTECT signal from the flexible disk drive.
4	T0	TRACK 0 - This bit is used to indicate the status of the Track 0 signal from the flexible disk drive.
5	RY	READY - This bit is always be a logic 1. The drive is presumed to be ready.
6	WP	WRITE PROTECTED - This bit is used to indicate the status of the WRITE PROTECT signal from the flexible disk drive.
7	0	Not used, this bit is always set to 0.

FDC to Disk Drive Interface

The flexible disk drive interface with the controller is through control signals, status signals and data signals. Flexible disk drives are daisy-chained; two drive connectors on each flexible disk drive cable. There are two system interface connectors (J46 and J42) on the System Board for flexible disk drive cables which means that up to four flexible disk devices (flexible disk drives and an internal tape drive) are supported. Figure 7-1 shows the pin assignments for the two 34-pin connectors. The input or output direction of the signals are referenced from the System Board.

In the HP Vectra system the HP 45812A 1.2 MB 5.25-inch Internal Flexible Disk Drive (standard) and the HP D1667A 1.44 MB 3.5-inch Internal Flexible Disk Drive (optional) provide a DISK CHG* signal on the drive interface pin 34. This signal indicates the detection of the drive door being opened. The HP 45811A 360 KB Internal Flexible Disk Drive does not provide a DISK CHG* signal on pin 34. Thus, this signal will always be inactive and the system cannot detect a possible media change via hardware. (Refer to the *HP Vectra 486/33T BIOS Technical Reference Manual* for the appropriate techniques to accomplished this detection.)

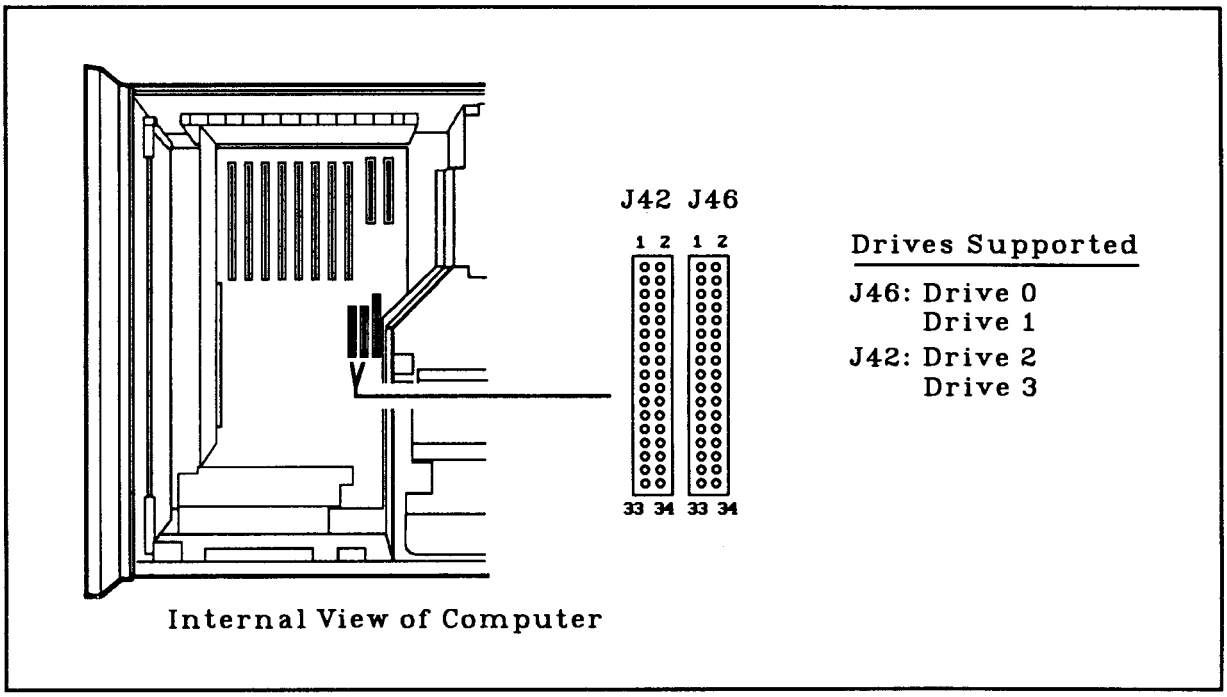


Figure 7-1. Flexible Disk Drive Interface Connectors

Table 7-11 shows signal descriptions for the interface between the controller and the supported flexible disk drives.

Table 7-11. Interface Connector Signal Definitions

Pin	Direction	Signal	Description
Odd pins (1-33)	—	Ground. (Pin 5 is not connected.)	
2	O	LDENSEL*	Selects read channel filters and write current on the 1.2 MB drive and the 1.44 MB drive. For the 1.2 MB drive, it is active only when the PLL is in 300 kHz mode and inactive for the 500 kHz and 250 kHz modes. For the 1.44 MB drive (in EISA mode) this signal is active when the PLL is in 250 kHz mode and inactive when in the 500 kHz mode.
4	I	MEDIA-ID	Identifies flexible disk drive type (1.2 MB, 360 KB, 1.44 MB).
6	—	Not connected.	
8	I	INDEX*	The selected drive supplies one pulse per disk revolution on this line.
10,16	O	MOTEN1*, MOTEN2* or MOTEN3*, MOTEN4*	MOTEN* controls the spindle motor of the drive (1 or 2 on connector J46, and 3 or 4 on connector J42) associated with each line. The line controls the spindle motor such that it starts when the line becomes active and stops when the line becomes inactive.
12,14	O	DRSEL2*, DRSEL1* or DRSEL4*, DRSEL3*	Drive select lines 1 or 2 (connector J46), and drive select lines 3 or 4 (connector J42) are used by drives 1, 2, 3, and 4 to degate all adapter and receiver drivers from the attachment, except MOTEN*, when the line associated with a drive is inactive.
18	O	DIRECTION*	Active level indicates the read/write head to move toward the spindle. Inactive level instructs the heads to move away from the spindle.
20	O	STEP*	The selected drive moves the read/write head one cylinder in or out per the direction line for each pulse present on this line. Motion is started each time the signal changes from an active to inactive level.
22	O	WRITE DATA*	For each inactive to active transition of this line while write enable is active, the selected drive causes a flux change to be stored on the disk.
24	O	WRITE ENABLE*	Disables write current in the drive head unless this line is active.
26	I	TRACK 00*	The selected drive makes this line active if the read/write head is over track 0.

Table 7-11. Interface Connector Signal Definitions (continued)

Pin	Direction	Signal	Description
28	I	WRITE PROTECT*	The selected drive makes this line active if a write-protected disk is mounted in the drive.
30	I	READ DATA*	The selected drive supplies a pulse on the line for each flux change encountered on the disk.
32	O	HDSEL 1*	Head 1 will be selected when this line is active. An inactive level will select the head on side 0.
34	I	DISK CHG*	Indicates that the drive door has been opened.

Notes:

- The direction of input and output signals is in reference to the FDC.
- An asterisk (*) indicates an active low signal.

Flexible Disk Drive support

The System Board provides a standard interface for up to four compatible internal flexible disk devices such as:

- HP 45811A 360 KB Internal Flexible Disk Drive (5.25-inch)
- HP 45812A 1.2 MB Internal Flexible Disk Drive (5.25-inch)
- HP D1667A 1.44 MB Internal Flexible Disk Drive (3.25-inch)
- HP D2045A 120 MB Internal Tape Drive

Flexible Disk Drive Performance

For the HP 45811A (360 KB) and 45812A (1.2 MB) flexible disk drives:

	360 KB Drive	1.2 MB Drive
Formatted Capacity	360 KB	1.2 MB
Recommended Media	HP 92190A	HP 92190X
Track density	48 tpi	96 tpi
Tracks/surface	40	80
Sectors/track	8 or 9	15
Bytes/sector	512	512
Track-to-track seek time	6 msec	3 msec
Average access time	93msec	93msec
Data transfer rate	250 kHz	500 kHz
Motor Start time	500 ms	500 ms
Rotation speed	300 rpm	360 rpm

For the HP D1667A (1.44 MB) flexible disk drive:

	720 KB Mode	1.44 MB Mode
Formatted Capacity	720 KB	1.44 MB
Recommended Media	HP 92192A	HP 92192X
Track density	135 tpi	135 tpi
Tracks/surface	80	80
Sectors/track	9	18
Track-to-track seek time	3 ms	3 ms
Average access time	175 ms	175 ms
Data transfer rate	250 kHz	500 kHz
Motor Start time	750 ms	750 ms
Rotation speed	300 rpm	300 rpm

Hard Disk Subsystem Interface Options

There are three hard disk interface options available for the HP Vectra 486/33T. They are:

1. the HP 1664A ESDI Controller Board is a separate board that takes up an accessory slot and supports up to two of the following high performance disk drives:

   - HP D1445A 108 MB (Megabyte),
   - HP D1446A 152 MB,
   - HP D1660A 330 MB,
   - HP D1661A 670 MB,

   or,

2. the System Board hard disk connector (on all models) that supports "embedded" hard disks, such as the HP D1666A 84 MB Embedded-AT Hard Disk Drive. Hard disks that have their own on-drive controlling subsystem are referred to as "embedded" drives. Up to two of these drives may be daisy-chained to the system via a single double-connector data/control cable.

   or,

3. an optional SCSI hard disk subsystem. This subsystem is currently available with three hard disk sizes, 440 MB, 670 MB, and 1,000 MB. Host Bus Adapters are available in both EISA (D1681A) and ISA (D1682A) configurations, and can daisy-chain up to 8 drives from a single controller.

   Complete information on the SCSI subsystem is provided in the Accessories Technical Reference Manual.

Note

The HP Vectra 486/33T will not support the use of *both* interface options at the same time (i.e., if you have a hard disk connected through the HP 1664A Hard Disk Controller Board, you may not also connect and use the HP D1666A 84 MB Embedded-AT Hard Disk Drive and visa versa).

Compatibility

Although different with respect to hardware make-up, the hard disk controller used with the HP 1664A ESDI Hard Disk Controller board and the embedded-AT hard disks is logically similar to an industry standard hard disk controller subsystem. To the host, these hard disk controllers appear identical.

In addition, if the HP D1664A ESDI Hard Disk Controller Board is used, only another ESDI hard disk drive (such as the 108, 152, 330, or 670 MB hard disk drives) may be added to the system. Figure 7-2 shows the HP D1664A ESDI Hard Disk Controller Board. Conversely, if the System Board's embedded controller hard disk interface is used (for example, with the HP D1666A 84 MB hard disk drive), only another embedded controller hard disk drive may be used.

Hard Disk Controller Configuration

The HP D1664A Hard Disk Controller Board and the HP D1666A 84 MB Embedded-AT Hard Disk are auto-configured via the system's configuration utility, EASY CONFIG. Through the systems configuration utility, the following may also be configured individually:

- set controller to the primary hard disk drive address (1F0h through 1F7h, 3F6h, and 3F7h). This is the default.

  (Note that since the HP Vectra 486/33T supports the use of only one hard disk controller at a time, the use of the secondary hard disk I/O address is not supported.)

- set controller for logical drive splitting. This feature is allows operating systems that use the ROM BIOS, such as DOS or OS/2, to utilize more than 528 MB of the HP D1661A 670 MB hard disk (UNIX based systems do not require this feature). With drive splitting enabled, the 670 MB drive is split in two: one 528 MB and one 142 MB disk. The default setting for the controller is with drive splitting enabled.

  You cannot add a second hard disk drive with drive splitting enabled.

In addition, on the HP D1664A Hard Disk Controller Board itself, there are two supported jumper configurations (see figure below): one which enables logical hard disk drive splitting, and one which disables the read-ahead cache (may be necessary for use with some applications). The default jumper position is to enable logical drive splitting (the user may disable this feature via the system's configuration utility) and the read-ahead cache.

Caution

Disabling drive splitting on a drive previously used with drive splitting enabled may result in the disk data becoming unusable. Before disabling drive splitting on a drive with data on it, backup all drive partitions.

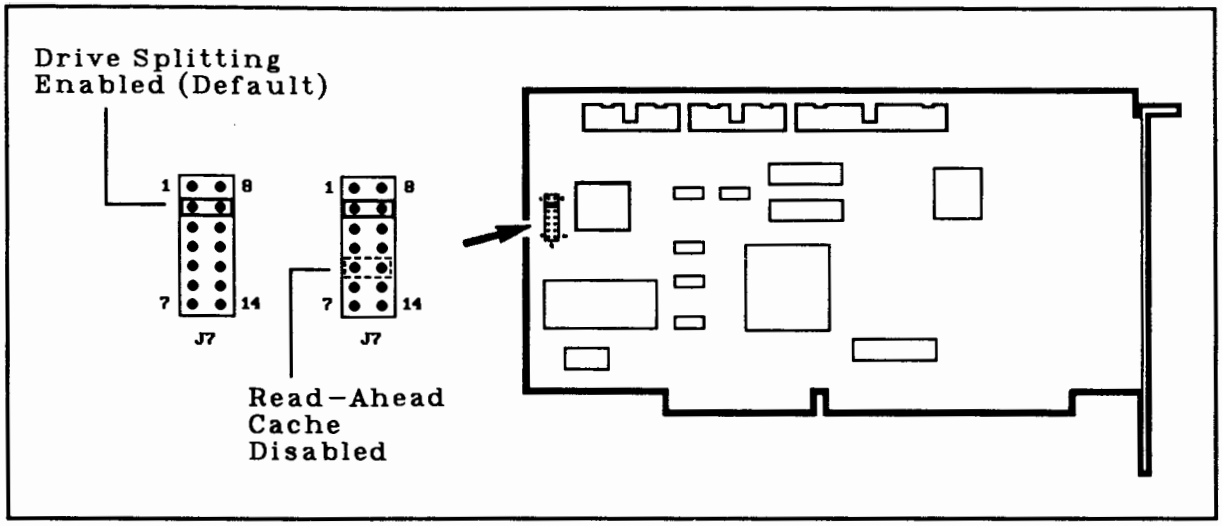


Figure 7-2. HP D1664A ESDI Hard Disk Controller

System to Hard Disk Controller Subsystem Interface

The hard disk controller subsystem interfaces to the system bus through the address, data and programmed I/O control signals. All data transfers are 16 bits wide and occur between the bus and the data buffer memory. Control, status and ECC byte transfers are 8 bits wide and use lower data byte lines SD07-SD00 only. The bus interrupt request level is fixed at IRQ14.

The hard disk controller subsystem (for both the HP D1664A Hard Disk Controller Board and the HP D1666A 84 MB Embedded-AT Hard Disk Drive) contains the common compatible registers for control and data transfer listed in Table 7-12.

Table 7-12. Hard Disk Controller Accessible Registers

Register	R/W	I/O Address Primary/Secondary
Data Register (16 bits)	R/W	1F0h/170h
Write Pre-compensation Cylinder *	W	1F1h/171h
Error Register	R	1F1h/171h
Sector Count	R/W	1F2h/172h
Sector Number	R/W	1F3h/173h
Cylinder Number - Low Byte	R/W	1F4h/174h
Cylinder Number - High Byte	R/W	1F5h/175h
Sector Size, Drive/Head Select	R/W	1F6h/176h
Status Register	R	1F7h/177h
Command Register	W	1F7h/177h
Hard Disk Register	W	3F6h/376h
Alternate Status Register	R	3F6h/376h
Digital Input Register	R	3F7h/377h

* This register is not used.

Note The HP Vectra 486/33T only supports one hard disk controller in the system at a time, therefore there is no support for the use of the secondary address.

Data Register (Read/Write)

The data register provides access to the sector buffer for read and write operations in the programmed I/O mode. This register must not be accessed unless a Read or Write command is being executed. The register provides a 16-bit path into the sector buffer for normal Read and Write commands. When a R/W Long is issued, 4 ECC bytes are transferred in low order byte with at least 2 microseconds between transfers. "Data Request" (DRQ) must be active before the transferring of the ECC bytes.

Error Register (Read only)

This is a read-only register that contains specific information related to the previous command. The data are valid only when the error bit in the Status Register is set in the operating mode or the hard disk controller subsystem is in diagnostic mode (the state immediately after power is switched on, or after a Diagnose command). In diagnostic mode the register must be checked regardless of the status register indicator. The following are error code bit values in the error register definitions for the diagnostic mode.

Table 7-13. Error Register Definitions

Code	Definition
00	Not used, undefined
01	No error
02	Control processor ROM checksum or RAM data error
03	Sector RAM buffer data error
04	Servo hardware error
05	Fatal HDA error
06-FF	Not used, undefined

The following are bit definitions for the operational mode.

Table 7-14. Error Register Operational Mode Definitions

Bit	Data	Definition
0	1	Data Address Mark Not Found within 16 bytes of the ID field
1	1	Track 000 error
2	1	Aborted command due to drive status error or invalid command code.
3	—	Not used
4	1	ID field not found
5	—	Not used
6	1	Data ECC error
7	1	Bad block detected

Sector Count Register (Read/Write)

This register defines the number of sectors to be transferred during a Verify, Read, Write, or Format command. The value written into this register is decremented by one after each sector is transferred to or from the sector buffer. Note: a 0 represents a 256 sector transfer, a 1 represents one sector, etc. This register is disregarded when a single sector command is specified.

Sector Number Register (Read/Write)

This register holds the number of the desired sector for Read, Write, and Verify commands. The starting sector number is loaded into this register for multi-sector operation. It is incremented by one after each sector has been transferred to or from the sector buffer.

Cylinder Number Registers - Low Byte (Read/Write)

This register holds the least signification 8 bits of the desired cylinder number.

Cylinder Number Register - High Byte (Read/Write)

The least three signification bits of this register contains the three most signification bits of the desired cylinder number.

Sector Size, Drive/Head Select Register (Read/Write)

This register contains the following information:

Table 7-15. Sector Size, Drive/Head Select Register Definitions

Bit	Data	Definition
0-3		Head select, 4 bit binary number. HS3 is the Most Signification Bit, and HS0 is the Least Significant Bit.
4	0	First hard disk drive selected.
	1	Second hard disk drive selected.
5	1	Set to 1 to select 512 bytes per sector. (See also bit 6.)
6	0	Set to 0 to select 512 bytes per sector. (See also bit 5.)
7	1	Set to 1 to select ECC mode for the data field.

Status Register (Read only)

The hard disk controller sets up the status register with the command status after execution. A read of the status register clears interrupt request 14. The following defines the bits of the status register.

Table 7-16. Status Register Definitions

Bit	Definition	
0	1	Error. Set to 1, this bit indicates that the command ended in an error, and Error Register bits are set. The next command resets the error bit.
1	1	Index. This bit is set to 1 at each revolution of the disk.
2	1	Corrected Data. Each time the error data read from disk is corrected by the ECC algorithm, this bit is set to 1.
3		Data Request. This bit indicates that the sector buffer requests servicing during a read of write command. If either this bit or bit 7 (Busy) are set to 1, a command is being executed. Upon receiving a command this bit is reset to 0.
4	1	Seek Completed. This bit is set to 1 when the read/write heads have completed a seek operation.
5	1	Write Fault. A 1 on this bit indicates improper operation of the drive; read, write, or seek is inhibited.
6	1	Drive Ready. When both this bit and the Seek Completed bit (bit 4) are set to 1, the hard disk drive is ready to read, write, or seek.
7	1	Busy. This bit is set to 1 when the hard disk controller is executing a command and no other status register bit is valid.

Command Register (Write only)

To maintain industry compatibility, the Command Register accepts the following commands and command attributes to perform hard disk operations; however, some hard disks may accept additional commands not listed here. Any code not defined in a hard disk's Command Register causes an Aborted Command error. The following are acceptable commands to the command register. (The commands Diagnose through Read Drive Parameters in the following table are extensions to the compatibility model, providing drive mechanism diagnostic capability and drive identification.) See Table 7-18 for descriptions of the commands.

Table 7-17. Command Register

Command	Hex	Bit 7 6 5 4 3 2 1 0
Restore to Cylinder 0	1X	0 0 0 1 X X X X
Read Sector(s)	20-23	0 0 1 0 0 0 L R
Write Sector(s)	30-33	0 0 1 1 0 0 L R
Read Verify	40-41	0 1 0 0 0 0 0 R
Format Track	50	0 1 0 1 0 0 0 0
Seek	7X	0 1 1 1 X X X X
Diagnose	90	1 0 0 1 0 0 0 0
Set Parameters	91	1 0 0 1 0 0 0 1
Read Drive Parameters	EC	1 1 1 0 1 1 0 0

Notes:

L = 0 (Normal mode, normal ECC functions)
L = 1 (Long mode)
R = 0 (Error retries are enabled)
R = 1 (Retries are disabled)
X = Don't care

Table 7-18. Command Register Descriptions

Command	Definition
RESTORE TO CYLINDER 0:	This command is used to move the R/W heads to the cylinder 00 position. If cylinder 00 is not true, the Error bit in the Status Register is set and a cylinder 00 error is posted in the Error Register. If the DRIVE READY* signal is de-asserted or ATTENTION* is asserted, this command terminates with the error bit set in the Status Register, and the Error Register reports an aborted command.
READ SECTOR:	A number of sectors (1-256) can be read from the selected drive with this command. The sector count register in the task file determines the number of sector to be transferred. Multi-sector reads may cross head and cylinder boundaries. If the R/W heads are not positioned over the target track, the controller performs an implied seek to the proper cylinder. The optional long bit (L set to 1 enables Read Long) informs the disk controller whether of not to include the four ECC bytes. These four ECC bytes are transferred as individual bytes, not words, as is the data field information. The data request bit in the data register must be valid before each byte transferred and at least 2 μsec will pass between each byte transferred. Data errors up to 22 bits in length will be automatically corrected on normal Read commands. If an uncorrectable error occurs, the data transfer will still take place. A multi-sector read, however, will terminate after the sector in error is read by the system.

Table 7-18. Command Descriptions (continued)

Command	Definition
	The optional retry bit (T set to 1 disables retries) disables or enables retries. The hard disk controller performs up to 10 automatic retries when the retry bit is enabled. The hard disk controller properly sets the error and status registers if the retries are unsuccessful. Disabling retries allows only two automatic retries before the hard disk controller sets the error and status registers. For ECC errors, eight retries are made at reading before a soft uncorrectable error is reported. A retry results in the reissuing of the hard disk controller Read Sector command. The hard disk controller Read Sector command attempts to verify the sector eight times before returning an error. ECC correctable data errors are corrected after two consecutive matching ECC syndromes are detected. If the error is an uncorrectable error or an error is reported by the hard disk controller, the command terminates. Interrupts occur as each sector is ready to be read by the system. No interrupt is generated at the end of the command. If the DRIVE READY* signal is de-asserted or WRITE FAULT* asserted, this command terminates with the error bit set in the status register and the error register reports an aborted command.
WRITE SECTOR:	A number of sectors (1-256) can be written to the selected drive. The sector count register in the task file determines the number of sectors to be transferred. A Multi-sector write may cross head and cylinder boundaries. The optional long bit (L set to 1 enables Write Long) informs the controller whether or not to append the host supplied ECC bytes. These four bytes are transferred as individual bytes, not words, as is data field information. The data request bit in the data register must be valid before each byte transferred and at least 2 μsec will pass between each byte transferred. The optional retry bit (T set to 1 disables retries) disables or enables retries. The hard disk controller performs up to 10 automatic retries when the retry bit is enabled. The hard disk controller properly sets the error and status registers if the retries are unsuccessful. Disabling retries allows only two automatic retries before the hard disk controller sets the error and status registers. The controller interrupt is generated as the data for each sector is required to be transferred into the sector buffer (except the first sector) and at the end of the command. The first sector may be written to the buffer immediately after the command has been sent, and the data request status is set. If the DRIVE READY* signal is de-asserted or WRITE FAULT* is asserted, this command terminates with the error bit set in the status register and the error register reports an aborted command.
READ VERIFY:	This command functions similarly to a normal Read command except that data are not output to the host. One to 256 sectors may be verified at on time. The generated ECC bytes are compared with the recorded ECC bytes for data verification. A signal interrupt is generated upon completion of the command or in the even of an error.

Table 7-18. Command Descriptions (continued)

Command	Definition
	For ECC errors, eight retries are made at reading before a soft uncorrectable error is reported. A retry results in the reissuing of the hard disk drive controller Read Sector command. The Read Sector command attempts to verify the sector eight times before returning an error. ECC correctable data errors are corrected after two consecutive matching ECC syndromes are detected. If the error is an uncorrectable error or an error is reported by the disk drive controller, the command terminates. The WRITE FAULT* and DRIVE READY* inputs are checked throughout the command's execution.
FORMAT TRACK:	The track specified by the task file is formatted with ID and Data fields according to the interleave table transferred to the buffer. The interleave table consists of two bytes per sector, the 1st byte is to indicate a bad block with a 80h (otherwise is 00h) and the second byte is for the logical sector address. The data transfer must be 512 bytes even though the table contains fewer bytes. The sector count register must be loaded with the number of sectors per track before each Format Track command. The Format Track command supports no error reporting. A bad block may be specified by replacing a 00 table entry with an 80h. When switching between drives, a Restore command must be executed prior to attempting a format. Command completion will leave all data fields initialized to zeros. The completion interrupt is generated after each track has been formatted.
SEEK:	This command moves the R/W heads to the cylinder specified in the task file cylinder high and low registers. An interrupt is generated at the completion of the command. If the DRIVE READY* signal is de-asserted or ATTENTION* is asserted, this command is terminated with the error bit set in the Status Register, and the Error Register reports an aborted command.
DIAGNOSE:	This command causes the controller to perform an onboard diagnostic and to report the result in the Error Register. An interrupt is generated upon completion of the command. The Diagnose command performs tests on the onboard microprocessor's internal ROM and RAM, host bus interface circuit, hard disk controller and the Sector buffer. If any component fails, the appropriate error code is loaded into the error register. Error code / Definition 00 — Not used, undefined 01 — No errors 02 — Hard disk controller error 03 — Sector RAM buffer data error 04 — Servo hardware or ECC device error 05 — Control processor ROM checksum or RAM data error 06-FF — Not used, undefined The sector count register is reset to one while the cylinder high and cylinder low and SDH register are all set to zero.

Table 7-18. Command Descriptions (continued)

Command	Definition
SET PARAMETERS:	This command sets up the drive parameters regarding the maximum number of heads and sectors per tracks. The hard disk controller uses these two parameters when performing multiple sector operations. The SDH task file register specifies the drive affected. The sector count and SDH registers must be set up before this command is issued. An interrupt is set at the completion of the command. This command must be issued before any multiple sector operations are undertaken. By setting the SDH register for each of the two possible drives, this command allows the controller to support two drives with different characteristics. If the parameters set with the SET PARAMETERS command are not identical to drive's physical parameters, the controller enters translation mode, where logical host sector addresses are translated into physical drive sector addresses. Translation is conceptualized as follows: **Variables Used:** LC: Logical Cylinder PC: Physical Cylinder LH: Logical Head PH: Physical Head LS: Logical Sector PS: Physical Sector **Constants Used:** SH: Set Parameters Heads/Cylinder SS: Set Parameters Sectors/Track DH: Actual Drive heads/cylinder DS: Actual Drive sectors/track **Intermediate Variable:** AS : Absolute Sector AS:= (LC * SH + LH) * SS + LS PC:= AS DIV (DH * DS) PH:= (AS - PC * DH * DS) DIV DS PS:= AS - PC * DH * DS - DH * DS This logical-to-physical translation enables drives whose physical parameters exceed BIOS limits to be supported if the capacity of the physical drive is greater than or equal to the capacity of the drive specified in the Drive Type. The logical parameters of number of drive heads and number of sectors per track defined in the drive type are those used in SET PARAMETERS.

Table 7-18. Command Descriptions (continued)

Command	Definition
READ DRIVE PARAMETERS:	This command returns 49 bytes of the mass storage subsystem configuration to the host. (See below.) The configuration information covers both the controller and the drive specified in the SDH register. When configuration information is ready to send to the host, the controller issues IRQ 14h and sets DRQ. The returned information has the following format. Words 0 through 9 are the drive's actual responses to the referenced commands. 0 General configuration 1 Number of fixed cylinders 2 Number of removable cylinders 3 Number of heads: bits 15 - 8 = number of removable heads bits 7 - 0 = number of fixed heads 4 Minimum number of unformatted bytes/track 5 Minimum number of unformatted bytes/sector (hard sectored drives only) 6 Number of sectors/track bit bits 15 - 8 = reserved bits 7 - 0 = sectors/track 7 Minimum bytes in Inter Sector Gap (ISG) field, not including intersector speed tolerance bits 15 - 8 = ISG bytes after index/sector pulse to write splice bits 7 - 0 = bytes per ISG 8 Minimum bytes per Phase Lock Oscillator (PLO) sync field bits 15 - 8 = reserved bits 7 - 0 = bytes per PLO sync field required when read gate is asserted. 9 Number of words in vendor unique status bits 15 - 8 = reserved bits 7 - 0 = number of vendor unique status words 10-19 Board serial number. 20 ASCII characters. All "0" means that the board is unserialized. (Currently not used.) 20 Controller type: 0 : Unspecified 1 : Single ported, single sector buffer 2 : Dual ported, multisector buffer 3 : Dual ported, multisector buffer, cache 21 Number of 512 byte per sector pages of controller buffer. 22 Number of ECC bytes transferred on long operations. 23-26 Controller firmware revision, 8 ASCII characters. 27-46 Controller model number, 40 ASCII characters. 47-255 Reserved

Hard Disk Register (Write only)

This register allows programmed hard disk subsystem reset and provides enable/disable control of the hard disk interrupt.

Table 7-19. Hard Disk Register Definitions

Bit	Data	Definition
0	—	Reserved
1	0	Enable interrupt (IRQ14).
	1	Interrupt disabled.
2	1	Reset subsystem.
3	—	Reserved
4	—	Reserved
5	—	Reserved
6	—	Reserved
7	—	Reserved.

Alternate Status Register (Read only)

The Alternate Status Register is an exact duplicate of the Status Register at location 1F7h.

Digital Input Register (Read only)

The digital input register contains the current state of the hard disk drive select, head select and drive write gate signals. The bit definition is shown below.

Table 7-20. Digital Input Register Definitions

Bit	Definition	
0	0	Drive select 0
1	0	Drive select 1
2-5		a 4-bit binary number whose one's complement represents the number of the head currently being used (i.e., 1010 represents head 5)
6	0	Write Gate On
7	—	Reserved

D1664A ESDI Controller Disk Drive Interface

The D1664A Hard Disk Controller subsystem interfaces to a maximum of two hard disk drives through one 34-pin daisy-chain control cable and two 20-pin radial data cables, in conformance with ESDI signal definitions. The control cable can connect two drives; for each drive a separate data cable is used.

Figure 7-3 shows the card's interface connector pinout for the control cable.

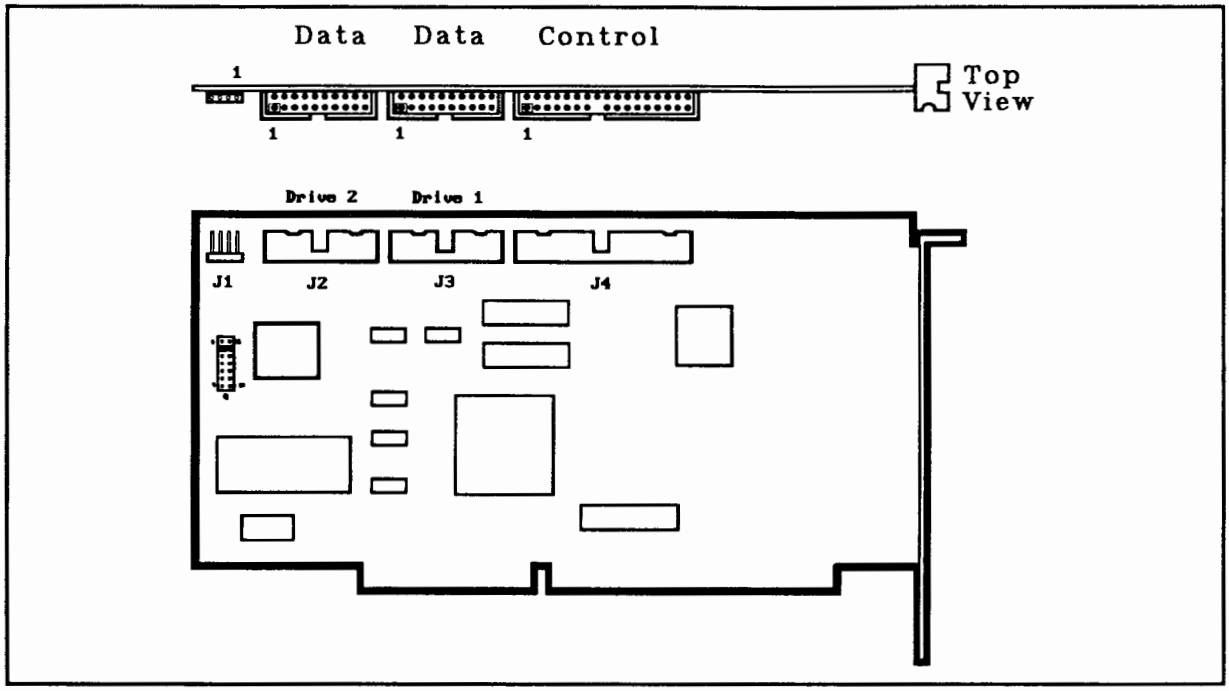


Figure 7-3. Control Cable Interface Connector

Table 7-21 gives the pin assignments and descriptions for the connector.

Table 7-21. Control Connector Pin Assignments

Pin	Direction	Signal	Description
2, 4, 18, 14	O	HEAD SELECT3*, HEAD SELECT2*, HEAD SELECT1*, HEAD SELECT0*	These four signals are decoded to select one of up to sixteen read/write heads.
6	O	WRITE GATE*	This signal is asserted when valid data is to be written on a disk. It is deasserted when WRITE* Fault signal from the selected drive is detected.
8	I	CONF/STATUS*	Status or data from commands sent from the controller to the drive over the COMMAND line.
10	I	TRANSFER ACK*	Handshake signal for serial command transfer to and status reception from an ESDI drive.
12	I	ATTENTION*	This signal informs the disk controller that some extraordinary event occurred in the drive requiring attention or error recovery.
16	I	SECTOR*	Asserted by the drive when an address mark is detected.
20	I	INDEX*	It indicates the start of a track. It pulses once at each disk revolution.

Table 7-21. Control Connector Pin Assignments (continued)

Pin	Direction	Signal	Description
22	I	READY*	It is asserted by the selected drive to inform the controller that the drive motor is up to speed.
24	O	TRANSFER REQ*	Indicates that the controller wishes to transfer command/status information to/from the drive. This signal is used with TRANSFER ACK* to handshake Command Serial Data.
26, 28	O	DS0*, DS1*	Drive select lines 0 or 1 are used to select drive 0 and drive 1 respectively.
30	—	—	Reserved.
32	O	READ GATE*	Informs the drive that a series of 0's has been detected as in the case of a sync field.
34	O	COMMAND*	Serial commands sent by the controller.

Notes:

- Pin 15 reserved to polarize the connector.
- The direction is in reference to the controller on the HP D1664A Hard Disk Controller Board. "I" is to the controller, "O" is to the host.
- An asterisk (*) indicates an active low signal.

Data Interface Pin Assignments and Description

The data lines between the hard disk controller and the two disk drives are connected to J2 and J3 at board upper edge. Both J2 and J3 have the same pin assignment and definition. The data lines are defined in the following table.

Table 7-22. Data Interface Connector Pin Assignments

Pin	Signal Name (J3)	Pin	Signal Name (J2)
1	DRIVE SELECTED 0*	1	DRIVE SELECTED 1*
2	Not connected	2	Not connected
3	COMMAND COMPLETED 0*	3	COMMAND COMPLETED 1*
4	Not connected	4	Not connected
5, 6, 9, 12, 15, 16, 19, 20	Ground	5, 6, 9, 12, 15, 16, 19, 20	Ground
7	+WRITE CLOCK 0	7	+WRITE CLOCK 1
8	−WRITE CLOCK 0	8	−WRITE CLOCK 1
10	+READ CLOCK 0	10	+READ CLOCK 1
11	−READ CLOCK 0	11	−READ CLOCK 1
13	+WRITE DATA 0	13	+WRITE DATA 1
14	−WRITE DATA 0	14	−WRITE DATA 1
17	+READ DATA 0	17	+READ DATA 1
18	−READ DATA 0	18	−READ DATA 1

Hard Disk Drive Activity LED Connector

The LED connector at J1 is a 4-pin header that connects with a reversible cable to the computer's front panel. The hard disk controller lights the LED when the hard disk drive is busy or the host asserts RESET. The following shows the connector pin assignments and functions.

Table 7-23. Activity LED Signals

Pin	Signal Name
1, 4	+LED; connects to LED anode.
2, 3	−LED; connects to LED cathode.

D1666A Embedded-AT Disk Drive Interface

A maximum of two D1666A 84 MB Embedded-AT Hard Disk Drives may be interfaced to the system through one 40-pin daisy-chain cable with connectors for two drives. Note that the maximum supported cable length is 23 inches (cable part number D1665-60005).

Figure 7-4 shows the connector (J47) on the System Board.

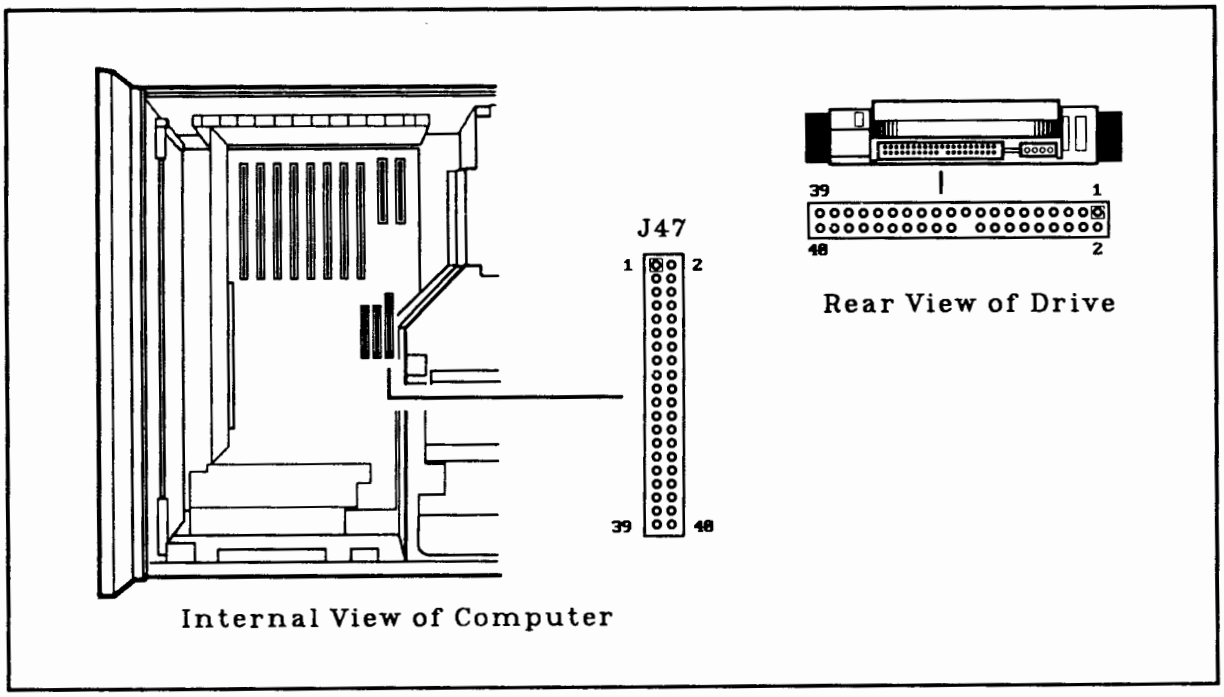


Figure 7-4. System Interface Connector for Embedded-Controller Hard Disk Drives

Table 7-24 gives the pin assignments and descriptions for the connector.

Table 7-24. Embedded-AT Signal Definition

Pin(s)	Direction	Signal	Description
1	I	RESET*	Reset signal from the host system; active low during system power up.
2, 19, 22, 24, 26, 30, 40	—	GROUND	Ground between the drive and the host system.
3-18	I/O	HD0-15	16 bit bi-directional data bus between the host system and the drive. HD0-7 are used to access registers and ECC. (HD7 is disabled when the host reads the Digital Input Register.) All 16 bit are used for data transfers. These are tri-state lines with 24mA drive capability.
20	—	Key	Unused pin for keying ribbon cable to the drive.
21 and 27	O	IOCHRDY	Enables host wait state generation to lengthen the I/O read and write cycles. Driven LOW by the drive immediately upon detecting a valid I/O host address select. These two pins are connected together.
23	I	IOW*	Write strobe. On the rising edge, it clocks data from the host to the drive over data lines HD0-7 and/or HD8-15 into a register or the data register of the drive.
25	I	IOR*	Read strobe. When it is low, enables data from a register or the data register of the drive onto the host data bus, HD0-7 and/or HD8-15. The rising edge latches data from the drive at host.
28	I	ALE	Address Latch Enable signal from the host. It is not currently used on the drive.
29 and 34	—	Reserved	Reserved
31	O	IRQ14	Interrupt signal to the host. Active high only when the drive is selected and the drive IEN* interrupt enable, bit of the digital output register is low. This signal goes to a high impedance state when the drive is not selected or the IEN* bit is high, inactive. The interrupt is cleared upon receiving the next command, or when the status register is read or when the drive is reset.
32	O	IO16*	Informs the host that the drive data register has been enabled and that the drive is prepared to perform a 16-bit I/O transfer. Open collector output with 24mA driver.

Table 7-24. Embedded-AT Signal Definition (continued)

Pin(s)	Direction	Signal	Description
33	I	HA1	Address line from the host to the drive that is used to select a register on the drive.
35	I	HA0	Address line from the host to the drive that is used to select a register on the drive.
36	I	HA2	Address line from the host to the drive that is used to select a register on the drive.
37	I	CS0*	Decoded address select from the host indicating that access to one of the 8 task file registers is desired. This signal is used with HA0-2 for selecting the task file register.
38	I	CS1*	Decoded address select from the host indicating that access to one of the 3 disk function registers is desired. This signal is used with HA0-2 for selecting the register.
39	O	DRV_ACT*	Active low signal from the drive. It is low when the drive is busy. This signal can be used for the drive LED. It has 20mA driver.

Notes:

- The direction is in reference to the embedded controller/drive subsystem. "I" is to the subsystem, "O" is to the host.
- CS0* is asserted when the host address is in the 1FO-1F7 range.
- CS1* is asserted when the host address is in the 3F6-3F7 range. DRV_ACT* is a drive activity light output or second drive indication.
- HA0-HA2 are low order I/O address bits and are used with CS0* and CS1* to decode I/O addresses.
- Pin 20 is reserved for the Key slot to prevent incorrect installation of mating connector.
- Pins 21 and 27 are tied together (HP D1666A 84 MB Embedded-AT Hard Disk Drive only).
- An asterisk (*) indicates an active low signal.

Hard Disk Drive Support

The HP D1664A ESDI Hard Disk Controller Board supports up to two hard disk drives. These drives provide fast access and high-capacity storage. These drives automatically retract the read and write heads to a parking zone when powered off, which protects the disk against damage during handling or moving. If you have one 670 MB hard disk with drive splitting enabled, you may not add a second (physical) hard disk.

Drives supported include:

- HP D1445A 108 MB
- HP D1446A 152 MB
- HP D1660A 330 MB
- HP D1661A 670 MB

Note

The drive splitting feature affects the configuration of the HP D1661A 670 MB hard disk in the following ways:

- If drive splitting is enabled, the first 528 MB of the hard disk is disk type 28, and the last 142 MB of the hard disk is disk type 29.
- If drive splitting is disabled, the entire 670 MB of the hard disk is disk type 30.

Without the HP 1664A Hard Disk Controller Board (such as with Model 1) the System Board hard disk connector supports up to two hard disks that have their own embedded controller subsystems, such as the HP D1666A 84 MB Embedded-AT hard disk drive. The HP D1666A also has automatically parking read and write heads to prevent disk damage when handling or moving.

Hard Disk Performance

For the HP D1660A (330 MB) and D1661A (670 MB) hard disk drives:

Data transfer rate:	20 Mbits per sec maximum (single track burst) 15.6 Mbits per sec (single track sustained) 13.0 Mbits per sec (continuous)
Sector format:	512 bytes/sector
Interleave:	1:1
Drives supported:	2 maximum
Heads supported:	8 (330 MB drive), 16 (670 MB drive)
Cylinders supported:	1456
Hard error rate:	less than 10 per 10(E13) bits read
Soft error rate:	less than 10 per 10(E11) bits read
Seek error rate:	less than 1 per 10(E06) seeks
Average random seek time:	17 ms
Average random (aggressive*) seek time:	16 ms
Maximum seek time:	32 ms
Track to track seek time:	3.5 ms
Average latency time:	7.47 ms (± 5%)
Rotational speed:	4,002 rpm

* These drives support aggressive seeks which allow the controller to begin reading before the heads are settled. This feature improves the subsystem's read performance, but is not used for disk writes.

For the HP D1445A (108 MB) and D1446A (152 MB) hard disk drives:

Data rate:	10.0 Mbit per Sec maximum
Sector format:	512 bytes/sector
Interleave:	1:1
Drives supported:	2 maximum
Heads supported:	5
Cylinders supported:	1249
Hard error rate:	less than 1 per 10(E12) bits read
Soft error rate:	less than 1 per 10(E10) bits read
Seek error rate:	less than 1 per 10(E06) seeks

Average random seek time:	16 ms
Maximum seek time:	37 ms
Track to track seek time:	4 ms
Average latency time:	8.33 ms
Rotational speed:	3,600 rpm

For the HP D1666A (84 MB) Embedded-AT Hard Disk Drive:

Data rate:	10.0 Mbit per Sec maximum
Sector format:	512 bytes/sector
Interleave:	1:1
Drives supported:	2 maximum
Heads supported:	10
Cylinders supported:	965
Hard error rate:	less than 1 per 10(E13) bits read
Soft error rate:	less than 1 per 10(E11) bits read
Seek error rate:	less than 1 per 10(E07) seeks
Average random seek time:	19 ms
Maximum seek time:	20 ms
Track to track seek time:	7 ms
Average latency time:	8.2 ms
Rotational speed:	3,662 rpm

8

Serial and Parallel I/O

Introduction

This chapter describes the serial and parallel I/O (Input/Output) subsystems. The serial datacomm interface provides asynchronous RS-232C communication. The parallel datacomm interface provides eight bits of parallel data at standard TTL levels.

Parallel/Serial Port Board

A 50-pin connector on the System Interface Board is provided for mounting the Parallel/Serial Port Board (see Figure 8-1). The pin signal definitions are given in the table following. There is an additional serial connector on the Keyboard/Mouse/Serial Board (the serial interface for both connectors is described in this chapter). For more information on the 50-pin system interface Board connector for the Mouse/Keyboard/Serial Port Board, see chapter 6.

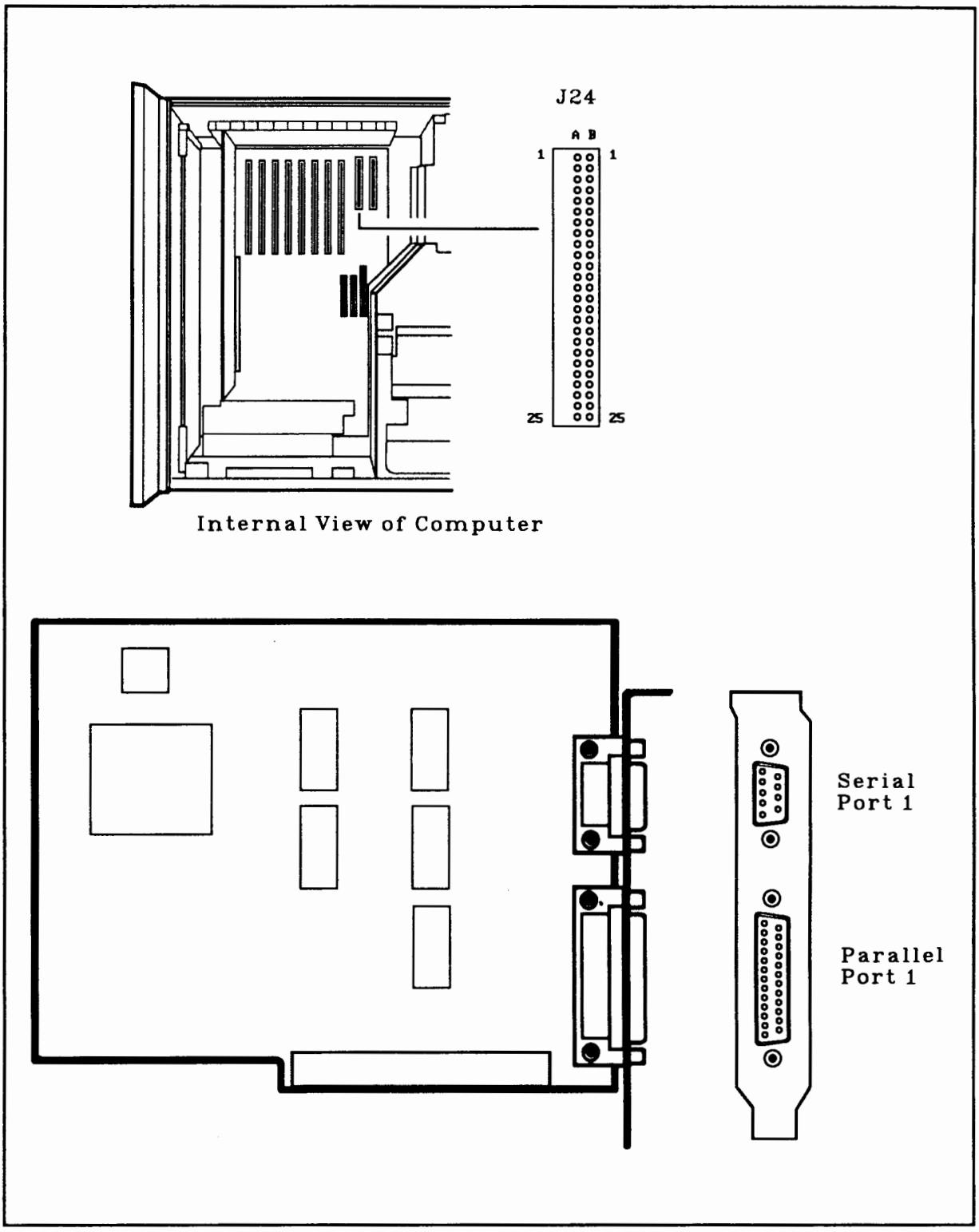


Figure 8-1. Parallel/Serial Port Board Connectors

Table 8-1. Parallel/Serial Port Board to System Signal Definitions[1]

Connector Pin	Direction	Signal Name	Signal Definition
A12	—	-12V	-12 Volts
B13	—	+12V	+12 Volts
B9	O	BDO	Bus Buffer Output
A3	I	CTS1*	Clear to Send No. 1
A2	I	DSR1*	Data Set Ready No. 1
B2	O	DTR1*	Data Terminal Ready No. 1
B10	I	EXMODE	Extended Mode
A1, 6, 8, 19, 22, 25	—	GND	Ground
B1, 5, 19, 22, 25	—	GND	Ground
B17	—	HICRD2ND	Identifier for Serial/Parallel
A11	O	IORD*	I/O Read
B12	I	PRACS*	Parallel Port Chip Select
A14	O	PRAINT	Parallel Port Interrupt
B6	I	RI1*	Ring Indicator No. 1
A5	I	RLSD1*	Received Line Signal Detect No. 1
A9	I	RSTDRV*	Reset
B8	O	RTS1*	Request to Send No. 1
A7	I	RXD1*	Receive Data No. 1
B14	I	SRACS*	Serial Port No. 1 Chip Select
A15	O	SRAINT	Serial Port No. 1 Interrupt
A13	I	SRBCS*	Serial Port No. 2 Chip Select
B15	O	SRBINT	Serial Port No. 2 Interrupt
B4	O	TXD1	Transmit Data No. 1
A4, 16	—	VCC	+5 Volts
B3, 7, 16	—	VCC	+5 Volts
A18	I	XA <0>	X-Bus Address Line 0
B18	I	XA <1>	X-Bus Address Line 1
A17	I	XA <2>	X-Bus Address Line 2
A24	I/O	XD <0>	X-Bus Data Line 0
B24	I/O	XD <1>	X-Bus Data Line 1
A23	I/O	XD <2>	X-Bus Data Line 2
B23	I/O	XD <3>	X-Bus Data Line 3
A21	I/O	XD <4>	X-Bus Data Line 4
B21	I/O	XD <5>	X-Bus Data Line 5
A20	I/O	XD <6>	X-Bus Data Line 6
B20	I/O	XD <7>	X-Bus Data Line 7
B11	I	XIOW*	X-Bus I/O Write

Notes:

1. The direction of the signal inputs (I) and outputs (O) is in reference to the WD16C522 ACE. An asterisk (*) indicates an active low signal.

Serial Datacomm Interface Subsystem

The two ports of the serial data communication subsystem provides asynchronous communications at RS-232C levels. Other features include: a programmable baud rate of 75 to 19200 bps (110 bps is the minimum for the BIOS initialization), character length of 5, 6, 7, or 8 bits, a stop bit of 1, 1 1/2, or 2, a parity bit, modem control signals, full double buffering, and automatic adding or deletion of start, stop, and parity bits.

Serial Port Configuration

The two standard serial output ports can be addressed as either communications port 1, 2, 3 or 4 via the computer's configuration utility (EASY CONFIG) so that additional serial ports (up to two additional for a total of four) can be added. These four ports are mapped to different I/O addresses and have different interrupt levels:

- Port 1 has I/O address 3F8h through 3FFh and interrupt level 4.
- Port 2 has I/O address 2F8h through 2FFh and interrupt level 3.
- Port 3 has I/O address 3E8h through 3EFh and interrupt level 10.
- Port 4 has I/O address 2E8h through 2EFh and interrupt level 11.

Serial Port Selection Through I/O Register C41

Selection of the serial port address is done programmatically using I/O register C41h. The bits of this register are described below.

Table 8-2. Serial Port Selection I/O Register C41[1]

Bits	Bit Format	Description
0, 1		Set first serial port: 00 = COM1 (3F8h-3FFh) 01 = COM2 (2F8h-2FFh) 10 = COM3 (3E8h-3EFh) 11 = COM4 (2E8h-2EFh)
2	0	Disable first serial port
	1	Enable first serial port
3, 4		Set second serial port: 00 = COM1 01 = COM2 10 = COM3 11 = COM4
5	0	Disable second serial port
	1	Enable second serial port
6	0	Enable write to CMOS
	1	Disable write to CMOS
7	0	Disable extend I/O recovery time
	1	Enable extend I/O recovery time

Note:

1. Power-on values for all bits are zero.

Serial Port Interface

The serial port uses an RS-232C connector D-type 9-pin male. Table 8-3 defines the pin assignments and signal descriptions for this connector.

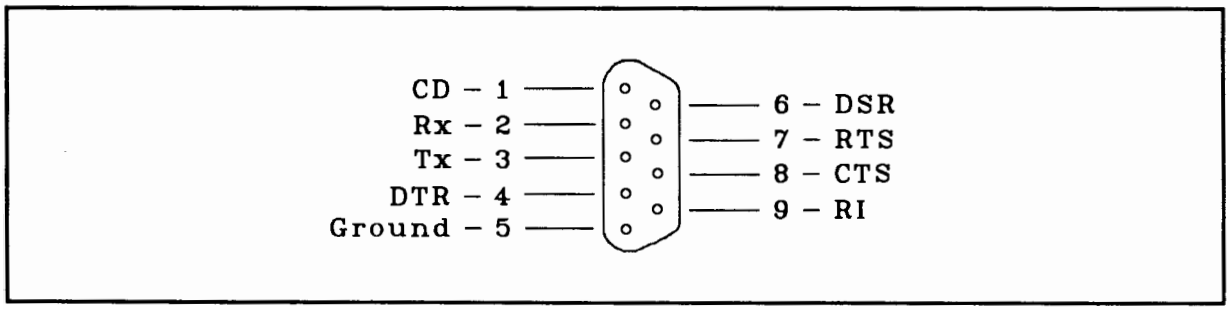


Figure 8-2. Serial Connector

Table 8-3. Serial Port Pin Assignments

Pin	I/O	Signal	Description
1	I	CF (CD)	When active, it indicates the device has detected the data carrier.
2	I	BB (RX)	Rx data
3	O	BA (TX)	Tx data
4	O	CD (DTR)	When active, it informs the device that the card is available to communicate.
5		AB (Gnd)	GND, ground
6	I	CC (DSR)	When active, it indicates the device is ready to establish the communications link in order to transfer data.
7	O	CA (RTS)	When active, it informs the device that the card is ready to transmit data.
8	I	CB (CTS)	When active, it indicates the device is available to receive data.
9	I	CE (RI)	When active, it indicates the device has a telephone ringing signal.

Registers

The serial datacomm subsystem has 11 programmable registers. The system programmer may gain access or control any of the registers through the system CPU. Table 8-4 defines the registers and their addresses.

Table 8-4. Serial Datacomm Subsystem Registers

Register	R/W	Port 1/Port 2/Port 3/Port 4
Transmit Buffer[1]	W	3F8h/2F8h/3E8h/2E8h
Receive Buffer[1]	R	3F8h/2F8h/3E8h/2E8h
Divisor Latch LSB[1]	R/W	3F8h/2F8h/3E8h/2E8h
Divisor Latch MSB[1]	R/W	3F9h/2F9h/3E9h/2E9F
Interrupt Enable Register[1]	R/W	3F9h/2F9h/3E9h/2E9h
Interrupt ID Register	R	3FAh/2FAh/3EAh/2EAh
FIFO Control Register	W	3FAh/2FAh/3EAh/2EAh
Line Control Register	R/W	3FBh/2FBh/3EBh/2EBh
Modem Control Register	R/W	3FCh/2FCh/3ECh/2ECh
Line Status Register	R/W	3FDh/2FDh/3EDh/2EDh
Modem Status Register	R/W	3FEh/2FEh/3EEh/2EEh
Reserved		3FFh/2FFh/3EFh/2EFh

Notes:

1. Access to these registers is determined by the state of the Line Control Register.

Transmit Buffer Register (Write only)

This register contains the characters to be transmitted via the serial connector. Data bit 0, the least significant bit (LSB), is transmitted first and data bit 7, the most significant bit (MSB), is the last bit transmitted.

Receive Buffer Register (Read only)

This register contains the characters received via the serial connector. Data bit 0, the least significant bit (LSB), is received first and data bit 7, the most significant bit (MSB), is the last bit received.

Bit 7 of the Line Control Register determines whether the Transmit Buffer Register or the Divisor Latch Register LSB is accessed and whether the Interrupt Enable Register or the Divisor Latch Register (MSB) is accessed.

Divisor Latch Registers LSB and MSB (Read/Write)

The read/write divisor latch LSB and the divisor latch MSB registers are used to control the baud-rate of the transmitted and received data.

This subsystem has a clock of 1.8432MHz. This frequency is divided by any divisor from 1 to 65,525 as set on the two divisor latches. The output frequency is 16 times the baud-rate.

The two divisor latches must be loaded to define the baud-rate before attempting to transmit or receive data. When either of the latches is loaded, a 16-bit baud-rate counter is immediately loaded to prevent long counts on the first load.

Table 8-5 gives examples of loading the divisor latch registers to determine the baud-rate.

Table 8-5. Divider Latch Values

Baud Rate	Divisor	MSB (XF9H) Bits 7 6 5 4 3 2 1 0	(XF8H) Bits LSB 7 6 5 4 3 2 1 0
75	1536	0 0 0 0 0 1 1 0	0 0 0 0 0 0 0 0
110	1047	0 0 0 0 0 1 0 0	0 0 0 1 0 1 1 1
300	384	0 0 0 0 0 0 0 1	1 0 0 0 0 0 0 0
600	192	0 0 0 0 0 0 0 0	1 1 0 0 0 0 0 0
1200	96	0 0 0 0 0 0 0 0	0 1 1 0 0 0 0 0
2400	48	0 0 0 0 0 0 0 0	0 0 1 1 0 0 0 0
4800	24	0 0 0 0 0 0 0 0	0 0 0 1 1 0 0 0
9600	12	0 0 0 0 0 0 0 0	0 0 0 0 1 1 0 0
19200	6	0 0 0 0 0 0 0 0	0 0 0 0 0 1 1 0

Interrupt Enable Register (Read/Write)

This register enables and disables the four types of interrupt from the serial datacomm subsystem.

Table 8-6. Interrupt Enable Register Definitions

Bit	Data	Interrupt
0	1	Enable the received data available interrupt.
	0	Disable interrupt.
1	1	Enable transmitter holding register empty interrupt.
	0	Disable interrupt.
2	1	Enable the receiver line status interrupt.
	0	Disable interrupt.
3	1	Enable the modem status interrupt.
	0	Disable interrupt.
4-7	0	Always set to 0.

When the enabled interrupt signal is received it activates the chip interrupt (INTRPT) output signal which is sent to the system. When all interrupts are disabled, the Interrupt Enable Register and the INTRPT output signal are disabled. The other registers are not affected.

Bit 7 of the Line Control Register determines whether the divisor latch MSB or the Interrupt Enable Register is accessed.

Interrupt Identification Register (Read only)

This register identifies the highest priority pending interrupt signal. When this register is addressed it inhibits the highest priority interrupt. No other interrupts are acknowledged until this inhibited interrupt is cleared.

Table 8-7. Interrupt Identification Register Definitions

Bit	Data	Definition
0	0	Interrupt pending.
	1	No interrupt pending.
1-2		Identifies the pending interrupt with the highest priority as in the following: **Bit** **1 2 Interrupt Condition** **----------** **1 1 Receiver line status** **1 0 Received data available** **0 1 Trans. buffer empty** **0 0 Modem status**
3-7	0	Not Used.

The following defines the interrupt priorities.

Table 8-8. Interrupt Priorities

Interrupt	Priority	Interrupt Source	Clear Interrupt
Receiver line status	1 (Highest)	Overrun error or parity error or framing error or break (200mS space on receive data line).	Reading the line status register.
Received data available	2 (Second)	Data available in receive buffer.	Reading the receive buffer register.
Transmitter buffer register empty	3 (Third)	Data transmitted from transmit buffer.	Reading the interrupt identification register (if source), or writing to the transmit buffer register.
Modem status	4 (Fourth)	CB, CC, CE or CF signal received.	Reading the modem status register.

FIFO Control Register (Write only)

The FIFO (First In First Out) register is used for the following:

- enable FIFO mode
- clear FIFOs
- set RCVR FIFO trigger levels
- set DMA signaling mode

The following table defines the different bits of this register.

Table 8-9. FIFO Control Register Definitions

Bit	Data	Definition
0	1	Enable FIFO mode. Enables XMIT and RCVR FIFOs (required in order to program other FIFO bits). This bit must be reset when changing from Character to FIFO, or from FIFO to Character mode in order to clear data in the FIFOs.
	0	Reset to Character mode.
1	1	Resets RCVR FIFO counters to zero and then sets this bit to 0.
2	1	Resets XMIT FIFO counters to zero and then sets this bit to 0.
3	1	With Bit 0 set to 1, this bit will cause TXRDY and RXRDY pins (used to signal DMA transfer to the CPU) of the WD16C552 to change from Mode 0 to Mode 1 (*see note 1*).
	0	Reset TXRDY and RXRDY pins to Mode 0 (*see note 2*).
4-5		Reserved.
6-7		The trigger level of the RCVR FIFO interrupt are controlled by these bits as follows:

Bit 6	Bit 7	Trigger Level (Bytes)
0	0	01
1	0	04
0	1	08
1	1	14

Notes:

1. Mode 1: RXRDY goes active low when time-out or trigger level has been reached, and inactive when FIFO is empty; TXRDY goes active low if there are unfilled positions (one or more) in XMIT FIFO, and inactive when FIFO is full.
2. Mode 0: whether in Character or FIFO modes, RXRDY goes active low if a character is in the RCVR holding or RCVR FIFO register, and inactive when the RCVR FIFO (when in FIFO mode), or RCVR holding register (in Character mode) is empty; whether in Character of FIFO modes, TXRDY goes active low if the XMIT FIFO (FIFO mode), or XMIT holding register is empty, and inactive after the first character is loaded.

Line Control Register (Read/Write)

This register controls the format of the data communications.

Table 8-8. Line Control Register Definitions

Bit	Data	Definition

0-1		Specifies the number of bits in each transmitted or received character as in the following:
		
		`Bit  Character Length in bits`
		`0 1`
		`----------`
		`1 1  8`
		`1 0  7`
		`0 1  6`
		`0 0  5`

2	0	One stop bit is generated or deleted in the data sent or received.
	1	1.5 stop bits are generated or deleted for 5-bit words. For a 6, 7, or 8-bit word, 2 stop bits are generated or deleted.
3	0	Disable parity bit.
	1	A parity bit is generated (transmit data) or deleted (receive data).
4	0	When bit 3 is 1, parity bits sent or checked odd.
	1	When bit 3 is 1, parity bits sent or checked even.
5	1	When bit 3 is 1, the parity bit is set 0 for even parity and 1 for odd parity.
	0	Stuck parity disabled.
6	1	Break bit. The transmit data line is set to the space state (0) and remains at that state regardless of the state of the output buffer register.
	0	Set-breaking is disabled.
7	1	Address selection bit. Set to gain access of the divisor latches of the baud-rate generator during a read/write operation.
	0	Reset to gain access of the receiver buffer register, the transmit buffer register, or the interrupt enable register.

Modem Control Register (Read/Write)

This register controls the modem signals. It also allows the serial Datacomm subsystem to be set into diagnostic mode. In the diagnostic mode, transmitted data is received immediately. The receiver and transmitter interrupts and the modem control interrupts are fully operational, allowing the interrupts to be tested.

Table 8-9. Modem Control Register Definitions

Bit	Data	Definition
0	1	Data terminal ready (CD) signal active.
	0	CD signal inactive.
1	1	Request to send (CA) signal active.
	0	CA signal inactive.
2	0 or 1	Controls the OUT1* signal from the controller chip. Can be 0 or 1.
3	1	Controls OUT2* signal from the controller chip. Enables INTRPT generated by the interrupt enable register.
	0	The OUT2* output is forced inactive.
4	1	Enables the modem loopback feature (diagnostic test) as follows: 1. Receiver serial input is disabled. 2. Transmitter serial output is set to the active state. 3. The output from the transmitter shift register is looped back to the receiver shift register. 4. The four modem control inputs to the modem status register are disabled. 5. The four modem control outputs from the modem control register are internally connected to the four modem control inputs
5-7	0	Always set to 0.

* Indicates an active low signal.

Line Status Register (Read/Write)

This register provides information on the data transfer. Bits 1 through 4 are error conditions that generate a receiver line status interrupt. This register is not to be written to.

Table 8-10. Line Status Register Definitions

Bit	Data	Definition
0	1	Set when a complete incoming character has been received and transferred into the receiver buffer register.
	0	Reset by reading the data in the receiver buffer register.
1	1	Indicates that data in the receive buffer register was not read by the processor before the next character was transferred into the register, thereby erasing the previous character.
	0	Reset when the host reads the Line Status Register.
2	1	Detection of a parity error.
	0	Reset when the host reads the Line Status Register.
3	1	Framing error has occurred, character does not have a valid stop bit.
	0	Reset when the host reads the Line Status Register.
4	1	The received data line was at a space state (0) for longer than a transmission time of a complete data character. (Including start, data, parity, and stop bits.)
	0	Reset when the host reads the Line Status Register.
5	1	Set when a character is transferred from the transmit buffer register to the transmit shift register indicating the card is available to transmit another character.
	0	Reset when the next character is written into the transmit buffer register.
6	1	Transmit buffer register and transmit shift register are empty.
	0	Reset when either register contains a character.
7	0	Not Used.

Modem Status Register (Read/Write)

This register provides information on the current state of the control lines from the modem or device.

Table 8-11. Modem Status Register Definitions

Bit	Data	Definition
0	1	Set if clear to send signal (CB) input changes state.
	0	Reset when the Modem Status Register is read.
1	1	Set if data set ready (CC) input changes state.
	0	Reset when the Modem Status Register is read.
2	1	Set when the ring indicator (CE) input changes from low to a high state.
	0	Reset when the modem status register is read.
3	1	Set if the received line signal detector (CF) input changes state.
	0	Reset when the Modem Status Register is read.
4	1	Set when the CB input is active. However, if the modem loopback is enabled, this bit is equivalent to CA of XFCH (bit 1).
	0	Reset when the CB input is inactive.
5	1	Set when the CC input is active. However, if the modem loopback is enabled, this bit is equivalent to CD of XFCH (bit 0).
	0	Reset when the CC input is inactive.
6	1	Set when the CE input is active. However, if the modem loopback is enabled, this bit is equivalent to OUT1* of XFCH (bit 2).
	0	Reset when the CE input is inactive.
7	1	Set when the CF input is active. However, if the modem loopback is enabled, this bit is equivalent to OUT2 of XFCH (bit 3).
	0	Reset when the CF input is inactive.

* Indicates an active low signal.

Parallel Interface Subsystem

The Parallel subsystem provides a single standard parallel port to attach devices that accept eight bits of parallel data at standard TTL levels. It includes programmable printer control such as automatic initialization, printer select, auto linefeed, and data strobe.

Parallel Port Configuration

The parallel output port can be addressed using the computer's configuration utility (EASY CONFIG) as either parallel Port 1 or 2. These two ports are mapped to different I/O addresses and have different interrupt levels. Port 1 is mapped to I/O address 378h through 37Fh and interrupt level 7. Port 2 is mapped to I/O address 278h through 27Fh and interrupt level 5.

Parallel Port Selection Through I/O Register C40

Selection of the parallel port address is done programmatically using I/O register C40h. The bits of this register are described below.

Table 8-12. Parallel Port Selection I/O Register C40

Bits	Data	Description
0	0	Disable hard disk controller
	1	Enable hard disk controller
1	0	Primary hard disk controller address (1F0h-1F8h)
	1	Secondary hard disk controller (170h-178h)
2	0	Disable flexible disk controller
	1	Enable flexible disk controller
3	0	Primary flexible disk controller address (3FOh-3F7h)
	1	Secondary flexible disk controller address (370h-377h)
4	0	Sete parallel port to LPT1 (378h-38Fh)
	1	Set parallel port to LPT2 (278h-27Fh)
5	0	Disable parallel port
	1	Enable parallel port
6	0	Disable parallel port bidirectionality (extended mode)
	1	Enable parallel port bidirectionality (extended mode)
7	0	Disable mouse port
	1	Enable mouse port

Note:

1. Power-on values for all bits are zero.

Parallel Port Interface

The parallel port uses a D-type 25-pin female connector. Table 8-13 gives the pin assignments.

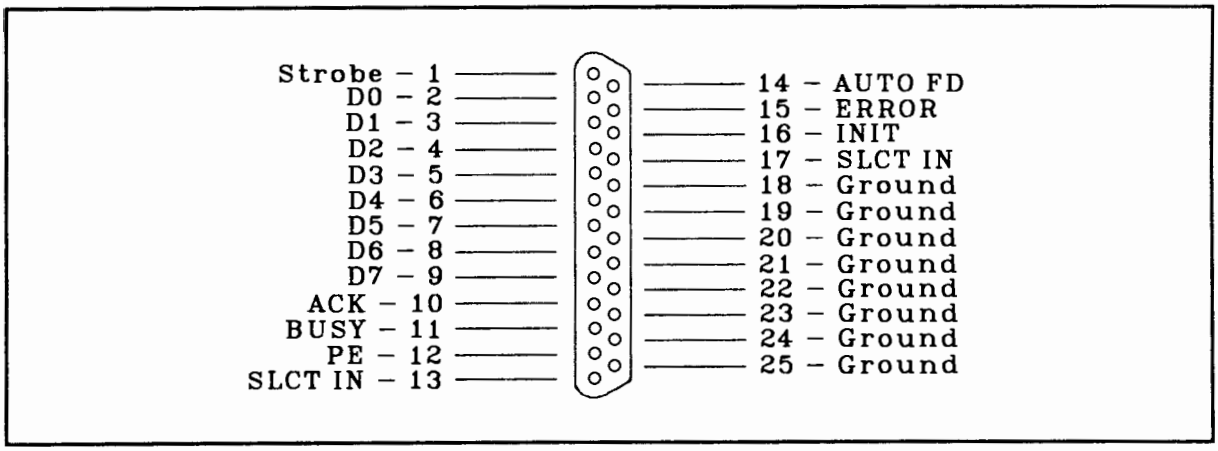


Figure 8-3. Parallel Connector

Table 8-13. Parallel Port Pin Assignments

Pin	I/O	Signal	Definition
1	O	STROBE*	Data strobe
2	O	Data 0	Data bit 0
3	O	Data 1	Data bit 1
4	O	Data 2	Data bit 2
5	O	Data 3	Data bit 3
6	O	Data 4	Data bit 4
7	O	Data 5	Data bit 5
8	O	Data 6	Data bit 6
9	O	Data 7	Data bit 7
10	I	ACK*	Printer has processed the received character.
11	I	BUSY	Printer is busy and will not accept more data.
12	I	PE	Printer detects end of paper.
13	I	SLCT	Printer select.
14	O	AUTO FD*	Printer to perform a linefeed after a line is printed.
15	I	ERROR*	Printer has encountered an error.
16	O	INIT*	Initialize printer.
17	O	SLCTIN*	Set low to enable the printer to accept data.
18-25		GND	Ground

* Indicates an active low signal.

Registers

The parallel subsystem has three programmable registers. The system programmer may gain access or control any of the registers in the parallel subsystem through the microprocessor. Table 8-14 defines the registers and their addresses.

Table 8-14. Accessible Registers

Register	R/W	Port 1	Port 2
Data Buffer Register	R/W	378h/37Ch	278h/27Ch
Printer Control Register	R/W	37Ah/37Eh	27Ah/27Eh
Printer Status Register	R	379h/37Dh	279h/27Dh

Data Buffer Register (Read/Write)

This register contains the data character. Reading from this address causes the character to be transferred from the port buffer to the host microprocessor. Writing to this address causes the character to be transmitted from the host microprocessor data bus to this port buffer and the parallel connector data line (DATA0-DATA7).

This register is cleared at power-up by the reset input signal.

Printer Control Register (Read/Write)

This register controls the printer signals. It is cleared at power-up by the reset input signal.

During a Write operation, bit 0 through 3 are output to parallel port STROBE* (pin 1), AUTOFD* (pin 14), INIT* (pin 16) and SLCTIN* (pin 17) respectively. During a Read operation, the levels on signals STROBE*, AUTOFD* and SLCTIN* are inverted and read as bits 0, 1 and 3. The level on signal INIT and IRQ are read as bit 2 and bit 4, respectively, directly with no inversion.

Table 8-15. Printer Control Register Definitions

Bit	Data	Definition
0	1	Generates an active low STROBE signal for a minimum of 500 ns. The STROBE signal clocks the data from the parallel port into the printer. The valid data must be present a minimum of 0.5 μs before and after the STROBE signal.
	0	STROBE inactive.
1	1	Generates the low AUTO FD signal.
	0	AUTO FD inactive.
2	0	Generates the low INIT* signal for a minimum of 50 μs.
	1	INIT* inactive.
3	1	Generates the low SLCTIN* signal.
	0	SLCTIN inactive.
4	1	Enables the IRQ when the ACK* input signal changes from true to false.
	0	Disable IRQ.

5		This bit is used to control the parallel port drivers. It works in conjunction with pin 1 (BIDEN) of the WD16C552 controller. The state of pin 1 is set via a positive edge triggered latch on the System Interface Board (port C40) to obtain the following results:
		`Port Mode  Pin 1  Bit 5  Direction`
		`Compatible Mode  0  0  Write`
		`Extended Mode  1  0  Write`
		`Extended Mode  1  1  Read`

6-7	0	Always set to 0.

Printer Status Register (Read Only)

This register provides information on the control lines from the printer. This register is cleared at power-up by the reset input signal.

Table 8-16. Printer Status Register Definition

Bit	Data	Definition
0-2		Not Used.
3	0	The ERROR* input is active. The printer is either at the paper end, at an off-line state, or an error state.
4	1	The SLCT input is active. The printer is selected.
5	1	The PE input is active. The printer is out of paper.
6	0	The ACK* input is active. The printer has received the character and is ready to accept another character.
7	0	The BUSY input is active. The printer is busy and not ready to accept data.

* Indicates an active low signal.

9

Power Supply

Introduction

This chapter provides information about the power supply's physical and electrical specifications. The power supply is a dc-switching power supply designed for continuous operation at approximately 264 Watts. It has a double pole fused 120 Vac input and provides 4 regulated dc output voltages. The dc outputs are over-voltage, over-current, open-circuit and short-circuit protected. The power supply has a single connection for system board power and six connectors for disk drives.

In addition to providing power, the power supply provides a reset signal necessary for system board, expansion board, and keyboard operation. In conjunction with the system board fan, the power supply fan provides cooling for the entire system.

The major components of the power supply are listed below:

- Ac input receptacle
- Fan
- Fuses and fuse holders
- On/off switch
- Power cables to disk drives
- Power connector to System Board

As shown in Figure 9-1, the power supply is installed in a zinc-plated steel enclosure, located inside the system processing unit (SPU). The power supply provides power to the following components:

- Backplane I/O
- Disk and tape drives
- Keyboard and other input devices
- Printed circuit boards

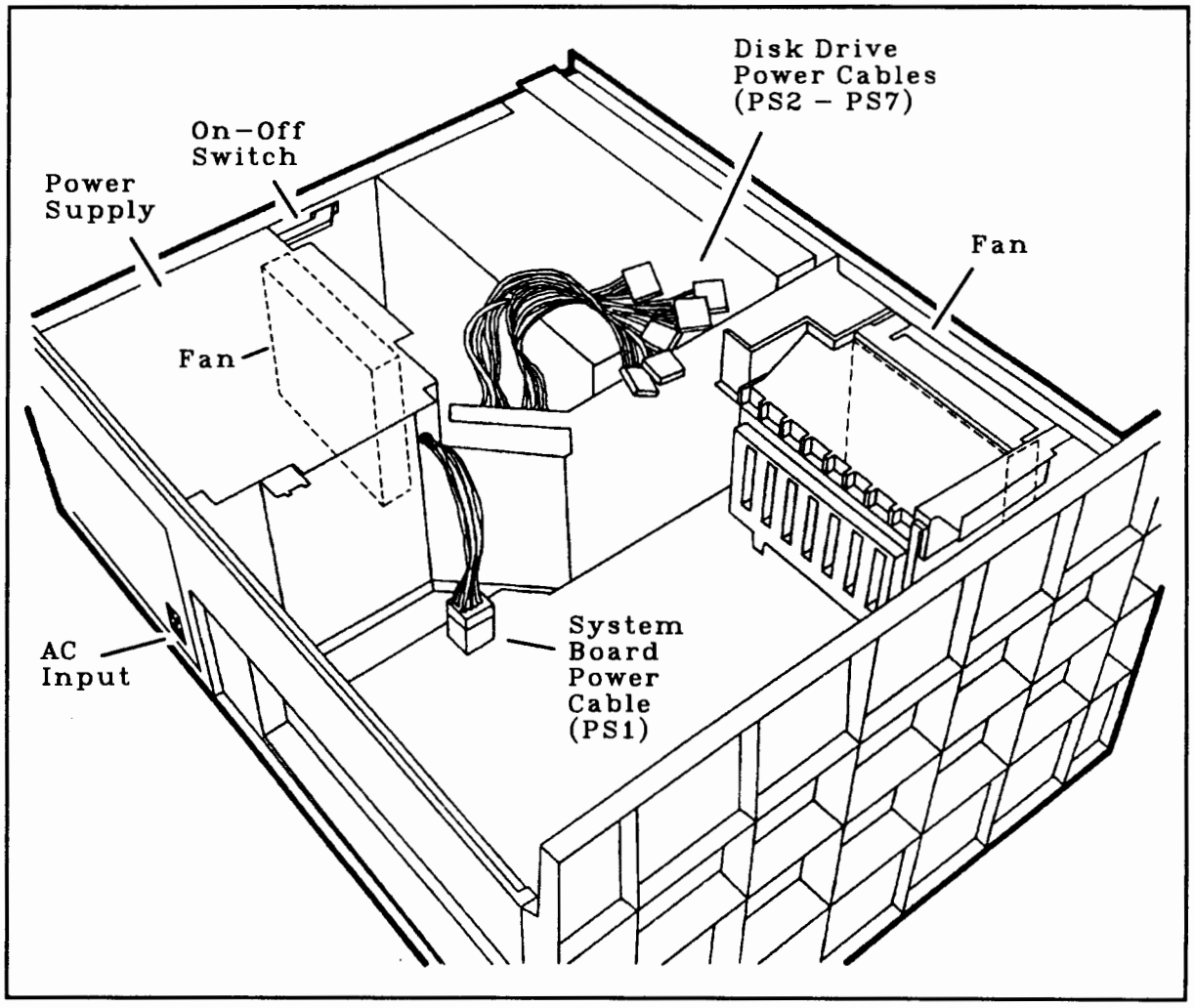


Figure 9-1. Power Supply and components

Power Supply Line Input

The auto-ranging, power supply senses the line voltage and frequency and automatically configures itself to operate with one of two voltages shown below:

Table 9-1. Power Supply Line Input Voltages and Frequencies

Line Voltage	Line Frequency	Configuration Voltage
90 to 132 Vac	47 to 63 Hz	115 Vac
198 to 264 Vac	47 to 63 Hz	230 Vac

Power Supply Line Output

Through PS1, the power supply provides ±5 Vdc and ±12 Vdc to the printed circuit assemblies. The PFAIL signal discussed in section "Power Supply Operating Status Indicator."

Through the six identical power cables (PS2 through PS7), the power supply provides +5 Vdc and +12 Vdc outputs to mass storage devices.

The following figure shows the pinouts for the power supply connectors and cables. Table 9-2 and Table 9-3 give for the power supply connectors: the pin assignments, the voltage output, the connector wire color, and the maximum current drawn per pin. Table 9-4 gives the power supply's output capabilities at power-up and 30 seconds after power-up. Table 9-5 gives total power at power-up and 30 seconds after power-up.

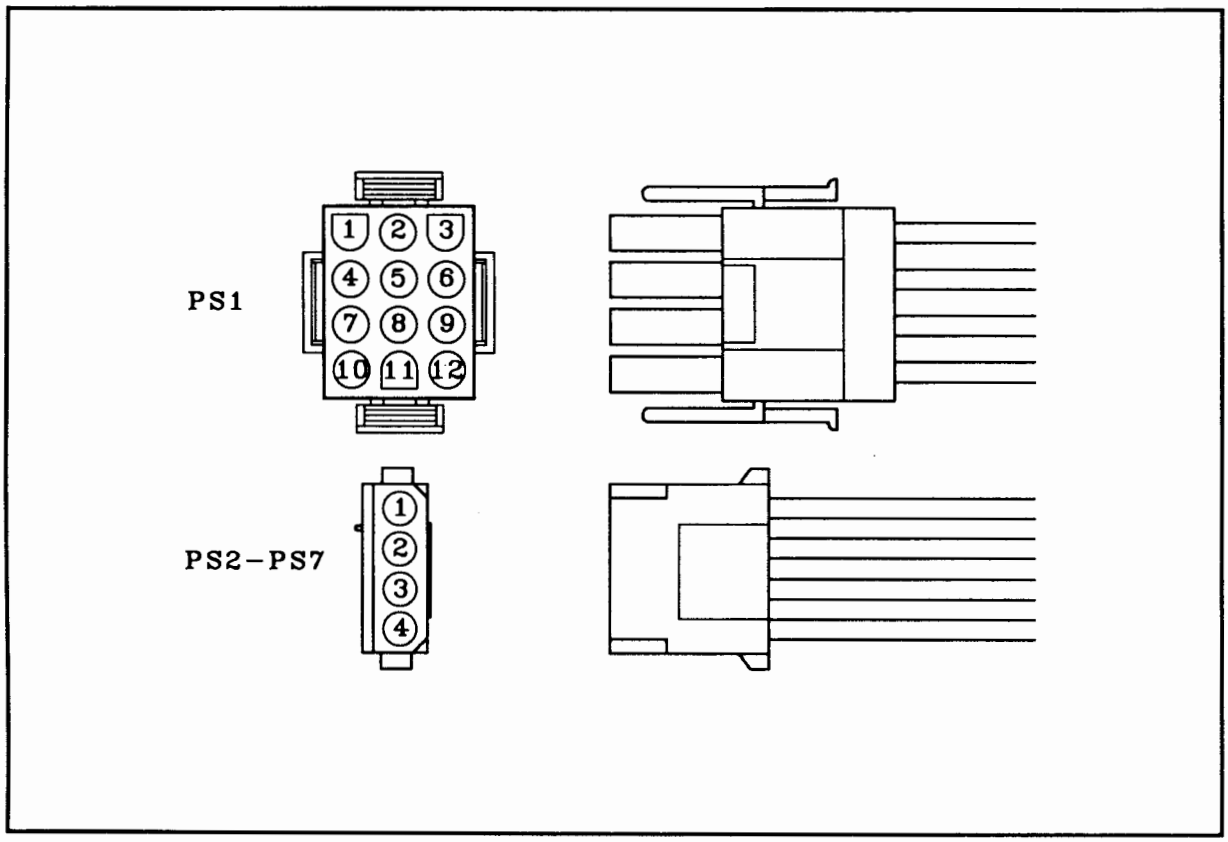


Figure 9-2. Pinouts for Power Supply Connector and Cables

Table 9-2. Power Supply Connector PS1 Pin Assignments

PS1 Pins	Vdc Output	Wire Color	Maximum Current Drawn (Per Pin)
1	Ground	Black	—
2	+12 Vdc	Yellow	2 A
3	+5 Vdc	Red	7 A
4	Ground	Black	—
5	PFAIL	Blue	(*see note 1*)
6	+5 Vdc	Red	7 A
7	Ground	Black	—
8	−12 Vdc	Green	1 A
9	+5 Vdc	Red	7 A
10	Ground	Black	—
11	−5 Vdc	Violet	0.3 A
12	+5 Vdc	Red	7 A

Table 9-3. Power Supply Connectors PS2-PS7 Pin Assignments

PS2-PS7 Pins	Vdc Output	Wire Color	Maximum Current Drawn (PS2-PS7 combined, *see note 2*)
1	+12 Vdc	Yellow	5 A
2, 3	Ground	Black	—
4	+5 Vdc	Red	5 A

Notes:

1. Refer to the "Power Supply Operating Status Indicator" section in this chapter for information.
2. During power-up, power supply connectors PS2-PS7 can provide a total maximum current of 13 Amps on the +12 Vdc output and 5 Amps on the +5 Vdc power supply.

Table 9-4. Power Supply Rated Output at Power-up and After Power-Up

Nominal Output (Vdc) From Power Supply	Voltage Range (Vdc) at Power-up and 30 Seconds After	Minimum Current (dc Amps) at Power-up and 30 Seconds After	Maximum Current (dc Amps) at Power-up	Maximum Current (dc Amps) 30 Seconds After Power-up	Maximum Voltage Ripple (mV p-p) at Power-up and 30 Seconds After
+5	+4.85 to +5.25	5.5	33	33	50
+12	+11.6 to +12.6	0	15	7	100
−5	−5.25 to −4.75	0	0.3	0.3	50
−12	−12.6 to −11.4	0	1.0	1.0	120

Table 9-5. Power Supply Total Power

Total Power (Watts) at Power-up	Total Power (Watts) 30 Seconds After Power-up
28 minimum to 360 maximum	28 minimum to 264 maximum

Power to Backplane I/O

Each of the +5 Vdc backplane I/O connector slots can support an average of eleven watts per connector. The following table gives the voltage for the backplane I/O connector slots, the maximum total current for the backplane I/O connector slots, and the average current per backplane I/O connector slot.

Table 9-6. Power to Backplane I/O Connector Slots

Backplane I/O Voltage	Maximum Total Current for Slots	Average / Slot
+ 5 Vdc	16 A	2 A
+12 Vdc	1.5 A	0.19 A
−5 Vdc	0.3 A	0.04 A
−12 Vdc	1 A	0.13 A

Power Supply Environmental Specifications

Temperature Specifications

Operating temperature: +5° to +40° C (+41° to +104° F)
Storage temperature (14 days maximum): −40° C to +70° C (−40° F to 158° F)

Humidity Specifications

Operating humidity: 15% to 80% relative humidity at operating temperature range

Altitude Specifications

Operating altitude: 0 to 4,600 meters (0 to 15,000 feet)
Storage altitude: 0 to 15,250 meters (0 to 50,000 feet)

Power Supply Protection

Line Dropout Protection

Under any condition, including low line power at rated load, the power supply will continue to deliver regulated outputs if a power line dropout of up to 20 milliseconds occurs at the input to the power supply.

AC Inrush Current Protection

Under any condition, AC inrush current for the power supply is limited to 55 Amps peak for up to 20 milliseconds after power-on from a completely powered-off state.

Undercurrent Protection

Any or all power supply outputs may be open (i.e., no loads on PS1-PS7) without damage to the power supply. (For the +5 Vdc power supply to operate normally, the minimum load is 5.5 Amps.) All dc outputs will not exceed overvoltage limits.

Overcurrent Protection

All power supply outputs are internally protected against a short-circuit. During a short-circuit condition, no power supply output will exceed 240 Vac. If a short circuit does occur, the power supply shuts off. In the event of a short-circuit, turn the power supply OFF, remove the short-circuit, and then turn the power supply ON.

Overvoltage Protection

Table 9-7 gives the voltage levels which cause a power supply overvoltage situation. If overvoltage occurs on the +5 or +12 Vdc line, within 500 microseconds, the power supply's switch controller turns off all the power supply outputs. To start the power supply again, turn the power supply OFF, correct the fault, and then turn the power supply ON.

If overvoltage occurs on the −5 or −12 Vdc line, the +5 and +12 Vdc outputs will not turn off, as the −5 and −12 Vdc outputs are internally protected. When the cause of the overvoltage situation is corrected, the −5 or −12 Vdc outputs return to normal operation.

Table 9-7. Voltage Levels Which Cause Power Supply Overvoltage

Nominal Power Supply Voltage	Maximum Over-Voltage Level
5 Vdc	+ 6.5 Vdc
+12 Vdc	+17.0 Vdc
−5 Vdc	*
−12 Vdc	*

(*) Inherent overvoltage protection at any voltage level.

Power Supply Fuses

To protect the computer from AC input overcurrent, the power supply has two fuses, rated at 10 Amps. These fuses are not user-replaceable items.

Power Supply Fan

To protect the computer from overheating, all models have a power supply fan that provides 0.65 cubic meters (23 cubic feet) of air per minute to cool the power supply and disk drives. In addition, a system fan in the SPU cools the printed circuit boards.

Power Supply Operating Status Indicator (PFAIL)

The PFAIL logic signal (pin 5 of the PS1 power connector) indicates the power supply's operating status. If the power supply is operating properly, within 300 milliseconds after the +5 Vdc output has reached its minimum level, the PFAIL signal goes low to indicate proper power supply operation.

If the power supply is not operating properly, the PFAIL signal goes high, which causes SPWROK to go low. When SPWROK is low, it resets the system and holds it reset until SPWROK goes high.

When the AC input voltage either has been removed or has dropped to an insufficient level, the rising edge of the PFAIL signal indicates to the real-time clock and CMOS RAM to go into a power-down mode and operate from the system's battery.

The PFAIL signal always indicates the correct power status even if output voltages do not fall to 0. The PFAIL timing diagram below shows voltage levels and timing.

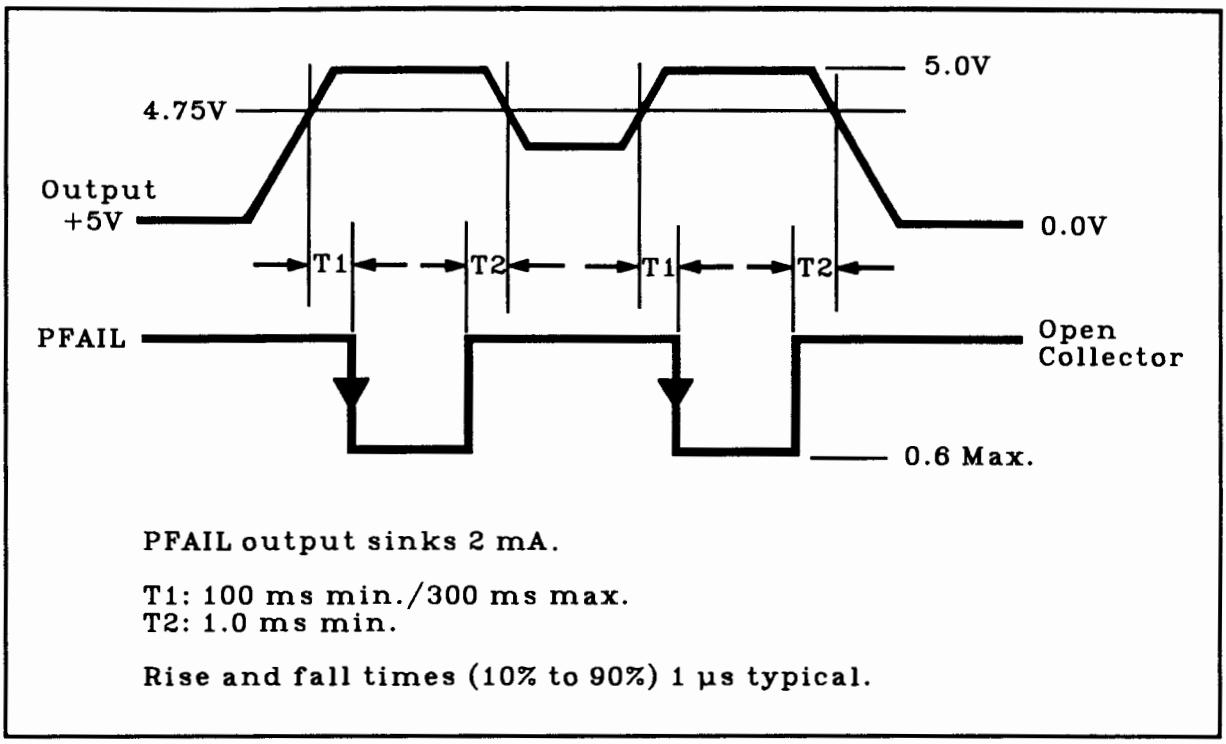


Figure 9-3. PFAIL Timing Diagram

Note: PFAIL output is open collector, and sinks 2 mA. In the OFF state, PFAIL current draw does not exceed 0.001 mA.

Battery Power

A system backup battery, located on the System Board, provides power to the real-time clock and CMOS RAM during power-down and power system failure. The 6-volt lithium battery has an average life of 2.3 years.

A

References

For general information on the HP Vectra 486/33T system, refer to the following:

- *HP Vectra 486/33T BIOS Technical Reference Manual*
  HP product number 5960-0749
  Technical reference information on the HP Vectra 486/33T's System BIOS.
- *HP Vectra 486/33T Configuration and Service Manual*
  HP product number 5960-0746
  Parts lists, installation, configuration, and troubleshooting information.
- *Dealer Configuration File Creation Guide*
  HP part number D2230-90001
  A reference for creating CFG files for use with EASY CONFIG.
- *Setting Up And Using Your HP Vectra 486/33T*
  Instructions on how to use the EXMODE command. Provided with the HP Vectra 486/33T PC.

  Hewlett-Packard provides a toll-free number (1-800-538-8787) for literature requests.
- *Inside the EISA Computers* by *Tony Dowden*
  An introduction to the EISA computers, including an overview of their operation, and technical discussions of both the hardware and software aspects of the EISA standard. Also includes DMA and I/O structures, video, serial and parallel interfaces, programming examples, and a glossary of terms. Published by:

  Addison-Wesley Publishing Company, Inc.
  ISBN 0-201-52397-3
- *EISA Specification* BCPR Services, Inc.
  Reference for EISA-specific information on the EISA bus, DMA, interrupt, memory, and the designing of EISA expansion boards and configuration files. Available from:

  BCPR Services, Inc.
  A Delaware Corporation
  1400 L Street, N.W.
  Washington, D.C. 20005-3502
  Phone: (202) 371-5921
- *CFG File Extensions Specification*
  Reference to configuration (CFG) file extensions specific to the system configuration utility. Available from:

  Micro Computer Systems Inc.
  2300 Valley View Lane
  Irving, Texas
  Phone: (214) 659-1514

- *Intel 82357 Integrated System Peripheral Data Sheet*
  Reference for the interrupt controller, DMA controller, system arbiter logic, interval counter/timers, and refresh.
- *Intel 82358 EISA Bus Controller Data Sheet*
  Reference for EISA bus controlling, ISA/EISA data conversion, interaction with the 82357 ISP for transfers of DMA between buses, I/O recovery logic, and integrated synchronous system reset.
- *Intel 80386 Programmer's Reference Manual*
  Reference for a description of the 80386 architecture used in the 80486, and for a description of the complete register set used in the 80386 and the 80486.
- *Intel i486 Microprocessor*
  Reference for the 80486 microprocessor, the 80486 on-chip code and data cache, the 80486 on-chip coprocessor (floating point unit), and the 80486 on-chip memory management unit.
- *Intel Microprocessor Handbook. Volume I—Microprocessor*
  Reference for the 80386 microprocessor.
- *i486 Microprocessor Data Sheet*
  Reference for a detailed description of the operation of the 80486 microprocessor.
- *Intel 486 Processor Bus Specification*
  Reference for a detailed description of the 486's on-chip cache memory.

  Intel publications may be obtained by writing to the following address:

  Intel Literature Sales
  P.O. Box 58130
  Santa Clara, CA 95052-8130

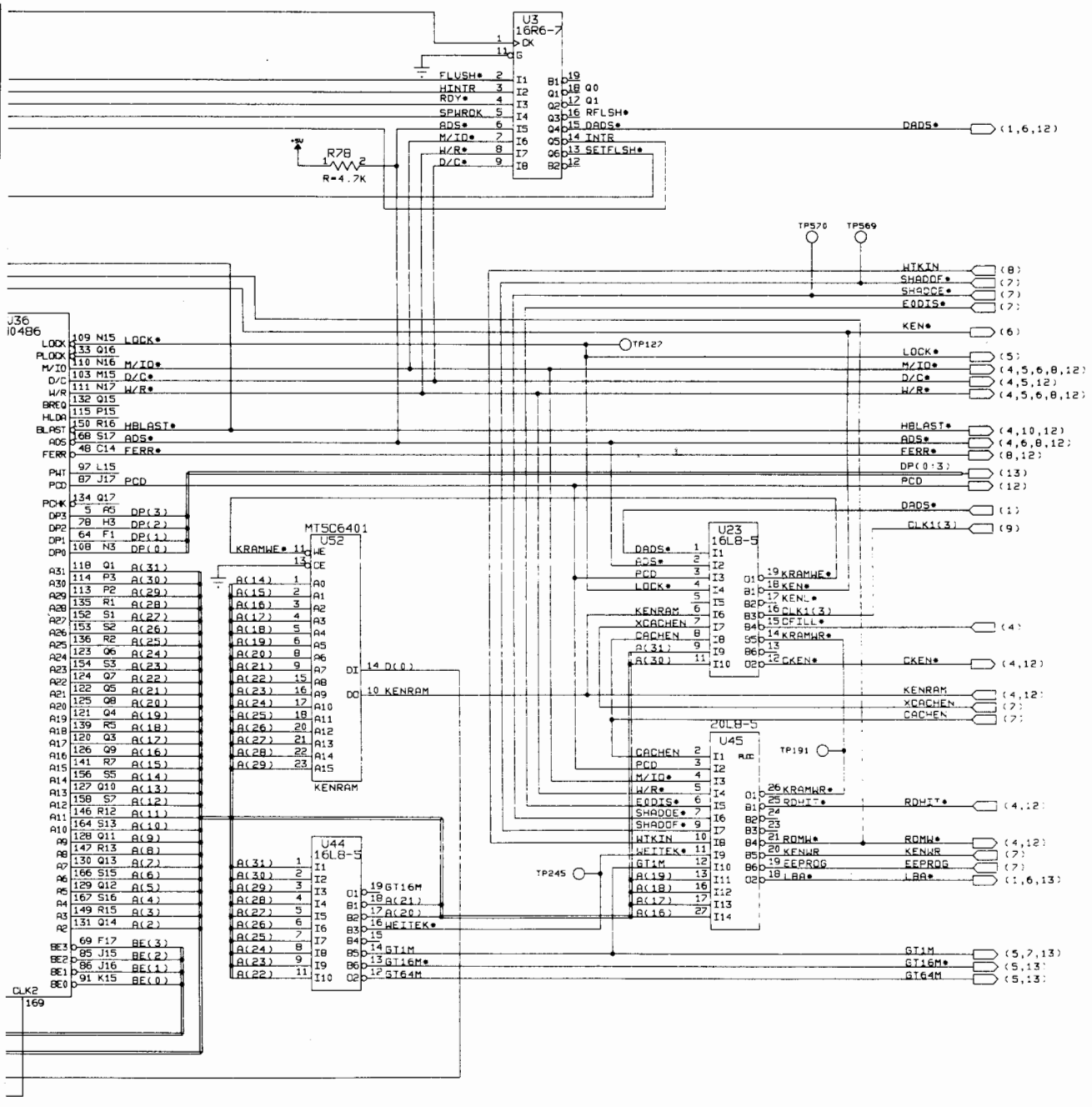
  Intel also provides a toll-free number (1-800-548-4725) for literature requests.
- *Motorola Single Chip Microcomputer Data, Section C.*
  Reference for the MC146818 real-time clock/CMOS RAM. Available from the Motorola Corporation.
- *Motorola MC1489, MC1489A Data Book*
  Reference for the RS-232C quad line receivers. Available from the Motorola Corporation.
- *Motorola MC145406 Data Book*
  Reference for the RS-232C/V.28 driver/receiver. Available from the Motorola Corporation.
- *Western Digital WD16C452/WD16C552 Asynchronous Communications Element Data Book*
  Reference to the Serial/Parallel I/O device controller for a bit-by-bit description of the WD16C522's registers. Available from the Western Digital Corporation.

B

SCHEMATICS

The following pages provide schematic drawings for the HP Vectra 486/33T PC.

Figure B-1. CPU Board (486 CPU)
Schematics B-3/B-4


U3
16R6-7
FLUSH•
HINTR
RDY•
SPWROK
ADS•
M/IO•
W/R•
D/C•
RFLSH•
DADS•
INTR
SETFLSH•
R78
R=4.7K
TP570
TP569
TP127
TP191
TP245
WTKIN
SHADOF•
SHADCE•
EODIS•
KEN•
LOCK•
M/IO•
D/C•
W/R•
HBLAST•
ADS•
FERR•
DP(0:3)
PCD
CLK1(3)
U36
10486
LOCK•
M/IO•
D/C•
W/R•
HLDA
BLAST•
FERR•
PWT
PCD
DP3
DP2
DP1
DP0
MT5C6401
U52
KRAMWE•
KENRAM
U23
16L8-5
KRAMWE•
KEN•
CLK1(3)
CFILL•
KRAMWR•
CKEN•
XCACHEN
CACHEN
U45
20L8-5
RDWIT•
ROMW•
KENWR
EEPROG
LBA•
U44
16L8-5
GT16M
GT1M
GT64M
WEITEK•
CLK2
169

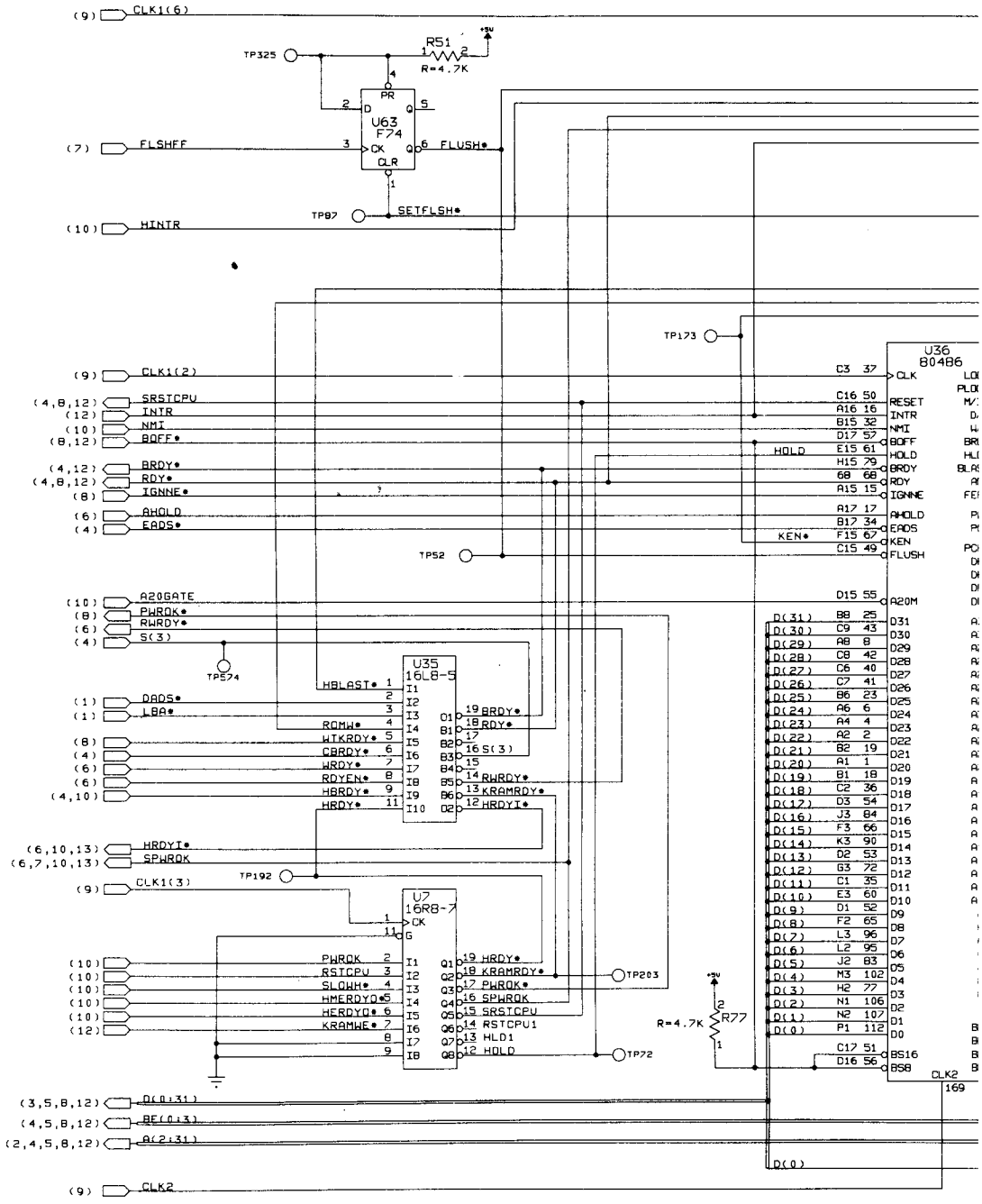
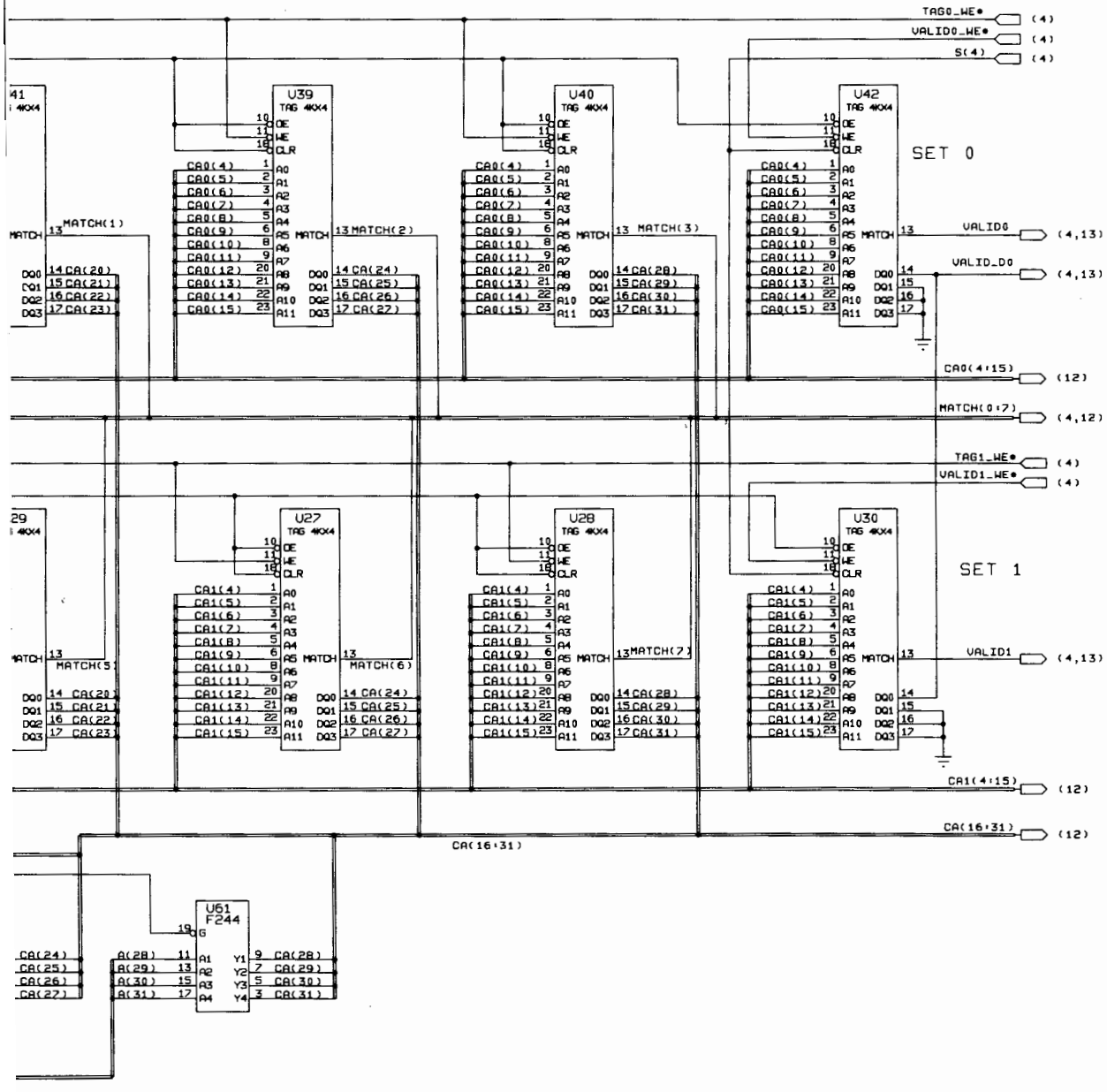

(9) CLK1(6)
TP325
R51
R=4.7K
U63
F74
(7) FLSHFF
FLUSH*
TP87
SETFLSH*
(10) HINTR
TP173
U36
80486
(9) CLK1(2)
(4,8,12) SRSTCPU
(12) INTR
(10) NMI
(8,12) BOFF*
(4,12) BRDY*
(4,8,12) RDY*
(8) IGNNE*
(6) AHOLD
(4) EADS*
HOLD
KEN*
TP52
(10) A20GATE
(8) PWRQK*
(6) RWRDY*
(4) S(3)
TP574
U35
16L8-5
HBLAST*
(1) DADS*
(1) LBA*
ROMW*
WTKRDY*
CBRDY*
WRDY*
RDYEN*
HBRDY*
HRDY*
BRDY*
RDY*
S(3)
RWRDY*
KRAMRDY*
HRDYI*
(6,10,13) HRDYI*
(6,7,10,13) SPWRQK
TP192
(9) CLK1(3)
U7
16R8-7
PWRQK
RSTCPU
SLOWW*
HMERDYD*
HERDYD*
KRAMWE*
HRDY*
KRAMRDY*
PWRQK*
SPWRQK
SRSTCPU
RSTCPU1
HLD1
HOLD
TP203
TP72
R=4.7K
R77
D(31) D31
D(30) D30
D(29) D29
D(28) D28
D(27) D27
D(26) D26
D(25) D25
D(24) D24
D(23) D23
D(22) D22
D(21) D21
D(20) D20
D(19) D19
D(18) D18
D(17) D17
D(16) D16
D(15) D15
D(14) D14
D(13) D13
D(12) D12
D(11) D11
D(10) D10
D(9) D9
D(8) D8
D(7) D7
D(6) D6
D(5) D5
D(4) D4
D(3) D3
D(2) D2
D(1) D1
D(0) D0
A20M
BS16
BS8
CLK2
169
(3,5,8,12) D(0:31)
(4,5,8,12) BE(0:3)
(2,4,5,8,12) A(2:31)
D(0)
(9) CLK2

Figure B-2. CPU Board (Cache Tag RAM)
Schematics B-5/B-6


TAG0_WE*
VALID0_WE*
S(4)
SET 0
U39
U40
U42
TAG 4KX4
VALID0
VALID_D0
CA0(4:15)
MATCH(0:7)
TAG1_WE*
VALID1_WE*
SET 1
U27
U28
U30
VALID1
CA1(4:15)
CA(16:31)
U61
F244

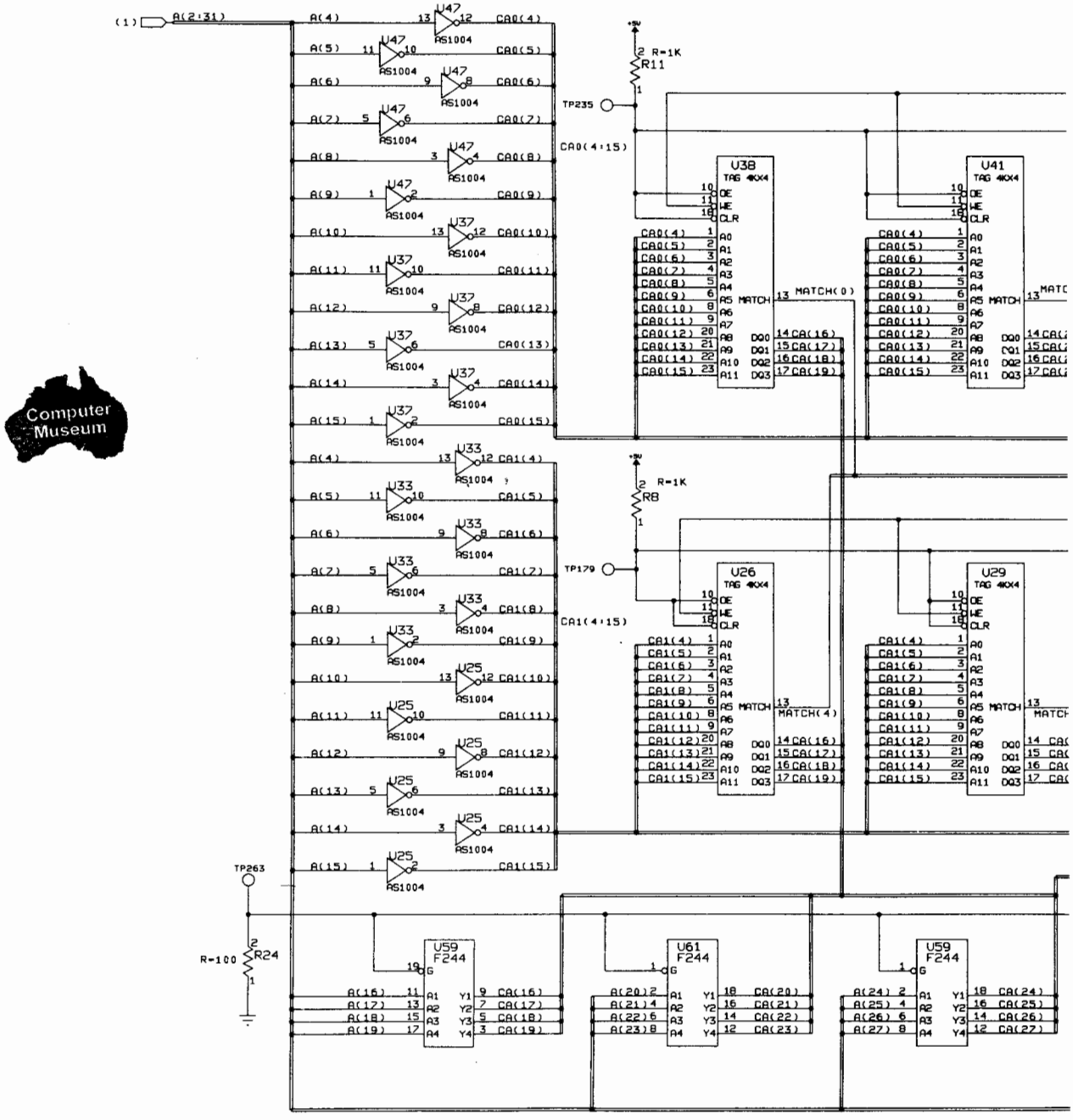
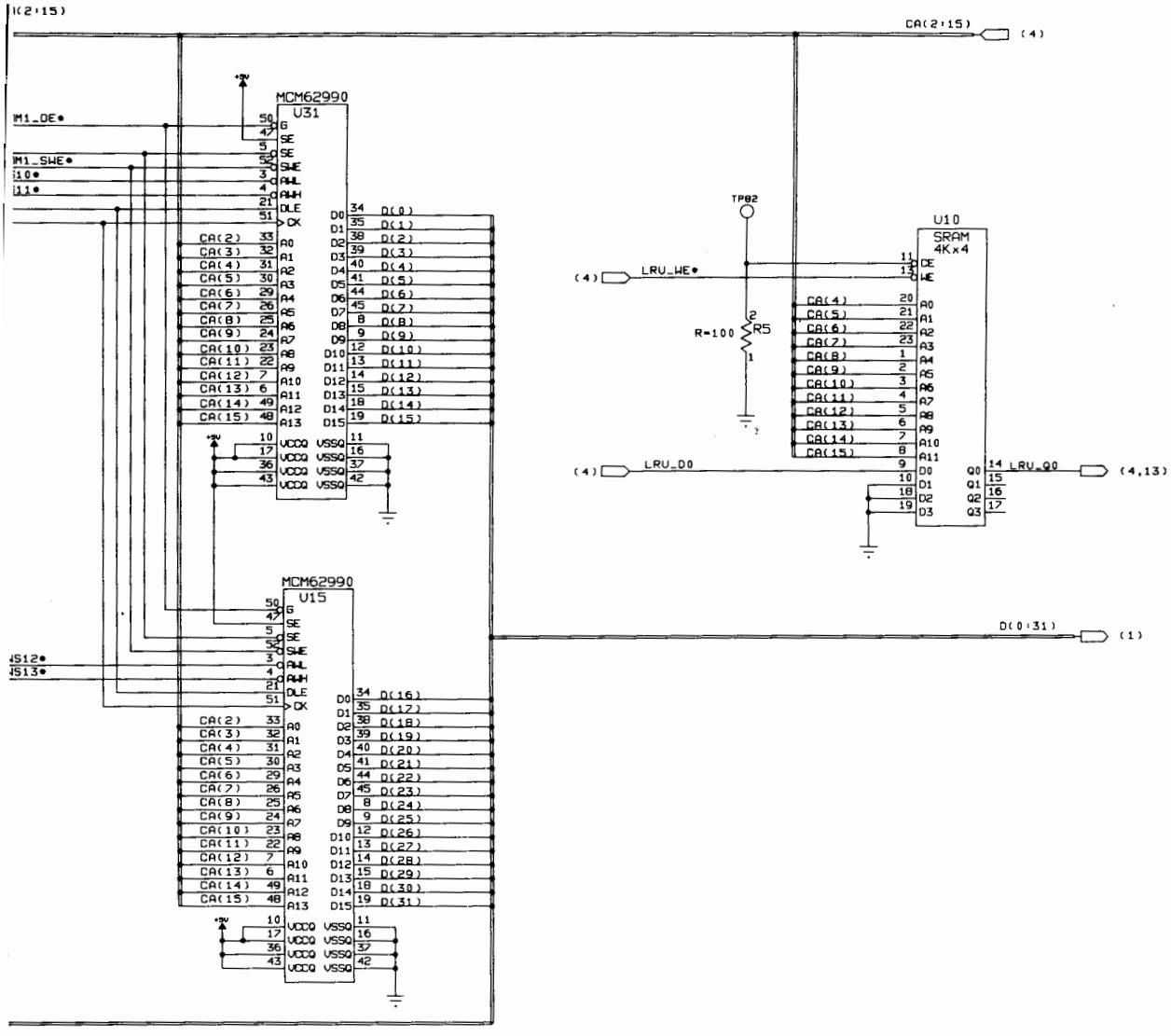


Figure B-3. CPU Board (Cache Data RAM)
Schematics B-7/B-8

SET 1


CA(2:15)
(4)
MCM62990
U31
M1_OE*
M1_SWE*
MCM62990
U15
D(0:31)
(1)
TP82
R=100
R5
LRU_WE*
LRU_DO
LRU_QO
(4,13)
U10
SRAM
4Kx4

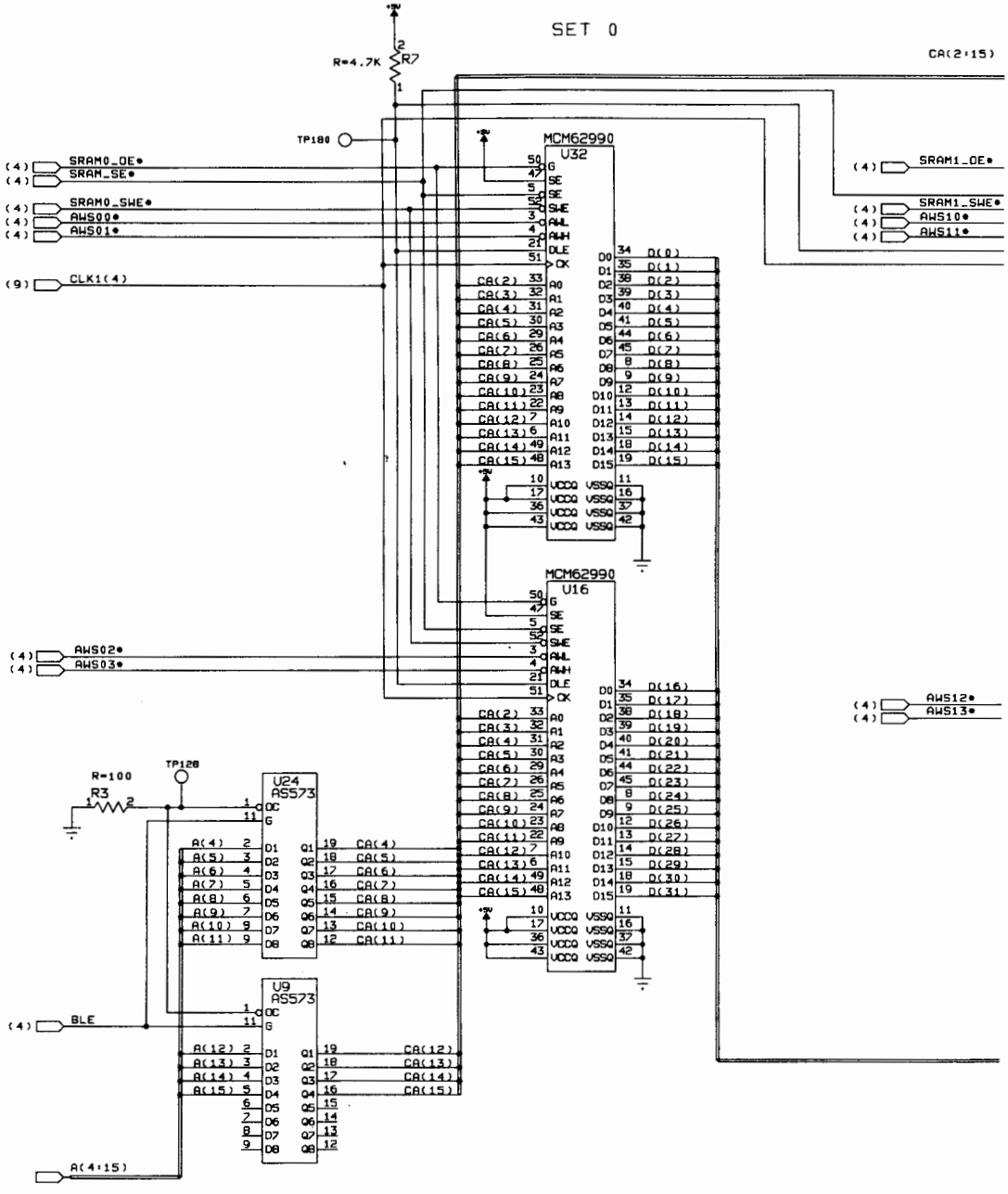
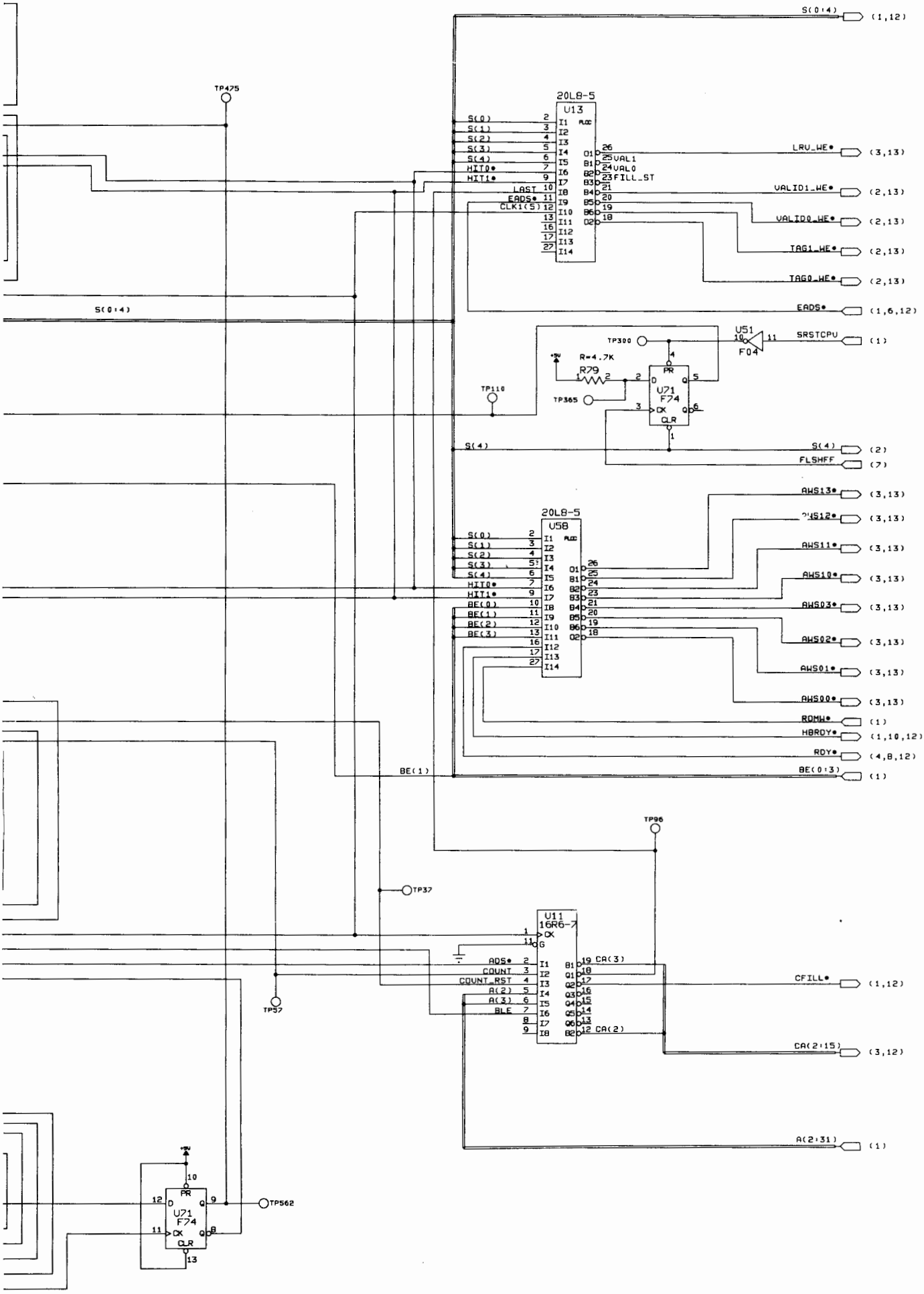

SET 0
R=4.7K
R7
CA(2:15)
TP180
MCM62990
U32
SRAM0_OE*
SRAM_SE*
SRAM0_SWE*
AWS00*
AWS01*
CLK1(4)
SRAM1_OE*
SRAM1_SWE*
AWS10*
AWS11*
MCM62990
U16
AWS02*
AWS03*
AWS12*
AWS13*
R=100
R3
TP128
U24
AS573
U9
AS573
BLE
A(4:15)

Figure B-4. CPU Board (Cache Controller)
Schematics B-9/B-10


S(0:4)
(1,12)
TP475
20L8-5
U13
S(0)
S(1)
S(2)
S(3)
S(4)
HIT0•
HIT1•
LAST
EADS•
CLK1(5)
VAL1
VAL0
FILL_ST
LRU_WE•
(3,13)
VALID1_WE•
(2,13)
VALID0_WE•
(2,13)
TAG1_WE•
(2,13)
TAG0_WE•
(2,13)
EADS•
(1,6,12)
S(0:4)
TP300
U51
F04
SRSTCPU
(1)
+5V
R=4.7K
R79
TP365
TP110
U71
F74
PR
CK
CLR
S(4)
S(4)
(2)
FLSHFF
(7)
20L8-5
U58
BE(0)
BE(1)
BE(2)
BE(3)
AWS13•
AWS12•
AWS11•
AWS10•
AWS03•
AWS02•
AWS01•
AWS00•
(3,13)
ROMW•
(1)
HBRDY•
(1,10,12)
RDY•
(4,8,12)
BE(1)
BE(0:3)
(1)
TP96
TP37
TP57
U11
16R6-7
G
ADS•
COUNT
COUNT_RST
A(2)
A(3)
BLE
CA(3)
CA(2)
CFILL•
(1,12)
CA(2:15)
(3,12)
A(2:31)
(1)
TP562

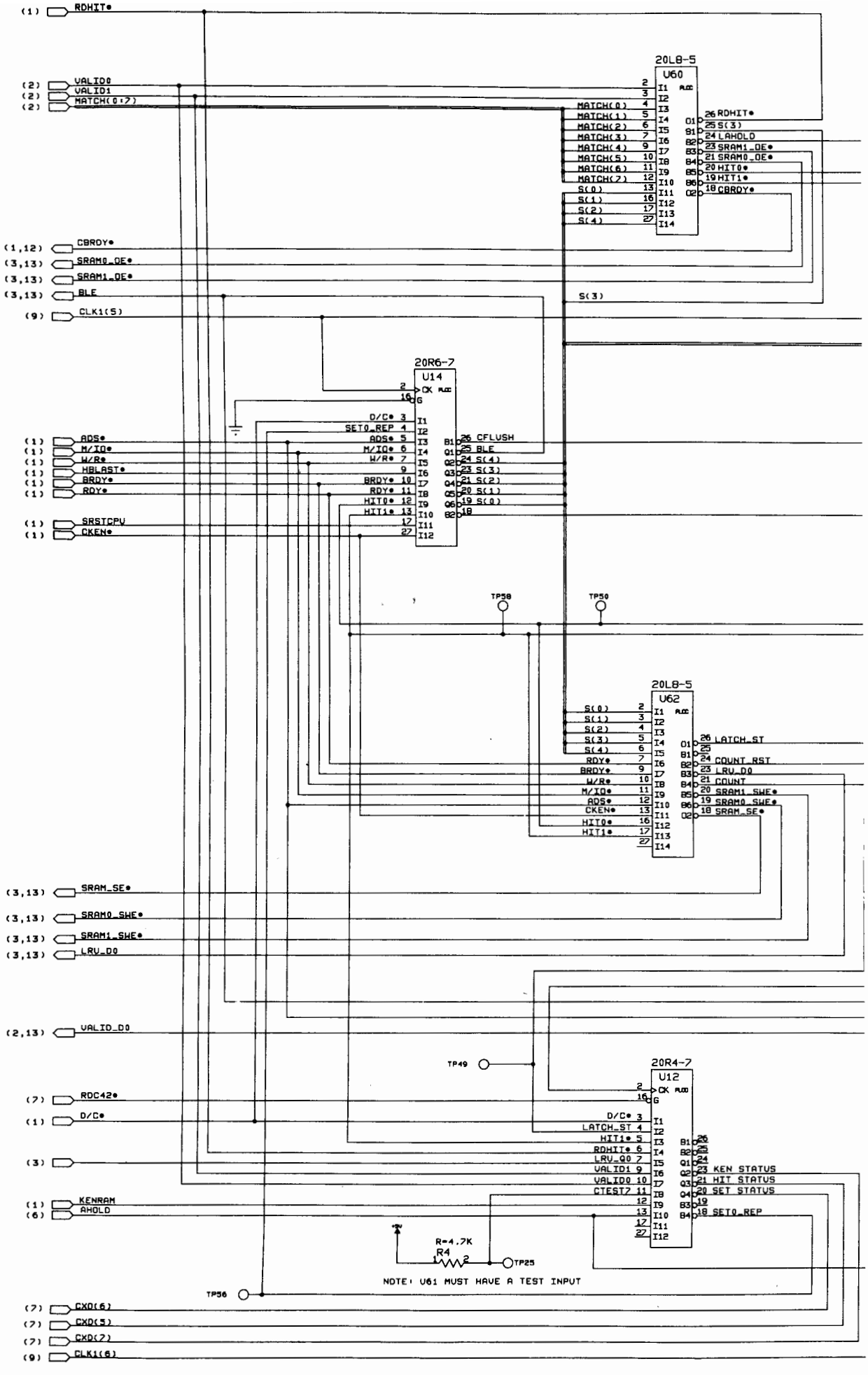
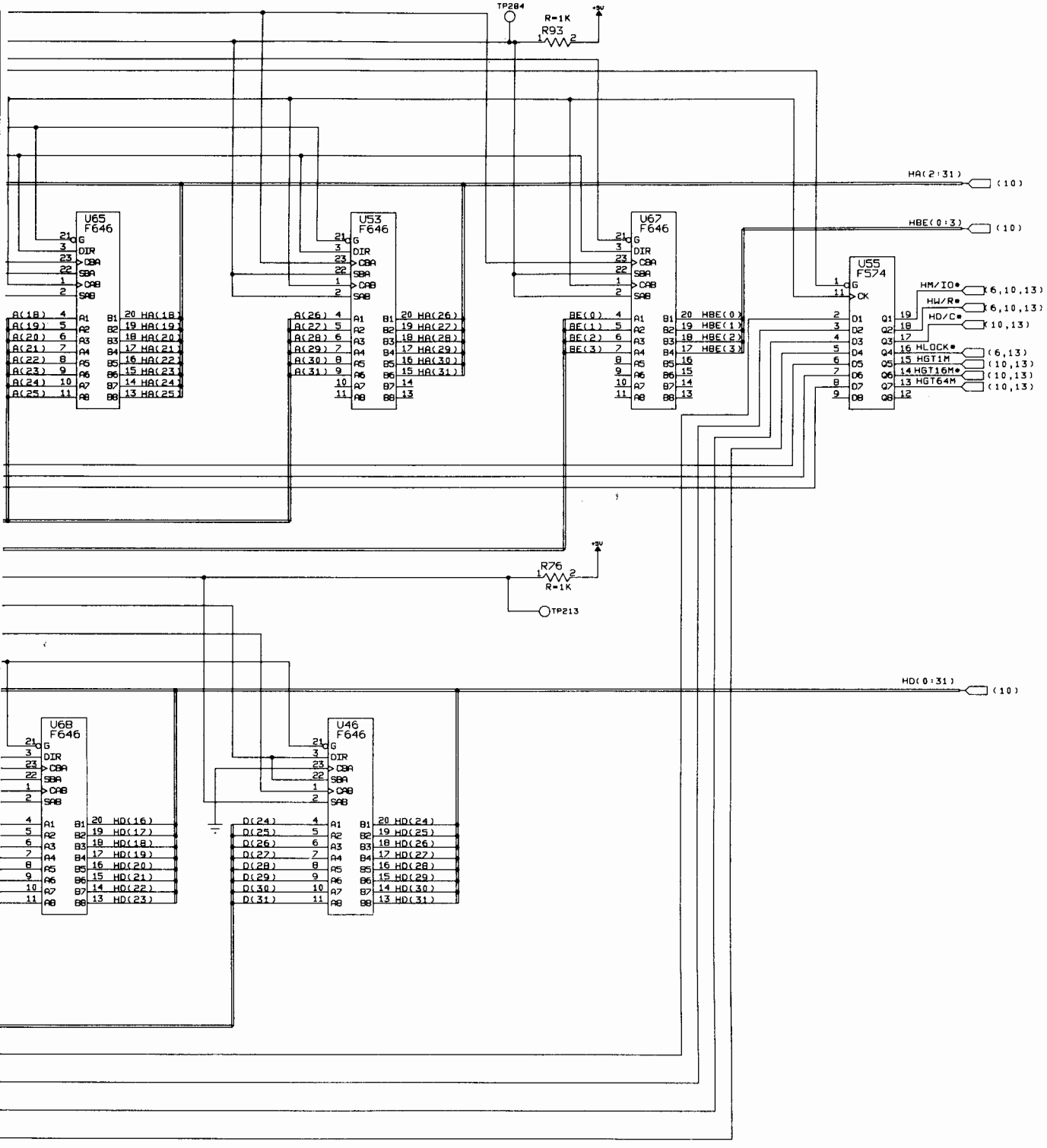

(1) RDHIT*
(2) VALID0
(2) VALID1
(2) MATCH(0:7)
20L8-5
U60
MATCH(0)
MATCH(1)
MATCH(2)
MATCH(3)
MATCH(4)
MATCH(5)
MATCH(6)
MATCH(7)
S(0)
S(1)
S(2)
S(4)
RDHIT*
S(3)
LAHOLD
SRAM1_OE*
SRAM0_OE*
HIT0*
HIT1*
CBRDY*
(1,12) CBRDY*
(3,13) SRAM0_OE*
(3,13) SRAM1_OE*
(3,13) BLE
(9) CLK1(5)
S(3)
20R6-7
U14
CK
G
D/C*
SET0_REP
ADS*
M/IO*
W/R*
BRDY*
RDY*
HIT0*
HIT1*
CFLUSH
BLE
S(4)
S(3)
S(2)
S(1)
S(0)
(1) ADS*
(1) M/IO*
(1) W/R*
(1) HBLAST*
(1) BRDY*
(1) RDY*
(1) SRSTCPU
(1) CKEN*
TP58
TP50
20L8-5
U62
S(0)
S(1)
S(2)
S(3)
S(4)
RDY*
BRDY*
W/R*
M/IO*
ADS*
CKEN*
HIT0*
HIT1*
LATCH_ST
COUNT_RST
LRU_D0
COUNT
SRAM1_SWE*
SRAM0_SWE*
SRAM_SE*
(3,13) SRAM_SE*
(3,13) SRAM0_SWE*
(3,13) SRAM1_SWE*
(3,13) LRU_D0
(2,13) VALID_D0
TP49
20R4-7
U12
CK
G
D/C*
LATCH_ST
HIT1*
RDHIT*
LRU_Q0
VALID1
VALID0
CTEST2
KEN STATUS
HIT STATUS
SET STATUS
SET0_REP
(7) RDC42*
(1) D/C*
(3)
(1) KENRAM
(6) AHOLD
+5V
R=4.7K
R4
TP25
NOTE: U61 MUST HAVE A TEST INPUT
TP56
(7) CXD(6)
(7) CXD(5)
(7) CXD(2)
(9) CLK1(6)

Figure B-5. CPU Board (Write Buffer)
Schematics B-11/B-12


TP284
R=1K
R93
+5V
HA(2:31)
(10)
HBE(0:3)
(10)
U65
F646
U53
F646
U67
F646
U55
F574
HM/IO*
HW/R*
HD/C*
HLOCK*
HGT1M
HGT16M*
HGT64M
(6,10,13)
(6,10,13)
(10,13)
(6,13)
(10,13)
(10,13)
(10,13)
R76
R=1K
TP213
HD(0:31)
(10)
U68
F646
U46
F646

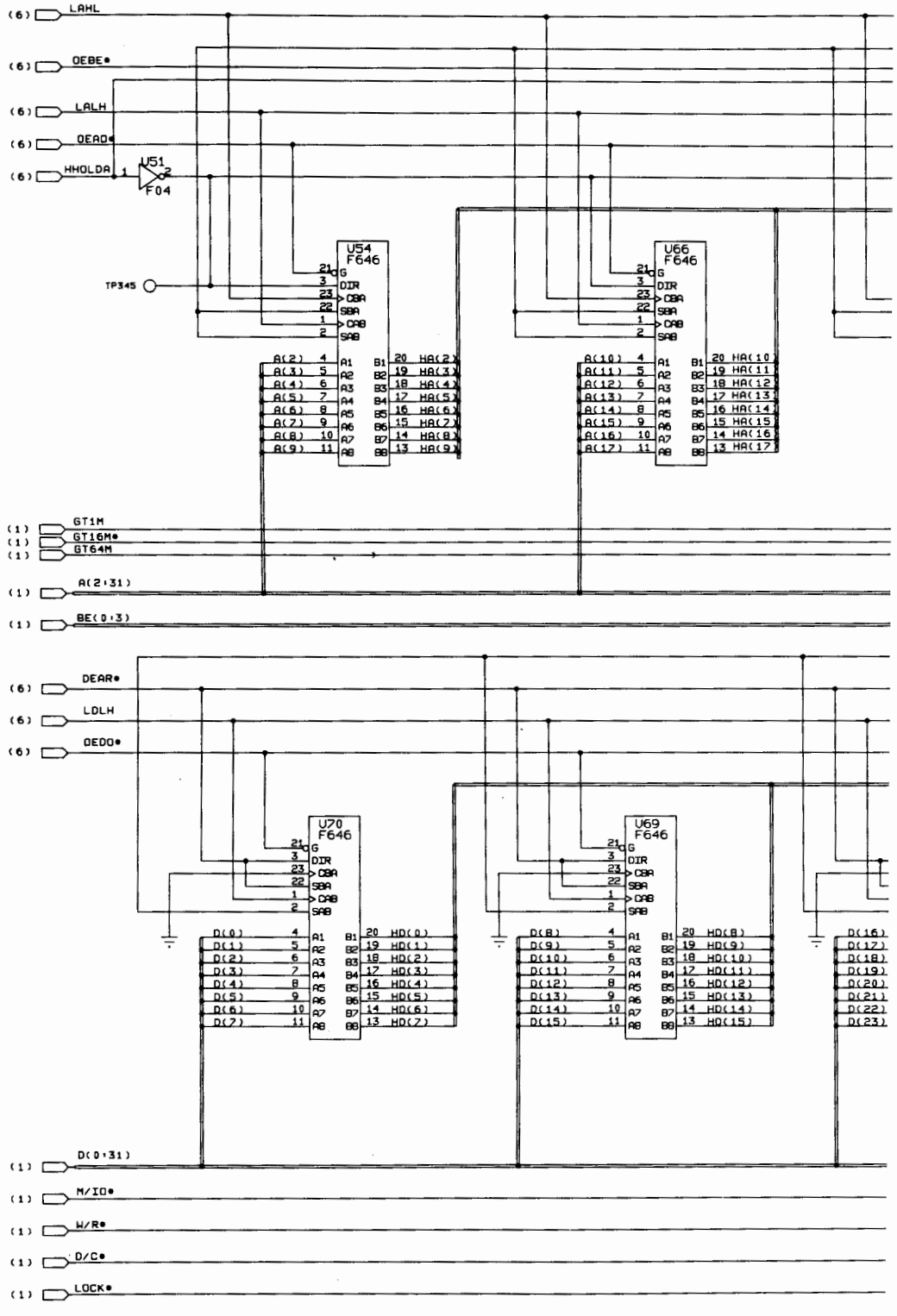
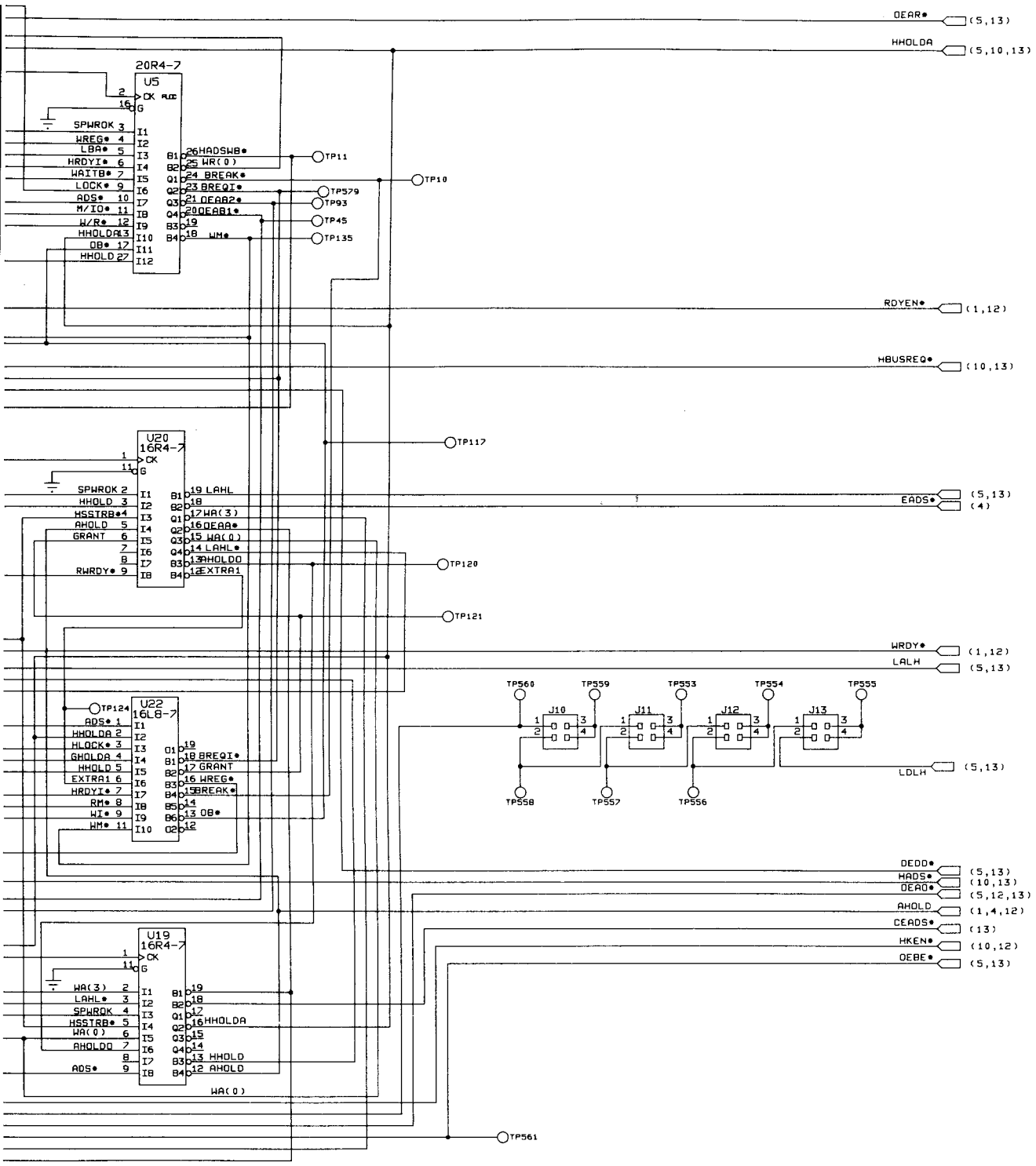


Figure B-6. CPU Board (Write Buffer Controller)
Schematics B-13/B-14


20R4-7
U5
SPWROK
WREG*
LBA*
HRDYI*
WAITB*
LOCK*
ADS*
M/IO*
W/R*
HHOLDA
OB*
HHOLD
HADSWB*
WR(0)
BREAK*
BREQI*
OEAB2*
OEAB1*
WM*
TP11
TP10
TP579
TP93
TP45
TP135
OEAR*
(5,13)
HHOLDA
(5,10,13)
RDYEN*
(1,12)
HBUSREQ*
(10,13)
U20
16R4-7
LAHL
WA(3)
OEAA*
WA(0)
LAHL*
AHOLDO
EXTRA1
TP117
TP120
TP121
EADS*
(5,13)
(4)
WRDY*
(1,12)
LALH
(5,13)
U22
16L8-7
TP124
ADS*
HHOLDA
HLOCK*
GHOLDA
HHOLD
EXTRA1
HRDYI*
RM*
WI*
WM*
BREQI*
GRANT
WREG*
BREAK*
OB*
TP560
TP559
TP553
TP554
TP555
J10
J11
J12
J13
TP558
TP557
TP556
LDLH
(5,13)
DEDD*
(5,13)
HADS*
(10,13)
DEAD*
(5,12,13)
AHOLD
(1,4,12)
CEADS*
(13)
HKEN*
(10,12)
OEBE*
(5,13)
U19
16R4-7
WA(3)
LAHL*
SPWROK
HSSTRB*
WA(0)
AHOLDO
ADS*
HHOLDA
HHOLD
AHOLD
WA(0)
TP561

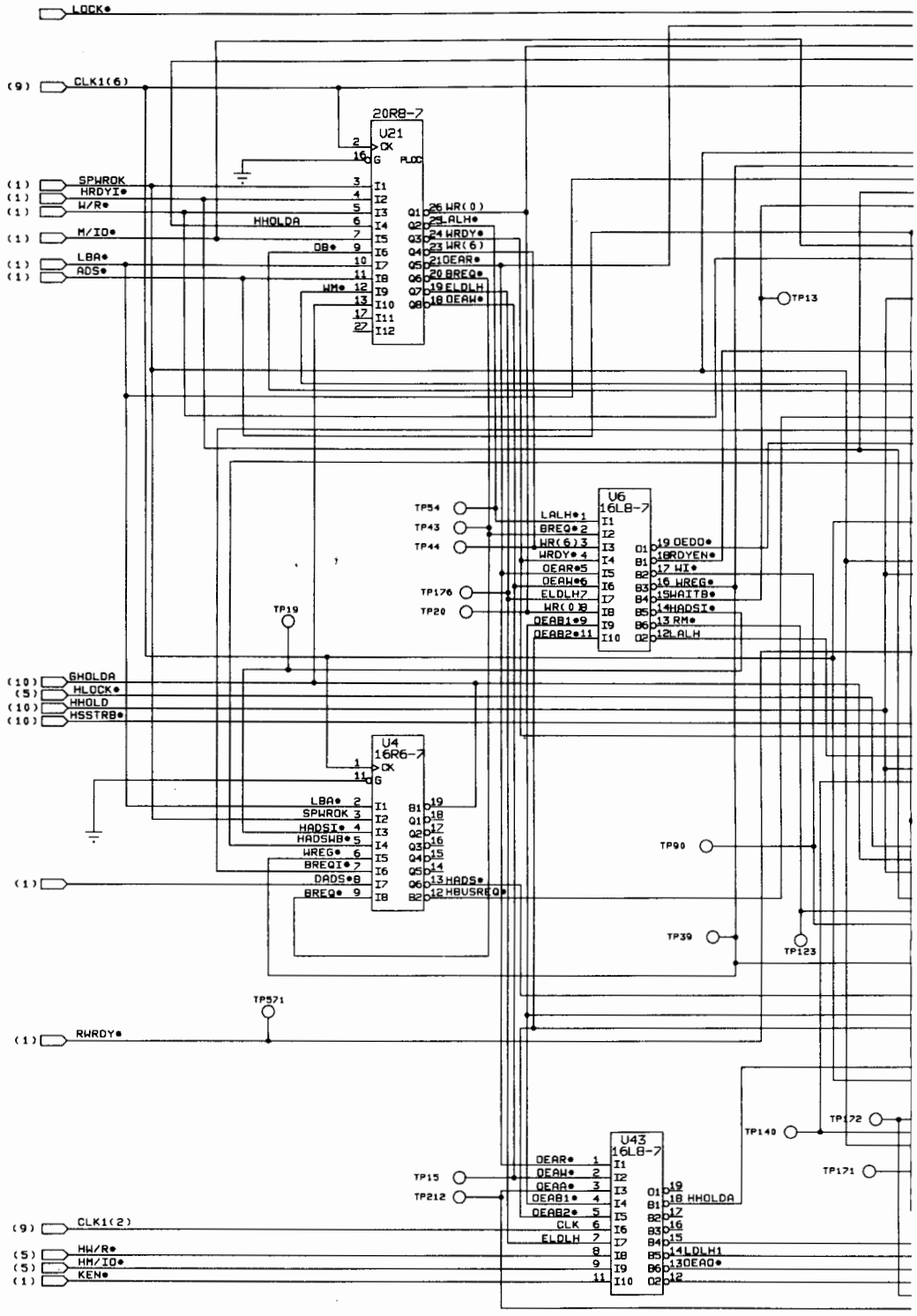
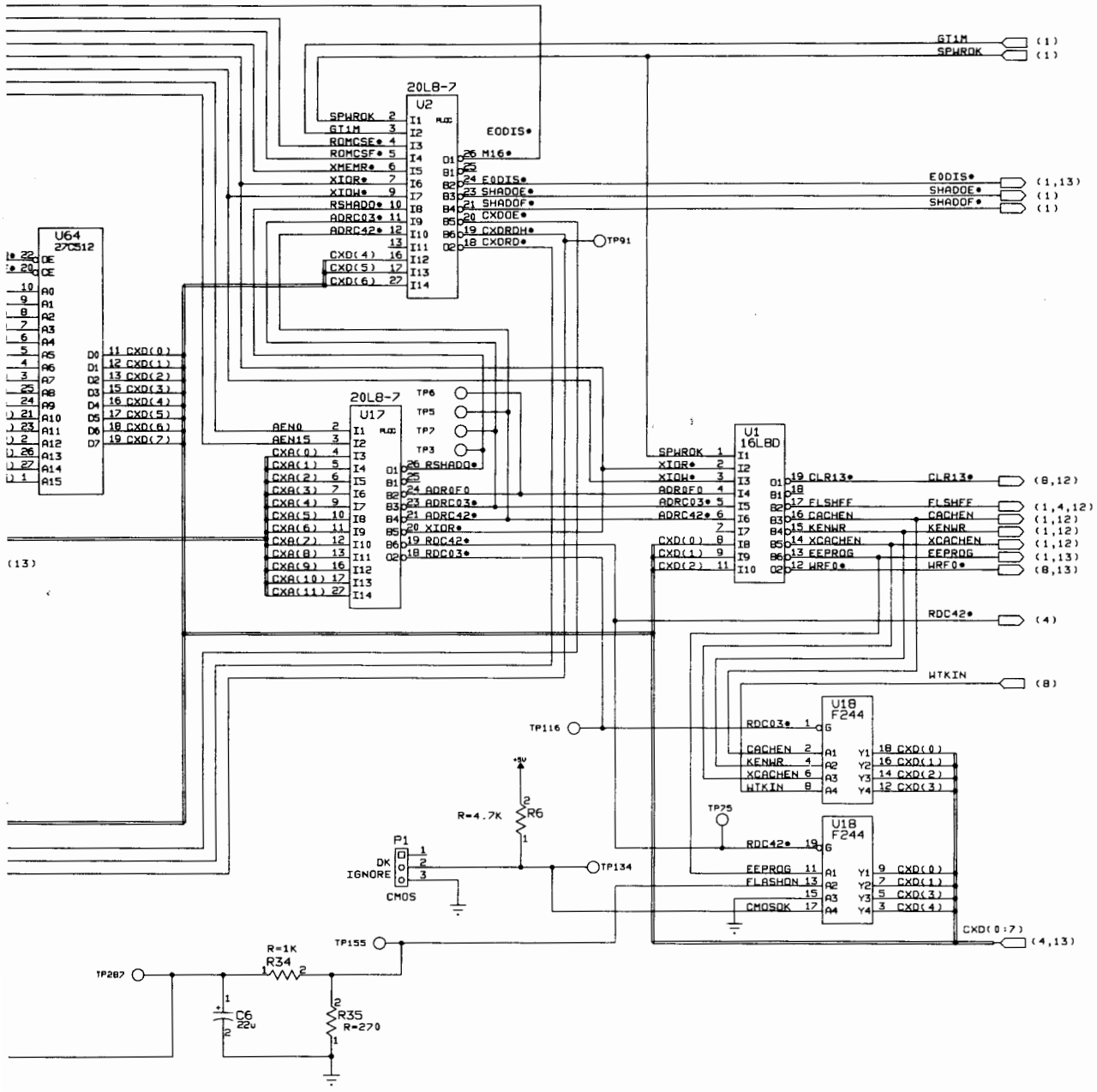

LOCK*
(9) CLK1(6)
20R8-7
U21
(1) SPWROK
(1) HRDYI*
(1) W/R*
HHOLDA
(1) M/IO*
DB* 9
(1) LBA*
(1) ADS*
WM* 12
WR(0)
ALH*
WRDY*
WR(6)
OEAR*
BREQ*
ELDLH
OEAW*
TP13
U6
16L8-7
TP54
TP43
TP44
TP176
TP20
TP19
LALH*1
BREQ*2
WR(6)3
WRDY*4
OEAR*5
OEAW*6
ELDLH7
WR(0)8
OEAB1*9
OEAB2*11
OEDO*
RDYEN*
WI*
WREG*
WAITB*
HADSI*
RM*
LALH
(10) GHOLDA
(5) HLOCK*
(10) HHOLD
(10) HSSTRB*
U4
16R6-7
LBA* 2
SPWROK 3
HADSI* 4
HADSWB* 5
WREG* 6
BREQI* 7
DADS*8
BREQ* 9
HADS*
HBUSREQ
(1)
TP90
TP39
TP123
TP571
(1) RWRDY*
U43
16L8-7
TP140
TP172
TP171
TP15
TP212
OEAR* 1
OEAW* 2
OEAA* 3
OEAB1* 4
OEAB2* 5
CLK 6
ELDLH 7
HHOLDA
LDLH1
OEAO*
(9) CLK1(2)
(5) HW/R*
(5) HM/IO*
(1) KEN*

Figure B-7. CPU (BIOS ROM I/O Ports)
Schematics B-15/16


20L8-7
U2
SPWROK 2
GT1M 3
ROMCSE• 4
ROMCSF• 5
XMEMR• 6
XIOR• 7
XIOW• 9
RSHADO• 10
ADRC03• 11
ADRC42• 12
CXD(4) 16
CXD(5) 17
CXD(6) 27
EODIS•
M16•
EODIS•
SHADOE•
SHADOF•
CXDOE•
CXDRDH•
CXDRD•
TP91
GT1M (1)
SPWROK (1)
EODIS• (1,13)
SHADOE• (1)
SHADOF• (1)
U64
27C512
CXD(0)
CXD(1)
CXD(2)
CXD(3)
CXD(4)
CXD(5)
CXD(6)
CXD(7)
(13)
20L8-7
U17
AEN0
AEN15
CXA(0)
CXA(1)
CXA(2)
CXA(3)
CXA(4)
CXA(5)
CXA(6)
CXA(7)
CXA(8)
CXA(9)
CXA(10)
CXA(11)
TP6
TP5
TP7
TP3
RSHADO•
ADR0F0
ADRC03•
ADRC42•
XIOR•
RDC42•
RDC03•
U1
16L8D
SPWROK
XIOR•
XIOW•
ADR0F0
ADRC03•
ADRC42•
CXD(0)
CXD(1)
CXD(2)
CLR13•
FLSHFF
CACHEN
KENWR
XCACHEN
EEPROG
WRF0•
CLR13• (8,12)
FLSHFF (1,4,12)
CACHEN (1,12)
KENWR (1,12)
XCACHEN (1,12)
EEPROG (1,13)
WRF0• (8,13)
RDC42• (4)
WTKIN (8)
U18
F244
RDC03•
CACHEN
KENWR
XCACHEN
WTKIN
CXD(0)
CXD(1)
CXD(2)
CXD(3)
TP116
TP25
RDC42•
EEPROG
FLASHON
CMOSOK
CXD(4)
CXD(7)
CXD(0:7)
(4,13)
+5V
R=4.7K
R6
P1
DK
IGNORE
CMOS
TP134
TP155
TP287
R=1K
R34
C6
22u
R35
R=270

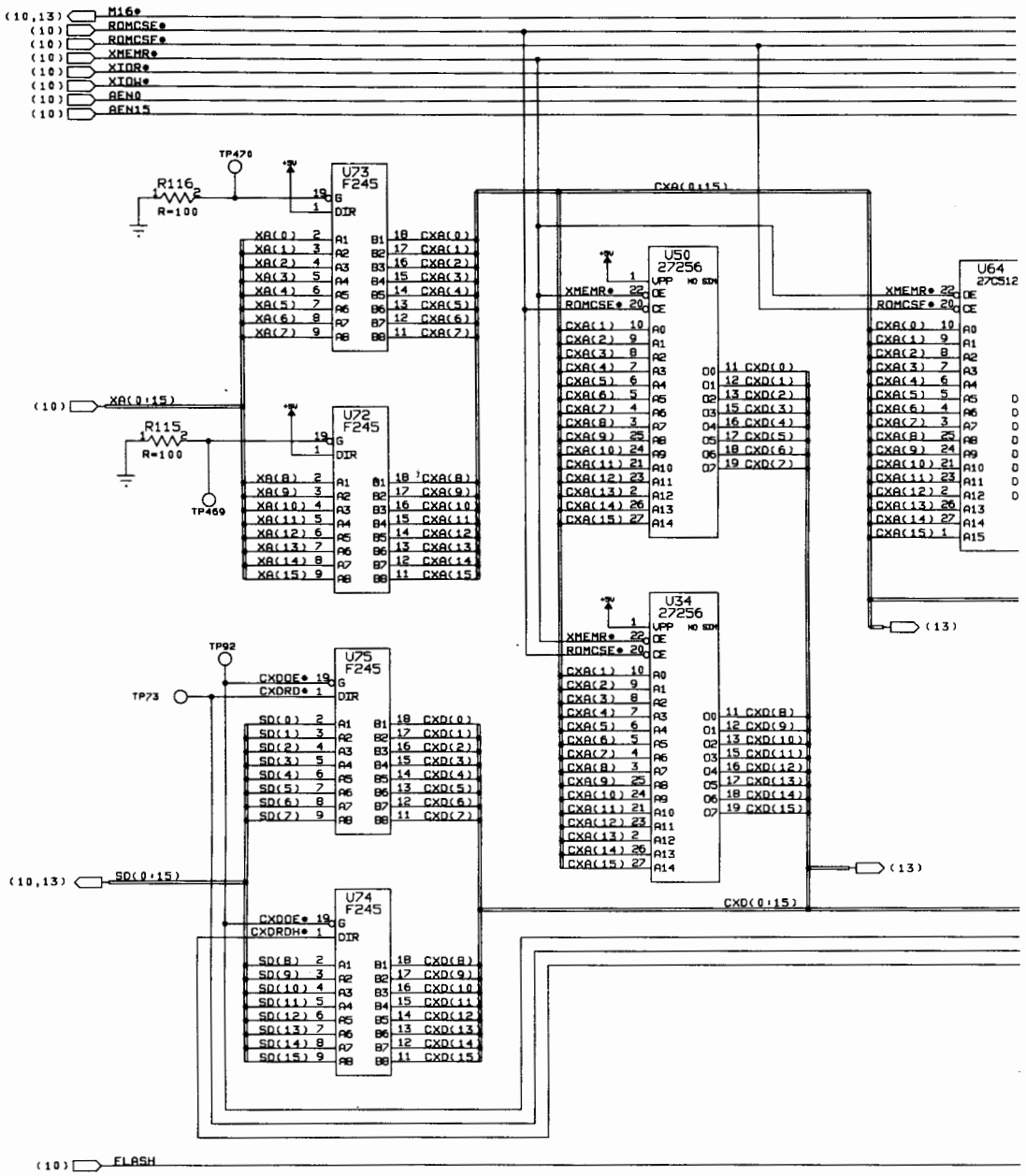
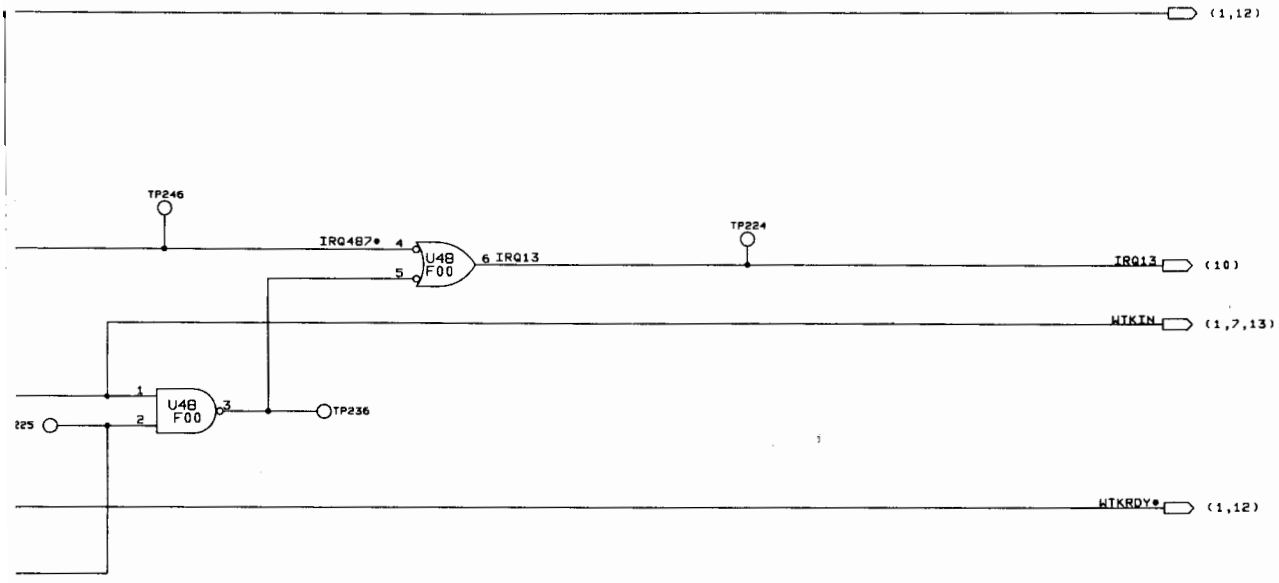

(10,13) M16*
(10) ROMCSE*
(10) ROMCSF*
(10) XMEMR*
(10) XIOR*
(10) XIOW*
(10) AEN0
(10) AEN15
TP470
R116
R=100
U73
F245
CXA(0:15)
XA(0:15)
R115
R=100
TP469
U72
F245
U50
27256
U64
27C512
(13)
U34
27256
TP92
TP73
U75
F245
SD(0:15)
U74
F245
CXD(0:15)
(10) FLASH

Figure B-8. CPU Board (Weitek Coprocessor)
Schematics B-17/B-18


(1,12)
TP246
IRQ487*
4
5
U48
F00
6 IRQ13
TP224
IRQ13
(10)
WIKIN
(1,7,13)
1
2
U48
F00
3
TP236
225
WIKRDY*
(1,12)

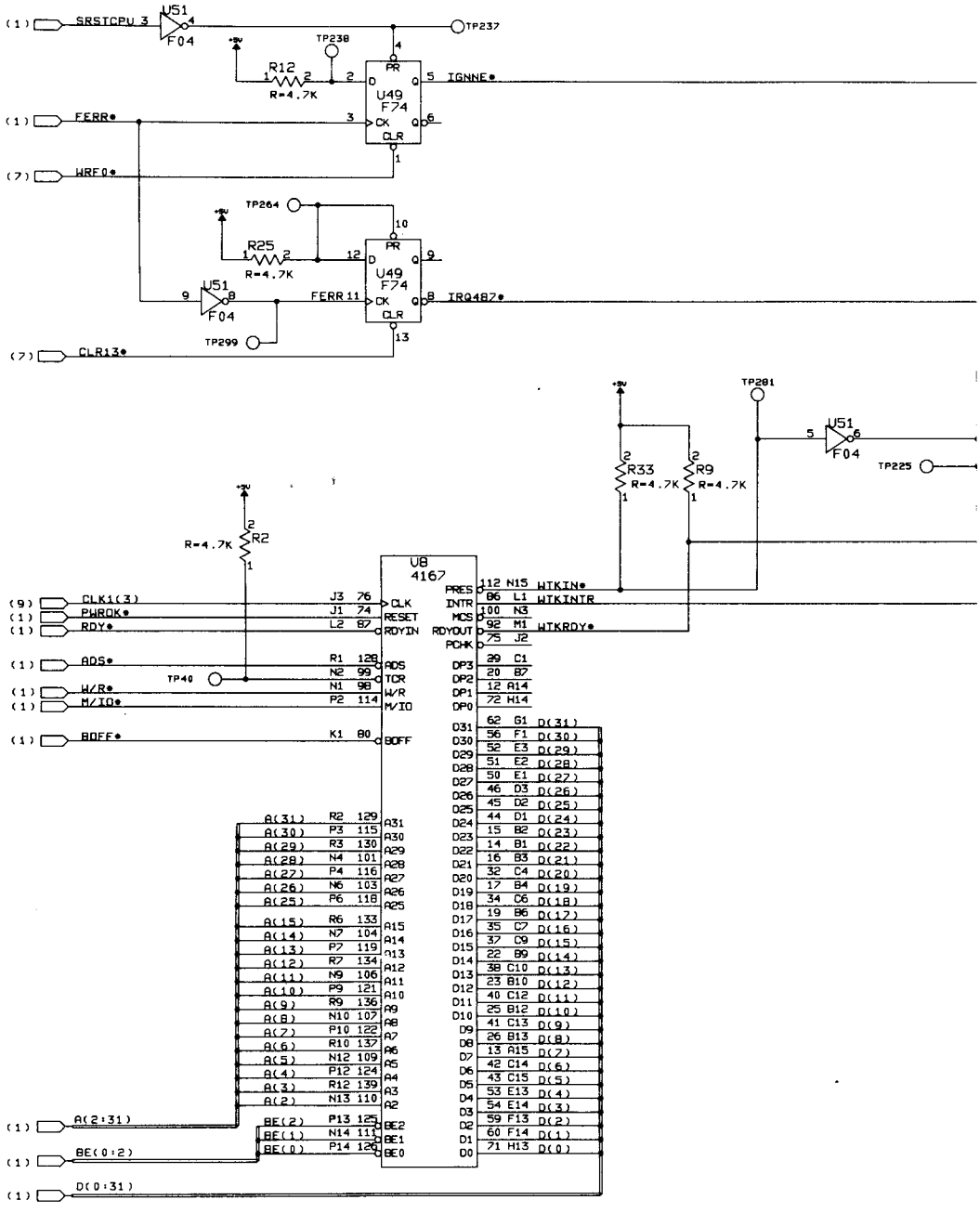
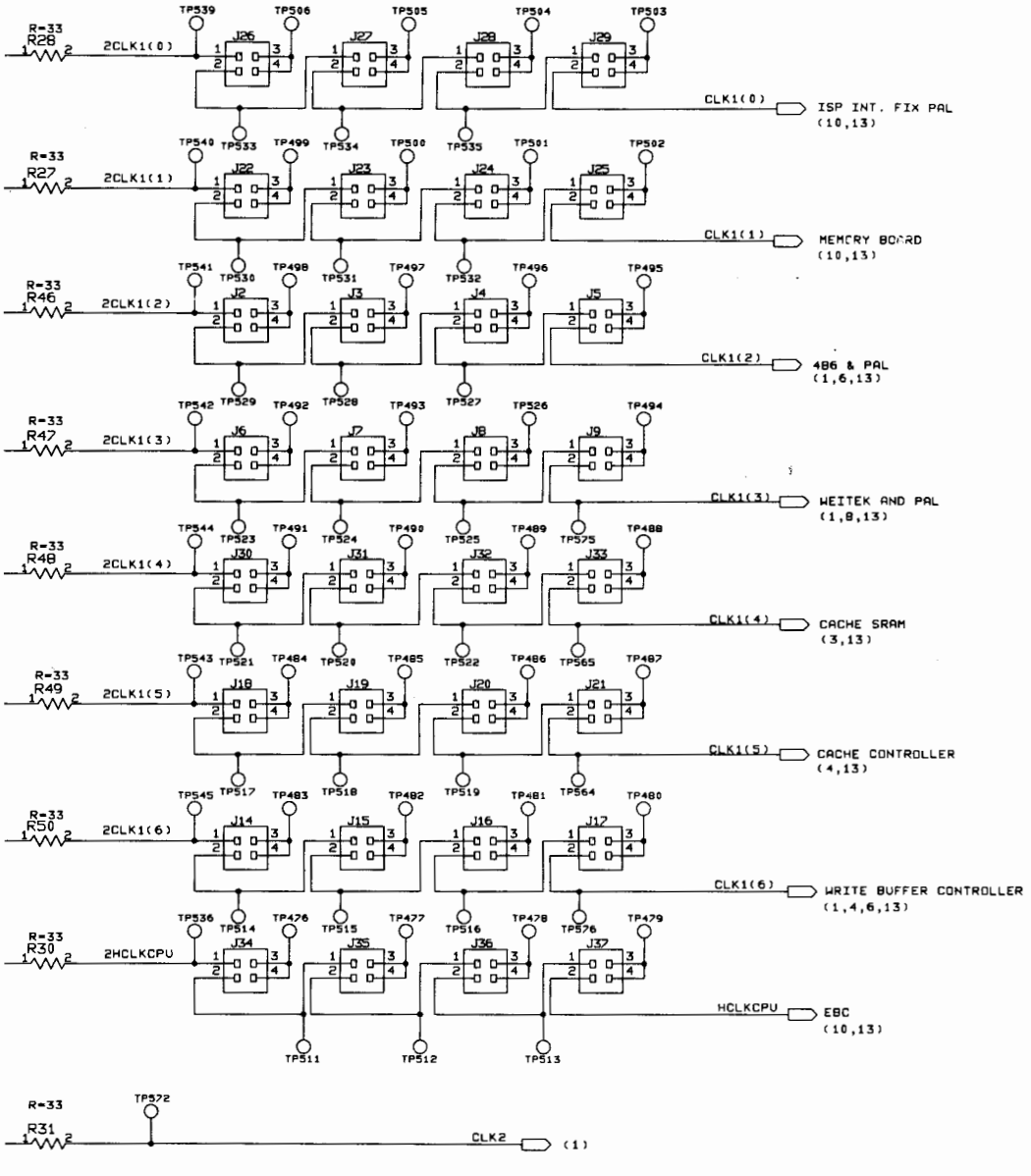


Figure B-9. CPU Board (Clock Generator)
Schematics B-19/B-20


R=33
R28
2CLK1(0)
TP539
TP506
TP505
TP504
TP503
J26
J27
J28
J29
CLK1(0)
ISP INT. FIX PAL
(10,13)
R=33
R27
2CLK1(1)
TP540
TP533
TP499
TP534
TP500
TP535
TP501
TP502
J22
J23
J24
J25
CLK1(1)
MEMORY BOARD
(10,13)
R=33
R46
2CLK1(2)
TP541
TP530
TP498
TP531
TP497
TP532
TP496
TP495
J2
J3
J4
J5
CLK1(2)
486 & PAL
(1,6,13)
R=33
R47
2CLK1(3)
TP542
TP529
TP492
TP528
TP493
TP527
TP526
TP494
J6
J7
J8
J9
CLK1(3)
WEITEK AND PAL
(1,8,13)
R=33
R48
2CLK1(4)
TP544
TP523
TP491
TP524
TP490
TP525
TP489
TP575
TP488
J30
J31
J32
J33
CLK1(4)
CACHE SRAM
(3,13)
R=33
R49
2CLK1(5)
TP543
TP521
TP484
TP520
TP485
TP522
TP486
TP565
TP487
J18
J19
J20
J21
CLK1(5)
CACHE CONTROLLER
(4,13)
R=33
R50
2CLK1(6)
TP545
TP517
TP483
TP518
TP482
TP519
TP481
TP564
TP480
J14
J15
J16
J12
CLK1(6)
WRITE BUFFER CONTROLLER
(1,4,6,13)
R=33
R30
2HCLKCPU
TP536
TP514
TP476
TP515
TP477
TP516
TP478
TP576
TP479
J34
J35
J36
J37
HCLKCPU
EBC
(10,13)
TP511
TP512
TP513
R=33
R31
TP572
CLK2
(1)

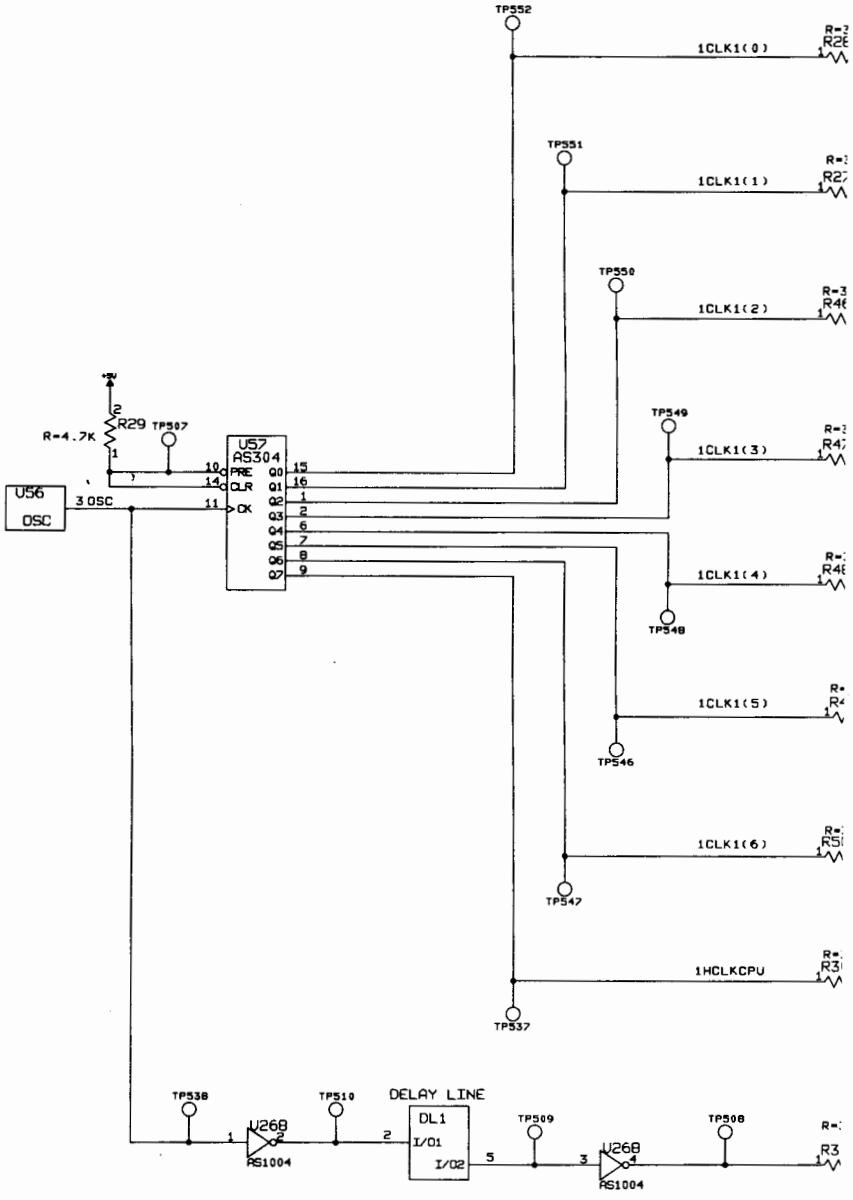
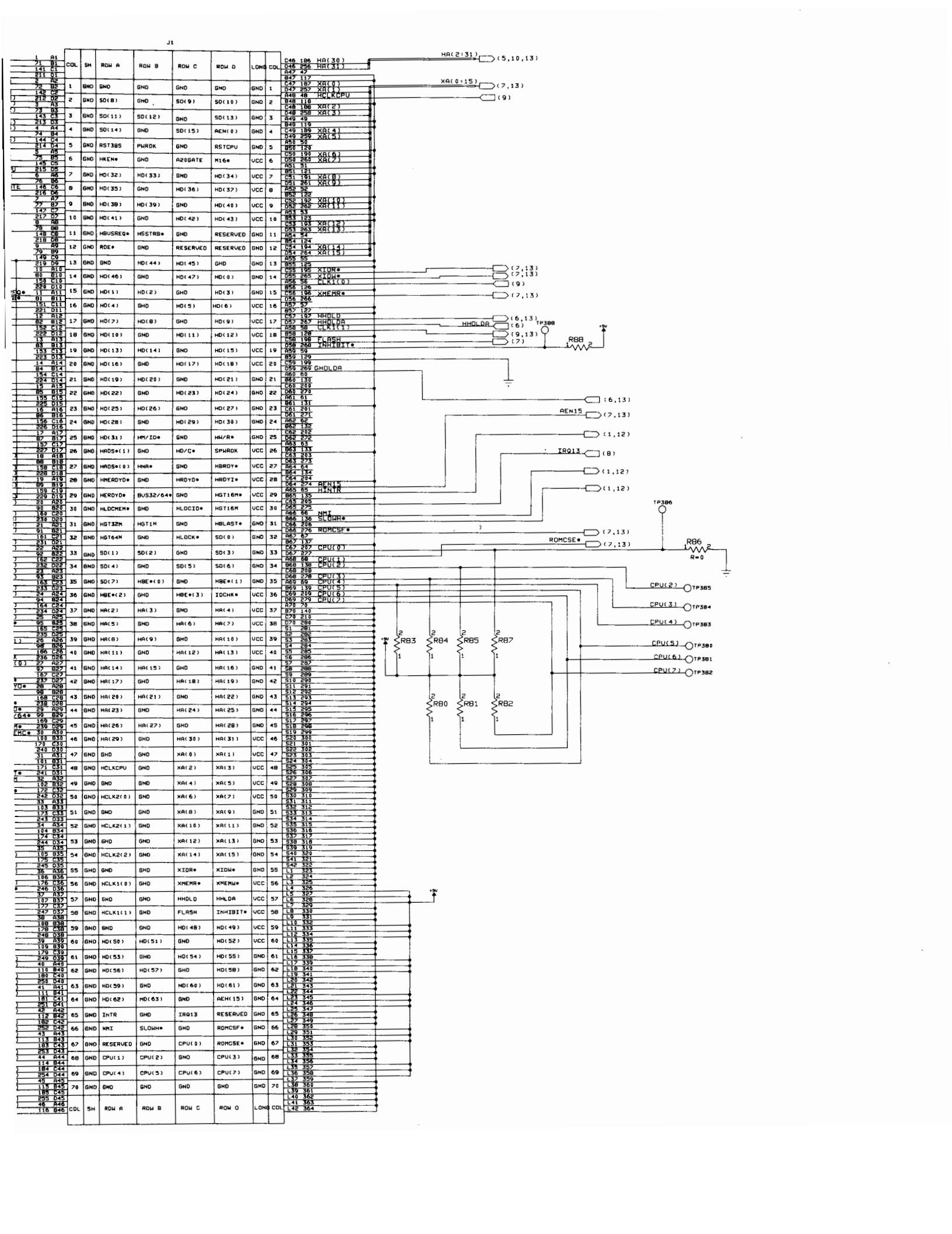

TP552
1CLK1(0)
TP551
1CLK1(1)
TP550
1CLK1(2)
TP549
1CLK1(3)
1CLK1(4)
TP548
1CLK1(5)
TP546
1CLK1(6)
TP547
1HCLKCPU
TP537
R=4.7K
R29
TP507
U57
AS304
PRE
CLR
CK
Q0
Q1
Q2
Q3
Q4
Q5
Q6
Q7
U56
OSC
3 OSC
TP538
TP510
DELAY LINE
DL1
I/01
I/02
TP509
TP508
U268
AS1004

Figure B-10. CPU Board (J1 Connector)
Schematics B-21/B-22



J1

COL	SH	ROW A	ROW B	ROW C	ROW D	LONG	COL
1	GND	GND	GND	GND	GND	GND	1
2	GND	SD(8)	GND	SD(9)	SD(10)	GND	2
3	GND	SD(11)	SD(12)	GND	SD(13)	GND	3
4	GND	SD(14)	GND	SD(15)	AEN(0)	GND	4
5	GND	RST385	PWROK	GND	RSTCPU	GND	5
6	GND	HKEN*	GND	A20GATE	M16*	VCC	6
7	GND	HD(32)	HD(33)	GND	HD(34)	VCC	7
8	GND	HD(35)	GND	HD(36)	HD(37)	VCC	8
9	GND	HD(38)	HD(39)	GND	HD(40)	VCC	9
10	GND	HD(41)	GND	HD(42)	HD(43)	VCC	10
11	GND	HBUSREQ*	HSSTRB*	GND	RESERVED	GND	11
12	GND	RDE*	GND	RESERVED	RESERVED	GND	12
13	GND	GND	HD(44)	HD(45)	GND	GND	13
14	GND	HD(46)	GND	HD(47)	HD(0)	GND	14
15	GND	HD(1)	HD(2)	GND	HD(3)	GND	15
16	GND	HD(4)	GND	HD(5)	HD(6)	VCC	16
17	GND	HD(7)	HD(8)	GND	HD(9)	VCC	17
18	GND	HD(10)	GND	HD(11)	HD(12)	VCC	18
19	GND	HD(13)	HD(14)	GND	HD(15)	VCC	19
20	GND	HD(16)	GND	HD(17)	HD(18)	VCC	20
21	GND	HD(19)	HD(20)	GND	HD(21)	GND	21
22	GND	HD(22)	GND	HD(23)	HD(24)	GND	22
23	GND	HD(25)	HD(26)	GND	HD(27)	GND	23
24	GND	HD(28)	GND	HD(29)	HD(30)	GND	24
25	GND	HD(31)	HM/IO*	GND	HW/R*	GND	25
26	GND	HADS*(1)	GND	HD/C*	SPWROK	VCC	26
27	GND	HADS*(0)	HNA*	GND	HBRDY*	VCC	27
28	GND	HMERDYD*	GND	HRDYD*	HRDYI*	VCC	28
29	GND	HERDYD*	BUS32/64*	GND	HGT16M*	VCC	29
30	GND	HLOCMEM*	GND	HLOCID*	HGT16M	VCC	30
31	GND	HGT32M	HGT1M	GND	HBLAST*	GND	31
32	GND	HGT64M	GND	HLOCK*	SD(0)	GND	32
33	GND	SD(1)	SD(2)	GND	SD(3)	GND	33
34	GND	SD(4)	GND	SD(5)	SD(6)	GND	34
35	GND	SD(7)	HBE*(0)	GND	HBE*(1)	GND	35
36	GND	HBE*(2)	GND	HBE*(3)	IOCHK*	VCC	36
37	GND	HA(2)	HA(3)	GND	HA(4)	VCC	37
38	GND	HA(5)	GND	HA(6)	HA(7)	VCC	38
39	GND	HA(8)	HA(9)	GND	HA(10)	VCC	39
40	GND	HA(11)	GND	HA(12)	HA(13)	VCC	40
41	GND	HA(14)	HA(15)	GND	HA(16)	GND	41
42	GND	HA(17)	GND	HA(18)	HA(19)	GND	42
43	GND	HA(20)	HA(21)	GND	HA(22)	GND	43
44	GND	HA(23)	GND	HA(24)	HA(25)	GND	44
45	GND	HA(26)	HA(27)	GND	HA(28)	GND	45
46	GND	HA(29)	GND	HA(30)	HA(31)	VCC	46
47	GND	GND	GND	XA(0)	XA(1)	VCC	47
48	GND	HCLKCPU	GND	XA(2)	XA(3)	VCC	48
49	GND	GND	GND	XA(4)	XA(5)	VCC	49
50	GND	HCLK2(0)	GND	XA(6)	XA(7)	VCC	50
51	GND	GND	GND	XA(8)	XA(9)	GND	51
52	GND	HCLK2(1)	GND	XA(10)	XA(11)	GND	52
53	GND	GND	GND	XA(12)	XA(13)	GND	53
54	GND	HCLK2(2)	GND	XA(14)	XA(15)	GND	54
55	GND	GND	GND	XIOR*	XIOW*	GND	55
56	GND	HCLK1(0)	GND	XMEMR*	XMEMW*	VCC	56
57	GND	GND	GND	HHDLD	HHLDA	VCC	57
58	GND	HCLK1(1)	GND	FLASH	INHIBIT*	VCC	58
59	GND	GND	GND	HD(48)	HD(49)	VCC	59
60	GND	HD(50)	HD(51)	GND	HD(52)	VCC	60
61	GND	HD(53)	GND	HD(54)	HD(55)	GND	61
62	GND	HD(56)	HD(57)	GND	HD(58)	GND	62
63	GND	HD(59)	GND	HD(60)	HD(61)	GND	63
64	GND	HD(62)	HD(63)	GND	AEN(15)	GND	64
65	GND	INTR	GND	IRQ13	RESERVED	GND	65
66	GND	NMI	SLOWH*	GND	ROMCSF*	GND	66
67	GND	RESERVED	GND	CPU(0)	ROMCSE*	GND	67
68	GND	CPU(1)	CPU(2)	GND	CPU(3)	GND	68
69	GND	CPU(4)	CPU(5)	CPU(6)	CPU(7)	GND	69
70	GND	GND	GND	GND	GND	GND	70
COL	SH	ROW A	ROW B	ROW C	ROW D	LONG	COL

HA(2:31) (5,10,13)

XA(0:15) (7,13)

(9)

(7,13)

(7,13)

(9)

(7,13)

(6,13)

HHOLDA (6)

(9,13)

(7)

TP388

R88

+5V

(6,13)

AEN15 (7,13)

(1,12)

IRQ13 (8)

(1,12)

(1,12)

TP386

R86 R=0

ROMCSE* (7,13)

CPU(2) TP385

CPU(3) TP384

CPU(4) TP383

CPU(5) TP380

CPU(6) TP381

CPU(7) TP382

RB3 RB4 RB5 RB7

RB0 RB1 RB2

+5V

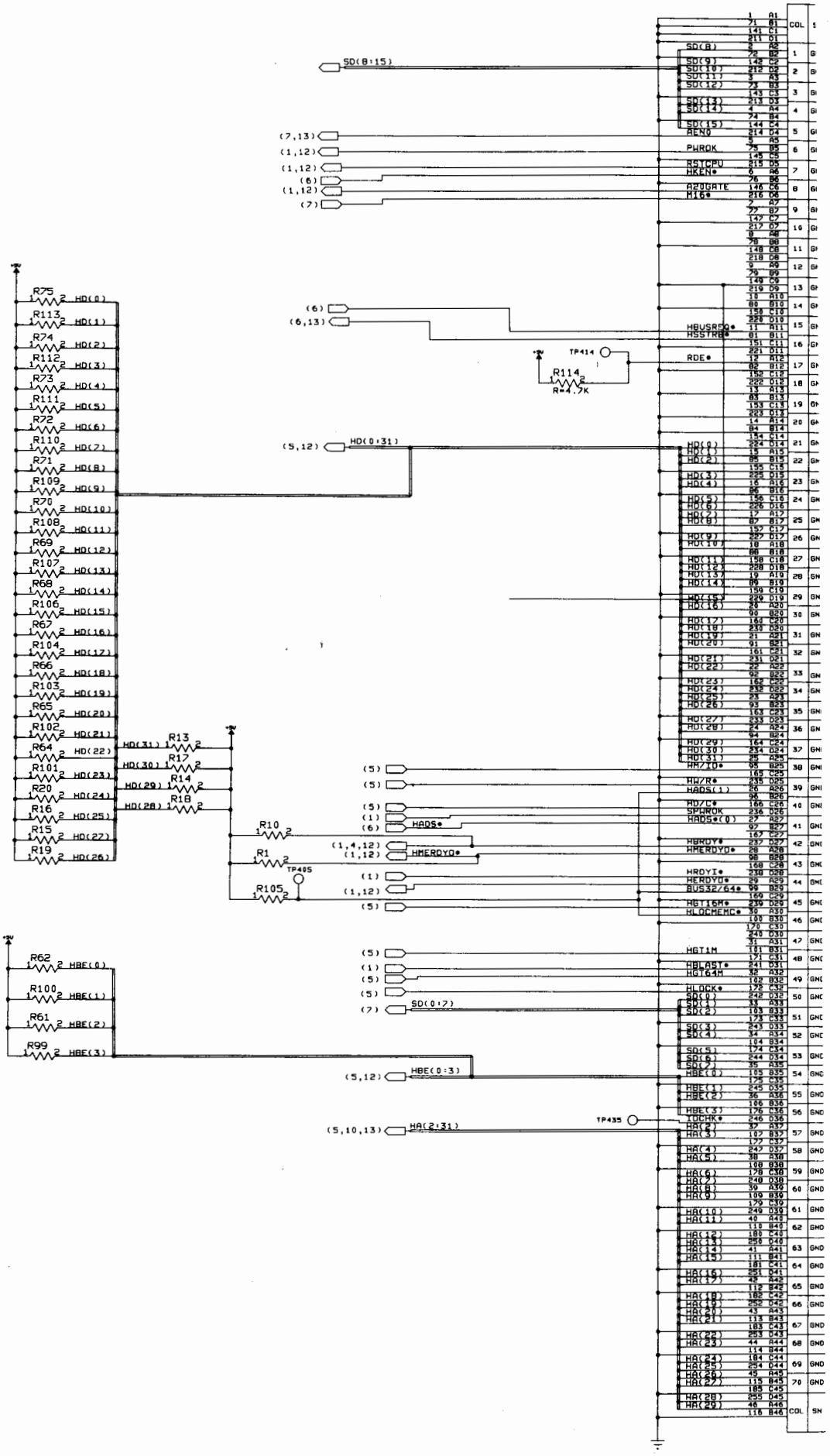
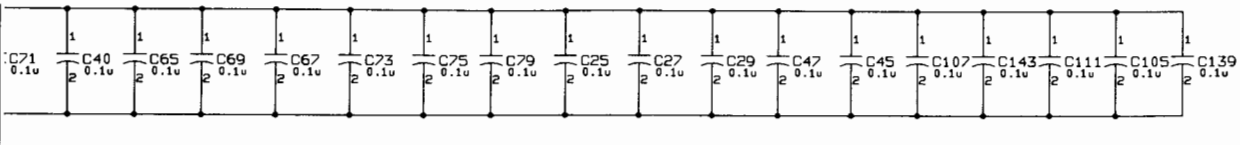
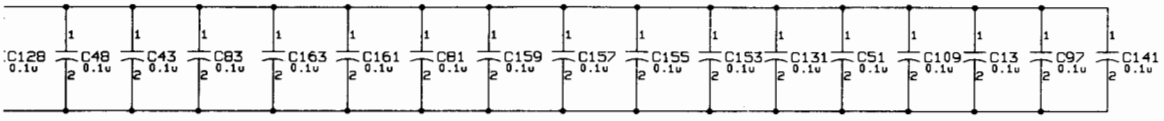
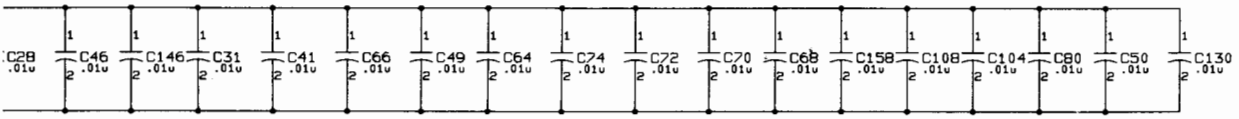
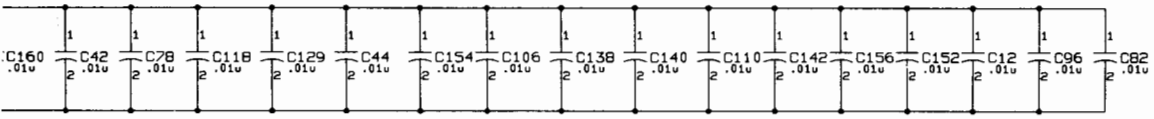

SD(8:15)
(7,13)
(1,12)
(1,12)
(6)
(1,12)
(7)
PWROK
RSTCPU
HKEN*
A20GATE
M16*
RENO
(6)
(6,13)
HBUSREQ*
HSSTRB*
TP414
R114
R=4.7K
RDE*
(5,12)
HD(0:31)
R75
R113
R74
R112
R73
R111
R72
R110
R71
R109
R70
R108
R69
R107
R68
R106
R67
R104
R66
R103
R65
R102
R64
R101
R20
R16
R15
R19
R13
R17
R14
R18
R10
R1
R105
TP405
HADS*
HMERDYO*
HBRDY*
HMERDYO*
HRDYI*
HERDYO*
BUS32/64*
HGT16M*
HLOCMEMC*
HD/R*
HADS(1)
HD/C*
SPWROK
HADS*(0)
HM/IO*
HGT1M
HBLAST*
HGT64M
HLOCK*
SD(0:7)
R62
R100
R61
R99
HBE(0:3)
TP435
IOCHK*
(5,10,13)
HA(2:31)

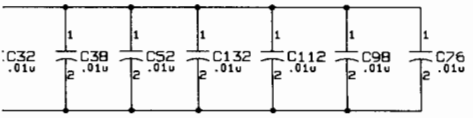
Figure B-11. CPU Board (Bypass Capacitors)
Schematics B-23/B-24

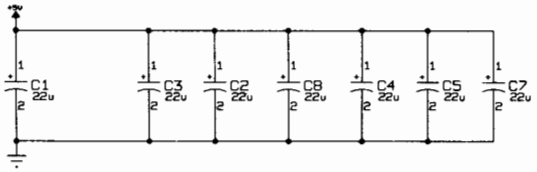

C71 0.1u
C40 0.1u
C65 0.1u
C69 0.1u
C67 0.1u
C73 0.1u
C75 0.1u
C79 0.1u
C25 0.1u
C27 0.1u
C29 0.1u
C47 0.1u
C45 0.1u
C107 0.1u
C143 0.1u
C111 0.1u
C105 0.1u
C139 0.1u

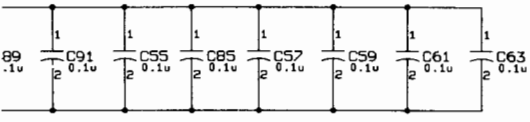

C128 0.1u
C48 0.1u
C43 0.1u
C83 0.1u
C163 0.1u
C161 0.1u
C81 0.1u
C159 0.1u
C157 0.1u
C155 0.1u
C153 0.1u
C131 0.1u
C51 0.1u
C109 0.1u
C13 0.1u
C97 0.1u
C141 0.1u


C28 .01u
C46 .01u
C146 .01u
C31 .01u
C41 .01u
C66 .01u
C49 .01u
C64 .01u
C24 .01u
C72 .01u
C70 .01u
C68 .01u
C158 .01u
C108 .01u
C104 .01u
C80 .01u
C50 .01u
C130 .01u


C160 .01u
C42 .01u
C78 .01u
C118 .01u
C129 .01u
C44 .01u
C154 .01u
C106 .01u
C138 .01u
C140 .01u
C110 .01u
C142 .01u
C156 .01u
C152 .01u
C12 .01u
C96 .01u
C82 .01u


C32 .01u
C38 .01u
C52 .01u
C132 .01u
C112 .01u
C98 .01u
C76 .01u


+5V
C1 22u
C3 22u
C2 22u
C8 22u
C4 22u
C5 22u
C7 22u


89 .1u
C91 0.1u
C55 0.1u
C85 0.1u
C57 0.1u
C59 0.1u
C61 0.1u
C63 0.1u

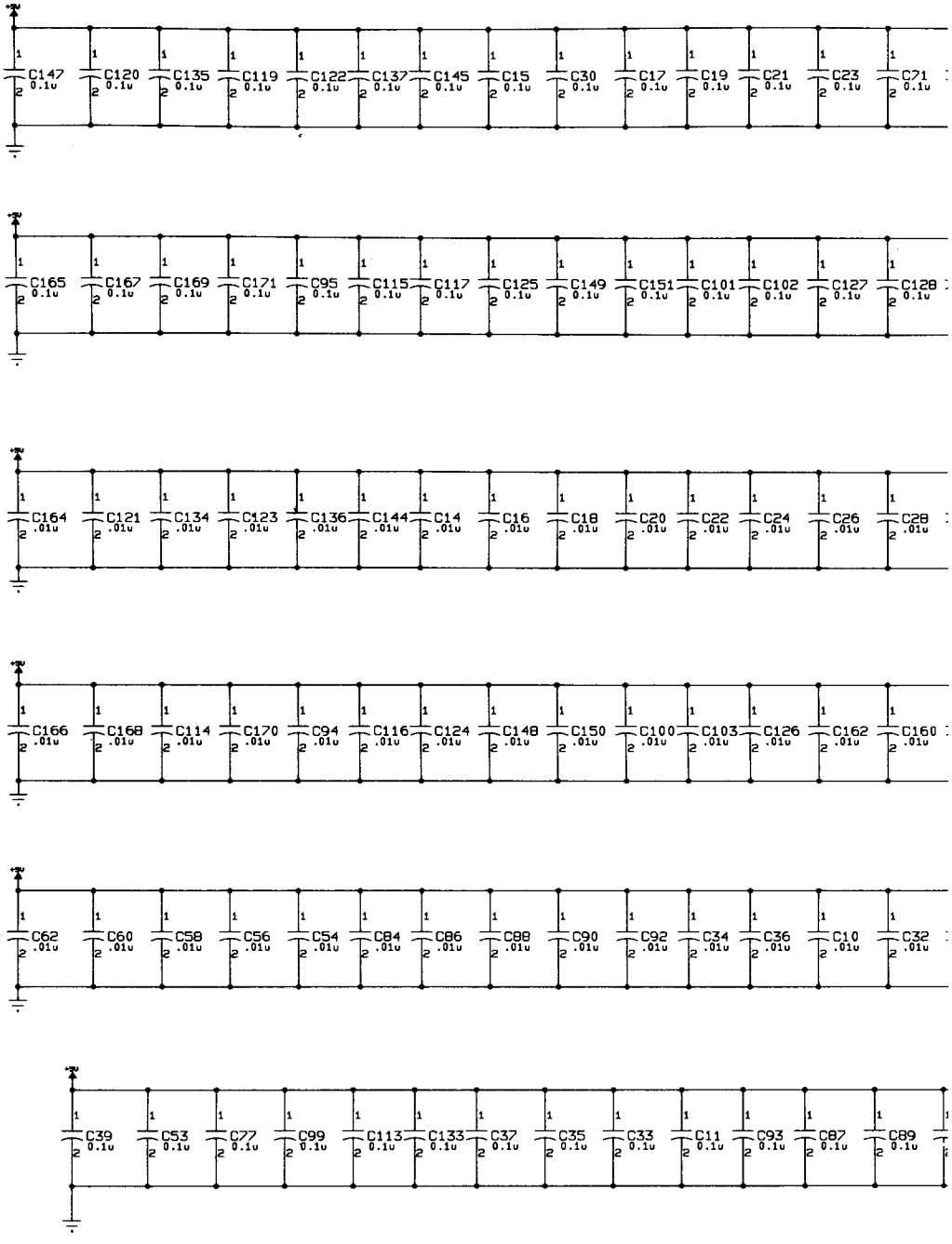
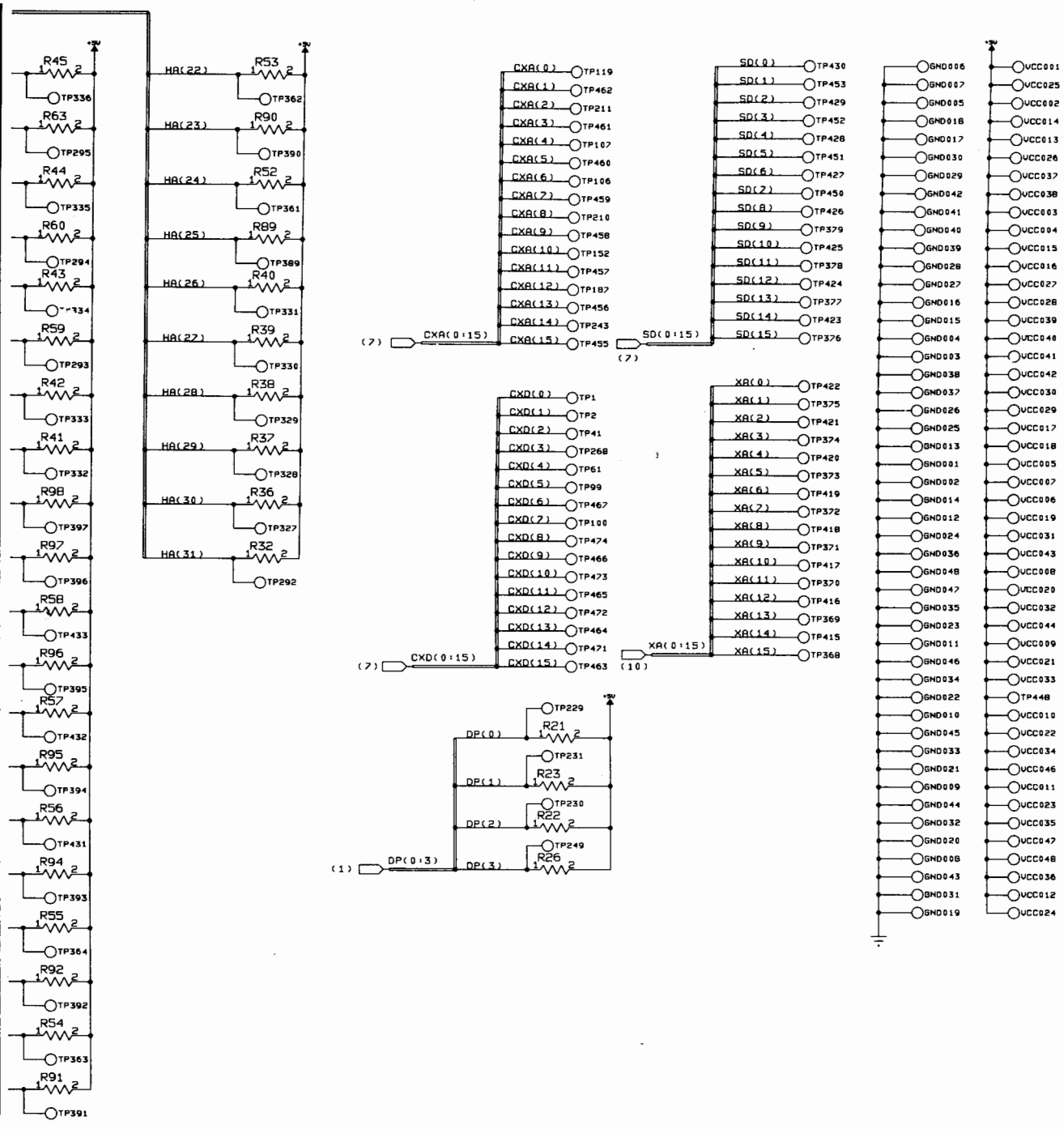

C147 0.1u
C120 0.1u
C135 0.1u
C119 0.1u
C122 0.1u
C137 0.1u
C145 0.1u
C15 0.1u
C30 0.1u
C17 0.1u
C19 0.1u
C21 0.1u
C23 0.1u
C71 0.1u
C165 0.1u
C167 0.1u
C169 0.1u
C171 0.1u
C95 0.1u
C115 0.1u
C117 0.1u
C125 0.1u
C149 0.1u
C151 0.1u
C101 0.1u
C102 0.1u
C127 0.1u
C128 0.1u
C164 .01u
C121 .01u
C134 .01u
C123 .01u
C136 .01u
C144 .01u
C14 .01u
C16 .01u
C18 .01u
C20 .01u
C22 .01u
C24 .01u
C26 .01u
C28 .01u
C166 .01u
C168 .01u
C114 .01u
C170 .01u
C94 .01u
C116 .01u
C124 .01u
C148 .01u
C150 .01u
C100 .01u
C103 .01u
C126 .01u
C162 .01u
C160 .01u
C62 .01u
C60 .01u
C58 .01u
C56 .01u
C54 .01u
C84 .01u
C86 .01u
C88 .01u
C90 .01u
C92 .01u
C34 .01u
C36 .01u
C10 .01u
C32 .01u
C39 0.1u
C53 0.1u
C77 0.1u
C99 0.1u
C113 0.1u
C133 0.1u
C37 0.1u
C35 0.1u
C33 0.1u
C11 0.1u
C93 0.1u
C87 0.1u
C89 0.1u

Figure B-12. CPU Board (Test Points)
Schematics B-25/B-26


R45
TP336
R63
TP295
R44
TP335
R60
TP294
R43
TP334
R59
TP293
R42
TP333
R41
TP332
R98
TP397
R97
TP396
R58
TP433
R96
TP395
R57
TP432
R95
TP394
R56
TP431
R94
TP393
R55
TP364
R92
TP392
R54
TP363
R91
TP391
HA(22)
R53
TP362
HA(23)
R90
TP390
HA(24)
R52
TP361
HA(25)
R89
TP389
HA(26)
R40
TP331
HA(27)
R39
TP330
HA(28)
R38
TP329
HA(29)
R37
TP328
HA(30)
R36
TP327
HA(31)
R32
TP292
CXA(0)
CXA(1)
CXA(2)
CXA(3)
CXA(4)
CXA(5)
CXA(6)
CXA(7)
CXA(8)
CXA(9)
CXA(10)
CXA(11)
CXA(12)
CXA(13)
CXA(14)
CXA(15)
TP119
TP462
TP211
TP461
TP107
TP460
TP106
TP459
TP210
TP458
TP152
TP457
TP187
TP456
TP243
TP455
(7)
CXA(0:15)
SD(0)
SD(1)
SD(2)
SD(3)
SD(4)
SD(5)
SD(6)
SD(7)
SD(8)
SD(9)
SD(10)
SD(11)
SD(12)
SD(13)
SD(14)
SD(15)
TP430
TP453
TP429
TP452
TP428
TP451
TP427
TP450
TP426
TP379
TP425
TP378
TP424
TP377
TP423
TP376
SD(0:15)
(7)
CXD(0)
CXD(1)
CXD(2)
CXD(3)
CXD(4)
CXD(5)
CXD(6)
CXD(7)
CXD(8)
CXD(9)
CXD(10)
CXD(11)
CXD(12)
CXD(13)
CXD(14)
CXD(15)
TP1
TP2
TP41
TP268
TP61
TP99
TP467
TP100
TP474
TP466
TP473
TP465
TP472
TP464
TP471
TP463
(7)
CXD(0:15)
XA(0)
XA(1)
XA(2)
XA(3)
XA(4)
XA(5)
XA(6)
XA(7)
XA(8)
XA(9)
XA(10)
XA(11)
XA(12)
XA(13)
XA(14)
XA(15)
TP422
TP375
TP421
TP374
TP420
TP373
TP419
TP372
TP418
TP371
TP417
TP370
TP416
TP369
TP415
TP368
XA(0:15)
(10)
TP229
R21
DP(0)
TP231
R23
DP(1)
TP230
R22
DP(2)
TP249
R26
DP(3)
DP(0:3)
(1)
GND006
GND007
GND005
GND018
GND017
GND030
GND029
GND042
GND041
GND040
GND039
GND028
GND027
GND016
GND015
GND004
GND003
GND038
GND037
GND026
GND025
GND013
GND001
GND002
GND014
GND012
GND024
GND036
GND048
GND047
GND035
GND023
GND011
GND046
GND034
GND022
GND010
GND045
GND033
GND021
GND009
GND044
GND032
GND020
GND008
GND043
GND031
GND019
VCC001
VCC025
VCC002
VCC014
VCC013
VCC026
VCC037
VCC038
VCC003
VCC004
VCC015
VCC016
VCC027
VCC028
VCC039
VCC040
VCC041
VCC042
VCC030
VCC029
VCC017
VCC018
VCC005
VCC007
VCC006
VCC019
VCC031
VCC043
VCC008
VCC020
VCC032
VCC044
VCC009
VCC021
VCC033
TP448
VCC010
VCC022
VCC034
VCC046
VCC011
VCC023
VCC035
VCC047
VCC048
VCC036
VCC012
VCC024

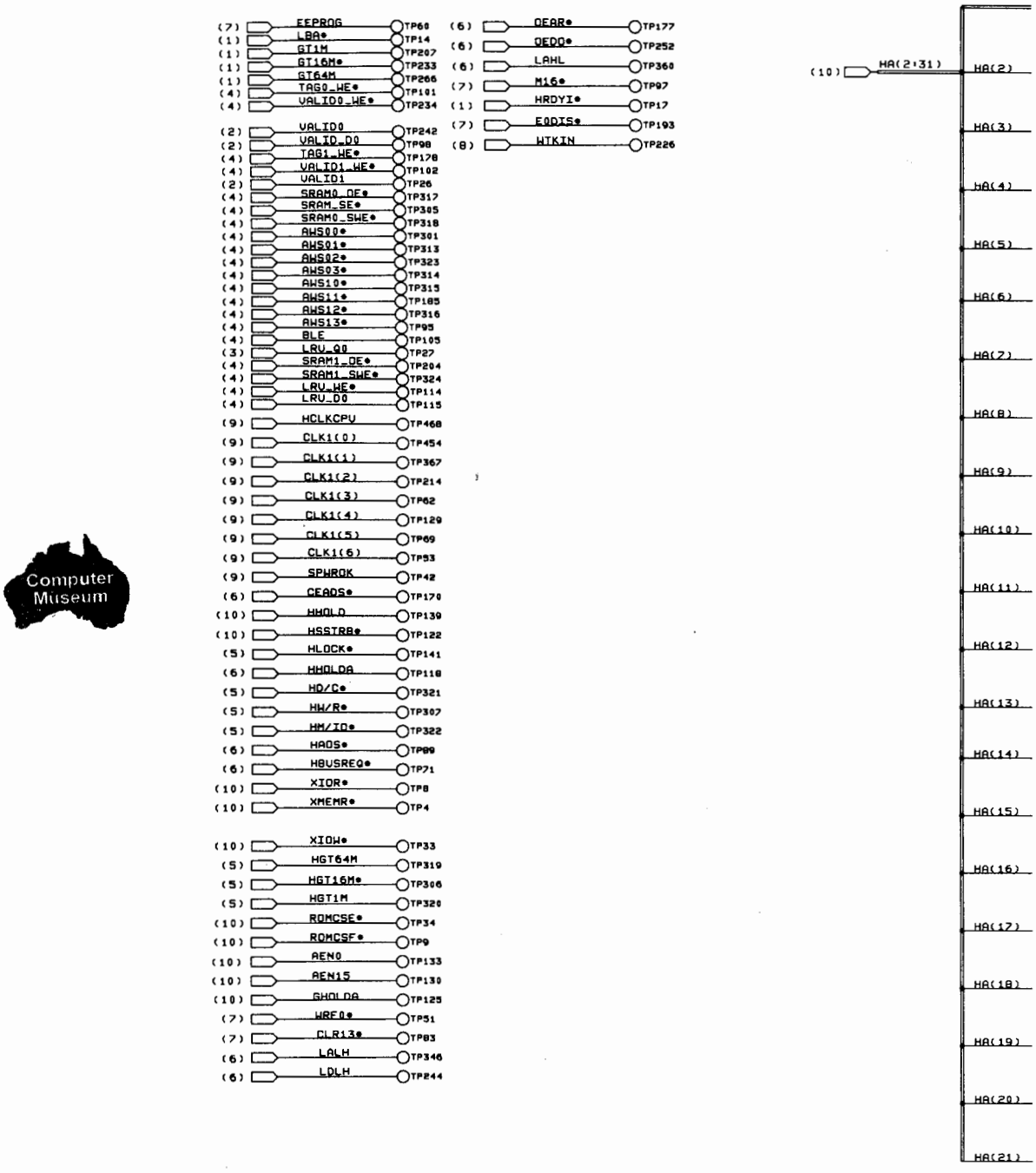
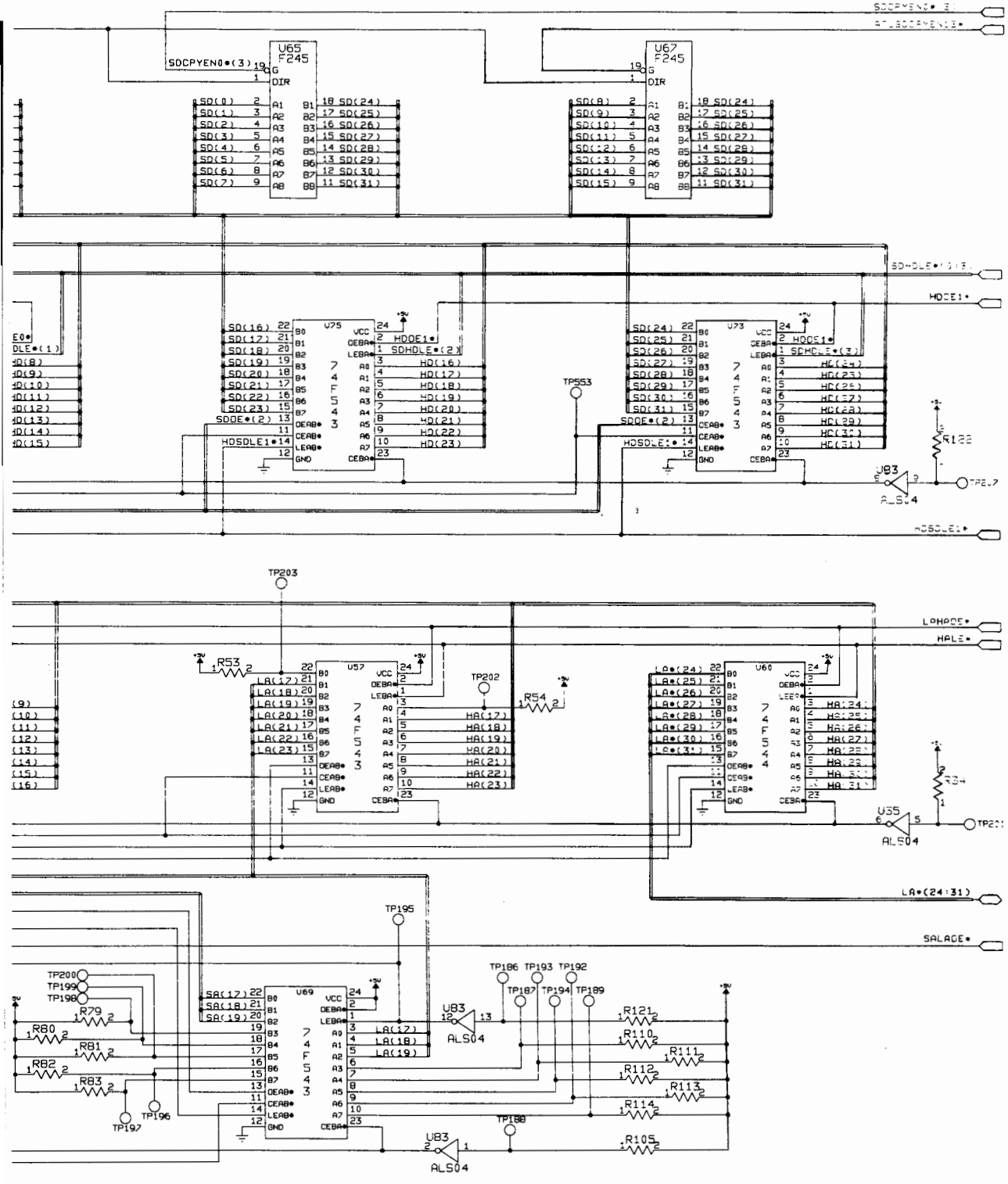


Figure B-13. System Board (Data Steering/Address Buffers)
Schematics B-27/B-28


U65
F245
U67
F245
SDCPYEN0*(3)
SD(0)
SD(1)
SD(2)
SD(3)
SD(4)
SD(5)
SD(6)
SD(7)
SD(8)
SD(9)
SD(10)
SD(11)
SD(12)
SD(13)
SD(14)
SD(15)
SD(16)
SD(17)
SD(18)
SD(19)
SD(20)
SD(21)
SD(22)
SD(23)
SD(24)
SD(25)
SD(26)
SD(27)
SD(28)
SD(29)
SD(30)
SD(31)
HDDE1*
SDHDLE*(2)
SDDE*(2)
HDSDLE1*
TP553
R122
U83
ALS04
TP203
TP202
R53
R54
LAHADE*
HALE*
LA*(24:31)
SALADE*
TP195
TP200
TP199
TP198
R79
R80
R81
R82
R83
TP197
TP196
TP186
TP193
TP192
TP187
TP194
TP189
R121
R110
R111
R112
R113
R114
R105
TP188

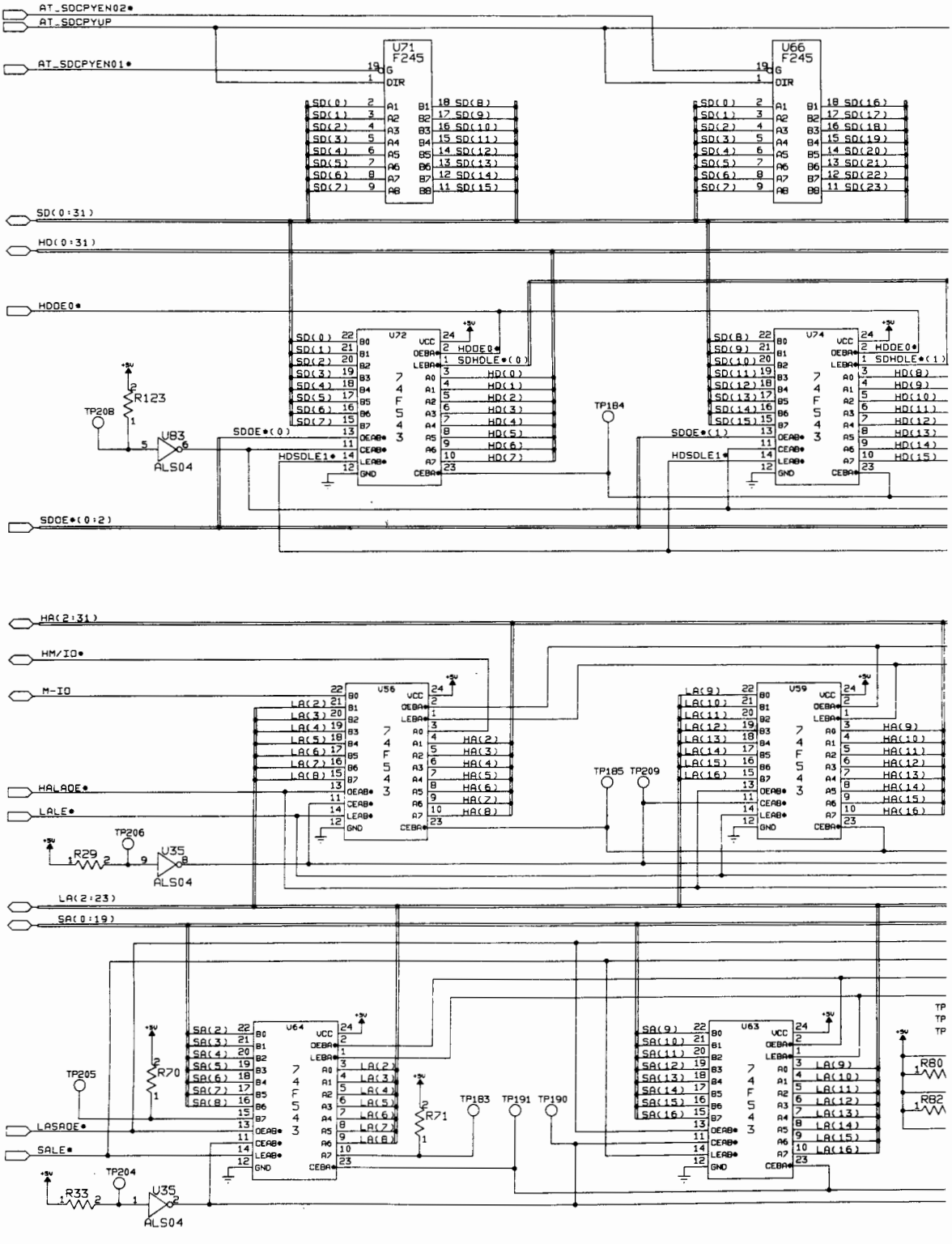
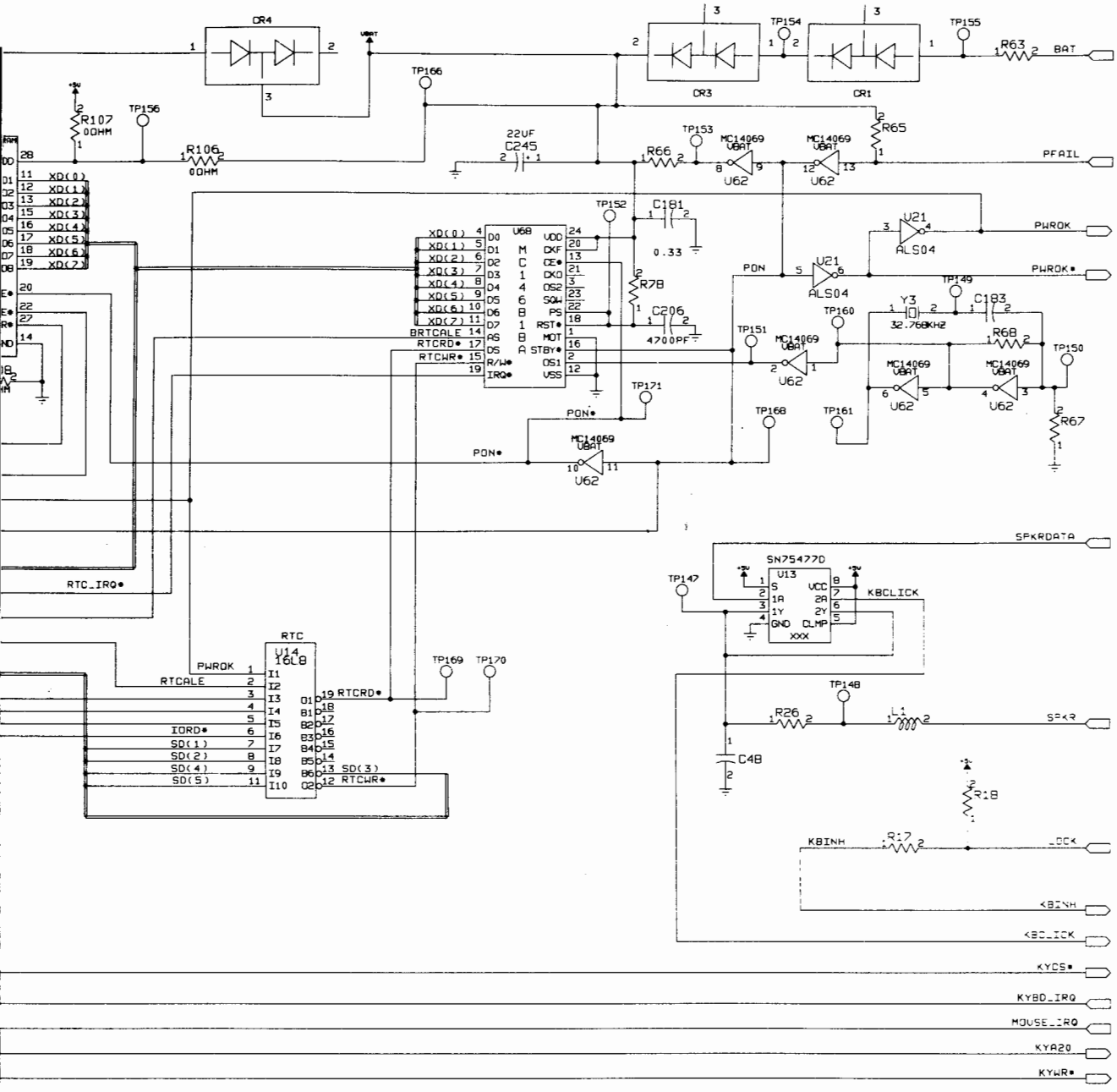

AT_SDCPYEN02*
AT_SDCPYUP
AT_SDCPYEN01*
U71
F245
U66
F245
SD(0:31)
HD(0:31)
HDDE0*
U72
U74
R123
TP208
U83
ALS04
SDDE*(0)
HDSDLE1*
TP184
SDDE*(1)
SDDE*(0:2)
HA(2:31)
HM/IO*
M-IO
U56
U59
HALADE*
LALE*
TP206
R29
U35
ALS04
TP185
TP209
LA(2:23)
SA(0:19)
U64
U63
TP205
R70
R71
TP183
TP191
TP190
LASADE*
SALE*
TP204
R33
U35
ALS04
R80
R82

Figure B-14. System Board (RTC, CMOS, Keyboard, Mouse, Speaker)
Schematics B-29/B-30


CR4
CR3
CR1
TP154
TP155
R63
BAT
TP166
R107
00HM
TP156
R106
00HM
22UF
C245
TP153
R66
MC14069
U62
R65
PFAIL
XD(0)
XD(1)
XD(2)
XD(3)
XD(4)
XD(5)
XD(6)
XD(7)
U68
TP152
C181
0.33
R78
C206
4700PF
U21
ALS04
PWROK
PWROK•
PON
TP149
Y3
32.768KHZ
C183
R68
TP150
TP151
TP160
TP161
TP168
TP171
PON•
R67
BRTCALE
RTCRD•
RTCWR•
IRQ•
RTC_IRQ•
RTC
U14
16L8
PWROK
RTCALE
IORD•
SD(1)
SD(2)
SD(4)
SD(5)
SD(3)
TP169
TP170
SPKRDATA
SN75477D
U13
TP147
KBCLICK
TP148
R26
SPKR
C48
R18
KBINH
R17
LOCK
KYCS•
KYBD_IRQ
MOUSE_IRQ
KYA20
KYWR•

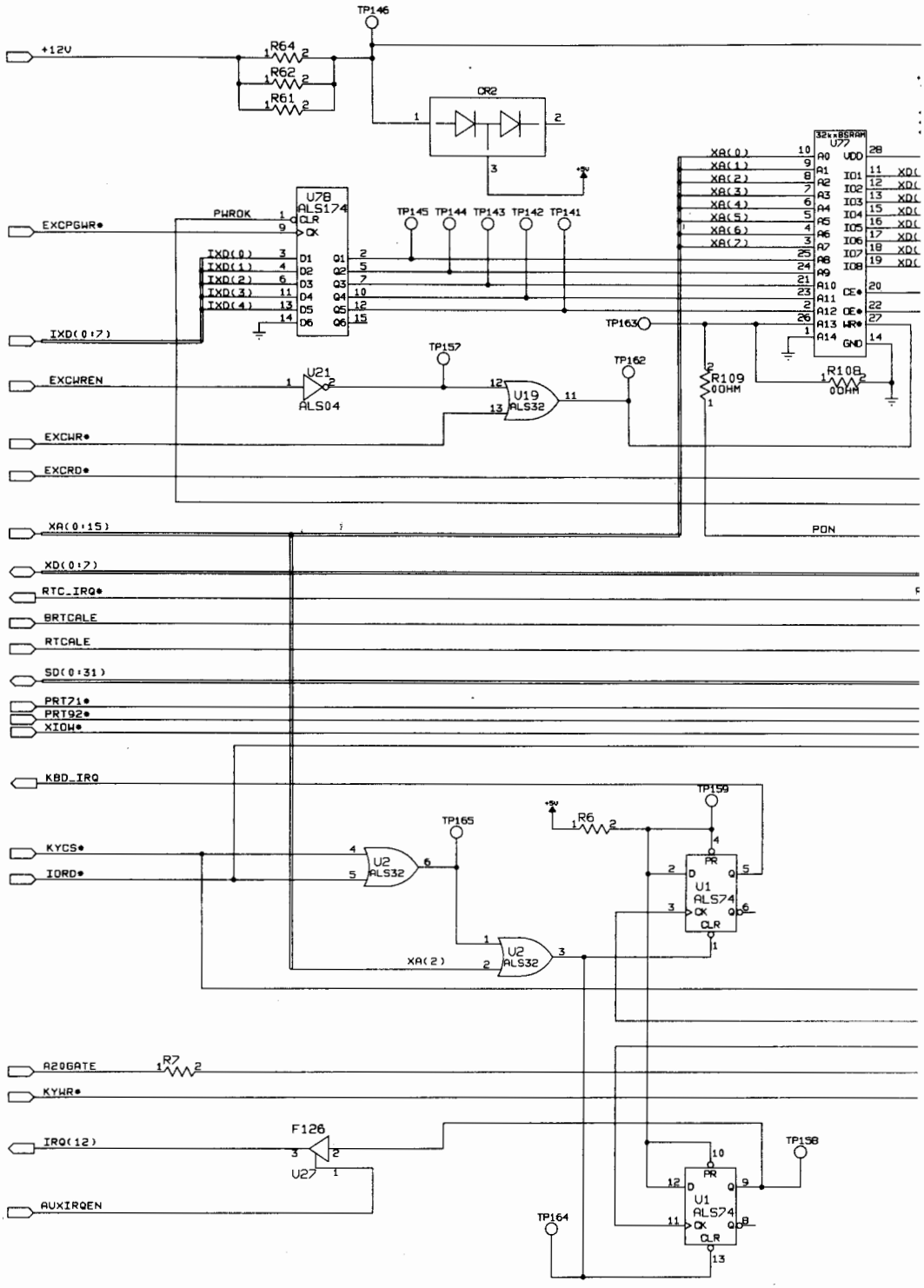
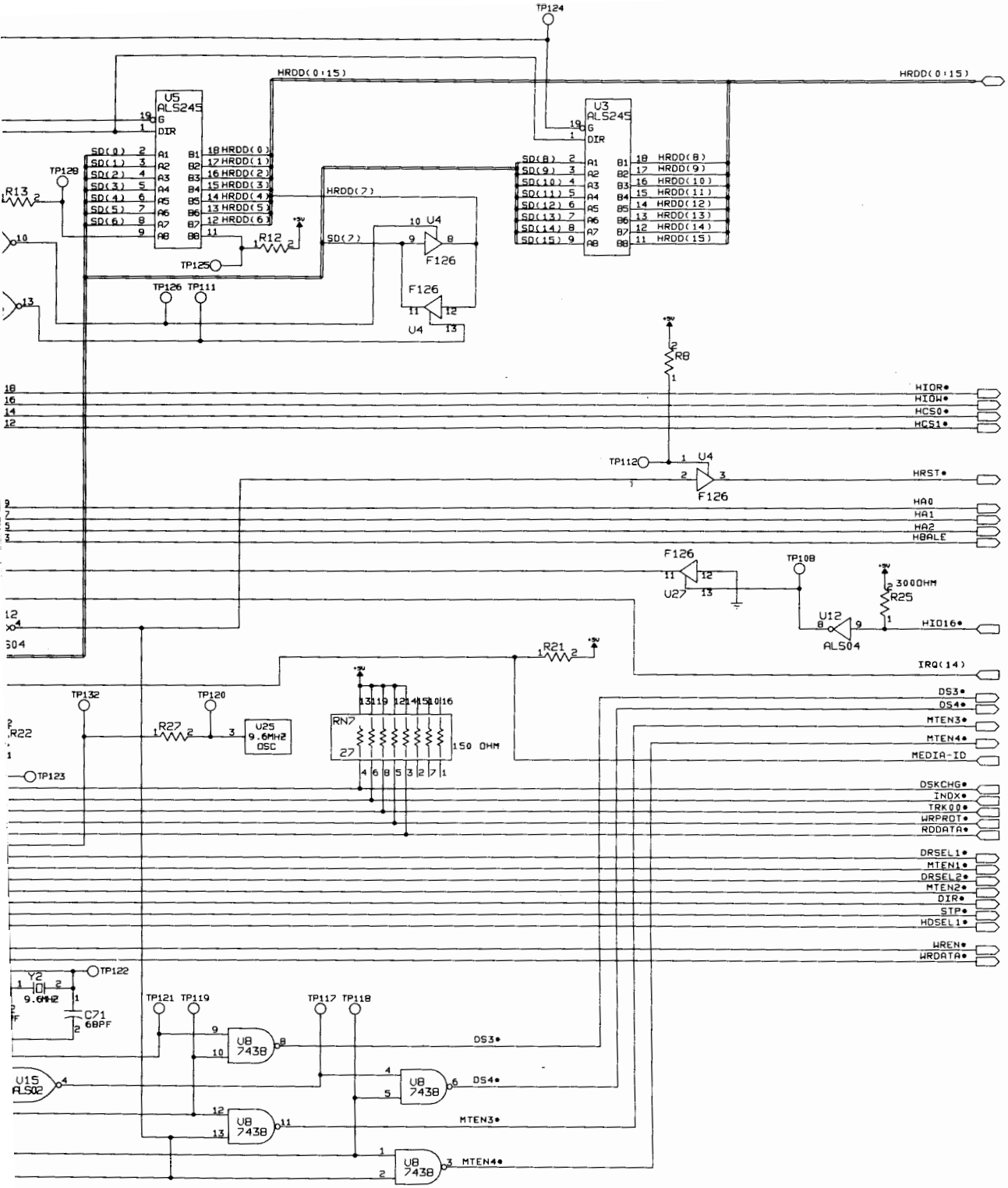

TP146
+12V
R64
R62
R61
DR2
+5V
U78
ALS174
PWROK
EXCPGWR•
IXD(0:7)
IXD(0)
IXD(1)
IXD(2)
IXD(3)
IXD(4)
TP145 TP144 TP143 TP142 TP141
32K×8SRAM
U77
XA(0)
XA(1)
XA(2)
XA(3)
XA(4)
XA(5)
XA(6)
XA(7)
TP163
TP157
TP162
EXCWREN
U21
ALS04
U19
ALS32
R109
00HM
R108
00HM
EXCWR•
EXCRD•
XA(0:15)
PON
XD(0:7)
RTC_IRQ•
BRTCALE
RTCALE
SD(0:31)
PRT71•
PRT92•
XIOW•
KBD_IRQ
TP159
TP165
R6
KYCS•
IORD•
U2
ALS32
U1
ALS74
XA(2)
A20GATE
R7
KYWR•
IRQ(12)
F126
U27
AUXIRQEN
TP158
TP164

Figure B-15. System Board (Flexible/Hard Disk Drive Support)
Schematics B-31/B-32


TP124
HRDD(0:15)
HRDD(0:15)
U5
ALS245
U3
ALS245
DIR
SD(0) SD(1) SD(2) SD(3) SD(4) SD(5) SD(6)
HRDD(0) HRDD(1) HRDD(2) HRDD(3) HRDD(4) HRDD(5) HRDD(6)
SD(8) SD(9) SD(10) SD(11) SD(12) SD(13) SD(14) SD(15)
HRDD(8) HRDD(9) HRDD(10) HRDD(11) HRDD(12) HRDD(13) HRDD(14) HRDD(15)
TP128
R13
R12
HRDD(7)
SD(7)
U4
F126
TP125
TP126
TP111
R8
HIOR•
HIOW•
HCS0•
HCS1•
TP112
HRST•
HA0
HA1
HA2
H8ALE
TP108
300OHM
R25
U12
ALS04
HI016•
IRQ(14)
R21
TP132
TP120
R27
R22
U25
9.6MHZ
OSC
RN7
27
150 OHM
DS3•
DS4•
MTEN3•
MTEN4•
MEDIA-ID
DSKCHG•
INDX•
TRK00•
WRPROT•
RDDATA•
DRSEL1•
MTEN1•
DRSEL2•
MTEN2•
DIR•
STP•
HDSEL1•
WREN•
WRDATA•
TP123
TP122
Y2
9.6MHZ
C71
68PF
TP121
TP119
TP117
TP118
U8
7438
U15
ALS02

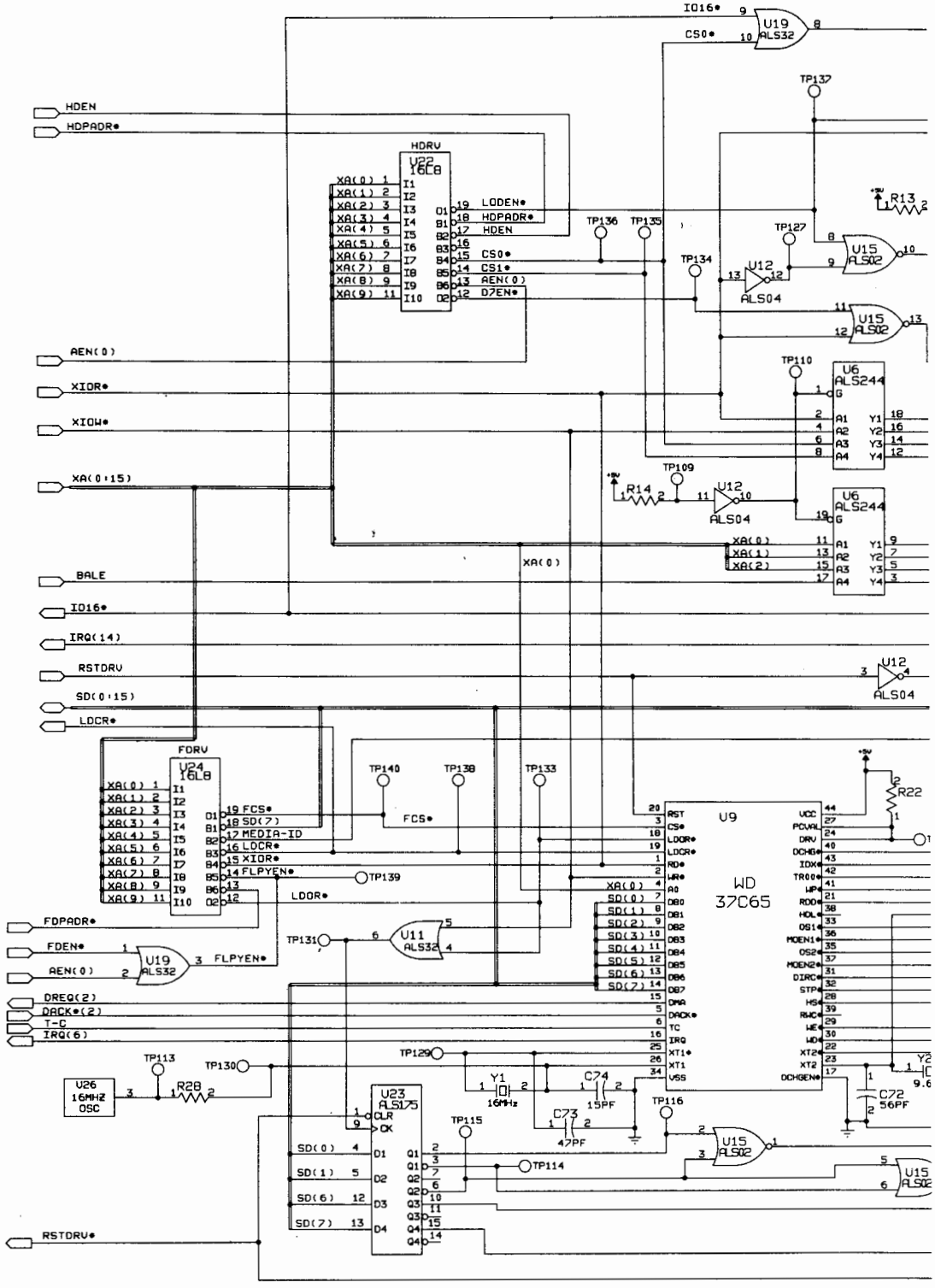
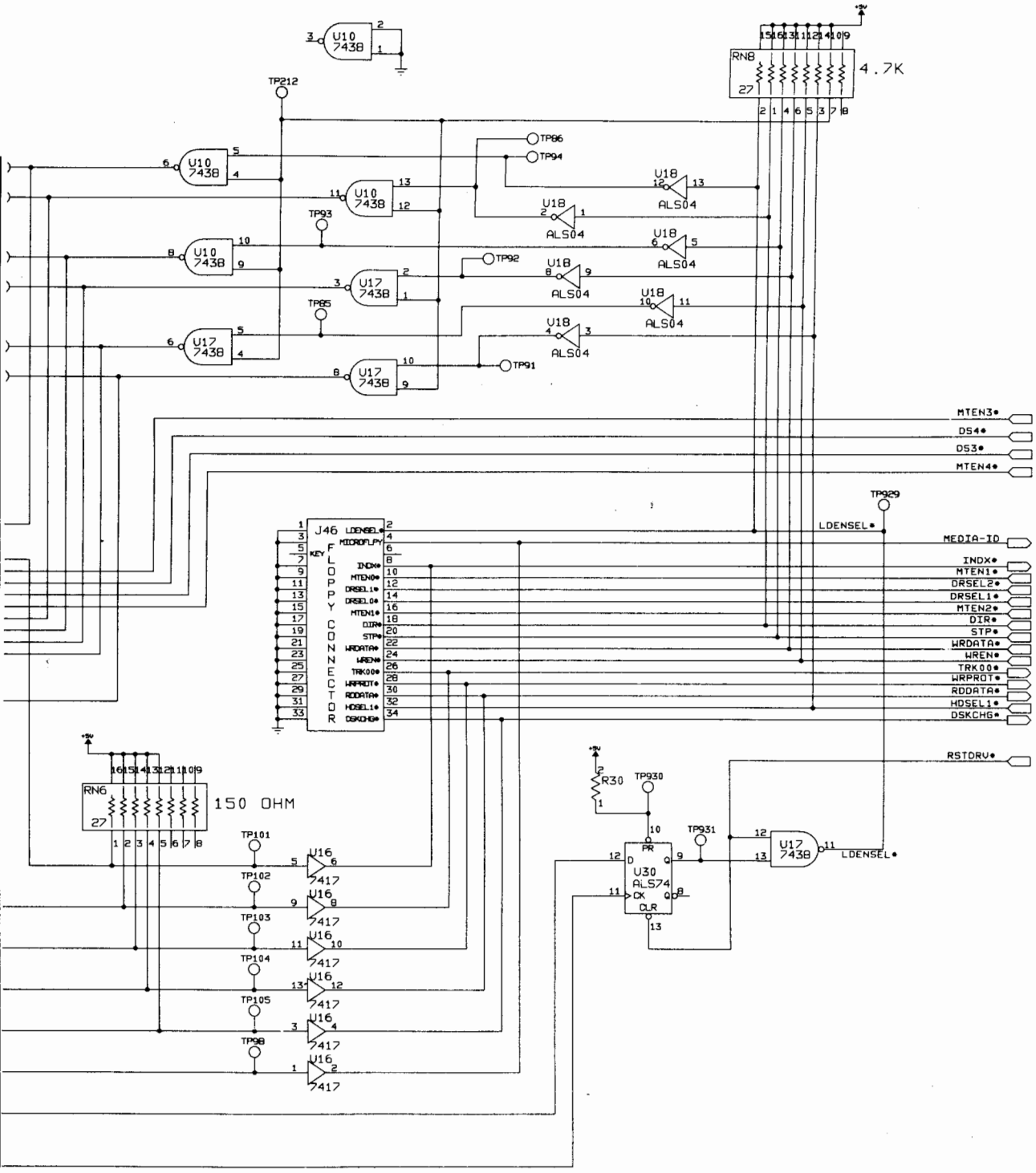

HDEN
HDPADR*
AEN(0)
XIOR*
XIOW*
XA(0:15)
BALE
IO16*
IRQ(14)
RSTDRV
SD(0:15)
LDCR*
FDPADR*
FDEN*
AEN(0)
DREQ(2)
DACK*(2)
T-C
IRQ(6)
RSTDRV*
HDRV
U22
16L8
LODEN*
HDPADR*
HDEN
CS0*
CS1*
AEN(0)
D2EN*
FDRV
U24
16L8
FCS*
MEDIA-ID
LDCR*
XIOR*
FLPYEN*
U19
ALS32
U11
ALS32
U12
ALS04
U15
ALS02
U6
ALS244
U9
WD
37C65
U23
ALS175
U26
16MHZ
OSC
Y1
16MHz
C74
15PF
C73
47PF
C72
56PF
R13
R14
R22
R28
TP109
TP110
TP113
TP114
TP115
TP116
TP127
TP129
TP130
TP131
TP133
TP134
TP135
TP136
TP137
TP138
TP139
TP140

Figure B-16. System Board (Connectors)
Schematics B-33/B-34


U10 7438
U17 7438
U18 ALS04
RN8
4.7K
RN6
150 OHM
TP212
TP86
TP94
TP93
TP92
TP85
TP91
TP929
TP930
TP931
TP101
TP102
TP103
TP104
TP105
TP98
J46
FLOPPY CONNECTOR
KEY
LDENSEL•
MICROFLPY
INDX•
MTEN0•
DRSEL1•
DRSEL0•
MTEN1•
DIR•
STP•
WRDATA•
WREN•
TRK00•
WRPROT•
RDDATA•
HDSEL1•
DSKCHG•
MTEN3•
DS4•
DS3•
MTEN4•
MEDIA-ID
MTEN2•
DRSEL2•
RSTDRV•
R30
U30 ALS74
PR
CK
CLR
U16 7417

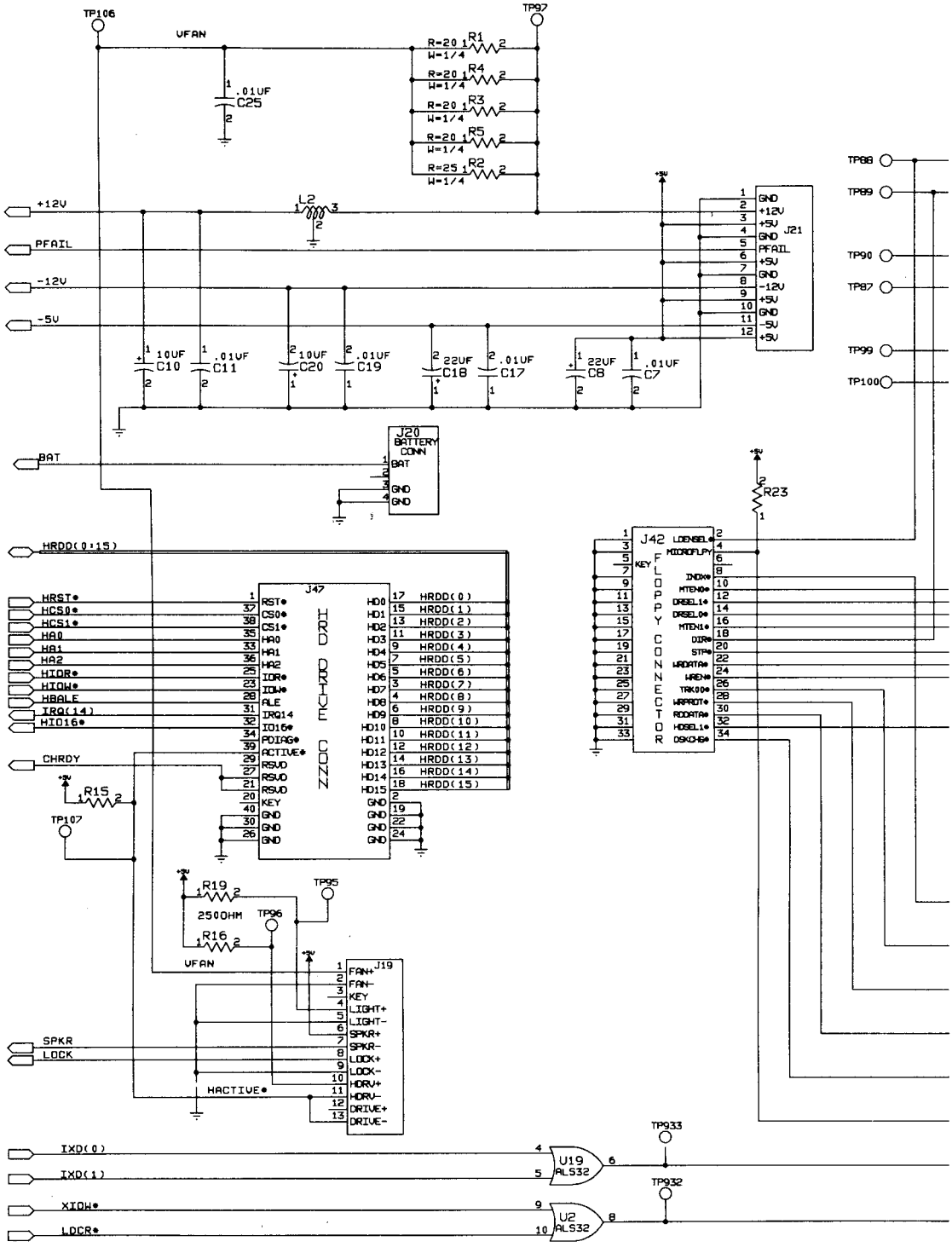
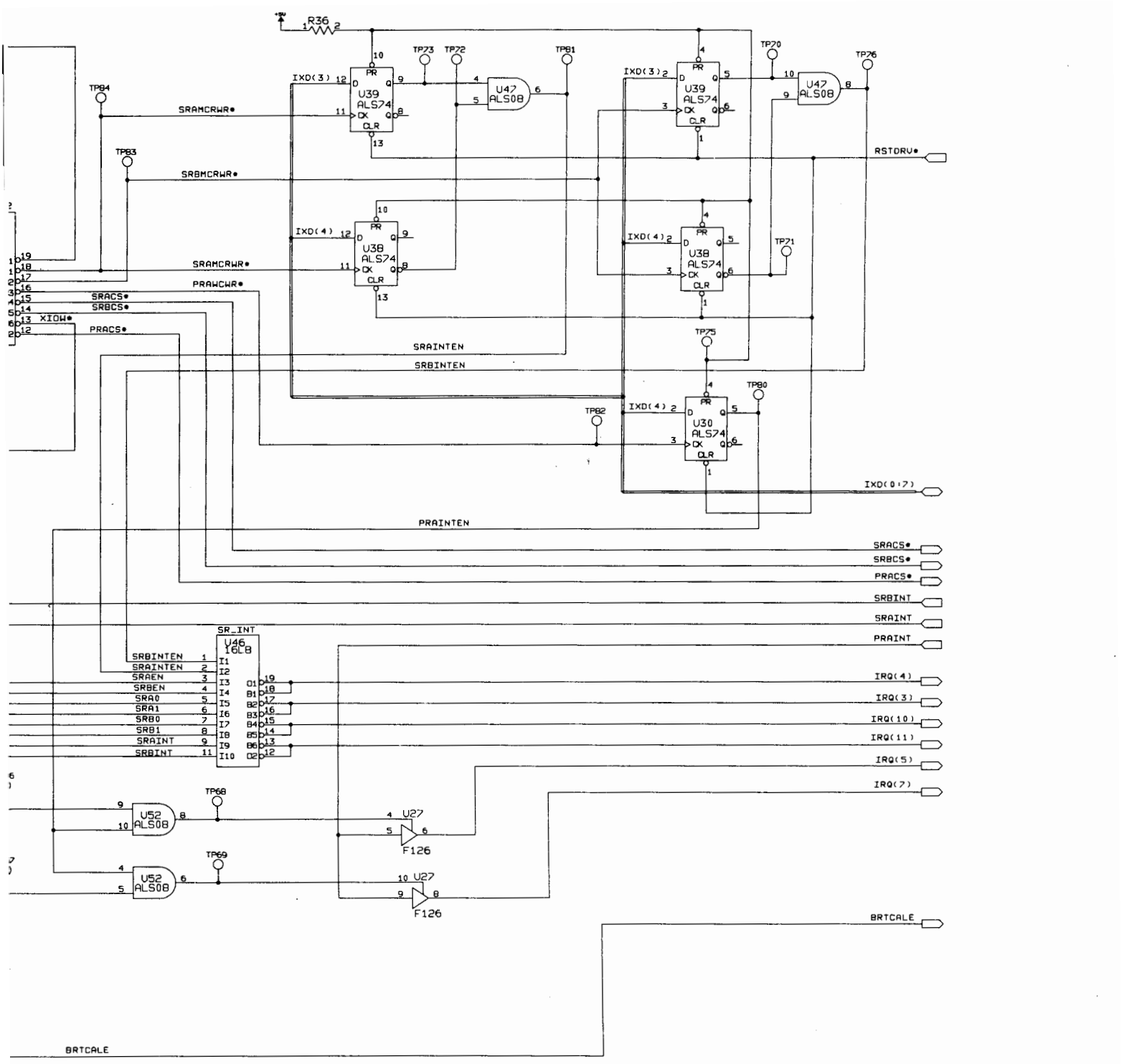

TP106
TP97
VFAN
.01UF
C25
R-20 1
W=1/4
R1
R4
R3
R5
R2
R-25 1
W=1/4
+12V
PFAIL
-12V
-5V
L2
J21
GND
+12V
+5V
GND
PFAIL
+5V
GND
-12V
+5V
GND
-5V
+5V
10UF
C10
.01UF
C11
10UF
C20
.01UF
C19
22UF
C18
.01UF
C17
22UF
C8
.01UF
C7
TP88
TP89
TP90
TP87
TP99
TP100
J20
BATTERY
CONN
BAT
GND
GND
BAT
R23
HRDD(0:15)
J47
HRST*
HCS0*
HCS1*
HA0
HA1
HA2
HIOR*
HIOW*
HBALE
IRQ(14)
HID16*
CHRDY
RST*
CS0*
CS1*
HA0
HA1
HA2
IOR*
IOW*
ALE
IRQ14
IO16*
PDIAG*
ACTIVE*
RSVD
RSVD
RSVD
KEY
GND
GND
GND
HD0
HD1
HD2
HD3
HD4
HD5
HD6
HD7
HD8
HD9
HD10
HD11
HD12
HD13
HD14
HD15
GND
GND
GND
GND
HRDD(0)
HRDD(1)
HRDD(2)
HRDD(3)
HRDD(4)
HRDD(5)
HRDD(6)
HRDD(7)
HRDD(8)
HRDD(9)
HRDD(10)
HRDD(11)
HRDD(12)
HRDD(13)
HRDD(14)
HRDD(15)
HARD DISK INTERFACE
J42
FLOPPY CONNECTOR
LDENSEL*
MICRODLPY
KEY
INDX*
MTEN0*
DRSEL1*
DRSEL0*
MTEN1*
DIR*
STP*
WRDATA*
WREN*
TRK00*
WRPROT*
RDDATA*
HDSEL1*
DSKCHG*
R15
TP107
R19
2500HM
R16
TP96
TP95
VFAN
J19
FAN+
FAN-
KEY
LIGHT+
LIGHT-
SPKR+
SPKR-
LOCK+
LOCK-
HDRV+
HDRV-
DRIVE+
DRIVE-
SPKR
LOCK
HACTIVE*
IXD(0)
IXD(1)
XIOW*
LDCR*
U19
ALS32
U2
ALS32
TP933
TP932

Figure B-17. System Board (Serial/Parallel Support Logic)
Schematics B-35/B-36


R36
TP84
TP83
TP73
TP72
TP81
TP70
TP76
TP71
TP75
TP80
TP82
TP68
TP69
U39 ALS74
U38 ALS74
U30 ALS74
U47 ALS08
U52 ALS08
U27 F126
U46 16L8
SR_INT
SRAMCRWR•
SRBMCRWR•
PRAWCWR•
SRACS•
SRBCS•
XIOW•
PRACS•
SRAINTEN
SRBINTEN
PRAINTEN
IXD(3)
IXD(4)
IXD(0:7)
RSTDRV•
SRBINTEN
SRAINTEN
SRAEN
SRBEN
SRA0
SRA1
SRB0
SRB1
SRAINT
SRBINT
SRBINT
SRAINT
PRAINT
IRQ(4)
IRQ(3)
IRQ(10)
IRQ(11)
IRQ(5)
IRQ(7)
BRTCALE
BRTCALE

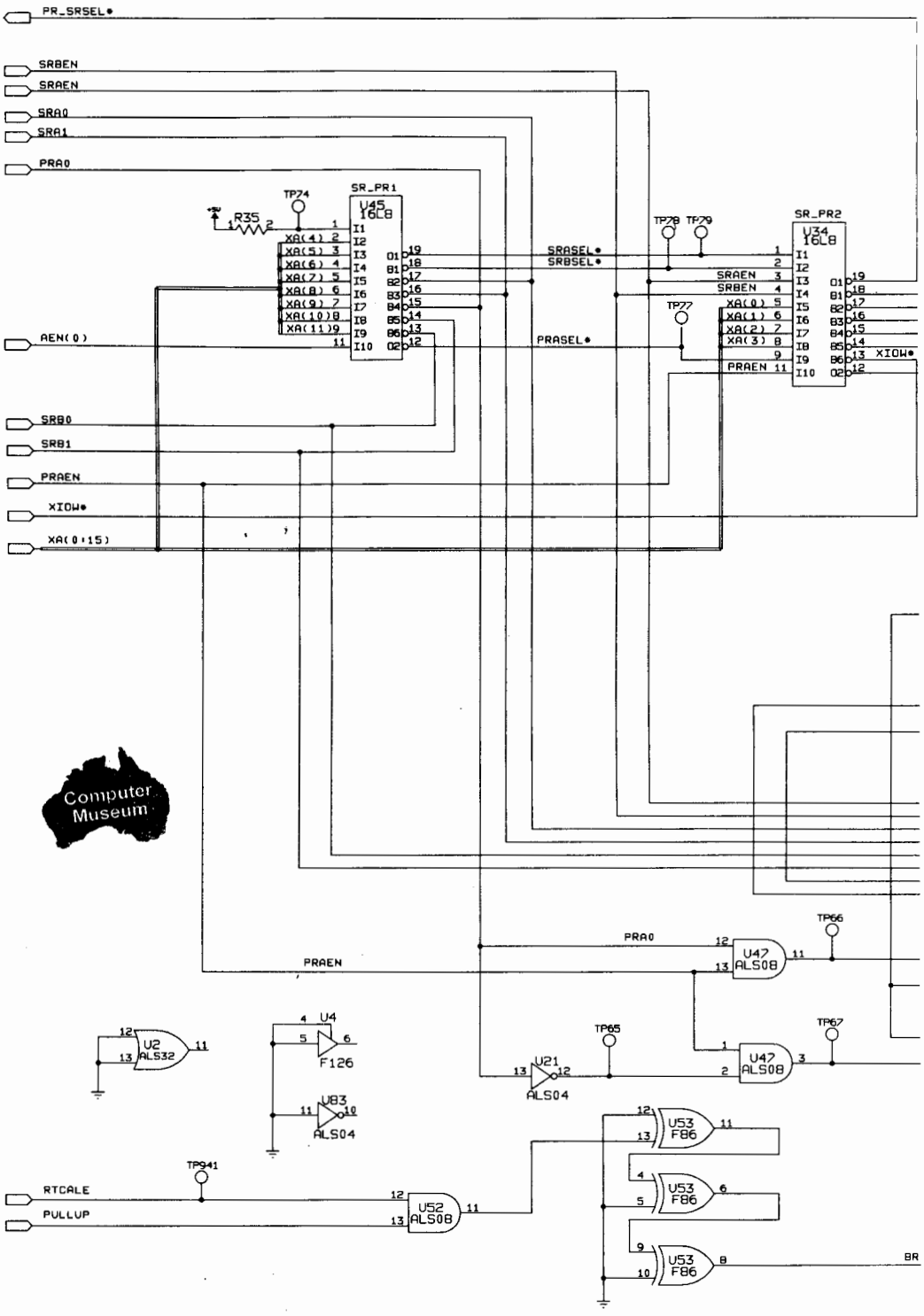
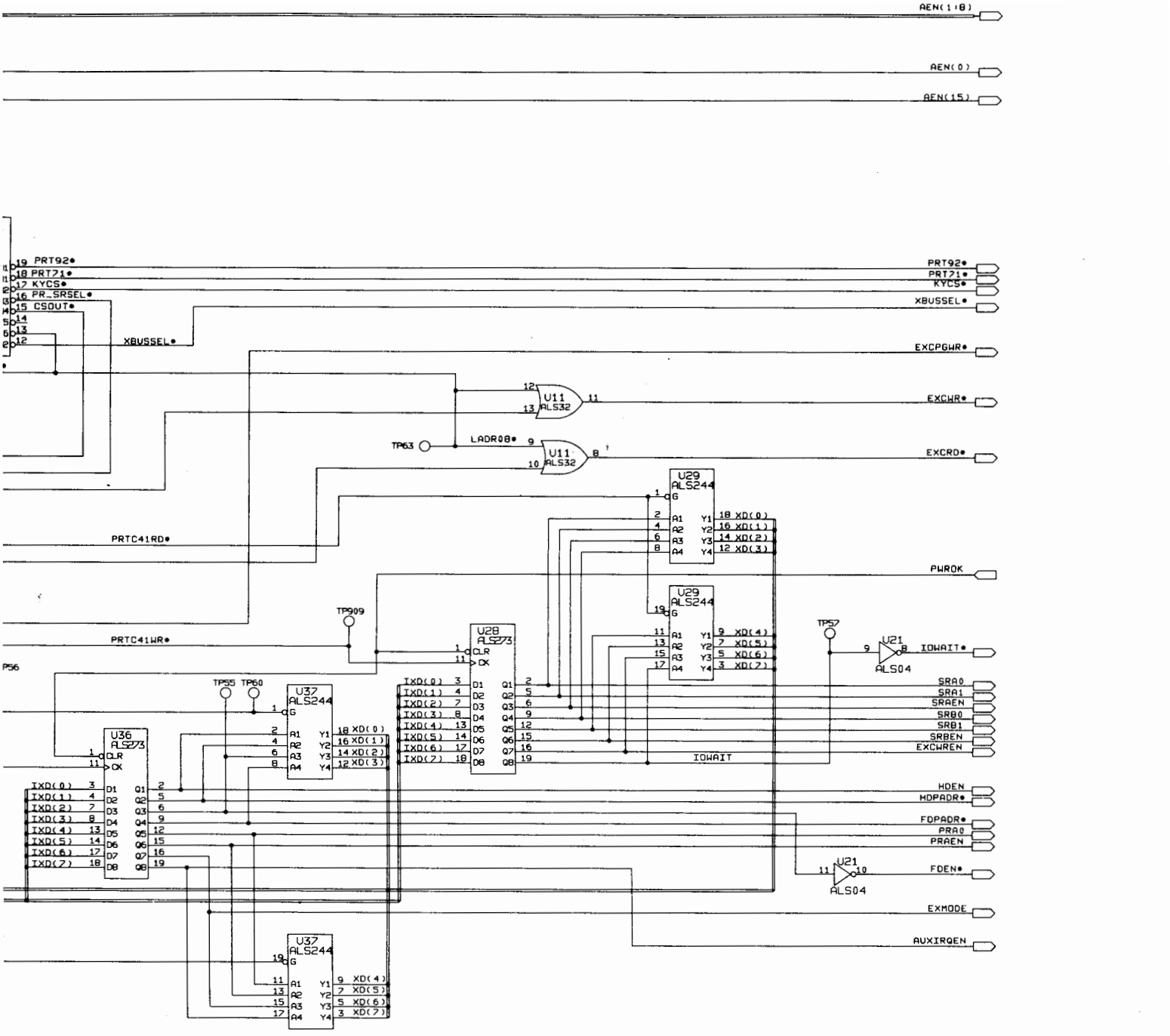

PR_SRSEL*
SRBEN
SRAEN
SRA0
SRA1
PRA0
AEN(0)
SRB0
SRB1
PRAEN
XIOW*
XA(0:15)
SR_PR1
V45
16L8
TP74
R35
SR_PR2
V34
16L8
TP78
TP79
TP77
SRASEL*
SRBSEL*
PRASEL*
SRAEN
SRBEN
PRAEN
XA(0)
XA(1)
XA(2)
XA(3)
XIOW*
Computer Museum
TP66
U47
ALS08
TP67
TP65
U21
ALS04
U2
ALS32
U4
F126
U83
ALS04
TP941
RTCALE
PULLUP
U52
ALS08
U53
F86
BR

Figure B-18. System Board (System Ports, Strobes, AEN Logic)
Schematics B-37/B-38



R/W PORT 0C40

R/W PORT 0C41

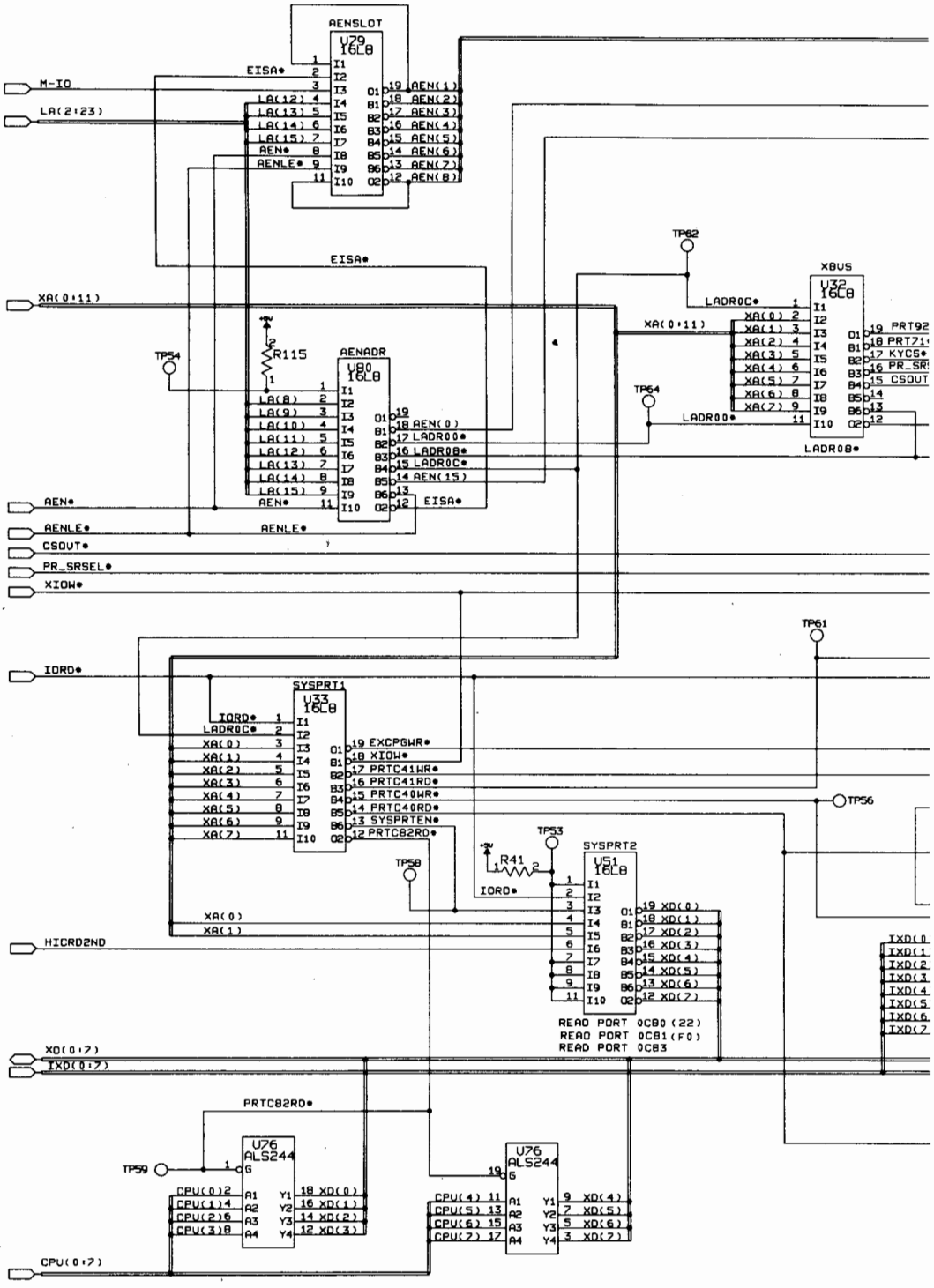
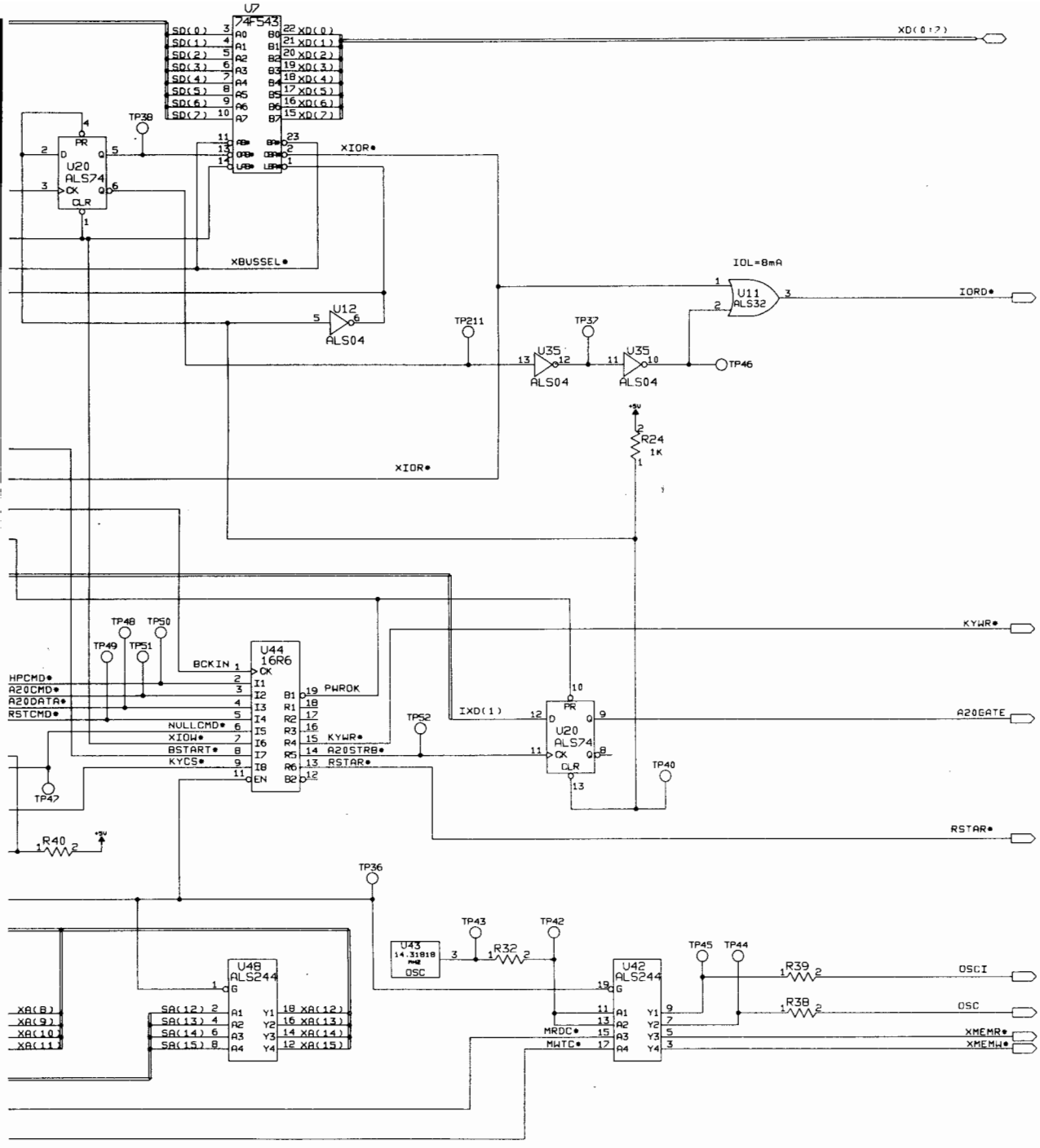

AENSLOT
U79
16L8
M-IO
LA(2:23)
EISA*
LA(12)
LA(13)
LA(14)
LA(15)
AEN*
AENLE*
AEN(1)
AEN(2)
AEN(3)
AEN(4)
AEN(5)
AEN(6)
AEN(7)
AEN(8)
TP62
XBUS
U32
16L8
XA(0:11)
LADR0C*
XA(0)
XA(1)
XA(2)
XA(3)
XA(4)
XA(5)
XA(6)
XA(7)
PRT92
KYCS*
PR_SRS
CSOUT
TP54
R115
AENADR
U80
16L8
LA(8)
LA(9)
LA(10)
LA(11)
LA(12)
LA(13)
LA(14)
LA(15)
AEN(0)
LADR00*
LADR08*
LADR0C*
AEN(15)
EISA*
TP64
LADR00*
LADR08*
AEN*
AENLE*
CSOUT*
PR_SRSEL*
XIOW*
IORD*
TP61
SYSPRT1
U33
16L8
EXCPGWR*
XIOW*
PRTC41WR*
PRTC41RD*
PRTC40WR*
PRTC40RD*
SYSPRTEN*
PRTC82RD*
TP56
TP58
TP53
SYSPRT2
U51
16L8
R41
XA(0)
XA(1)
HICRD2ND
XD(0)
XD(1)
XD(2)
XD(3)
XD(4)
XD(5)
XD(6)
XD(7)
READ PORT 0C80 (22)
READ PORT 0C81 (F0)
READ PORT 0C83
IXD(0:7)
XD(0:7)
PRTC82RD*
TP59
U76
ALS244
CPU(0)
CPU(1)
CPU(2)
CPU(3)
CPU(4)
CPU(5)
CPU(6)
CPU(7)
CPU(0:7)

Figure B-19. System Board (XD/SD Buffer, Fast A20/RST, I/O Wait)
Schematics B-39/B-40


U7
74F543
SD(0)
SD(1)
SD(2)
SD(3)
SD(4)
SD(5)
SD(6)
SD(7)
XD(0)
XD(1)
XD(2)
XD(3)
XD(4)
XD(5)
XD(6)
XD(7)
XD(0:7)
TP38
U20
ALS74
PR
D
CK
CLR
Q
XIOR•
XBUSSEL•
IOL=8mA
U11
ALS32
IORD•
U12
ALS04
TP211
TP37
U35
ALS04
TP46
+5V
R24
1K
KYWR•
TP48
TP50
TP49
TP51
U44
16R6
BCKIN
HPCMD•
A20CMD•
A20DATA•
RSTCMD•
NULLCMD•
XIOW•
BSTART•
KYCS•
PWROK
KYWR•
A20STRB•
RSTAR•
IXD(1)
TP52
A20GATE
TP40
TP47
R40
RSTAR•
TP36
U43
14.31818
OSC
TP43
TP42
R32
U48
ALS244
U42
ALS244
TP45
TP44
R39
R38
OSCI
OSC
XA(8)
XA(9)
XA(10)
XA(11)
SA(12)
SA(13)
SA(14)
SA(15)
XA(12)
XA(13)
XA(14)
XA(15)
MRDC•
MWTC•
XMEMR•
XMEMW•

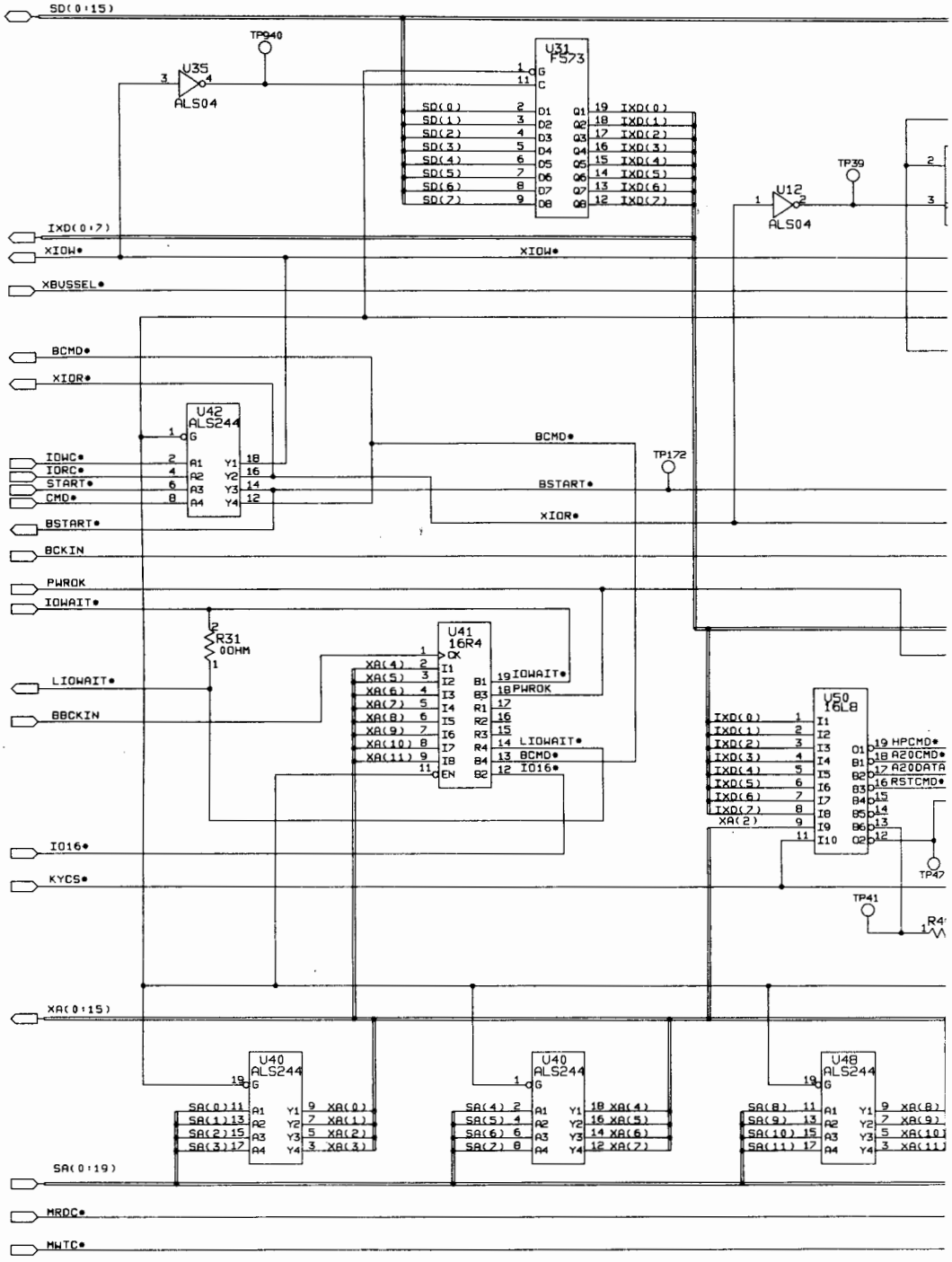
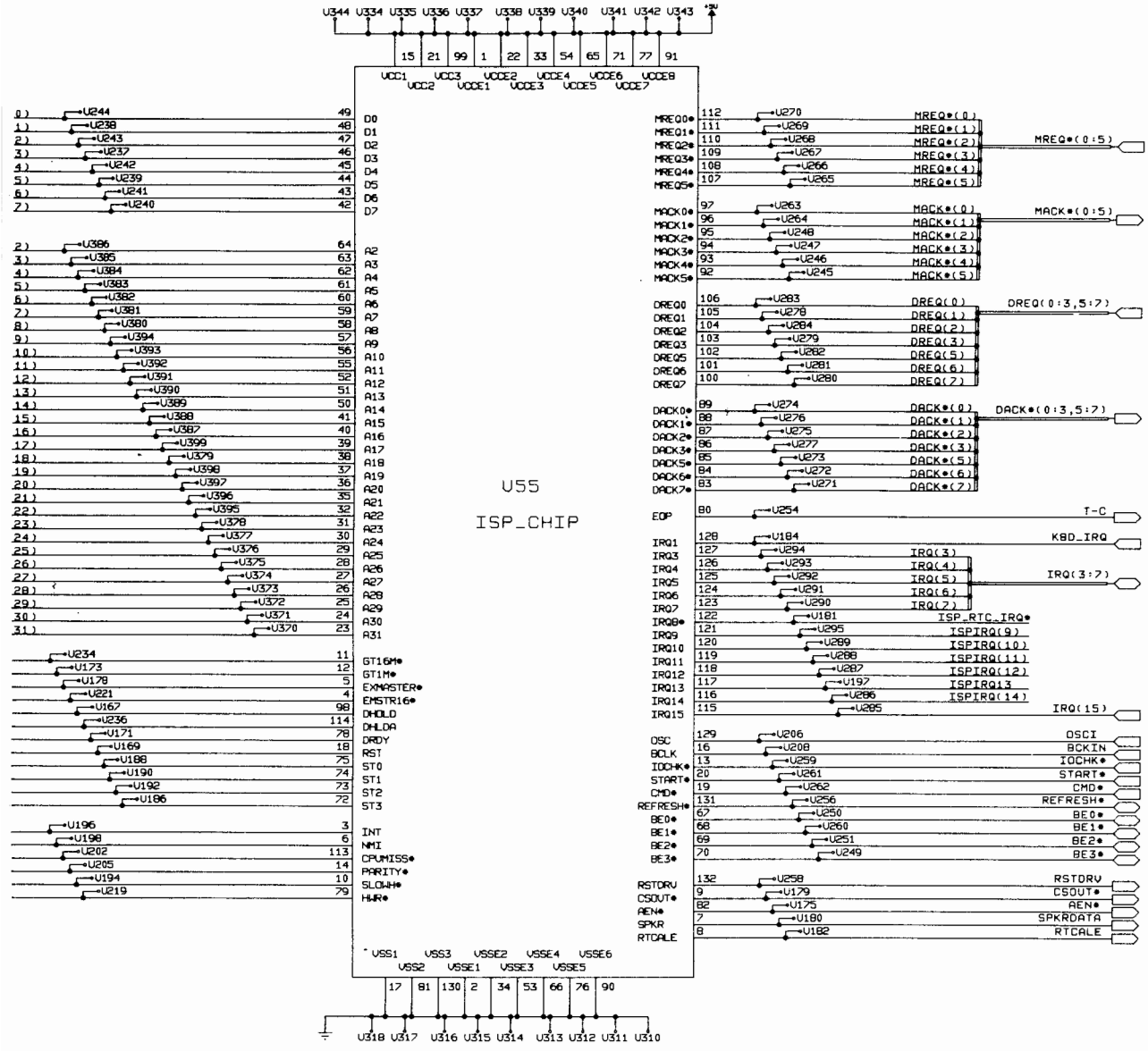
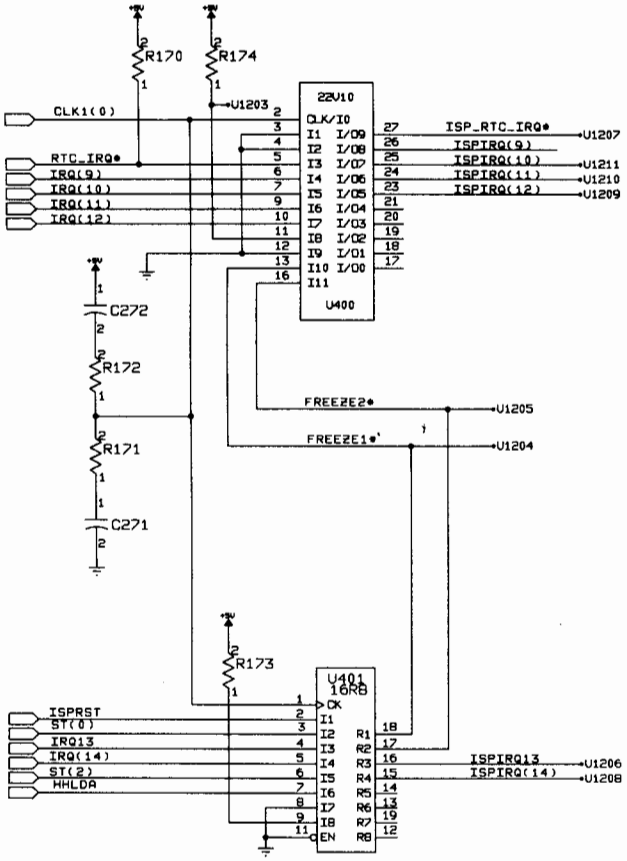

SD(0:15)
TP940
U35
ALS04
U31
F573
SD(0)
SD(1)
SD(2)
SD(3)
SD(4)
SD(5)
SD(6)
SD(7)
IXD(0)
IXD(1)
IXD(2)
IXD(3)
IXD(4)
IXD(5)
IXD(6)
IXD(7)
TP39
U12
ALS04
IXD(0:7)
XIOW•
XBUSSEL•
BCMD•
XIOR•
U42
ALS244
IOWC•
IORC•
START•
CMD•
BSTART•
BCKIN
PWROK
IOWAIT•
R31
00HM
LIOWAIT•
BBCKIN
U41
16R4
XA(4)
XA(5)
XA(6)
XA(7)
XA(8)
XA(9)
XA(10)
XA(11)
IOWAIT•
PWROK
LIOWAIT•
BCMD•
IO16•
TP172
U50
16L8
IXD(0)
IXD(1)
IXD(2)
IXD(3)
IXD(4)
IXD(5)
IXD(6)
IXD(7)
XA(2)
HPCMD•
A20CMD•
A20DATA
RSTCMD•
TP47
TP41
IO16•
KYCS•
XA(0:15)
U40
ALS244
SA(0)
SA(1)
SA(2)
SA(3)
XA(0)
XA(1)
XA(2)
XA(3)
U40
ALS244
SA(4)
SA(5)
SA(6)
SA(7)
XA(4)
XA(5)
XA(6)
XA(7)
U48
ALS244
SA(8)
SA(9)
SA(10)
SA(11)
XA(8)
XA(9)
XA(10)
XA(11)
SA(0:19)
MRDC•
MWTC•

Figure B-20. System Board (Integrated Systems Peripheral)
Schematics B-41/B-42


U55
ISP_CHIP
VCC1 VCC2 VCC3 VCCE1 VCCE2 VCCE3 VCCE4 VCCE5 VCCE6 VCCE7 VCCE8
VSS1 VSS2 VSS3 VSSE1 VSSE2 VSSE3 VSSE4 VSSE5 VSSE6
D0 D1 D2 D3 D4 D5 D6 D7
A2 A3 A4 A5 A6 A7 A8 A9 A10 A11 A12 A13 A14 A15 A16 A17 A18 A19 A20 A21 A22 A23 A24 A25 A26 A27 A28 A29 A30 A31
GT16M* GT1M* EXMASTER* EMSTR16* DHOLD DHLDA DRDY RST ST0 ST1 ST2 ST3
INT NMI CPUMISS* PARITY* SLOWH* HWR*
MREQ0* MREQ1* MREQ2* MREQ3* MREQ4* MREQ5*
MACK0* MACK1* MACK2* MACK3* MACK4* MACK5*
DREQ0 DREQ1 DREQ2 DREQ3 DREQ5 DREQ6 DREQ7
DACK0* DACK1* DACK2* DACK3* DACK5* DACK6* DACK7*
EOP
IRQ1 IRQ3 IRQ4 IRQ5 IRQ6 IRQ7 IRQ8* IRQ9 IRQ10 IRQ11 IRQ12 IRQ13 IRQ14 IRQ15
OSC BCLK IOCHK* START* CMD* REFRESH* BE0* BE1* BE2* BE3*
RSTDRV CSOUT* AEN* SPKR RTCALE
MREQ*(0:5)
MACK*(0:5)
DREQ(0:3,5:7)
DACK*(0:3,5:7)
T-C
KBD_IRQ
IRQ(3:7)
ISP_RTC_IRQ*
ISPIRQ(9)
ISPIRQ(10)
ISPIRQ(11)
ISPIRQ(12)
ISPIRQ13
ISPIRQ(14)
IRQ(15)
OSCI
BCKIN
IOCHK*
START*
CMD*
REFRESH*
BE0*
BE1*
BE2*
BE3*
RSTDRV
CSOUT*
AEN*
SPKRDATA
RTCALE


R170
R174
22V10
CLK1(0)
U1203
CLK/I0
I1 I/O9
I2 I/O8
I3 I/O7
I4 I/O6
I5 I/O5
I6 I/O4
I7 I/O3
I8 I/O2
I9 I/O1
I10 I/O0
I11
U400
ISP_RTC_IRQ*
U1207
ISPIRQ(9)
ISPIRQ(10)
U1211
ISPIRQ(11)
U1210
ISPIRQ(12)
U1209
RTC_IRQ*
IRQ(9)
IRQ(10)
IRQ(11)
IRQ(12)
C272
R172
R171
C271
FREEZE2*
U1205
FREEZE1*
U1204
R173
U401
16R8
ISPRST
ST(0)
IRQ13
IRQ(14)
ST(2)
HHLDA
ISPIRQ13
U1206
ISPIRQ(14)
U1208

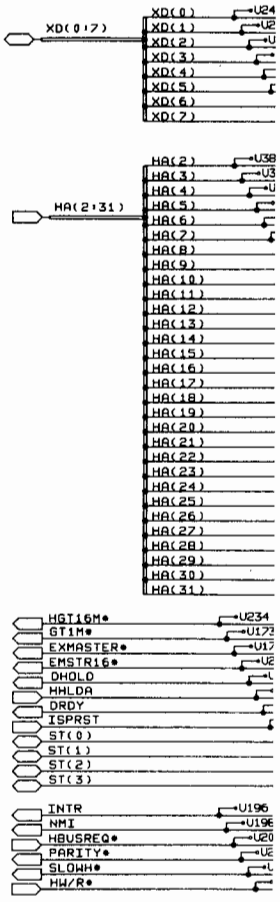
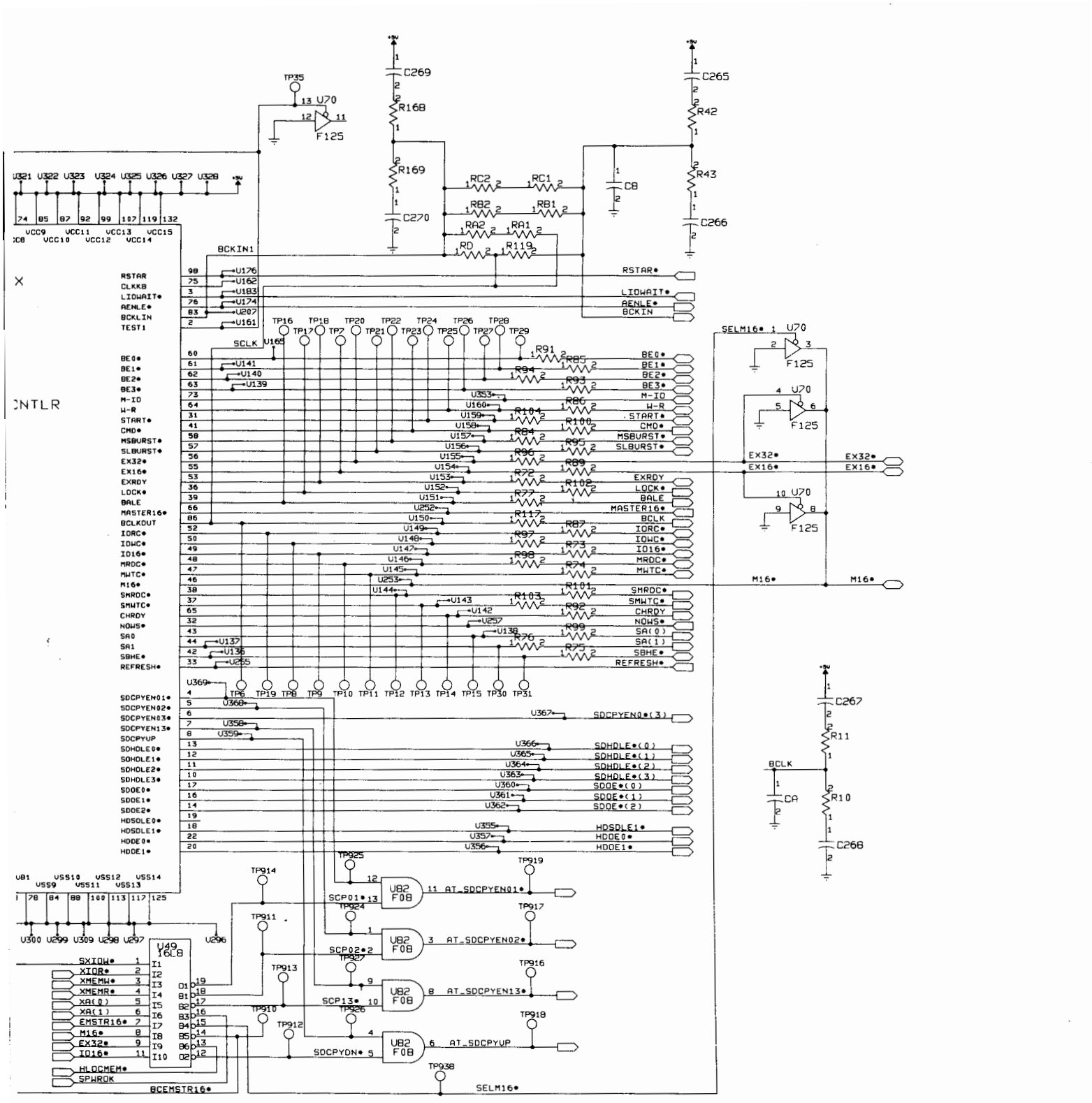

XD(0:7)
XD(0)
XD(1)
XD(2)
XD(3)
XD(4)
XD(5)
XD(6)
XD(7)
HA(2:31)
HA(2)
HA(3)
HA(4)
HA(5)
HA(6)
HA(7)
HA(8)
HA(9)
HA(10)
HA(11)
HA(12)
HA(13)
HA(14)
HA(15)
HA(16)
HA(17)
HA(18)
HA(19)
HA(20)
HA(21)
HA(22)
HA(23)
HA(24)
HA(25)
HA(26)
HA(27)
HA(28)
HA(29)
HA(30)
HA(31)
HGT16M*
GT1M*
EXMASTER*
EMSTR16*
DHOLD
HHLDA
DRDY
ISPRST
ST(0)
ST(1)
ST(2)
ST(3)
INTR
NMI
HBUSREQ*
PARITY*
SLOWH*
HW/R*

Figure B-21. System Board (EISA Bus Controller)
Schematics B-43/B-44


C269
R168
R169
C270
C265
R42
R43
C266
C8
RC2
RC1
RB2
RB1
RA2
RA1
RD
R119
TP35
U70
F125
BCKIN1
RSTAR
CLKKB
LIOWAIT
AENLE
BCKLIN
TEST1
SCLK
U165
BE0
BE1
BE2
BE3
M-IO
W-R
START
CMD
MSBURST
SLBURST
EX32
EX16
EXRDY
LOCK
BALE
MASTER16
BCLKOUT
IORC
IOWC
IO16
MRDC
MWTC
M16
SMRDC
SMWTC
CHRDY
NOWS
SA0
SA1
SBHE
REFRESH
SDCPYEN01
SDCPYEN02
SDCPYEN03
SDCPYEN13
SDCPYUP
SDHDLE0
SDHDLE1
SDHDLE2
SDHDLE3
SDOE0
SDOE1
SDOE2
HDSDLE0
HDSDLE1
HDDE0
HDDE1
SELM16
BCLK
C267
R11
CA
R10
C268
U49
16L8
SXIOW
XIOR
XMEMW
XMEMR
XA(0)
XA(1)
EMSTR16
M16
EX32
IO16
HLOCMEM
SPWROK
BCEMSTR16
U82
F08
SCP01
SCP02
SCP13
SDCPYON
AT_SDCPYEN01
AT_SDCPYEN02
AT_SDCPYEN13
AT_SDCPYUP
SDCPYEN0(3)
SDHDLE(0)
SDHDLE(1)
SDHDLE(2)
SDHDLE(3)
SDOE(0)
SDOE(1)
SDOE(2)
HDSDLE1
HDDE0
HDDE1

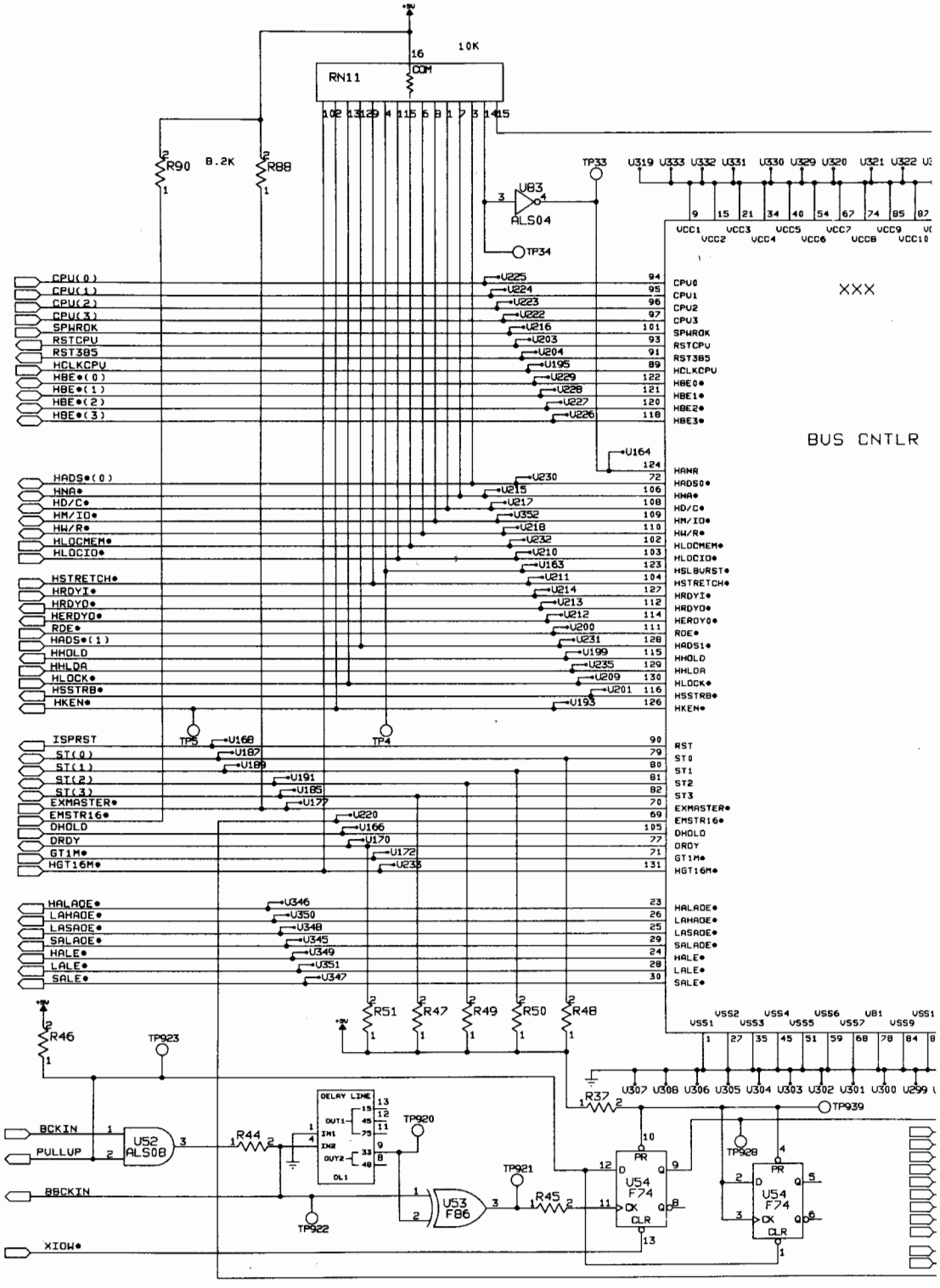
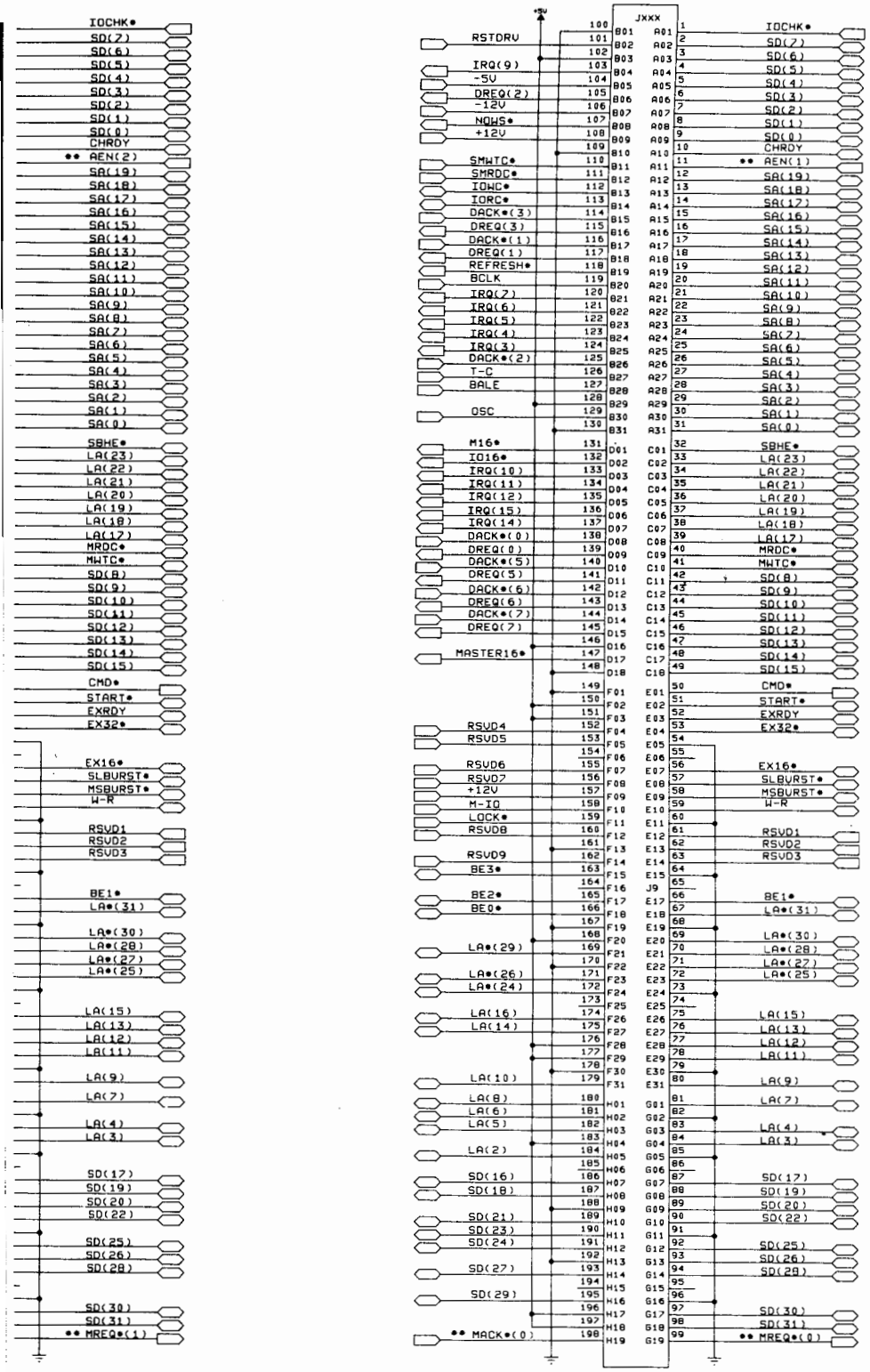
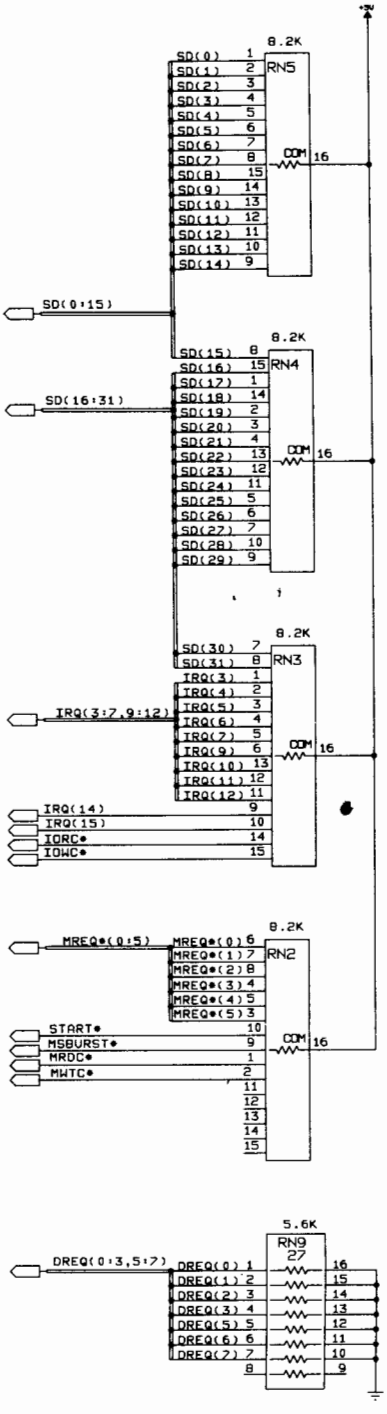

RN11
10K
R90
8.2K
R88
U83
ALS04
TP33
TP34
BUS CNTLR
XXX
CPU(0)
CPU(1)
CPU(2)
CPU(3)
SPWROK
RSTCPU
RST385
HCLKCPU
HBE*(0)
HBE*(1)
HBE*(2)
HBE*(3)
HADS*(0)
HNA*
HD/C*
HM/IO*
HW/R*
HLOCMEM*
HLOCIO*
HSTRETCH*
HRDYI*
HRDYO*
HERDYO*
RDE*
HADS*(1)
HHOLD
HHLDA
HLOCK*
HSSTRB*
HKEN*
ISPRST
ST(0)
ST(1)
ST(2)
ST(3)
EXMASTER*
EMSTR16*
DHOLD
DRDY
GT1M*
HGT16M*
HALADE*
LAHADE*
LASADE*
SALADE*
HALE*
LALE*
SALE*
R46
TP923
R51
R47
R49
R50
R48
R37
TP939
BCKIN
PULLUP
U52
ALS08
R44
DELAY LINE
TP920
TP921
TP922
BBCKIN
U53
F86
R45
U54
F74
TP928
XIOW*

Figure B-22. System Board (EISA Slots 1 & 2)
Schematics B-45/B-46


IOCHK•
SD(7)
SD(6)
SD(5)
SD(4)
SD(3)
SD(2)
SD(1)
SD(0)
CHRDY
•• AEN(2)
SA(19)
SA(18)
SA(17)
SA(16)
SA(15)
SA(14)
SA(13)
SA(12)
SA(11)
SA(10)
SA(9)
SA(8)
SA(7)
SA(6)
SA(5)
SA(4)
SA(3)
SA(2)
SA(1)
SA(0)
SBHE•
LA(23)
LA(22)
LA(21)
LA(20)
LA(19)
LA(18)
LA(17)
MRDC•
MWTC•
SD(8)
SD(9)
SD(10)
SD(11)
SD(12)
SD(13)
SD(14)
SD(15)
CMD•
START•
EXRDY
EX32•
EX16•
SLBURST•
MSBURST•
W-R
RSVD1
RSVD2
RSVD3
BE1•
LA•(31)
LA•(30)
LA•(28)
LA•(27)
LA•(25)
LA(15)
LA(13)
LA(12)
LA(11)
LA(9)
LA(7)
LA(4)
LA(3)
SD(17)
SD(19)
SD(20)
SD(22)
SD(25)
SD(26)
SD(28)
SD(30)
SD(31)
•• MREQ•(1)
+5V
RSTDRV
IRQ(9)
-5V
DREQ(2)
-12V
NOWS•
+12V
SMWTC•
SMRDC•
IOWC•
IORC•
DACK•(3)
DREQ(3)
DACK•(1)
DREQ(1)
REFRESH•
BCLK
IRQ(7)
IRQ(6)
IRQ(5)
IRQ(4)
IRQ(3)
DACK•(2)
T-C
BALE
OSC
M16•
IO16•
IRQ(10)
IRQ(11)
IRQ(12)
IRQ(15)
IRQ(14)
DACK•(0)
DREQ(0)
DACK•(5)
DREQ(5)
DACK•(6)
DREQ(6)
DACK•(7)
DREQ(7)
MASTER16•
RSVD4
RSVD5
RSVD6
RSVD7
+12V
M-IO
LOCK•
RSVD8
RSVD9
BE3•
BE2•
BE0•
LA•(29)
LA•(26)
LA•(24)
LA(16)
LA(14)
LA(10)
LA(8)
LA(6)
LA(5)
LA(2)
SD(16)
SD(18)
SD(21)
SD(23)
SD(24)
SD(27)
SD(29)
•• MACK•(0)
JXXX
IOCHK•
SD(7)
SD(6)
SD(5)
SD(4)
SD(3)
SD(2)
SD(1)
SD(0)
CHRDY
•• AEN(1)
SA(19)
SA(18)
SA(17)
SA(16)
SA(15)
SA(14)
SA(13)
SA(12)
SA(11)
SA(10)
SA(9)
SA(8)
SA(7)
SA(6)
SA(5)
SA(4)
SA(3)
SA(2)
SA(1)
SA(0)
SBHE•
LA(23)
LA(22)
LA(21)
LA(20)
LA(19)
LA(18)
LA(17)
MRDC•
MWTC•
SD(8)
SD(9)
SD(10)
SD(11)
SD(12)
SD(13)
SD(14)
SD(15)
CMD•
START•
EXRDY
EX32•
EX16•
SLBURST•
MSBURST•
W-R
RSVD1
RSVD2
RSVD3
BE1•
LA•(31)
LA•(30)
LA•(28)
LA•(27)
LA•(25)
LA(15)
LA(13)
LA(12)
LA(11)
LA(9)
LA(7)
LA(4)
LA(3)
SD(17)
SD(19)
SD(20)
SD(22)
SD(25)
SD(26)
SD(28)
SD(30)
SD(31)
•• MREQ•(0)


8.2K
RN5
SD(0)
SD(1)
SD(2)
SD(3)
SD(4)
SD(5)
SD(6)
SD(7)
SD(8)
SD(9)
SD(10)
SD(11)
SD(12)
SD(13)
SD(14)
COM
16
SD(0:15)
8.2K
RN4
SD(15)
SD(16)
SD(17)
SD(18)
SD(19)
SD(20)
SD(21)
SD(22)
SD(23)
SD(24)
SD(25)
SD(26)
SD(27)
SD(28)
SD(29)
SD(16:31)
8.2K
RN3
SD(30)
SD(31)
IRQ(3)
IRQ(4)
IRQ(5)
IRQ(6)
IRQ(7)
IRQ(9)
IRQ(10)
IRQ(11)
IRQ(12)
IRQ(3:7,9:12)
IRQ(14)
IRQ(15)
IORC*
IOWC*
8.2K
RN2
MREQ*(0:5)
MREQ*(0)
MREQ*(1)
MREQ*(2)
MREQ*(3)
MREQ*(4)
MREQ*(5)
START*
MSBURST*
MRDC*
MWTC*
5.6K
RN9
27
DREQ(0:3,5:7)
DREQ(0)
DREQ(1)
DREQ(2)
DREQ(3)
DREQ(5)
DREQ(6)
DREQ(7)

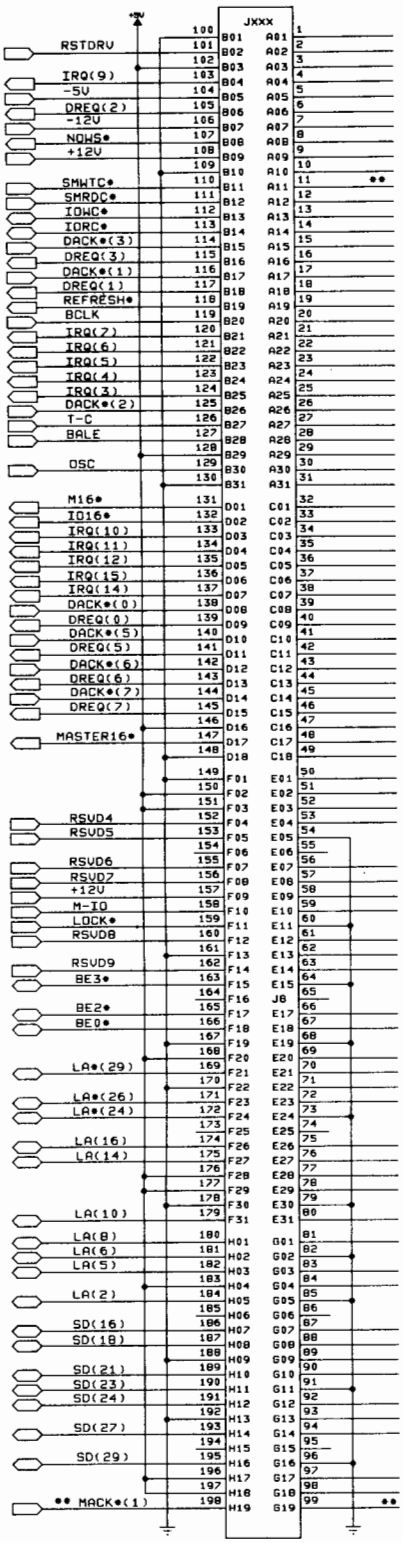
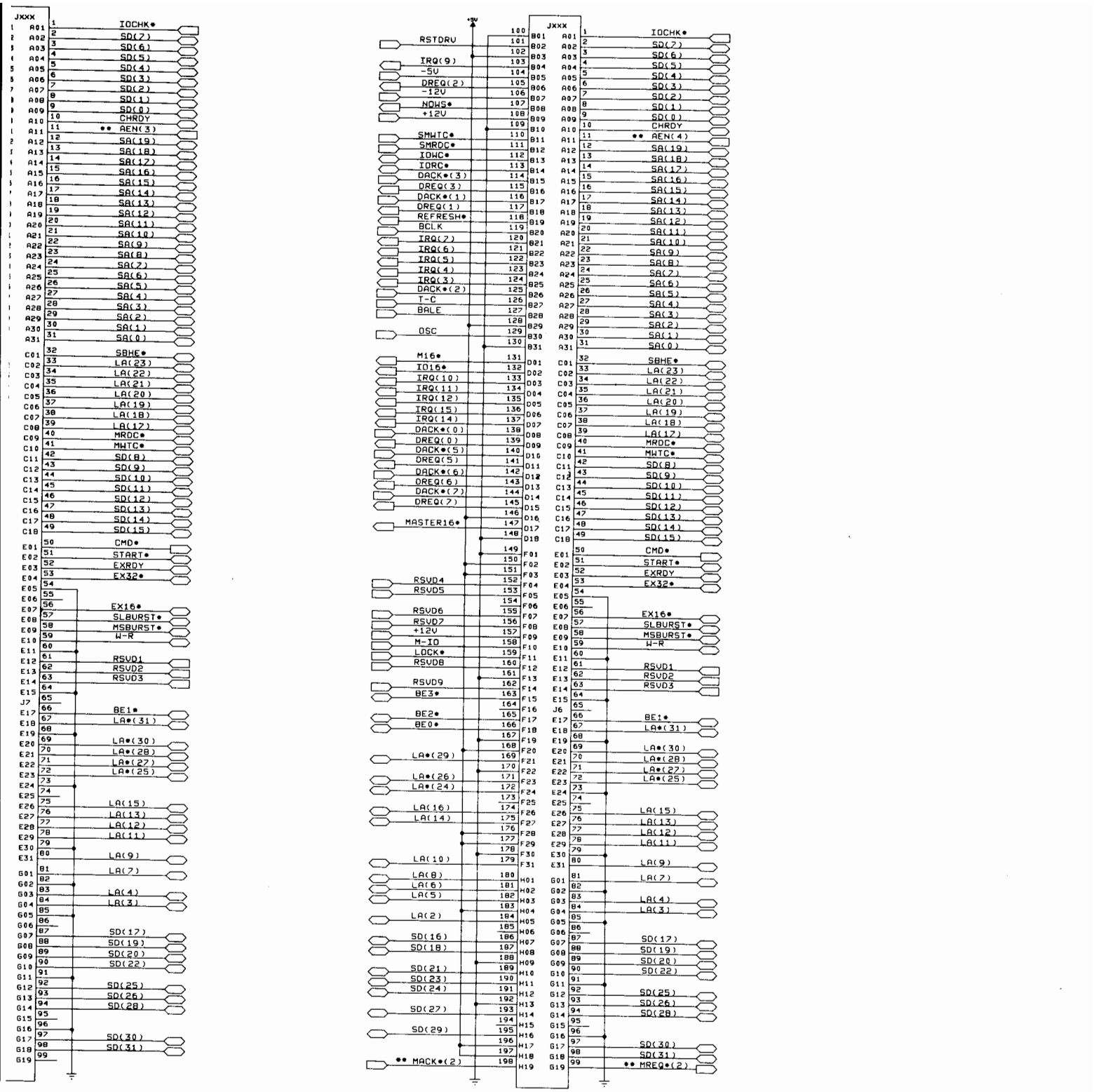
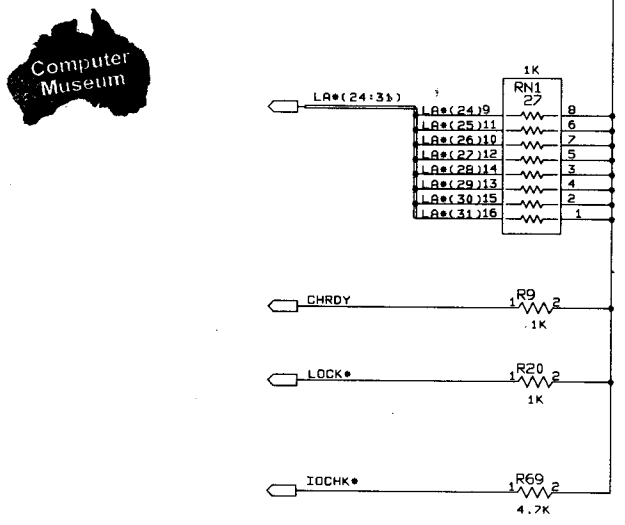

JXXX
RSTDRV
IRQ(9)
-5V
DREQ(2)
-12V
NOWS*
+12V
SMWTC*
SMRDC*
IOWC*
IORC*
DACK*(3)
DREQ(3)
DACK*(1)
DREQ(1)
REFRESH*
BCLK
IRQ(7)
IRQ(6)
IRQ(5)
IRQ(4)
IRQ(3)
DACK*(2)
T-C
BALE
OSC
M16*
IO16*
IRQ(10)
IRQ(11)
IRQ(12)
IRQ(15)
IRQ(14)
DACK*(0)
DREQ(0)
DACK*(5)
DREQ(5)
DACK*(6)
DREQ(6)
DACK*(7)
DREQ(7)
MASTER16*
RSVD4
RSVD5
RSVD6
RSVD7
+12V
M-IO
LOCK*
RSVD8
RSVD9
BE3*
BE2*
BE0*
LA*(29)
LA*(26)
LA*(24)
LA(16)
LA(14)
LA(10)
LA(8)
LA(6)
LA(5)
LA(2)
SD(16)
SD(18)
SD(21)
SD(23)
SD(24)
SD(27)
SD(29)
** MACK*(1)

Figure B-23. System Board (EISA Slots 3 & 4)
Schematics B-47/B-48


JXXX
IOCHK•
SD(7)
SD(6)
SD(5)
SD(4)
SD(3)
SD(2)
SD(1)
SD(0)
CHRDY
•• AEN(3)
SA(19)
SA(18)
SA(17)
SA(16)
SA(15)
SA(14)
SA(13)
SA(12)
SA(11)
SA(10)
SA(9)
SA(8)
SA(7)
SA(6)
SA(5)
SA(4)
SA(3)
SA(2)
SA(1)
SA(0)
SBHE•
LA(23)
LA(22)
LA(21)
LA(20)
LA(19)
LA(18)
LA(17)
MRDC•
MWTC•
SD(8)
SD(9)
SD(10)
SD(11)
SD(12)
SD(13)
SD(14)
SD(15)
CMD•
START•
EXRDY
EX32•
EX16•
SLBURST•
MSBURST•
W-R
RSVD1
RSVD2
RSVD3
BE1•
LA•(31)
LA•(30)
LA•(28)
LA•(27)
LA•(25)
LA(15)
LA(13)
LA(12)
LA(11)
LA(9)
LA(7)
LA(4)
LA(3)
SD(17)
SD(19)
SD(20)
SD(22)
SD(25)
SD(26)
SD(28)
SD(30)
SD(31)
+5V
RSTDRV
IRQ(9)
-5V
DREQ(2)
-12V
NOWS•
+12V
SMWTC•
SMRDC•
IOWC•
IORC•
DACK•(3)
DREQ(3)
DACK•(1)
DREQ(1)
REFRESH•
BCLK
IRQ(7)
IRQ(6)
IRQ(5)
IRQ(4)
IRQ(3)
DACK•(2)
T-C
BALE
OSC
M16•
IO16•
IRQ(10)
IRQ(11)
IRQ(12)
IRQ(15)
IRQ(14)
DACK•(0)
DREQ(0)
DACK•(5)
DREQ(5)
DACK•(6)
DREQ(6)
DACK•(7)
DREQ(7)
MASTER16•
RSVD4
RSVD5
RSVD6
RSVD7
+12V
M-IO
LOCK•
RSVD8
RSVD9
BE3•
BE2•
BE0•
LA•(29)
LA•(26)
LA•(24)
LA(16)
LA(14)
LA(10)
LA(8)
LA(6)
LA(5)
LA(2)
SD(16)
SD(18)
SD(21)
SD(23)
SD(24)
SD(27)
SD(29)
•• MACK•(2)
•• MREQ•(2)



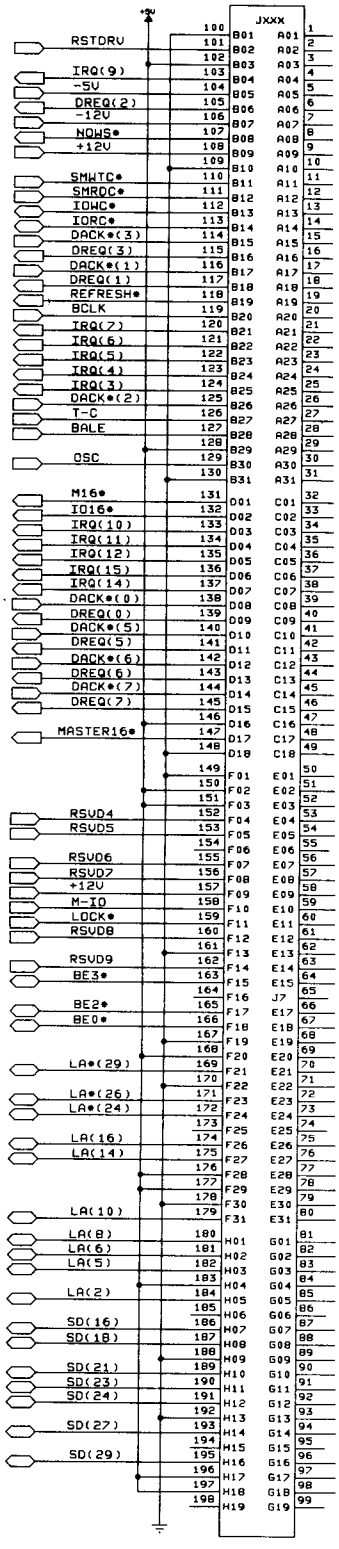
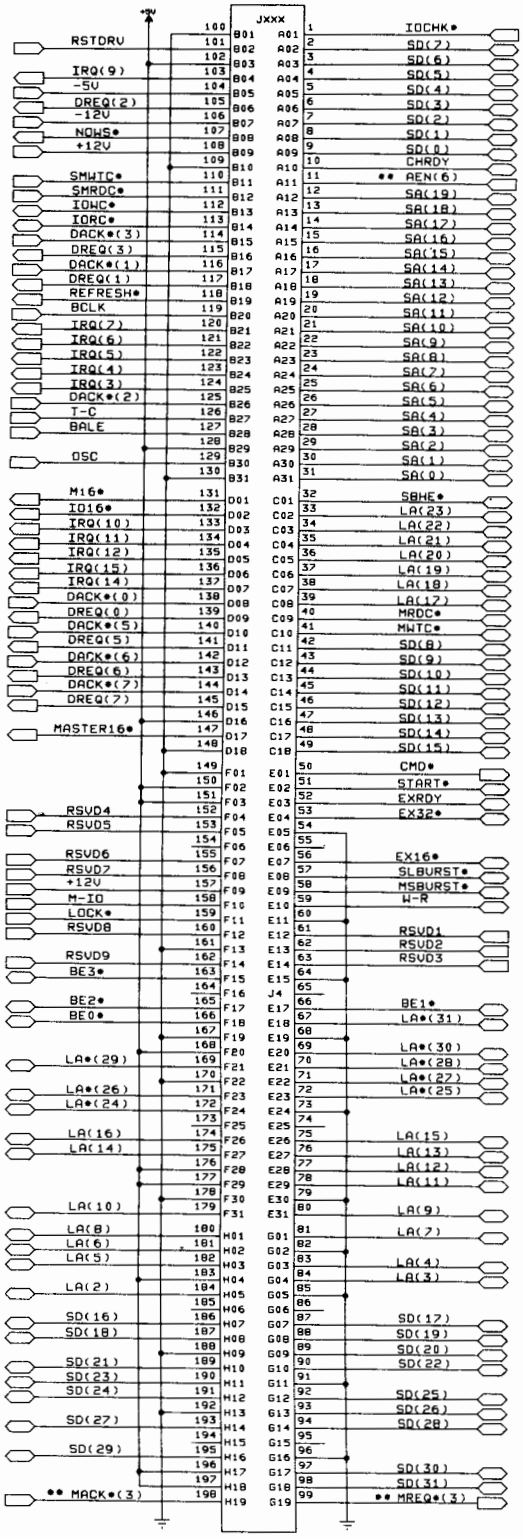


Figure B-24. System Board (EISA Slots 5 & 6)
Schematics B-49/B-50


JXXX
RSTDRV
IRQ(9)
-5V
DREQ(2)
-12V
NOWS*
+12V
SMWTC*
SMRDC*
IOWC*
IORC*
DACK*(3)
DREQ(3)
DACK*(1)
DREQ(1)
REFRESH*
BCLK
IRQ(7)
IRQ(6)
IRQ(5)
IRQ(4)
IRQ(3)
DACK*(2)
T-C
BALE
OSC
M16*
IO16*
IRQ(10)
IRQ(11)
IRQ(12)
IRQ(15)
IRQ(14)
DACK*(0)
DREQ(0)
DACK*(5)
DREQ(5)
DACK*(6)
DREQ(6)
DACK*(7)
DREQ(7)
MASTER16*
RSVD4
RSVD5
RSVD6
RSVD7
+12V
M-IO
LOCK*
RSVD8
RSVD9
BE3*
BE2*
BE0*
LA*(29)
LA*(26)
LA*(24)
LA(16)
LA(14)
LA(10)
LA(8)
LA(6)
LA(5)
LA(2)
SD(16)
SD(18)
SD(21)
SD(23)
SD(24)
SD(27)
SD(29)
** MACK*(3)
IOCHK*
SD(7)
SD(6)
SD(5)
SD(4)
SD(3)
SD(2)
SD(1)
SD(0)
CHRDY
** AEN(6)
SA(19)
SA(18)
SA(17)
SA(16)
SA(15)
SA(14)
SA(13)
SA(12)
SA(11)
SA(10)
SA(9)
SA(8)
SA(7)
SA(6)
SA(5)
SA(4)
SA(3)
SA(2)
SA(1)
SA(0)
SBHE*
LA(23)
LA(22)
LA(21)
LA(20)
LA(19)
LA(18)
LA(17)
MRDC*
MWTC*
SD(8)
SD(9)
SD(10)
SD(11)
SD(12)
SD(13)
SD(14)
SD(15)
CMD*
START*
EXRDY
EX32*
EX16*
SLBURST*
MSBURST*
W-R
RSVD1
RSVD2
RSVD3
BE1*
LA*(31)
LA*(30)
LA*(28)
LA*(27)
LA*(25)
LA(15)
LA(13)
LA(12)
LA(11)
LA(9)
LA(7)
LA(4)
LA(3)
SD(17)
SD(19)
SD(20)
SD(22)
SD(25)
SD(26)
SD(28)
SD(30)
SD(31)
** MREQ*(3)

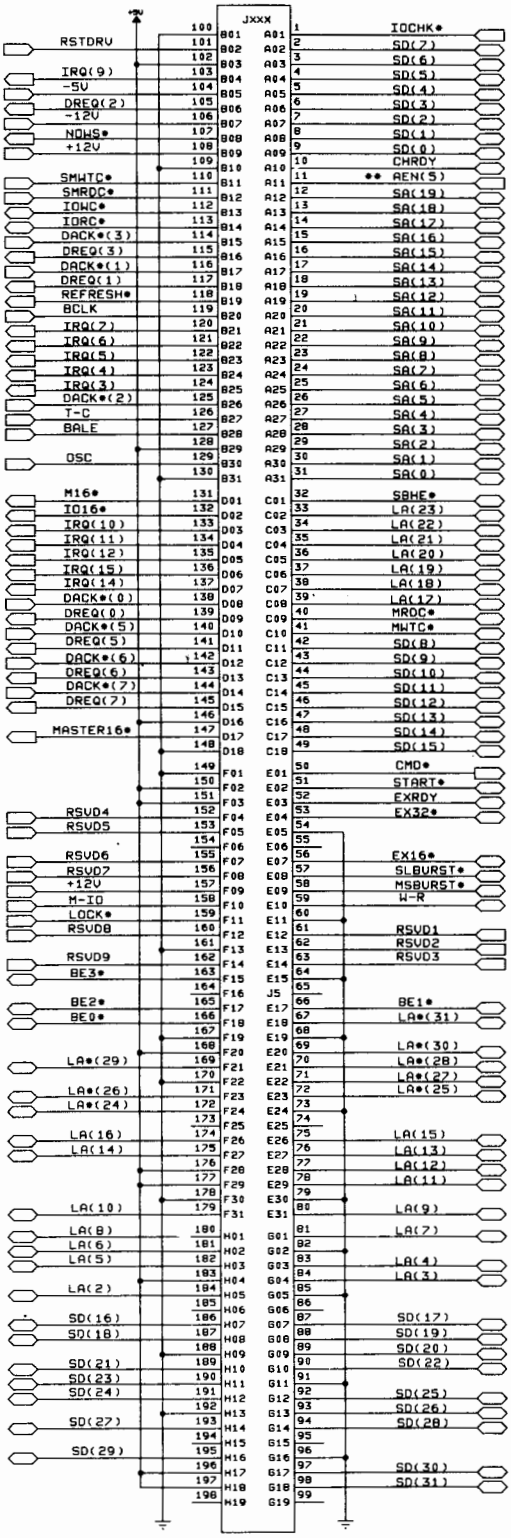
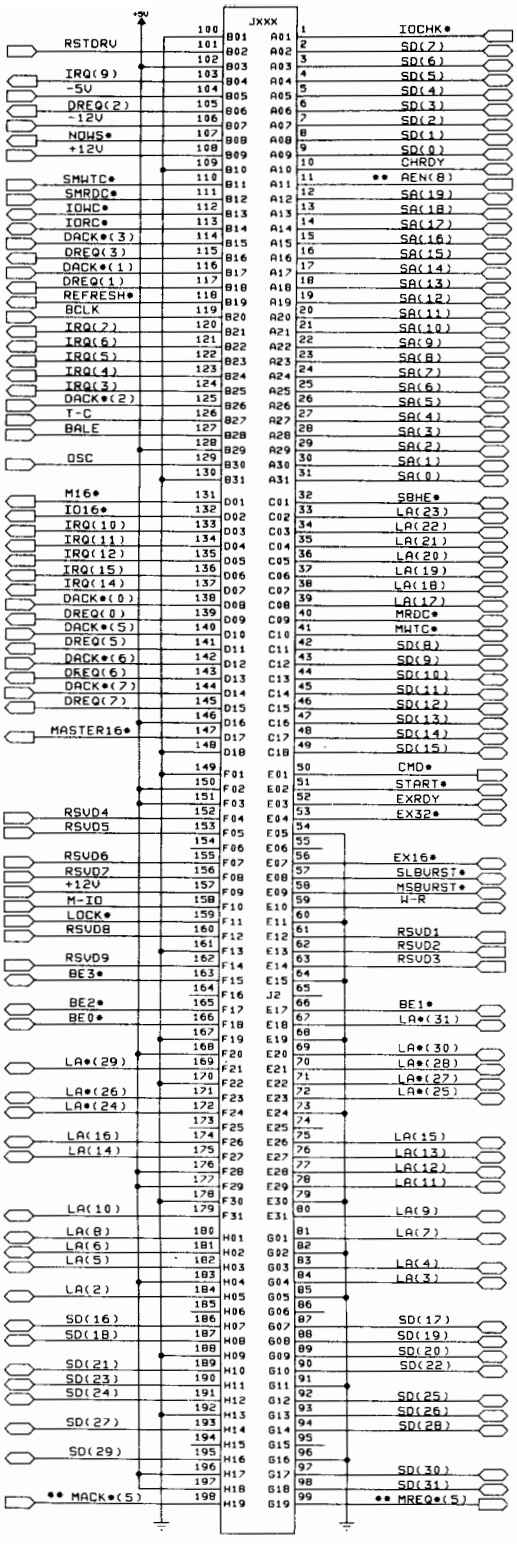

JXXX
RSTDRV
IRQ(9)
-5V
DREQ(2)
-12V
NOWS*
+12V
SMWTC*
SMRDC*
IOWC*
IORC*
DACK*(3)
DREQ(3)
DACK*(1)
DREQ(1)
REFRESH*
BCLK
IRQ(7)
IRQ(6)
IRQ(5)
IRQ(4)
IRQ(3)
DACK*(2)
T-C
BALE
OSC
M16*
IO16*
IRQ(10)
IRQ(11)
IRQ(12)
IRQ(15)
IRQ(14)
DACK*(0)
DREQ(0)
DACK*(5)
DREQ(5)
DACK*(6)
DREQ(6)
DACK*(7)
DREQ(7)
MASTER16*
RSVD4
RSVD5
RSVD6
RSVD7
+12V
M-IO
LOCK*
RSVD8
RSVD9
BE3*
BE2*
BE0*
LA*(29)
LA*(26)
LA*(24)
LA(16)
LA(14)
LA(10)
LA(8)
LA(6)
LA(5)
LA(2)
SD(16)
SD(18)
SD(21)
SD(23)
SD(24)
SD(27)
SD(29)
IOCHK*
SD(7)
SD(6)
SD(5)
SD(4)
SD(3)
SD(2)
SD(1)
SD(0)
CHRDY
** AEN(5)
SA(19)
SA(18)
SA(17)
SA(16)
SA(15)
SA(14)
SA(13)
SA(12)
SA(11)
SA(10)
SA(9)
SA(8)
SA(7)
SA(6)
SA(5)
SA(4)
SA(3)
SA(2)
SA(1)
SA(0)
SBHE*
LA(23)
LA(22)
LA(21)
LA(20)
LA(19)
LA(18)
LA(17)
MRDC*
MWTC*
SD(8)
SD(9)
SD(10)
SD(11)
SD(12)
SD(13)
SD(14)
SD(15)
CMD*
START*
EXRDY
EX32*
EX16*
SLBURST*
MSBURST*
W-R
RSVD1
RSVD2
RSVD3
BE1*
LA*(31)
LA*(30)
LA*(28)
LA*(27)
LA*(25)
LA(15)
LA(13)
LA(12)
LA(11)
LA(9)
LA(7)
LA(4)
LA(3)
SD(17)
SD(19)
SD(20)
SD(22)
SD(25)
SD(26)
SD(28)
SD(30)
SD(31)

Figure B-25. System Board (EISA Slots 7 & 8)
Schematics B-51/B-52


+5V
JXXX
RSTDRV
IRQ(9)
-5V
DREQ(2)
-12V
NOWS*
+12V
SMWTC*
SMRDC*
IOWC*
IORC*
DACK*(3)
DREQ(3)
DACK*(1)
DREQ(1)
REFRESH*
BCLK
IRQ(7)
IRQ(6)
IRQ(5)
IRQ(4)
IRQ(3)
DACK*(2)
T-C
BALE
OSC
M16*
IO16*
IRQ(10)
IRQ(11)
IRQ(12)
IRQ(15)
IRQ(14)
DACK*(0)
DREQ(0)
DACK*(5)
DREQ(5)
DACK*(6)
DREQ(6)
DACK*(7)
DREQ(7)
MASTER16*
RSVD4
RSVD5
RSVD6
RSVD7
+12V
M-IO
LOCK*
RSVD8
RSVD9
BE3*
BE2*
BE0*
LA*(29)
LA*(26)
LA*(24)
LA(16)
LA(14)
LA(10)
LA(8)
LA(6)
LA(5)
LA(2)
SD(16)
SD(18)
SD(21)
SD(23)
SD(24)
SD(27)
SD(29)
** MACK*(5)
IOCHK*
SD(7)
SD(6)
SD(5)
SD(4)
SD(3)
SD(2)
SD(1)
SD(0)
CHRDY
** AEN(8)
SA(19)
SA(18)
SA(17)
SA(16)
SA(15)
SA(14)
SA(13)
SA(12)
SA(11)
SA(10)
SA(9)
SA(8)
SA(7)
SA(6)
SA(5)
SA(4)
SA(3)
SA(2)
SA(1)
SA(0)
SBHE*
LA(23)
LA(22)
LA(21)
LA(20)
LA(19)
LA(18)
LA(17)
MRDC*
MWTC*
SD(8)
SD(9)
SD(10)
SD(11)
SD(12)
SD(13)
SD(14)
SD(15)
CMD*
START*
EXRDY
EX32*
EX16*
SLBURST*
MSBURST*
W-R
RSVD1
RSVD2
RSVD3
BE1*
LA*(31)
LA*(30)
LA*(28)
LA*(27)
LA*(25)
LA(15)
LA(13)
LA(12)
LA(11)
LA(9)
LA(7)
LA(4)
LA(3)
SD(17)
SD(19)
SD(20)
SD(22)
SD(25)
SD(26)
SD(28)
SD(30)
SD(31)
** MREQ*(5)

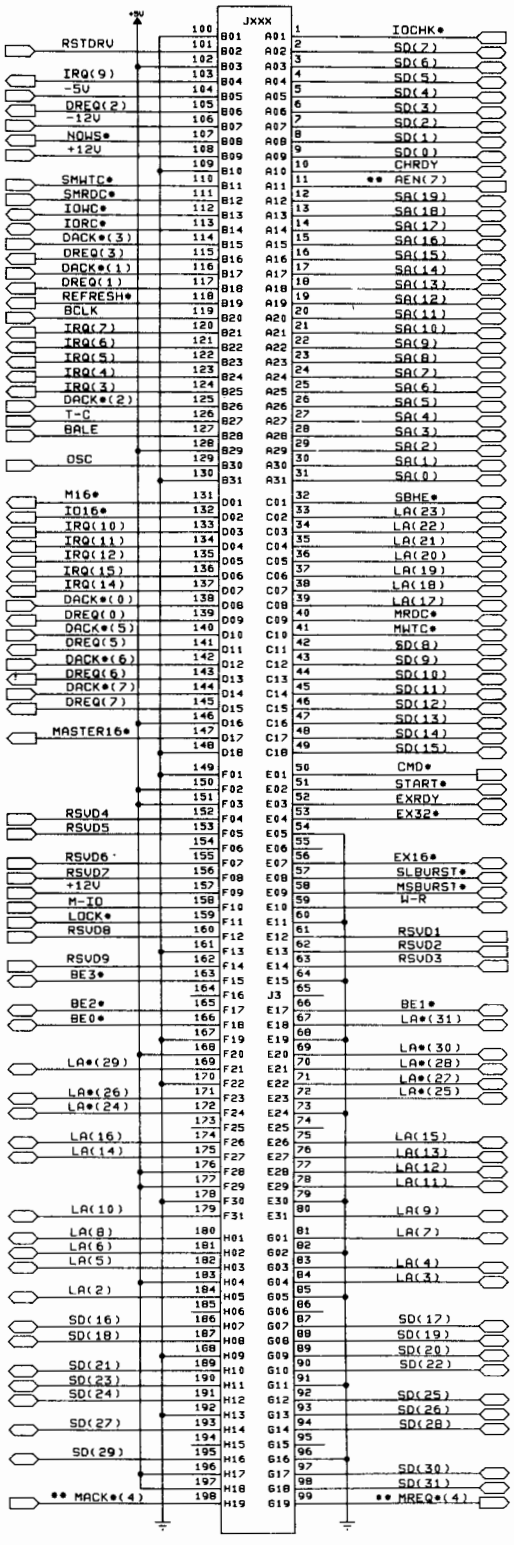
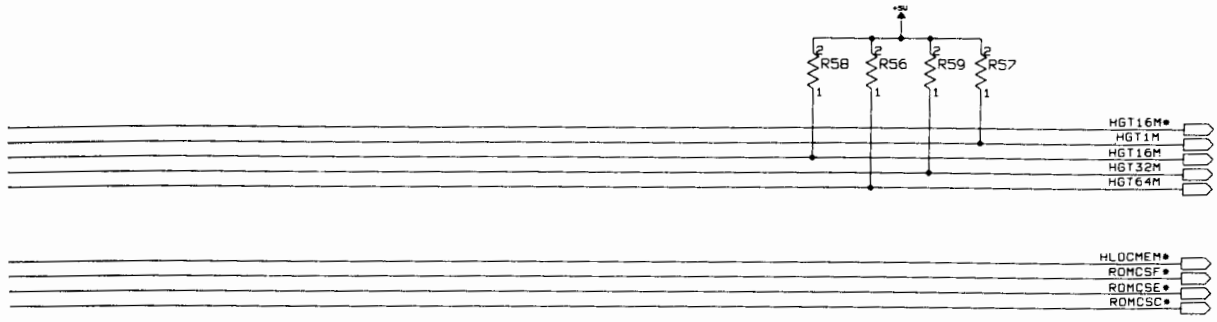
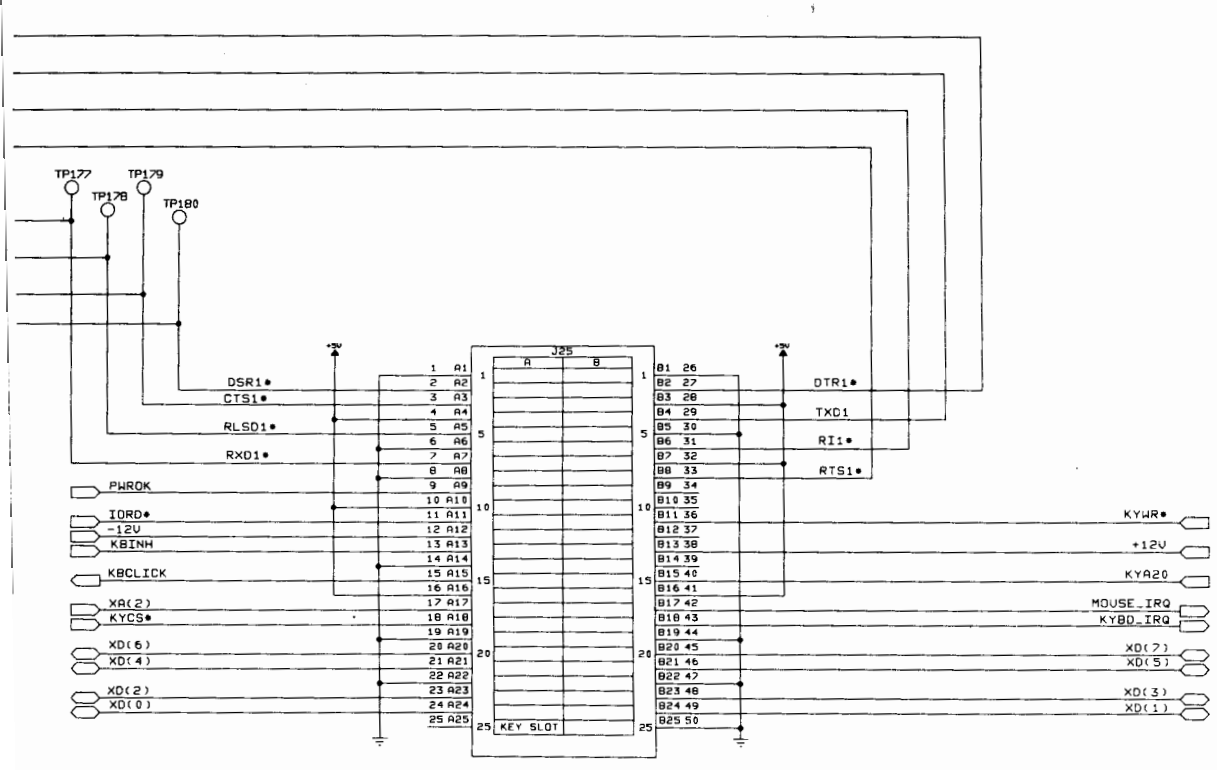

JXXX
+5V
RSTDRV
IRQ(9)
-5V
DREQ(2)
-12V
NOWS*
+12V
SMWTC*
SMRDC*
IOWC*
IORC*
DACK*(3)
DREQ(3)
DACK*(1)
DREQ(1)
REFRESH*
BCLK
IRQ(7)
IRQ(6)
IRQ(5)
IRQ(4)
IRQ(3)
DACK*(2)
T-C
BALE
OSC
M16*
IO16*
IRQ(10)
IRQ(11)
IRQ(12)
IRQ(15)
IRQ(14)
DACK*(0)
DREQ(0)
DACK*(5)
DREQ(5)
DACK*(6)
DREQ(6)
DACK*(7)
DREQ(7)
MASTER16*
RSVD4
RSVD5
RSVD6
RSVD7
+12V
M-IO
LOCK*
RSVD8
RSVD9
BE3*
BE2*
BE0*
LA*(29)
LA*(26)
LA*(24)
LA(16)
LA(14)
LA(10)
LA(8)
LA(6)
LA(5)
LA(2)
SD(16)
SD(18)
SD(21)
SD(23)
SD(24)
SD(27)
SD(29)
** MACK*(4)
IOCHK*
SD(7)
SD(6)
SD(5)
SD(4)
SD(3)
SD(2)
SD(1)
SD(0)
CHRDY
** AEN(7)
SA(19)
SA(18)
SA(17)
SA(16)
SA(15)
SA(14)
SA(13)
SA(12)
SA(11)
SA(10)
SA(9)
SA(8)
SA(7)
SA(6)
SA(5)
SA(4)
SA(3)
SA(2)
SA(1)
SA(0)
SBHE*
LA(23)
LA(22)
LA(21)
LA(20)
LA(19)
LA(18)
LA(17)
MRDC*
MWTC*
SD(8)
SD(9)
SD(10)
SD(11)
SD(12)
SD(13)
SD(14)
SD(15)
CMD*
START*
EXRDY
EX32*
EX16*
SLBURST*
MSBURST*
W-R
RSVD1
RSVD2
RSVD3
BE1*
LA*(31)
LA*(30)
LA*(28)
LA*(27)
LA*(25)
LA(15)
LA(13)
LA(12)
LA(11)
LA(9)
LA(7)
LA(4)
LA(3)
SD(17)
SD(19)
SD(20)
SD(22)
SD(25)
SD(26)
SD(28)
SD(30)
SD(31)
** MREQ*(4)

Figure B-26. System Board (Interface Card Connectors)
Schematics B-53/B-54


+5V
R58
R56
R59
R57
HGT16M*
HGT1M
HGT16M
HGT32M
HGT64M
HLOCMEM*
ROMCSF*
ROMCSE*
ROMCSC*


TP177
TP179
TP178
TP180
+5V
J25
A
B
KEY SLOT
DSR1*
CTS1*
RLSD1*
RXD1*
PWROK
IORD*
-12V
KBINH
KBCLICK
XA(2)
KYCS*
XD(6)
XD(4)
XD(2)
XD(0)
DTR1*
TXD1
RI1*
RTS1*
KYWR*
+12V
KYA20
MOUSE_IRQ
KYBD_IRQ
XD(7)
XD(5)
XD(3)
XD(1)

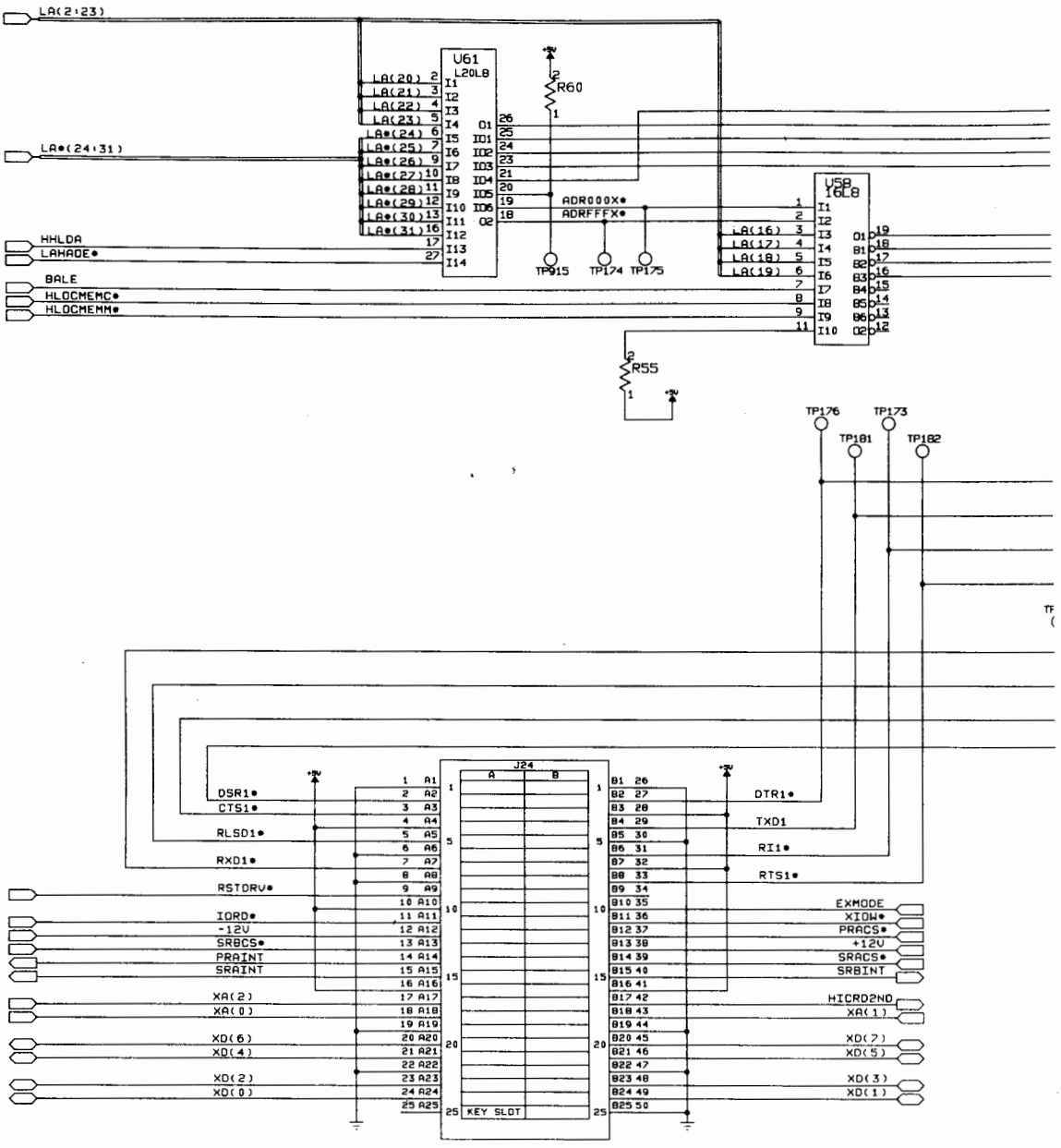
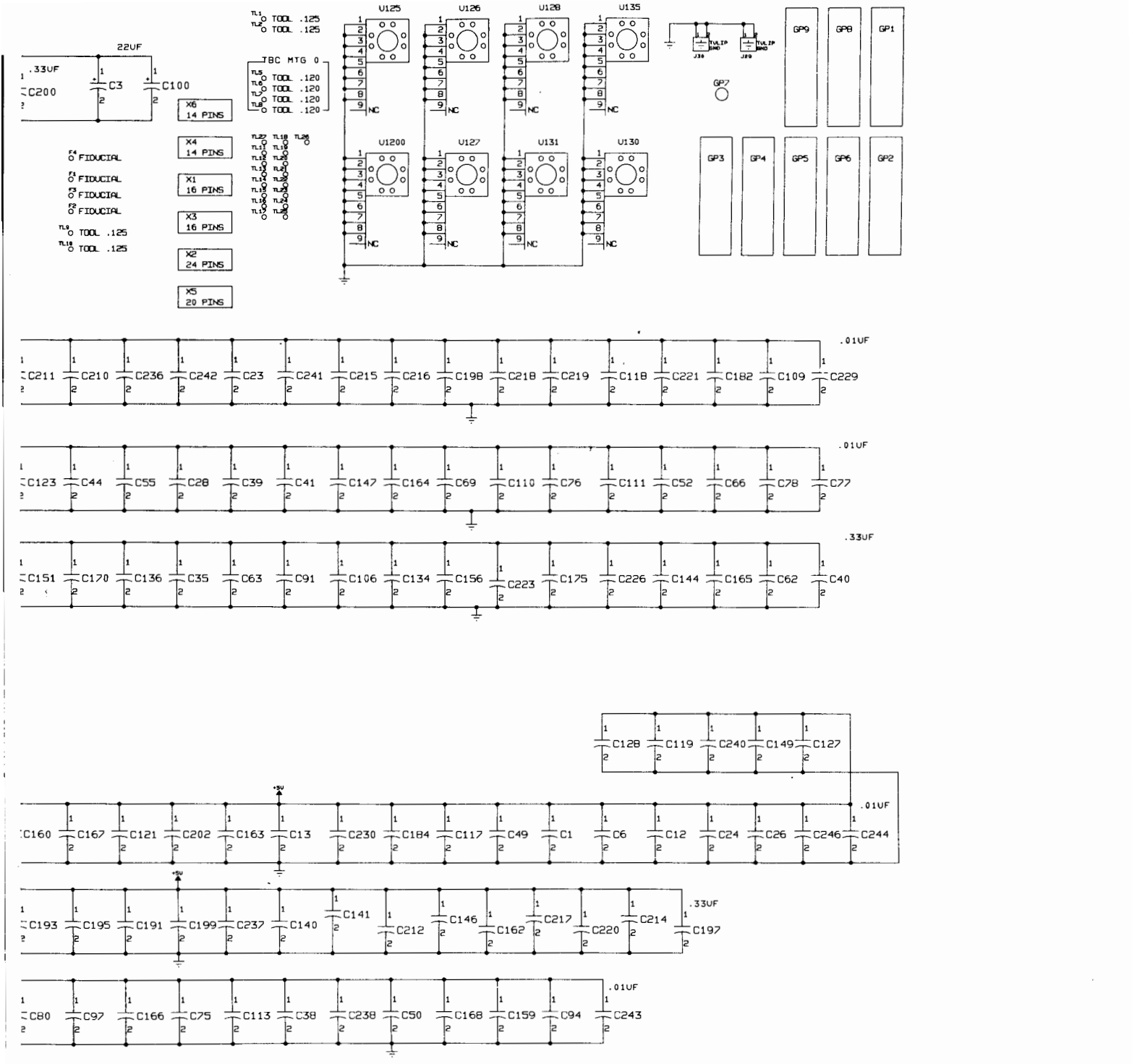

LA(2:23)
LA*(24:31)
HHLDA
LAHADE*
BALE
HLOCMEMC*
HLOCMEMM*
U61
L20L8
R60
R55
ADR000X*
ADRFFFX*
TP915
TP174
TP175
U58
16L8
LA(16)
LA(17)
LA(18)
LA(19)
TP176
TP173
TP181
TP182
J24
KEY SLOT
DSR1*
CTS1*
RLSD1*
RXD1*
RSTDRV*
IORD*
-12V
SRBCS*
PRAINT
SRAINT
XA(2)
XA(0)
XD(6)
XD(4)
XD(2)
XD(0)
DTR1*
TXD1
RI1*
RTS1*
EXMODE
XIOW*
PRACS*
+12V
SRACS*
SRBINT
HICRD2ND
XA(1)
XD(7)
XD(5)
XD(3)
XD(1)

Figure B-27. System Board (Bypass Capacitors)
Schematics B-55/B-56


22UF
.33UF
C200
C3
C100
X6
14 PINS
X4
14 PINS
X1
16 PINS
X3
16 PINS
X2
24 PINS
X5
20 PINS
FIDUCIAL
FIDUCIAL
FIDUCIAL
FIDUCIAL
TOOL .125
TOOL .125
TOOL .125
TOOL .125
TOOL .120
TOOL .120
TOOL .120
TOOL .120
JBC MTG 0
U125
U126
U128
U135
U1200
U127
U131
U130
NC
GP9
GP8
GP1
GP7
GP3
GP4
GP5
GP6
GP2
.01UF
C211
C210
C236
C242
C23
C241
C215
C216
C198
C218
C219
C118
C221
C182
C109
C229
.01UF
C123
C44
C55
C28
C39
C41
C147
C164
C69
C110
C76
C111
C52
C66
C78
C77
.33UF
C151
C170
C136
C35
C63
C91
C106
C134
C156
C223
C175
C226
C144
C165
C62
C40
C128
C119
C240
C149
C127
.01UF
C160
C167
C121
C202
C163
C13
C230
C184
C117
C49
C1
C6
C12
C24
C26
C246
C244
+5V
.33UF
C193
C195
C191
C199
C237
C140
C141
C212
C146
C162
C217
C220
C214
C197
.01UF
C80
C97
C166
C75
C113
C38
C238
C50
C168
C159
C94
C243

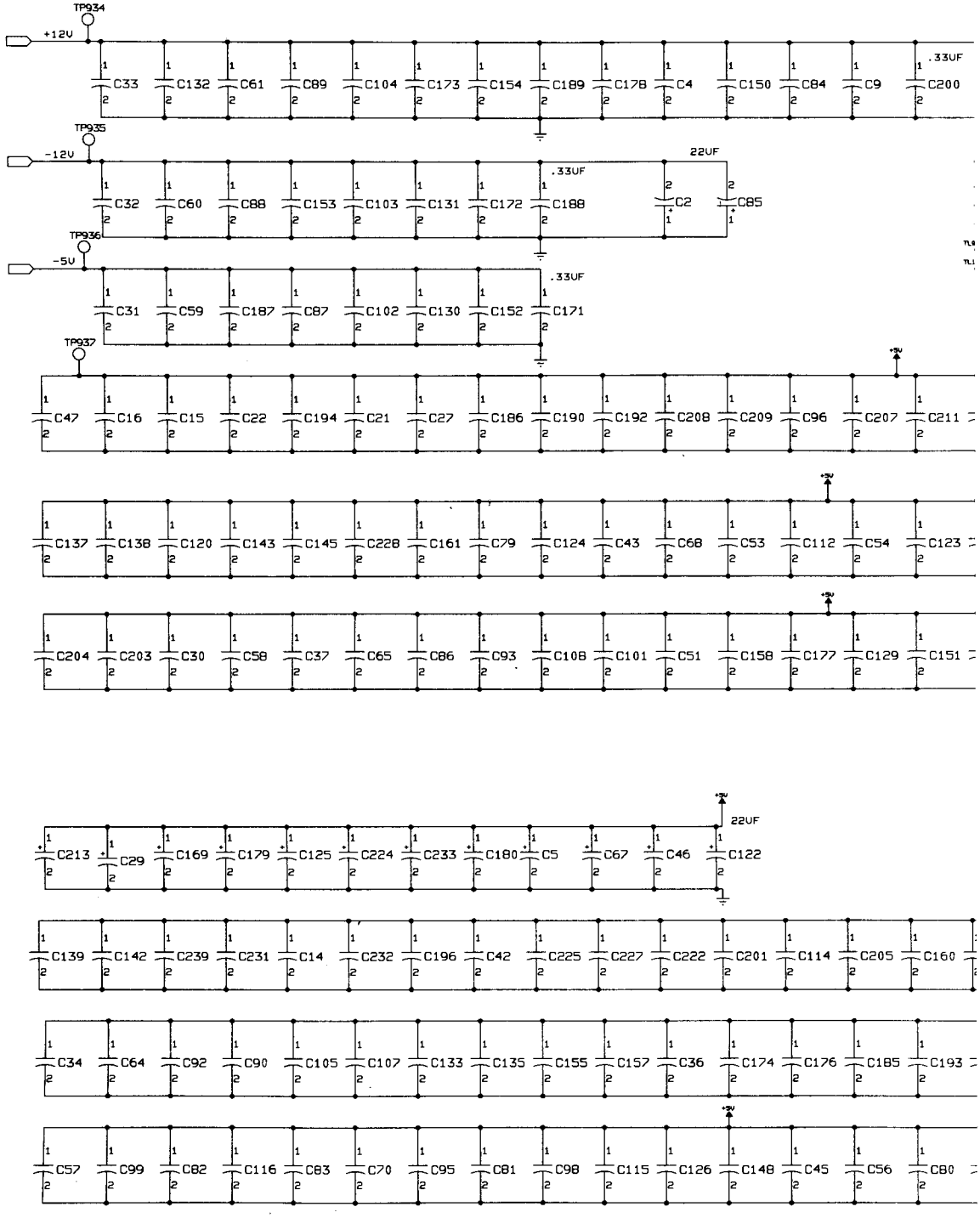

TP934
+12V
C33 C132 C61 C89 C104 C173 C154 C189 C178 C4 C150 C84 C9 C200
.33UF
TP935
-12V
C32 C60 C88 C153 C103 C131 C172 C188 C2 C85
.33UF
22UF
TP936
-5V
C31 C59 C187 C87 C102 C130 C152 C171
.33UF
TP937
+5V
C47 C16 C15 C22 C194 C21 C27 C186 C190 C192 C208 C209 C96 C207 C211
C137 C138 C120 C143 C145 C228 C161 C79 C124 C43 C68 C53 C112 C54 C123
C204 C203 C30 C58 C37 C65 C86 C93 C108 C101 C51 C158 C177 C129 C151
C213 C29 C169 C179 C125 C224 C233 C180 C5 C67 C46 C122
22UF
C139 C142 C239 C231 C14 C232 C196 C42 C225 C227 C222 C201 C114 C205 C160
C34 C64 C92 C90 C105 C107 C133 C135 C155 C157 C36 C174 C176 C185 C193
C57 C99 C82 C116 C83 C70 C95 C81 C98 C115 C126 C148 C45 C56 C80

Figure B-28. System Board (Memory Board Connector)
Schematics B-57/B-58

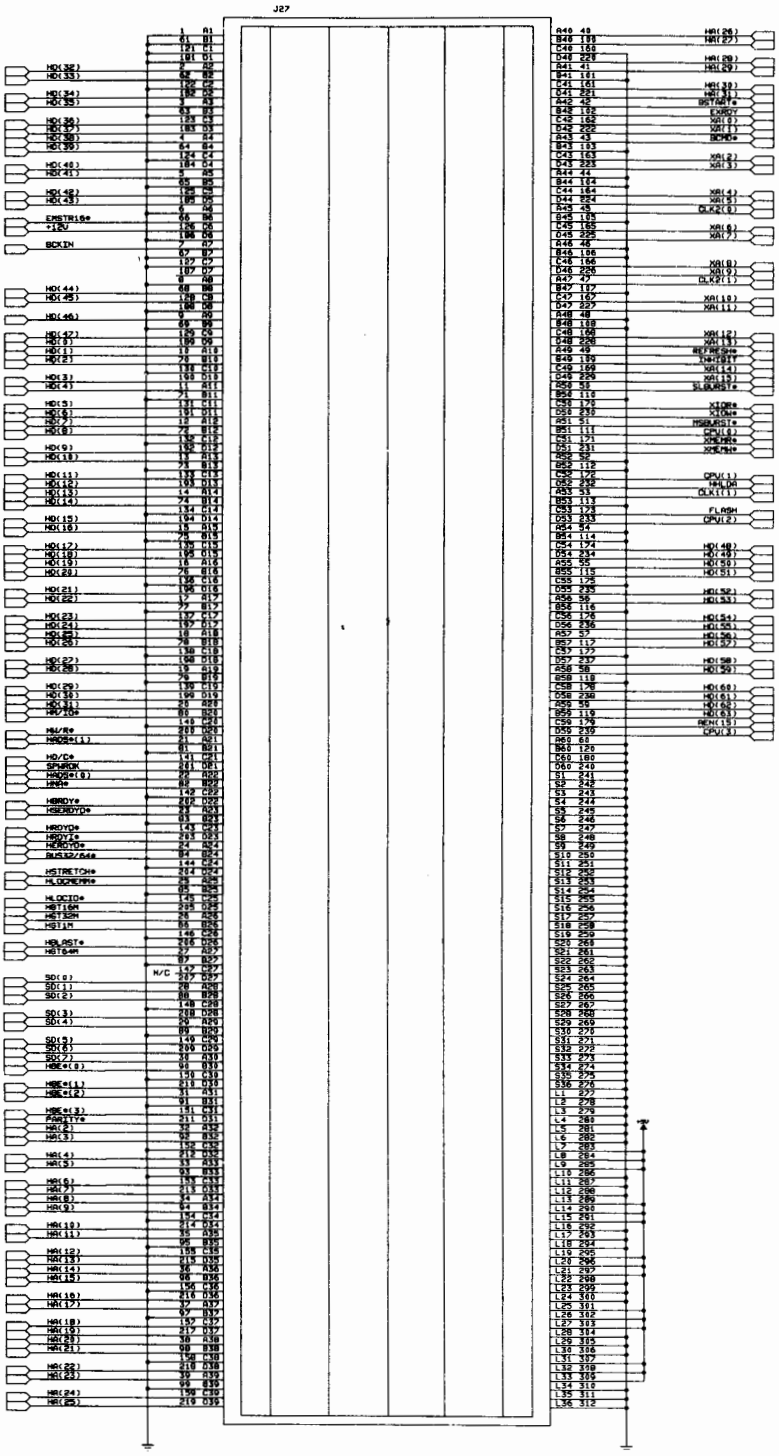

J27

Figure B-29. System Board (Processor Board Connector)
Schematics B-59/B-60

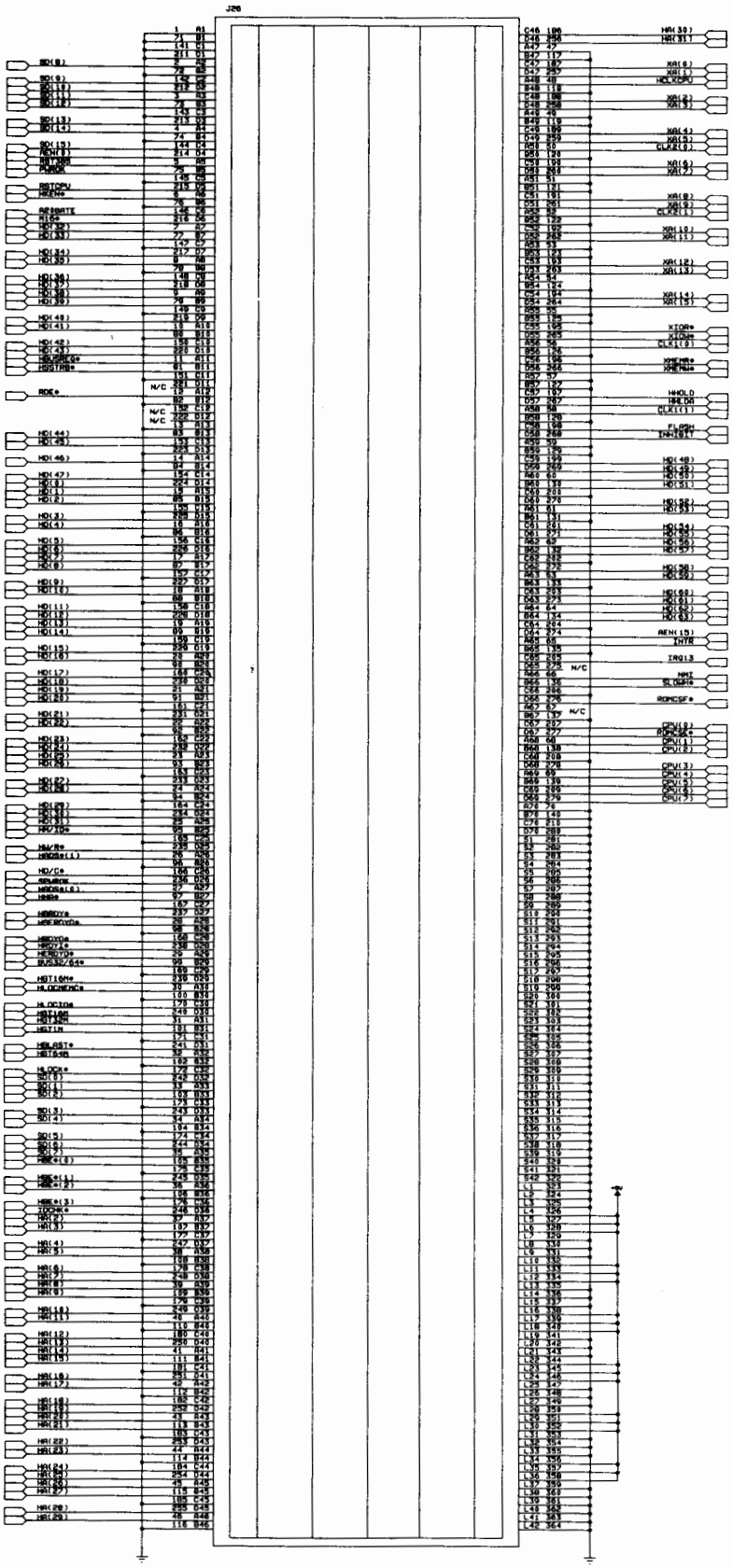
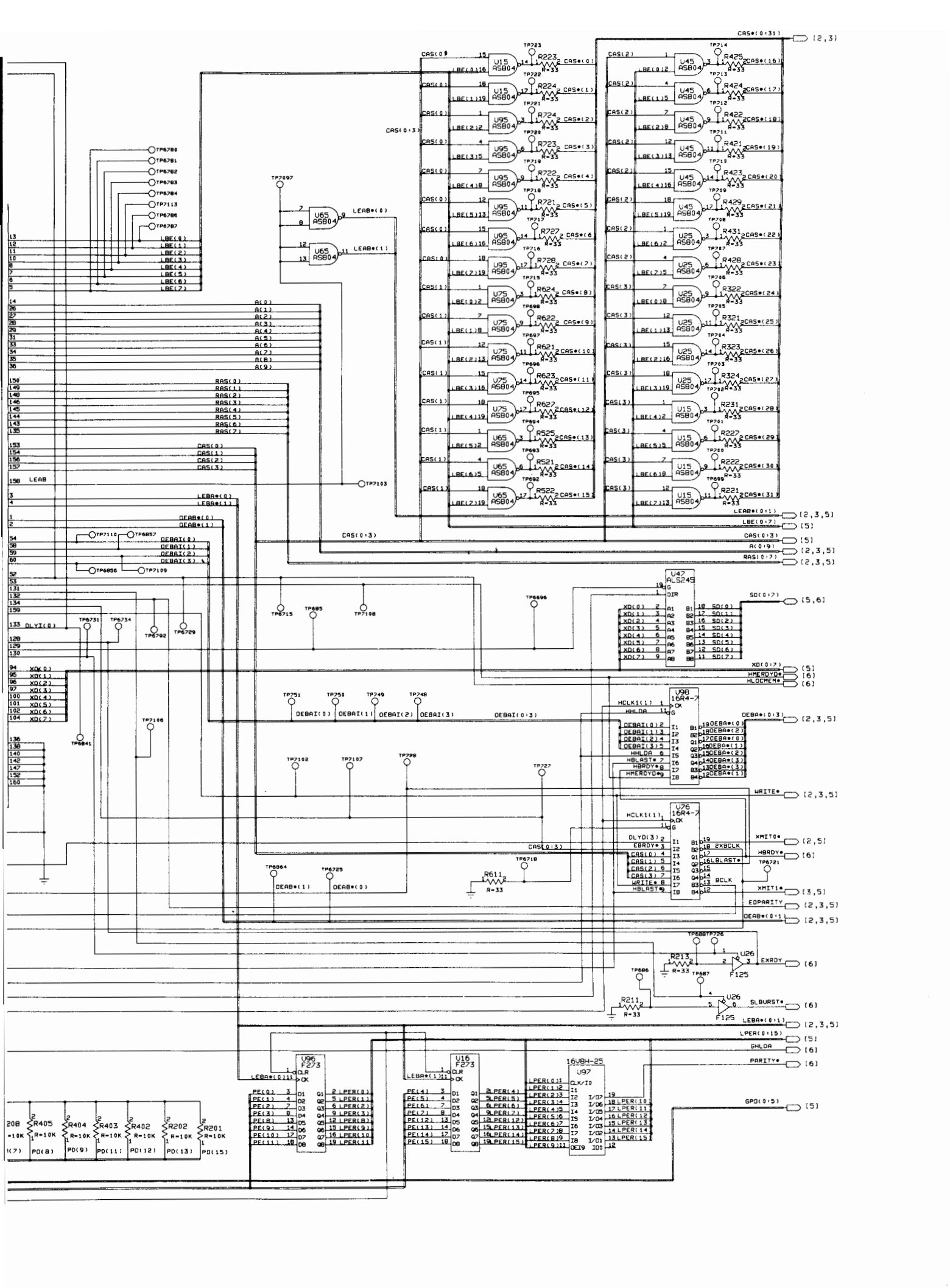


Figure B-30. Memory Board (Memory Controller)
Schematics B-61/B-62


CAS*(0:31) [2,3]
CAS(0:3)
LBE(0:7)
LEAB*(0:1) [2,3,5]
LBE(0:7) [5]
CAS(0:3) [5]
A(0:9) [2,3,5]
RAS(0:7) [2,3,5]
SD(0:7) [5,6]
XD(0:7) [5]
HMERDY* [6]
HLOCMEM* [6]
OEBA*(0:3) [2,3,5]
WRITE* [2,3,5]
XMIT0* [2,5]
HBRDY* [6]
XMIT1* [3,5]
EOPARITY [2,3,5]
OEAB*(0:1) [2,3,5]
EXRDY [6]
SLBURST* [6]
LEBA*(0:1) [2,3,5]
LPER(0:15) [5]
GHLDA [6]
PARITY* [6]
GPO(0:5) [5]
U42 ALS245
U98 16R4-7
U76 16R4-7
U96 F273
U16 F273
U97 16V8H-25
U26 F125
U15 AS804
U25 AS804
U45 AS804
U95 AS804
U75 AS804
U65 AS804
U65 AS804
R405 R404 R403 R402 R202 R201 R=10K
R611 R213 R211 R=33

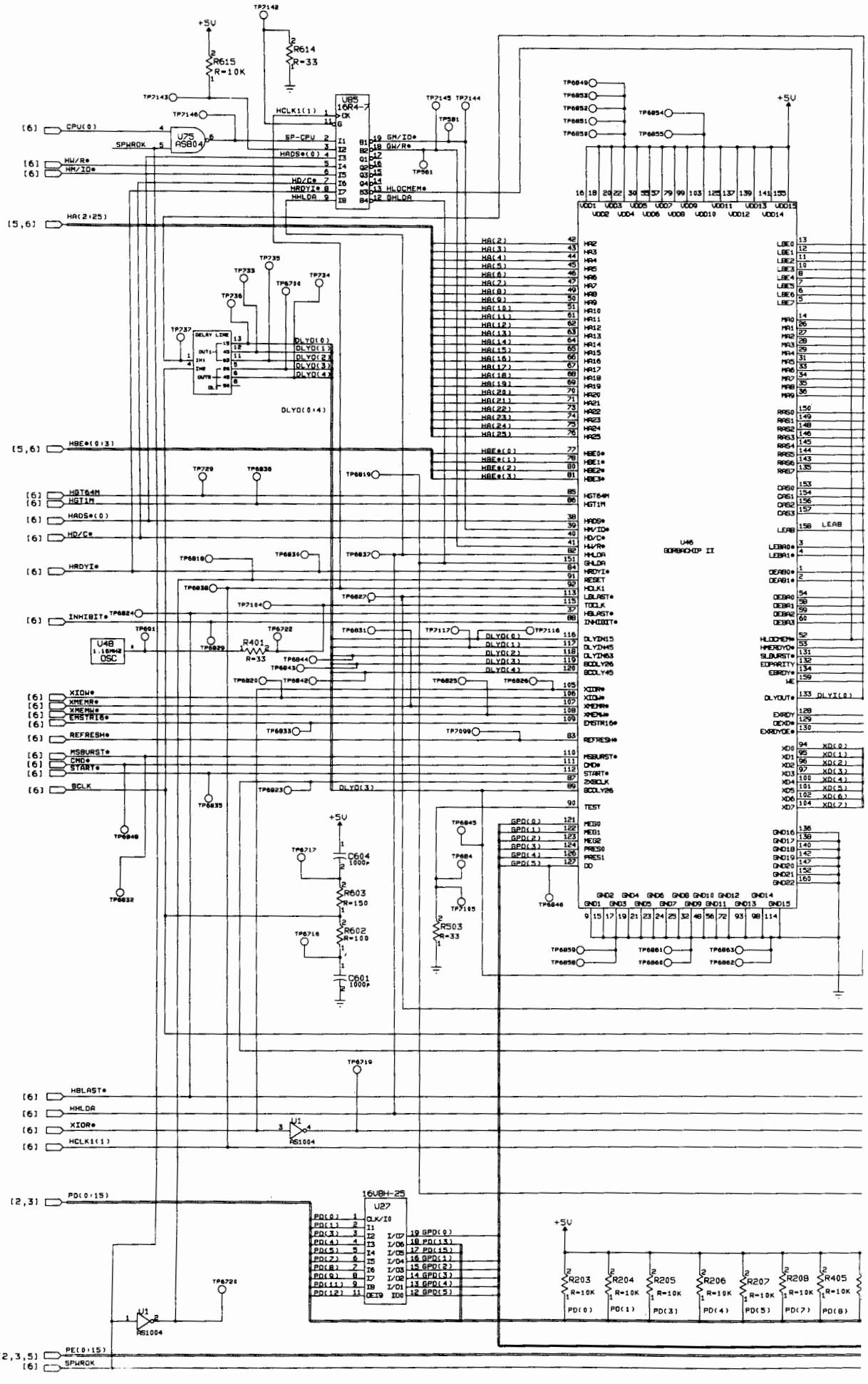
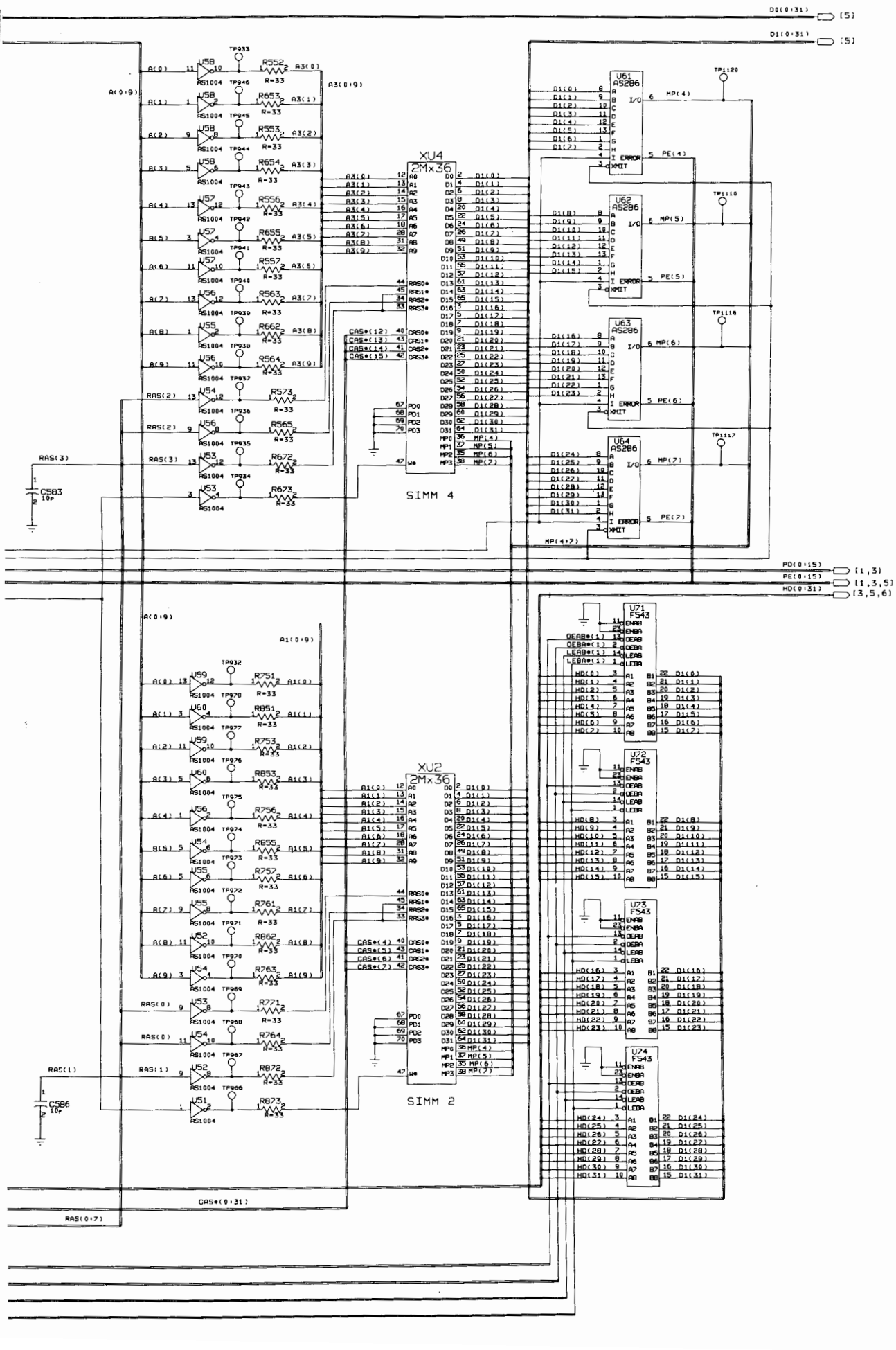

+5V
R615 R=10K
R614 R=33
U85 16R4-7
U75 AS804
SP-CPU
SPWROK
HCLK1(1)
HADS*(0)
HD/C*
HRDYI*
HHLDA
HW/R*
HM/IO*
HLOCMEM*
BHLDA
CPU(0)
HA(2:25)
HBE*(0:3)
HBT64M
HBT1M
INNIBIT*
U48 1.16MHZ OSC
R401 R=33
XIOW*
XMEMR*
XMEMW*
EMSTR16*
REFRESH*
MSBURST*
CMD*
START*
BCLK
DELAY LINE
DLY0(0)
DLY0(1)
DLY0(2)
DLY0(3)
DLY0(4)
DLY0(0:4)
U46 BORABACHIP II
C604 1000P
R603 R=150
R602 R=100
C601 1000P
R503 R=33
HBLAST*
HHLDA
XIOR*
HCLK1(1)
PD(0:15)
PE(0:15)
SPWROK
16V8H-25
U27
R203 R=10K
R204 R=10K
R205 R=10K
R206 R=10K
R207 R=10K
R208 R=10K
R405 R=10K

Figure B-31. Memory Board (Memory Module Sockets 1-4)
Schematics B-63/B-64


D0(0:31)
[5]
D1(0:31)
[5]
A(0:9)
A3(0:9)
XU4
2Mx36
SIMM 4
U61
AS286
U62
AS286
U63
AS286
U64
AS286
MP(4:7)
PD(0:15)
[1,3]
PE(0:15)
[1,3,5]
HD(0:31)
[3,5,6]
A1(0:9)
XU2
2Mx36
SIMM 2
U71
FS43
U72
FS43
U73
FS43
U74
FS43
RAS(3)
C583
10p
RAS(1)
C586
10p
CAS*(0:31)
RAS(0:7)

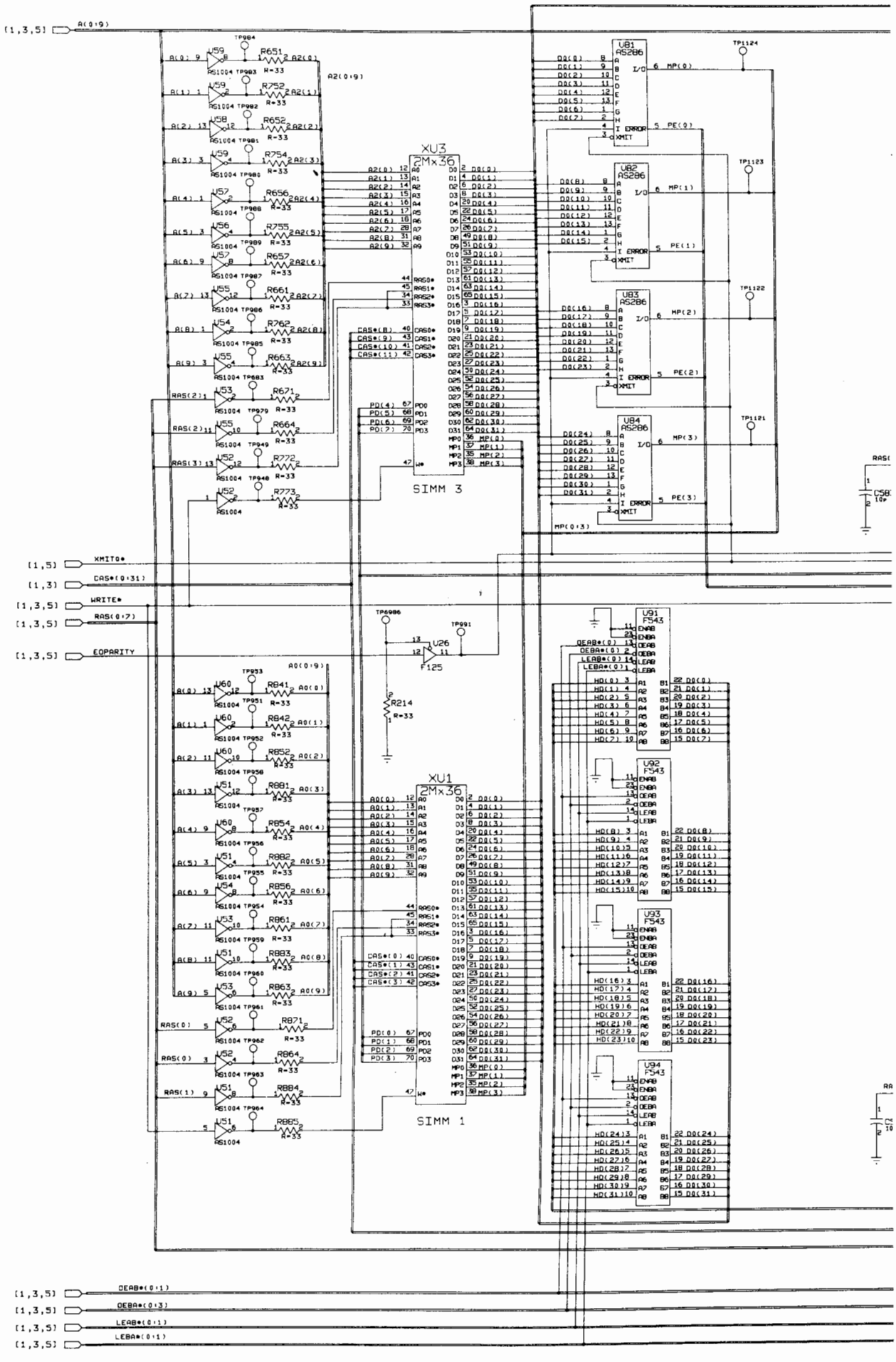
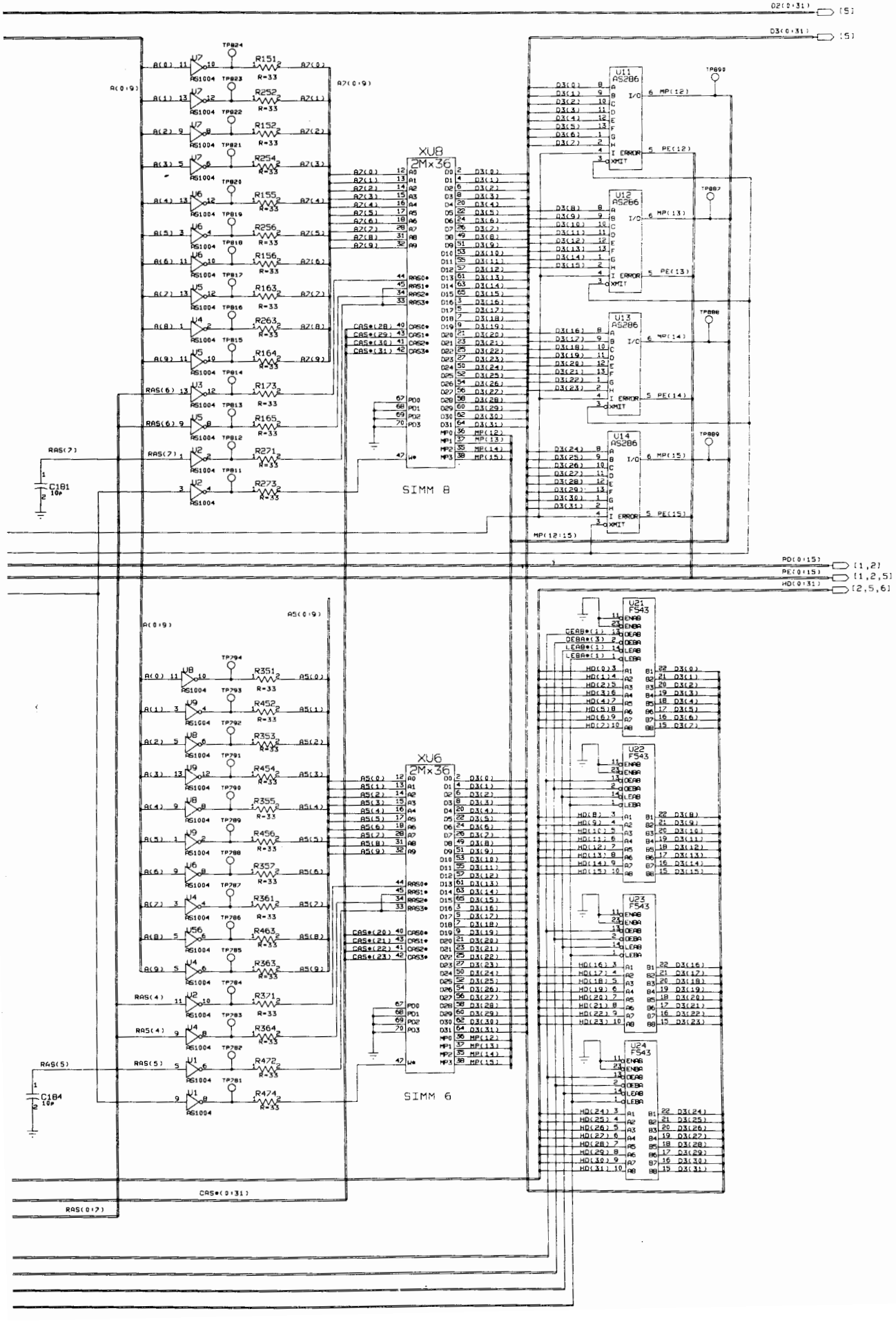

[1,3,5] A(0:9)
A2(0:9)
SIMM 3
SIMM 1
XU3
XU1
2Mx36
[1,5] XMITO*
[1,3] CAS*(0:31)
[1,3,5] WRITE*
[1,3,5] RAS(0:7)
[1,3,5] EOPARITY
A0(0:9)
[1,3,5] OEAB*(0:1)
[1,3,5] OEBA*(0:3)
[1,3,5] LEAB*(0:1)
[1,3,5] LEBA*(0:1)
MP(0:3)

Figure B-32. Memory Board (Memory Module Sockets 5-8)
Schematics B-65/B-66


D2(0:31)
D3(0:31)
XU8
2Mx36
SIMM 8
XU6
SIMM 6
U11 AS286
U12 AS286
U13 AS286
U14 AS286
MP(12:15)
PD(0:15)
PE(0:15)
HD(0:31)
U21 F543
U22 F543
U23 F543
U24 F543
C181 10p
C184 10p
CAS*(0:31)
RAS(0:7)

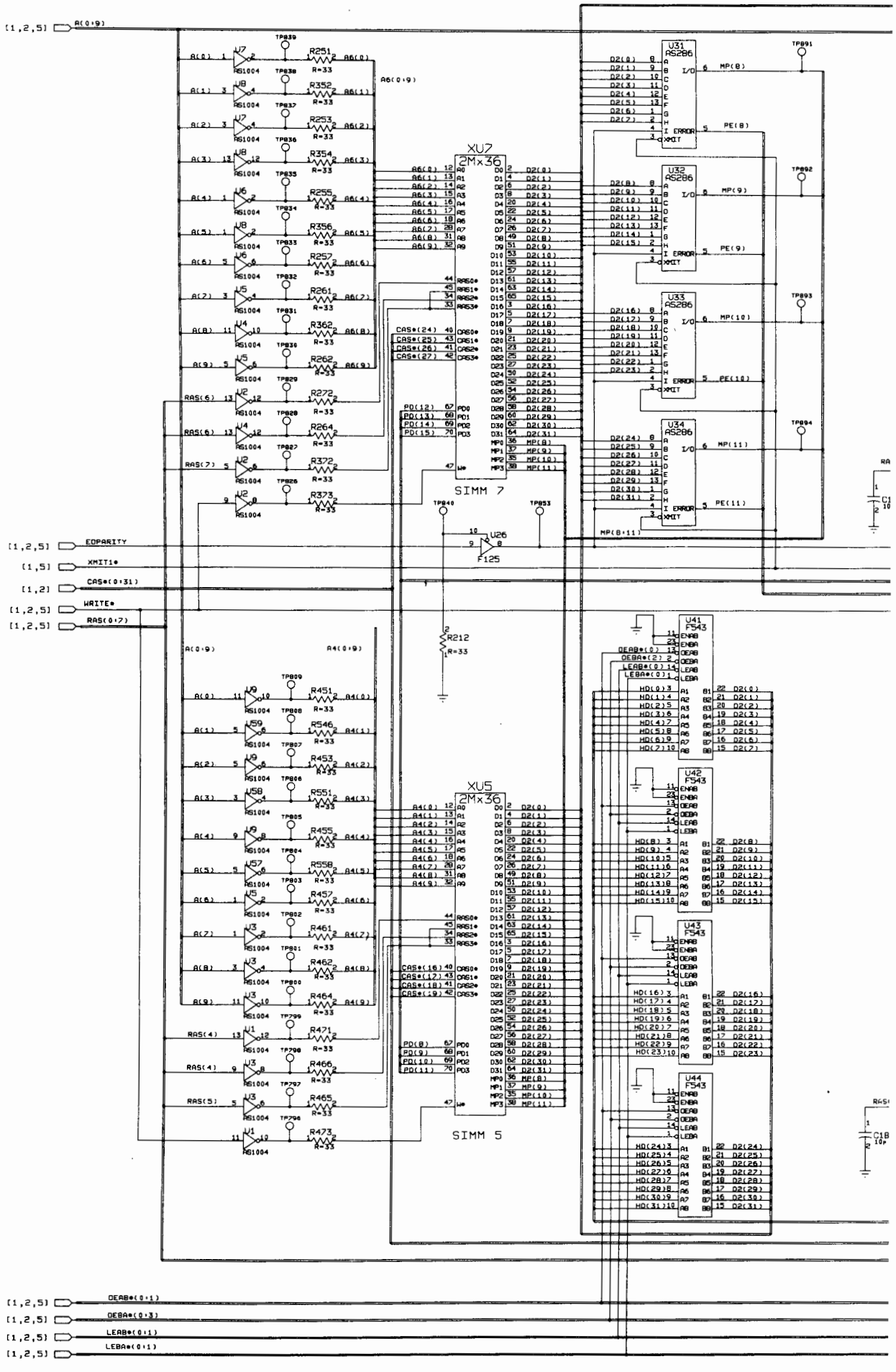
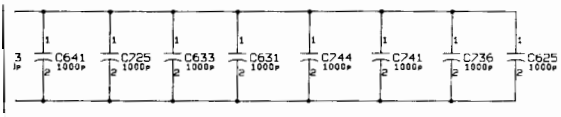
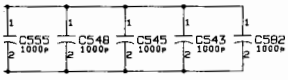
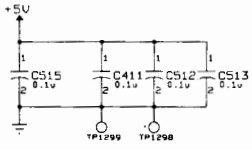

[1,2,5] A(0:9)
A6(0:9)
XU7
2Mx36
SIMM 7
XU5
SIMM 5
U31 AS286
U32 AS286
U33 AS286
U34 AS286
[1,2,5] EDPARITY
[1,5] XMIT1*
[1,2] CAS*(0:31)
[1,2,5] WRITE*
[1,2,5] RAS(0:7)
A4(0:9)
U41 F543
U42 F543
U43 F543
U44 F543
[1,2,5] DEAB*(0:1)
[1,2,5] DEBA*(0:3)
[1,2,5] LEAB*(0:1)
[1,2,5] LEBA*(0:1)

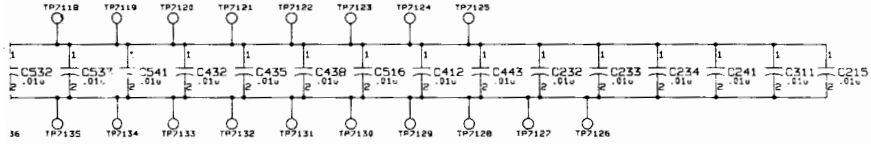
Figure B-33. Memory Board (Bypass Capacitors)
Schematics B-67/B-68

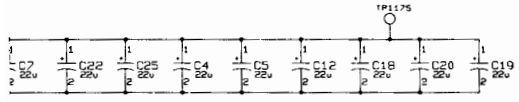

C641 1000p
C725 1000p
C633 1000p
C631 1000p
C744 1000p
C741 1000p
C736 1000p
C625 1000p


C555 1000p
C548 1000p
C545 1000p
C543 1000p
C582 1000p

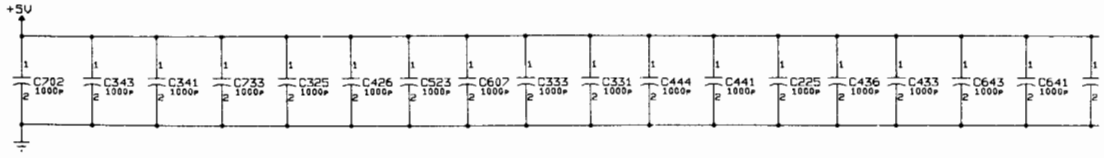

+5V
C515 0.1u
C411 0.1u
C512 0.1u
C513 0.1u
TP1299
TP1298

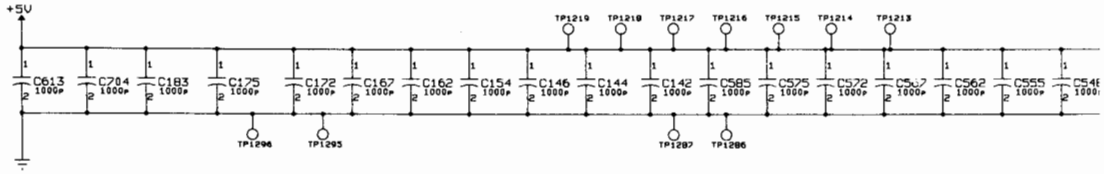

C547 .01u
C544 .01u
C644 .01u
C182 .01u
C174 .01u
C171 .01u
C166 .01u
C745 .01u
C742 .01u
C737 .01u
C734 .01u
C605 .01u
C703 .01u
C701 .01u
C161 .01u
C301 .01u
C502 .01u
C501 .01u
C514 .01u
C511 .01u
C612 .01u

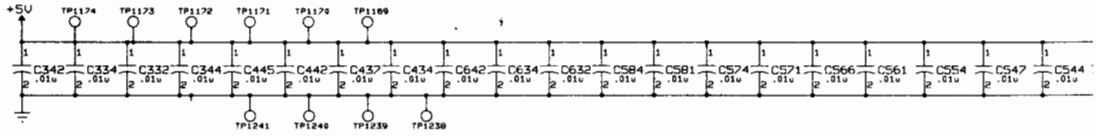

TP7118
TP7119
TP7120
TP7121
TP7122
TP7123
TP7124
TP7125
C532 .01u
C541 .01u
C432 .01u
C435 .01u
C438 .01u
C516 .01u
C412 .01u
C443 .01u
C232 .01u
C233 .01u
C234 .01u
C241 .01u
C411 .01u
C215 .01u
TP7135
TP7134
TP7133
TP7132
TP7131
TP7130
TP7129
TP7128
TP7127
TP7126

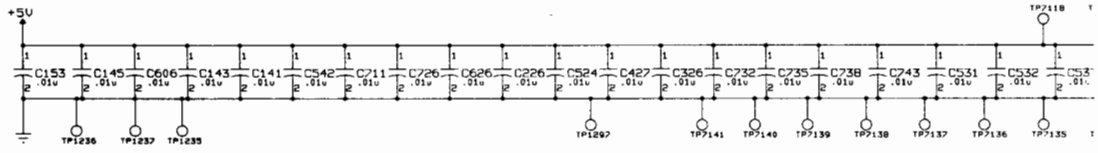

TP1175
C7 22u
C22 22u
C25 22u
C4 22u
C5 22u
C12 22u
C18 22u
C20 22u
C19 22u

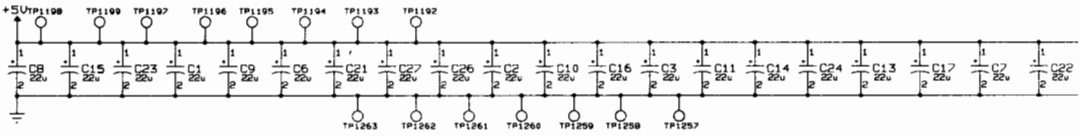

FLASH [6]


+5V
C702 1000p
C343 1000p
C341 1000p
C733 1000p
C325 1000p
C426 1000p
C523 1000p
C607 1000p
C333 1000p
C331 1000p
C444 1000p
C441 1000p
C225 1000p
C436 1000p
C433 1000p
C643 1000p
C641 1000p


+5V
TP1219
TP1218
TP1217
TP1216
TP1215
TP1214
TP1213
C613 1000p
C704 1000p
C183 1000p
C175 1000p
C172 1000p
C167 1000p
C162 1000p
C154 1000p
C146 1000p
C144 1000p
C142 1000p
C585 1000p
C575 1000p
C572 1000p
C562 1000p
C555 1000p
TP1296
TP1295
TP1287
TP1286

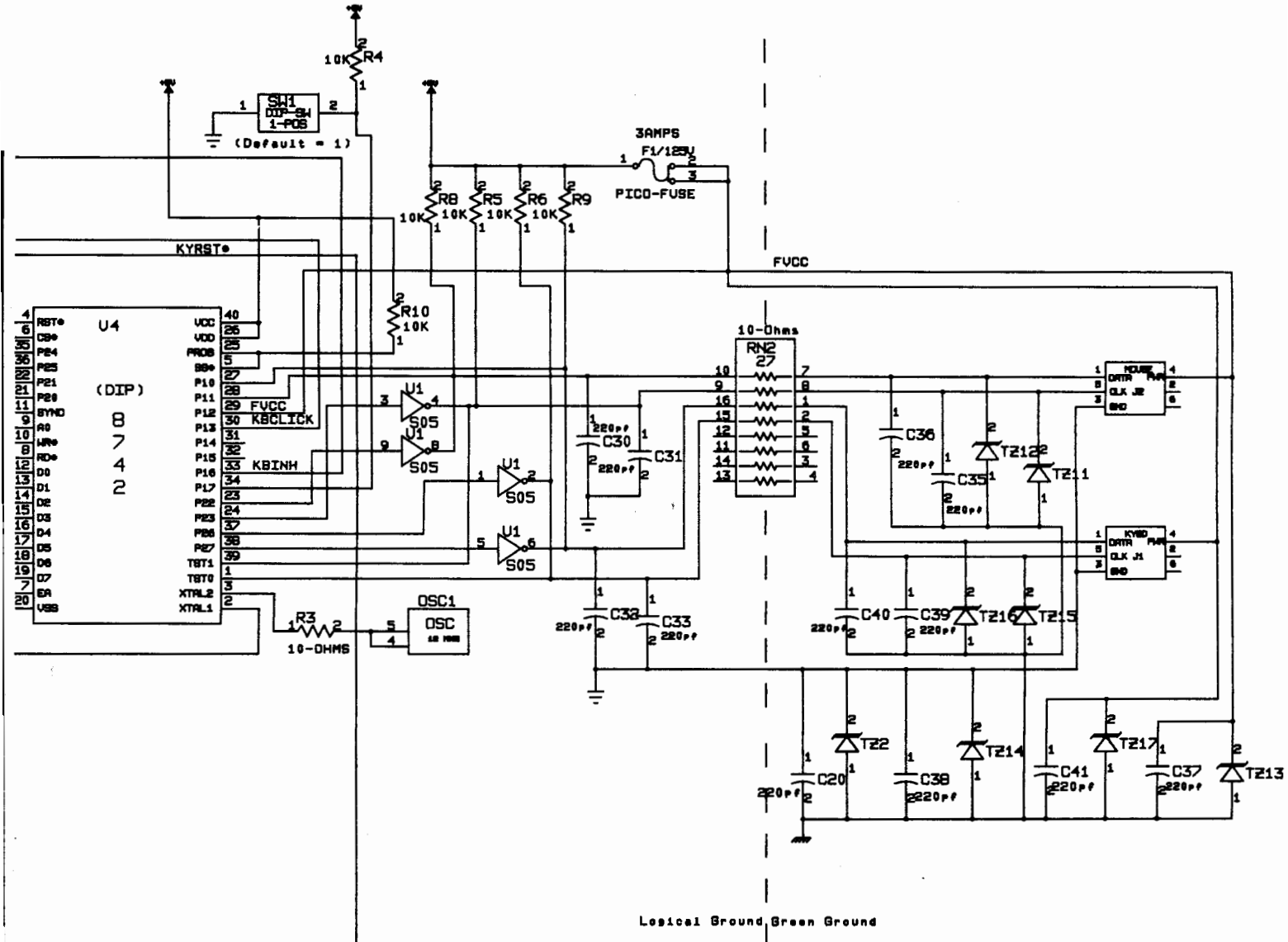

+5V
TP1174
TP1173
TP1172
TP1171
TP1170
TP1169
C342 .01u
C334 .01u
C332 .01u
C344 .01u
C445 .01u
C442 .01u
C437 .01u
C434 .01u
C642 .01u
C634 .01u
C632 .01u
C584 .01u
C581 .01u
C574 .01u
C571 .01u
C566 .01u
C561 .01u
C554 .01u
C547 .01u
TP1241
TP1240
TP1239
TP1238


+5V
TP7118
C153 .01u
C145 .01u
C606 .01u
C143 .01u
C141 .01u
C542 .01u
C711 .01u
C726 .01u
C626 .01u
C226 .01u
C524 .01u
C427 .01u
C326 .01u
C732 .01u
C735 .01u
C738 .01u
C743 .01u
C531 .01u
C532 .01u
TP1236
TP1237
TP1235
TP1297
TP7141
TP7140
TP7139
TP7138
TP7137
TP7136
TP7135


+5V
TP1198
TP1199
TP1197
TP1196
TP1195
TP1194
TP1193
TP1192
C8 22u
C15 22u
C23 22u
C1 22u
C9 22u
C6 22u
C21 22u
C27 22u
C26 22u
C2 22u
C10 22u
C16 22u
C3 22u
C11 22u
C14 22u
C24 22u
C13 22u
C17 22u
C7 22u
C22 22u
TP1263
TP1262
TP1261
TP1260
TP1259
TP1258
TP1257


[6]
+12V
C587 .01u
C588 0.1u
DIP-SW 2-POS
SW1
TP1161

Figure B-34. Mouse/Keyboard Serial Board (Mouse and Keyboard Interface)
Schematics B-69/B-70


10K R4
SW1 DIP-SW 1-POS
(Default = 1)
3AMPS F1/125V
PICO-FUSE
R8 10K
R5 10K
R6 10K
R9 10K
KYRST*
FVCC
R10 10K
U4
(DIP)
8 7 4 2
FVCC
KBCLICK
KBINH
U1 S05
10-Ohms
RN2 27
C30 220pf
C31 220pf
C36 220pf
C35 220pf
TZ12
TZ11
MOUSE DATA CLK J2 GND PWR
KYBD DATA CLK J1 GND PWR
C32 220pf
C33 220pf
C40 220pf
C39 220pf
TZ16
TZ15
OSC1
OSC
R3 10-OHMS
TZ22
TZ14
TZ17
TZ13
C20 220pf
C38 220pf
C41 220pf
C37 220pf
Logical Ground
Green Ground

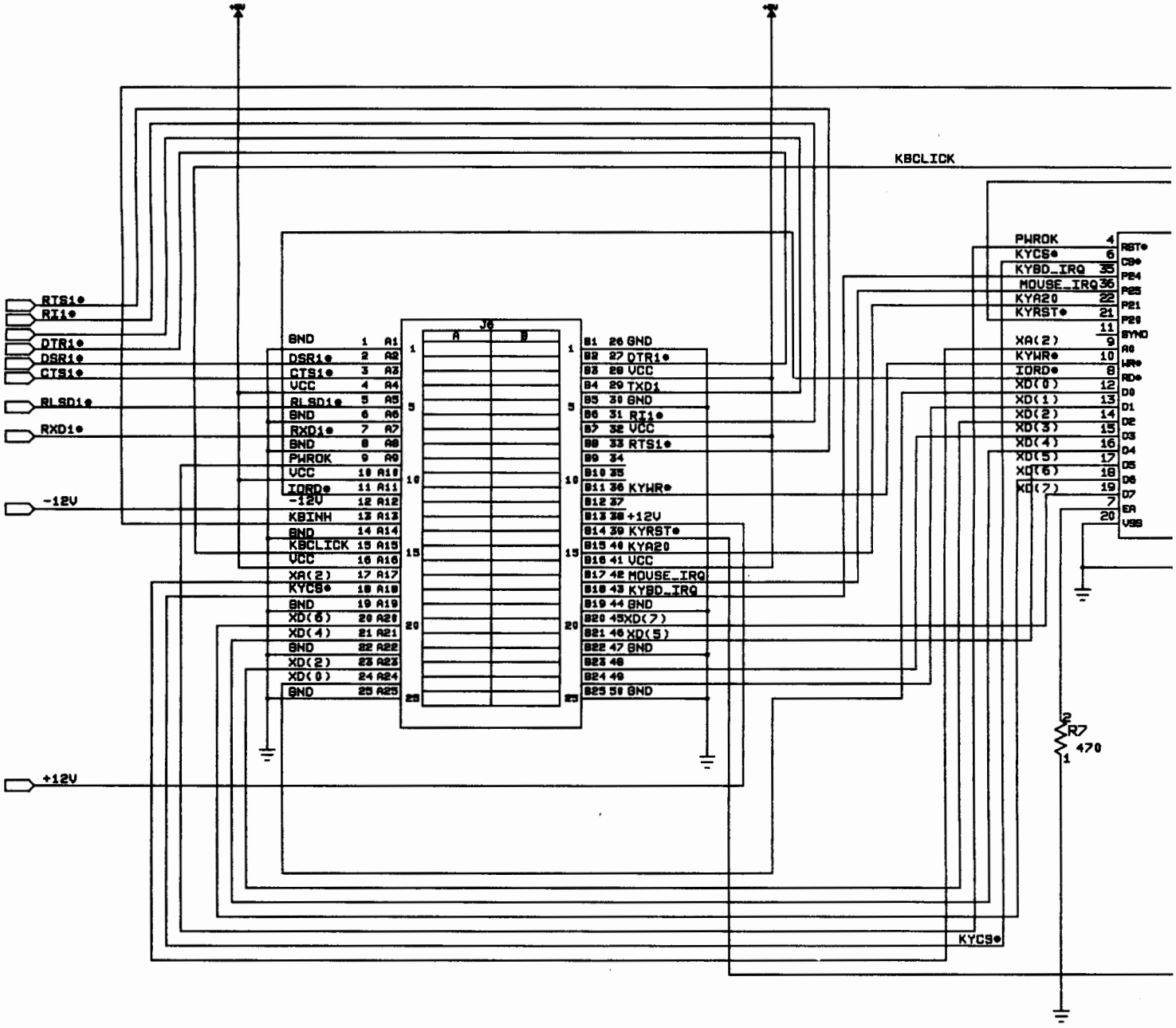
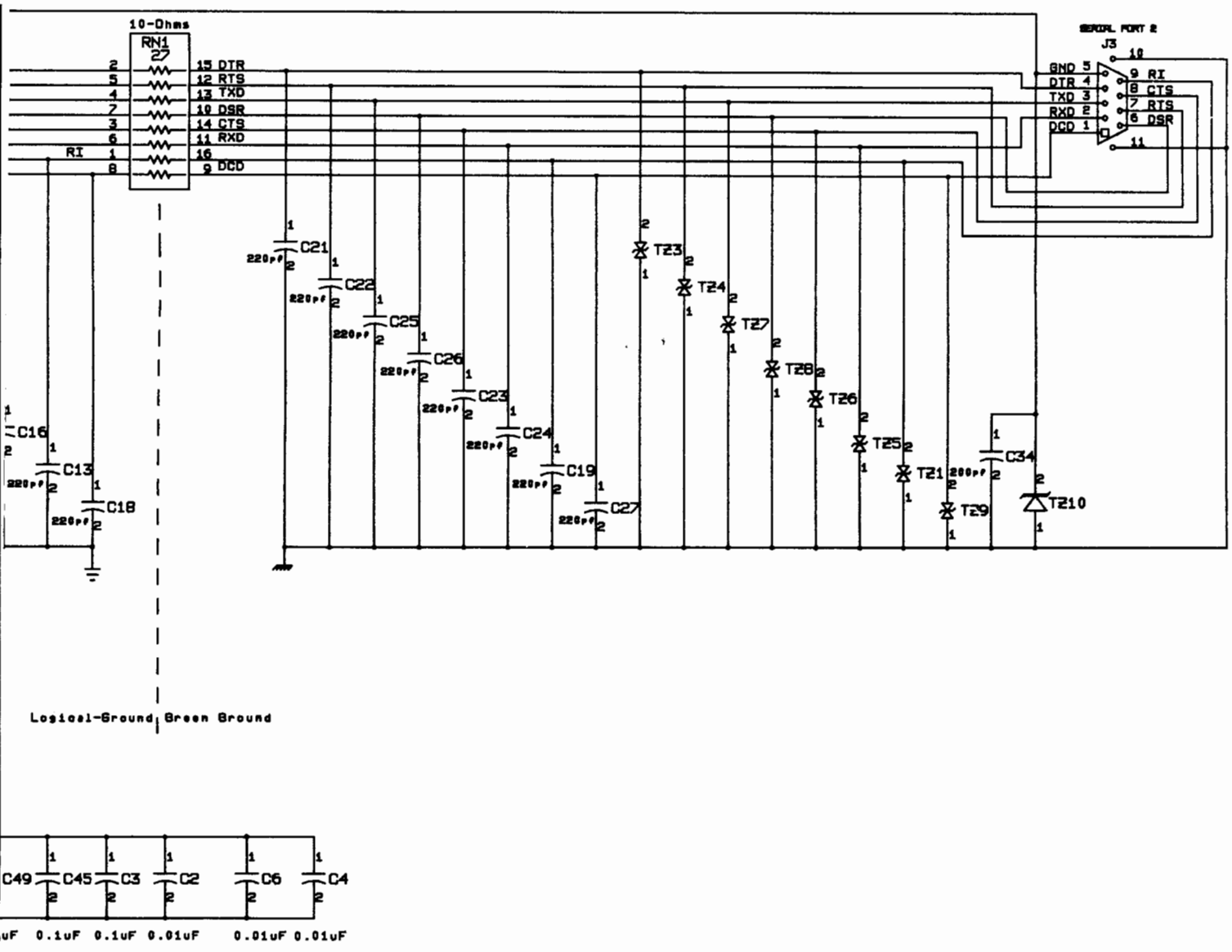

KBCLICK
RTS1●
RI1●
DTR1●
DSR1●
CTS1●
RLSD1●
RXD1●
-12V
+12V
KYCS●
R7
470
J6
A
B
GND 1 A1
DSR1● 2 A2
CTS1● 3 A3
VCC 4 A4
RLSD1● 5 A5
GND 6 A6
RXD1● 7 A7
GND 8 A8
PWROK 9 A9
VCC 10 A10
IORD● 11 A11
-12V 12 A12
KBINH 13 A13
GND 14 A14
KBCLICK 15 A15
VCC 16 A16
XA(2) 17 A17
KYCS● 18 A19
GND 19 A19
XD(6) 20 A20
XD(4) 21 A21
GND 22 A22
XD(2) 23 A23
XD(0) 24 A24
GND 25 A25
B1 26 GND
B2 27 DTR1●
B3 28 VCC
B4 29 TXD1
B5 30 GND
B6 31 RI1●
B7 32 VCC
B8 33 RTS1●
B9 34
B10 35
B11 36 KYWR●
B12 37
B13 38 +12V
B14 39 KYRST●
B15 40 KYA20
B16 41 VCC
B17 42 MOUSE_IRQ
B18 43 KYBD_IRQ
B19 44 GND
B20 45 XD(7)
B21 46 XD(5)
B22 47 GND
B23 48
B24 49
B25 50 GND
PWROK 4
KYCS● 6
KYBD_IRQ 35
MOUSE_IRQ 36
KYA20 22
KYRST● 21
XA(2) 9
KYWR● 10
IORD● 8
XD(0) 12
XD(1) 13
XD(2) 14
XD(3) 15
XD(4) 16
XD(5) 17
XD(6) 18
XD(7) 19
7
20
RST●
CS●
P24
P25
P21
P20
11
SYNC
A0
WR●
RD●
D0
D1
D2
D3
D4
D5
D6
D7
EA
VSS

Figure B-35. Mouse/Keyboard Serial Board (Serial Port 2 Interface) Schematics B-71/B-72


10-Ohms
RN1
27
2
5
4
7
3
6
RI 1
8
15 DTR
12 RTS
13 TXD
10 DSR
14 CTS
11 RXD
16
9 DCD
SERIAL PORT 2
J3
10
GND 5
DTR 4
TXD 3
RXD 2
DCD 1
9 RI
8 CTS
7 RTS
6 DSR
11
C21
C22
C25
C26
C23
C24
C19
C27
C16
C13
C18
C34
T23
T24
T27
T28
T26
T25
T21
T29
T210
220pF
Logical-Ground
Green Ground

C49
C45
C3
C2
C6
C4
uF 0.1uF 0.1uF 0.01uF 0.01uF 0.01uF

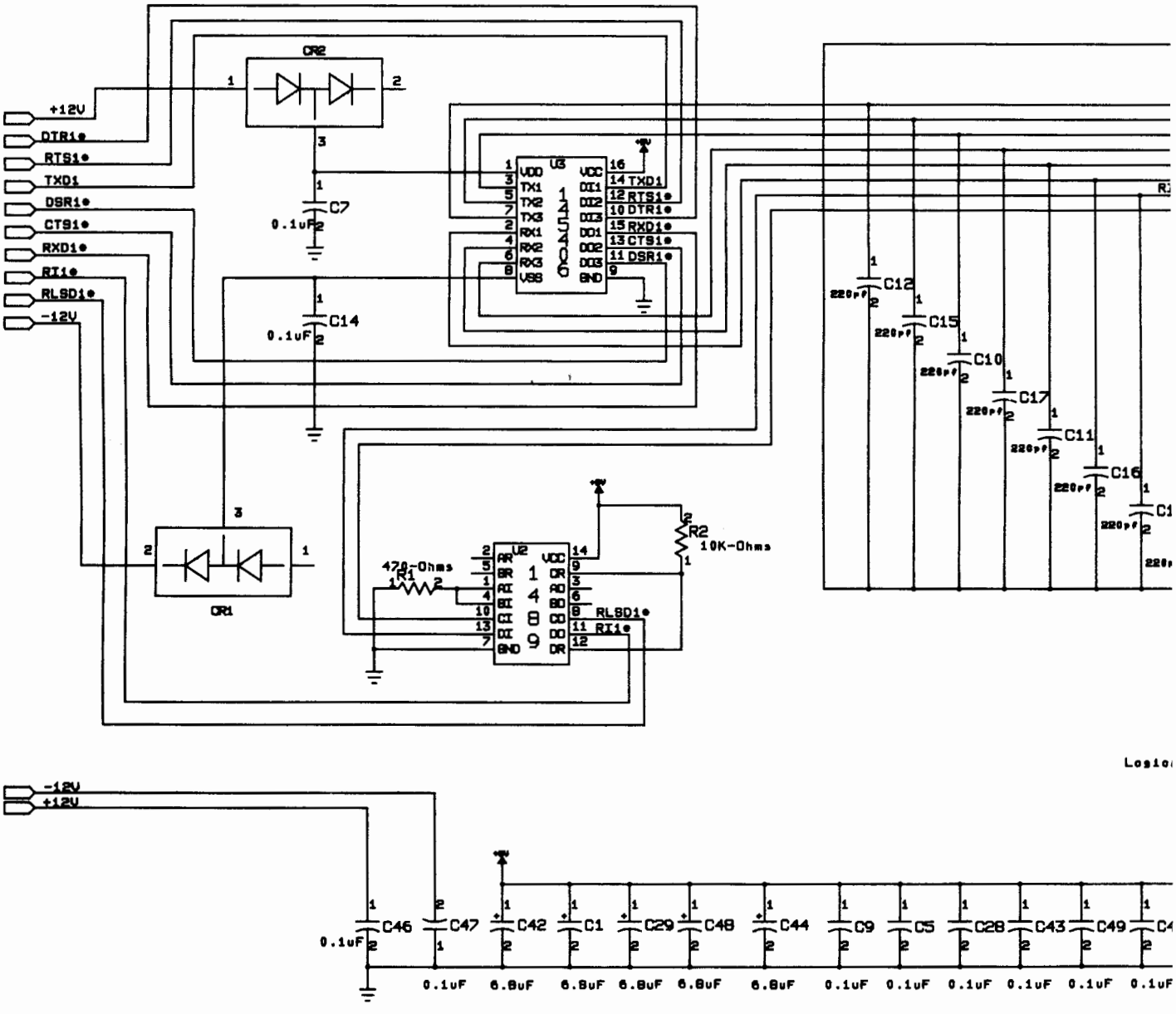
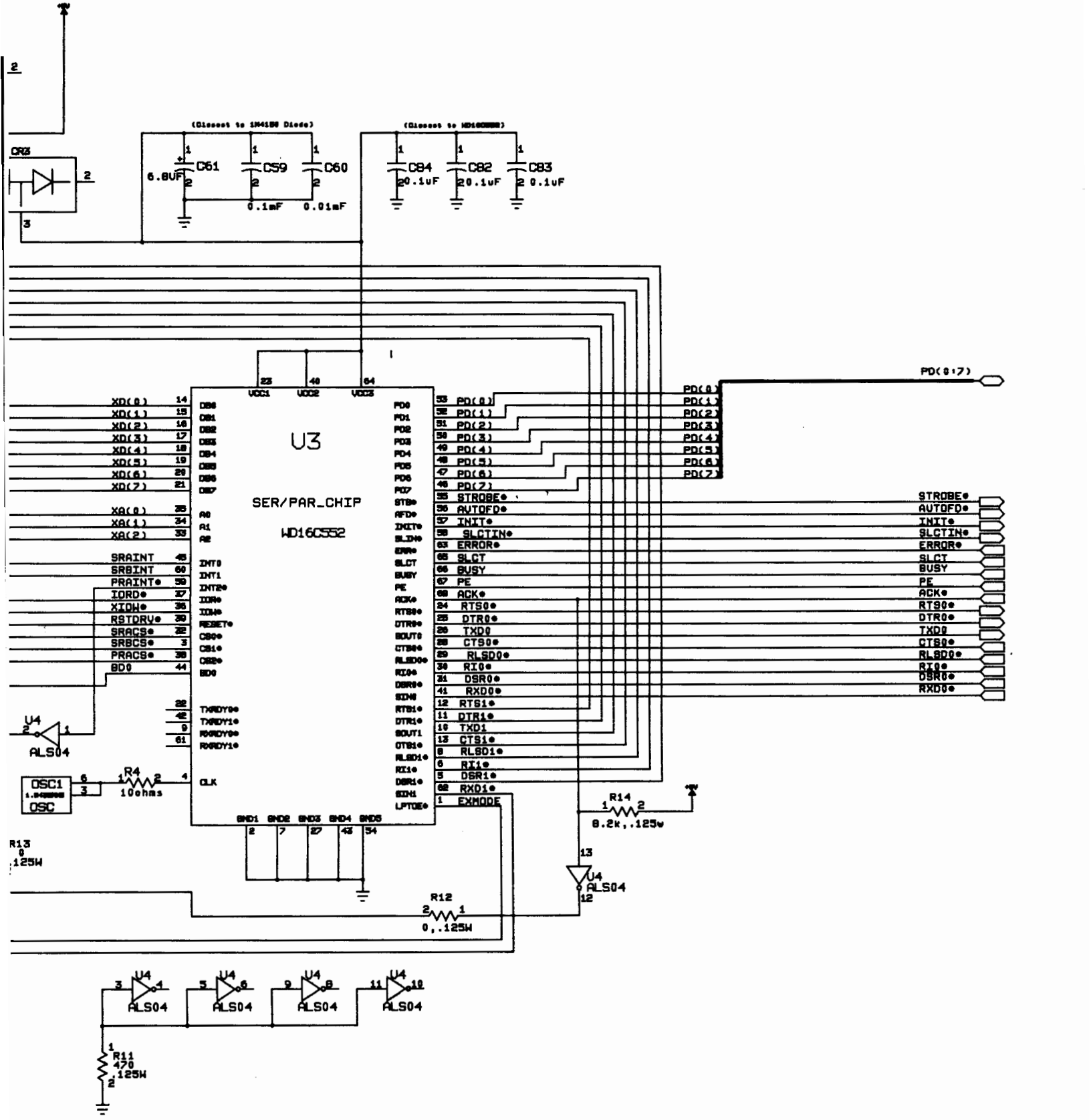

CR2
+12V
DTR1●
RTS1●
TXD1
DSR1●
CTS1●
RXD1●
RI1●
RLSD1●
-12V
C7
0.1uF
C14
0.1uF
U3
VDD
TX1
TX2
TX3
RX1
RX2
RX3
VSS
VCC
DO1
DO2
DO3
DO1
DO2
DO3
GND
TXD1
RTS1●
DTR1●
RXD1●
CTS1●
DSR1●
C12
C15
C10
C17
C11
C16
220pF
CR1
R1
470-Ohms
R2
10K-Ohms
U2
RLSD1●
RI1●
-12V
+12V
C46
C47
C42
C1
C29
C48
C44
C9
C5
C28
C43
C49
0.1uF
6.8uF
0.1uF

Figure B-36. Parallel/Serial Board (Parallel and Serial Interface)
Schematics B-73/B-74


+5V
2
CR5
3
(Closest to IN4150 Diode)
C61
6.8UF
C59
0.1mF
C60
0.01mF
(Closest to HD16C552)
C84
0.1uF
C82
0.1uF
C83
0.1uF
U3
SER/PAR_CHIP
WD16C552
VCC1
VCC2
VCC3
XD(0)
XD(1)
XD(2)
XD(3)
XD(4)
XD(5)
XD(6)
XD(7)
XA(0)
XA(1)
XA(2)
SRAINT
SRBINT
PRAINT*
IORD*
XIOW*
RSTDRV*
SRACS*
SRBCS*
PRACS*
BD0
PD(0:7)
STROBE*
AUTOFD*
INIT*
SLCTIN*
ERROR*
SLCT
BUSY
PE
ACK*
RTS0*
DTR0*
TXD0
CTS0*
RLSD0*
RI0*
DSR0*
RXD0*
RTS1*
DTR1*
TXD1
CTS1*
RLSD1*
RI1*
DSR1*
RXD1*
EXMODE
GND1
GND2
GND3
GND4
GND5
U4
ALS04
OSC1
OSC
R4
10ohms
R14
8.2k,.125w
R13
0
.125W
R12
0,.125W
R11
470
.125W

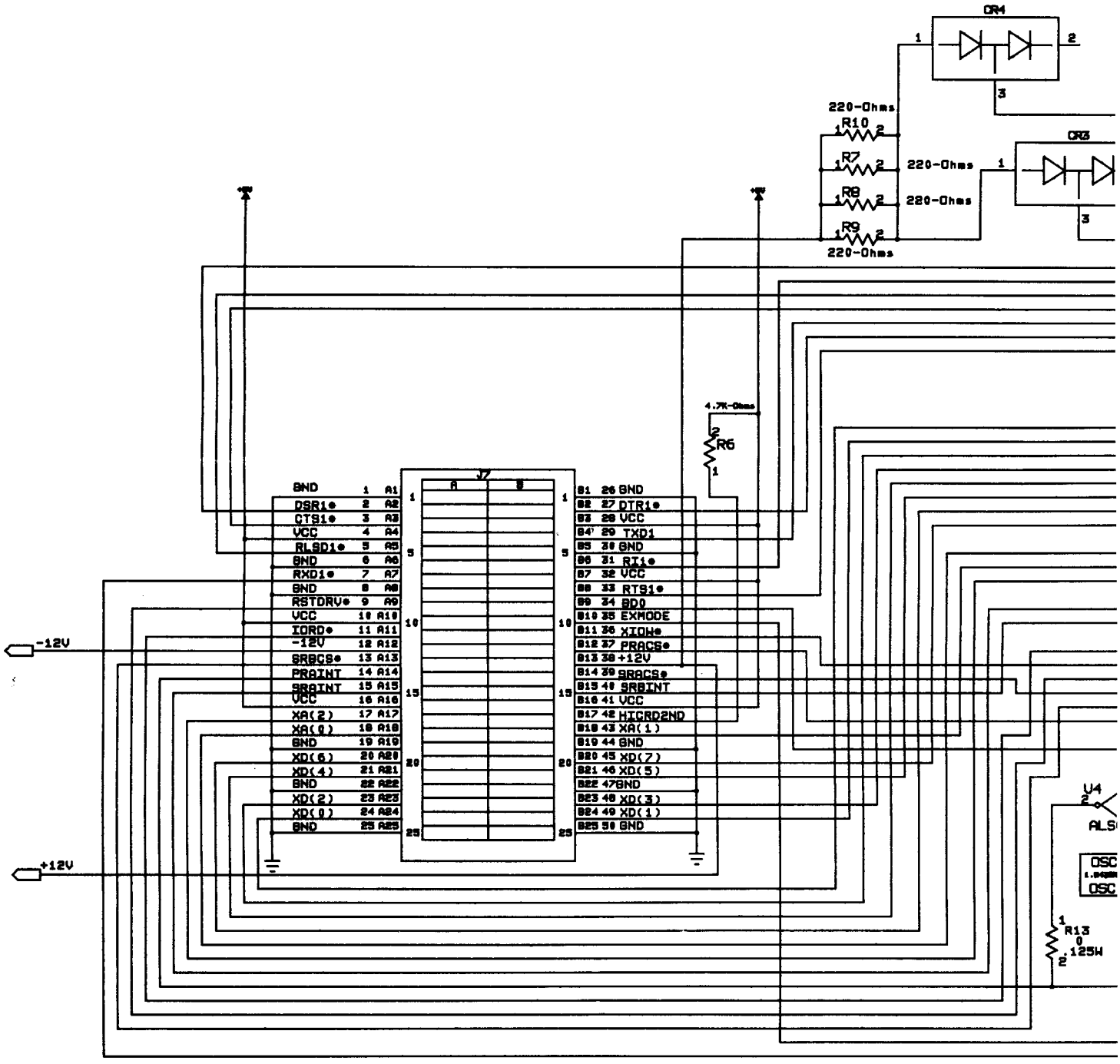
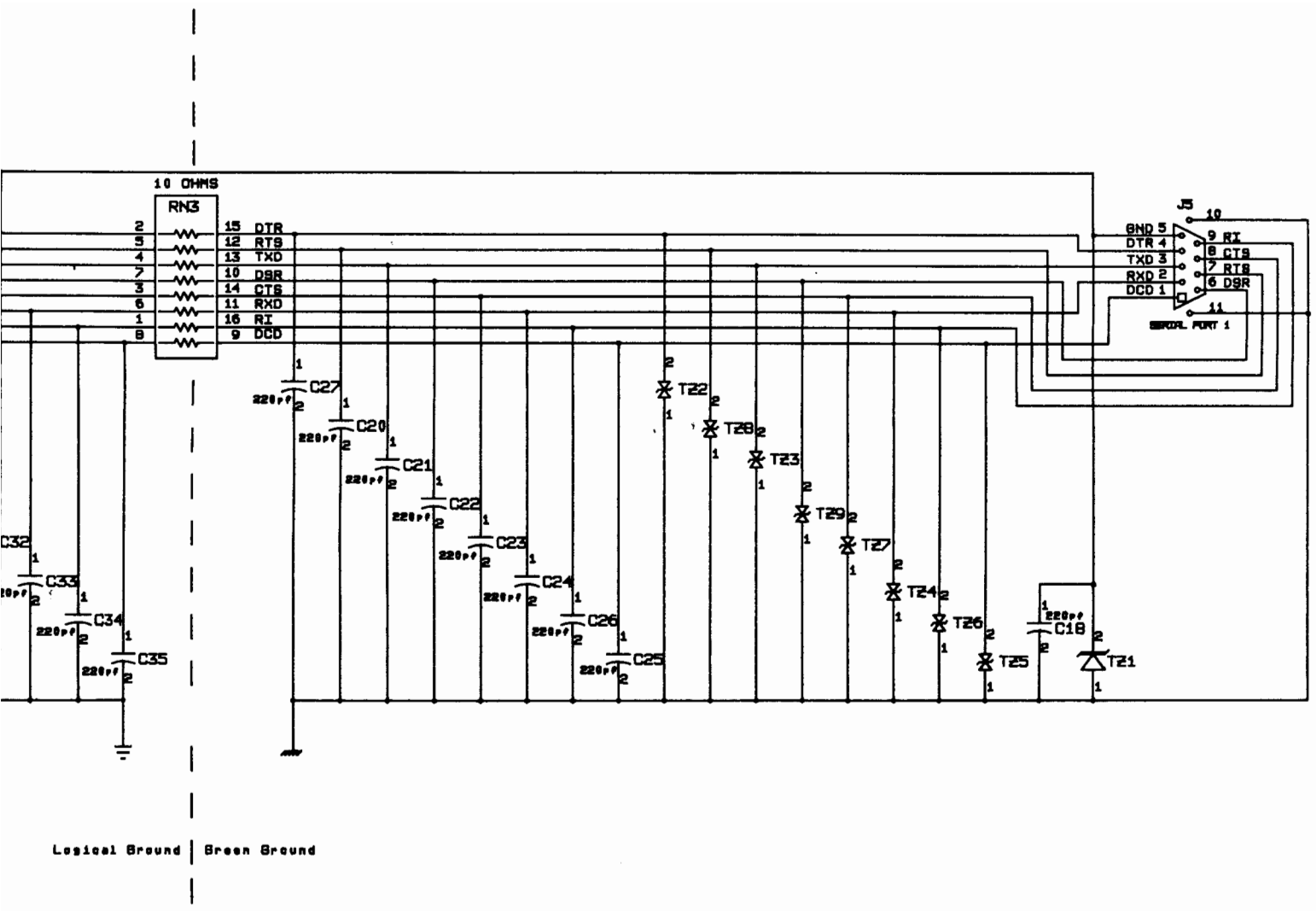
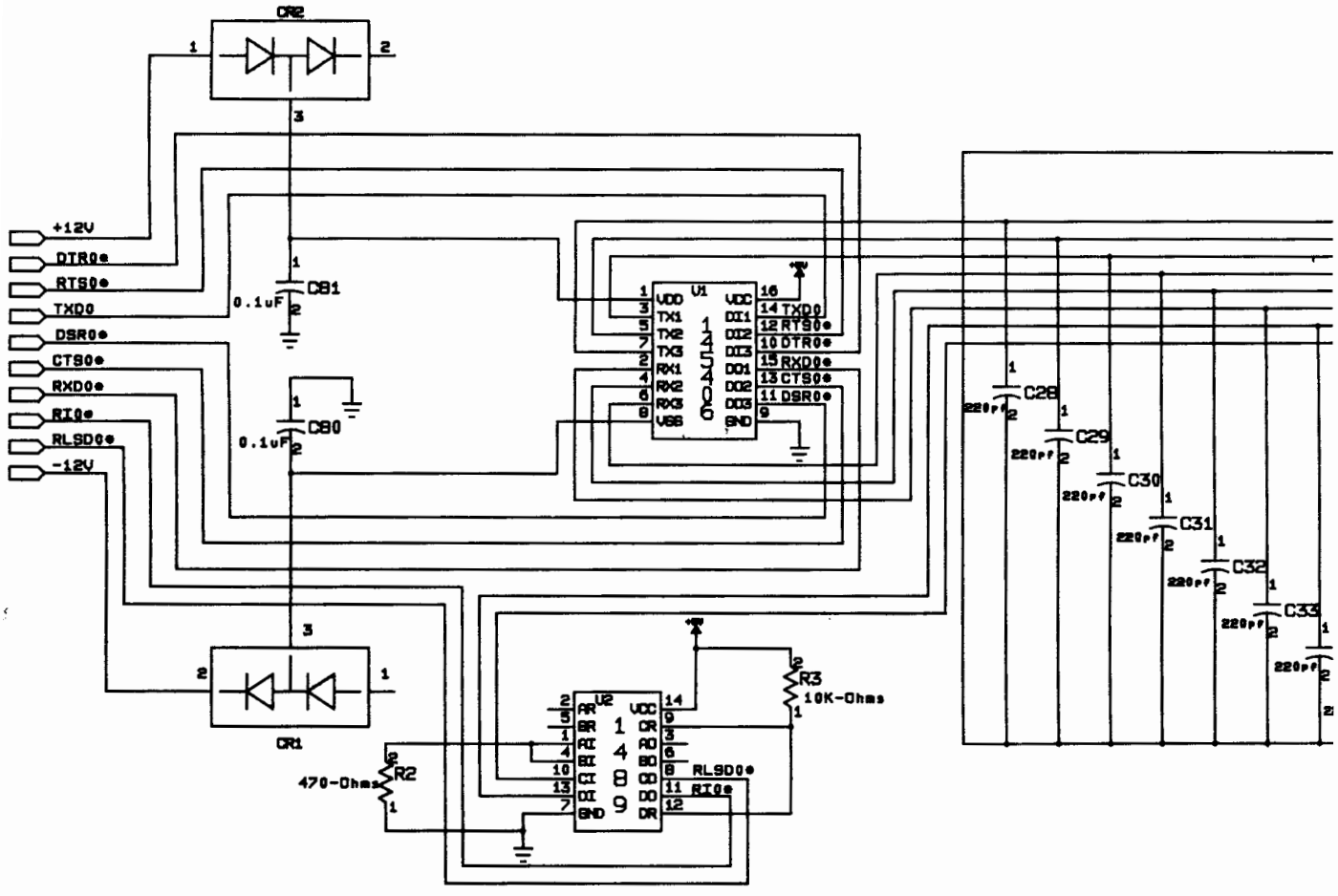

CR4
220-Ohms
R10
R7
220-Ohms
R8
220-Ohms
R9
220-Ohms
CR3
4.7K-Ohms
R6
J7
A
B
GND 1 A1
DSR1* 2 A2
CTS1* 3 A3
VCC 4 A4
RLSD1* 5 A5
GND 6 A6
RXD1* 7 A7
GND 8 A8
RSTDRV* 9 A9
VCC 10 A10
IORD* 11 A11
-12V 12 A12
SRBCS* 13 A13
PRAINT 14 A14
SRAINT 15 A15
VCC 16 A16
XA(2) 17 A17
XA(0) 18 A18
GND 19 A19
XD(6) 20 A20
XD(4) 21 A21
GND 22 A22
XD(2) 23 A23
XD(0) 24 A24
GND 25 A25
B1 26 GND
B2 27 DTR1*
B3 28 VCC
B4 29 TXD1
B5 30 GND
B6 31 RI1*
B7 32 VCC
B8 33 RTS1*
B9 34 BD0
B10 35 EXMODE
B11 36 XIOW*
B12 37 PRACS*
B13 38 +12V
B14 39 SRACS*
B15 40 SRBINT
B16 41 VCC
B17 42 HIGRD2ND
B18 43 XA(1)
B19 44 GND
B20 45 XD(7)
B21 46 XD(5)
B22 47 GND
B23 48 XD(3)
B24 49 XD(1)
B25 50 GND
-12V
+12V
U4
R13
0
.125W

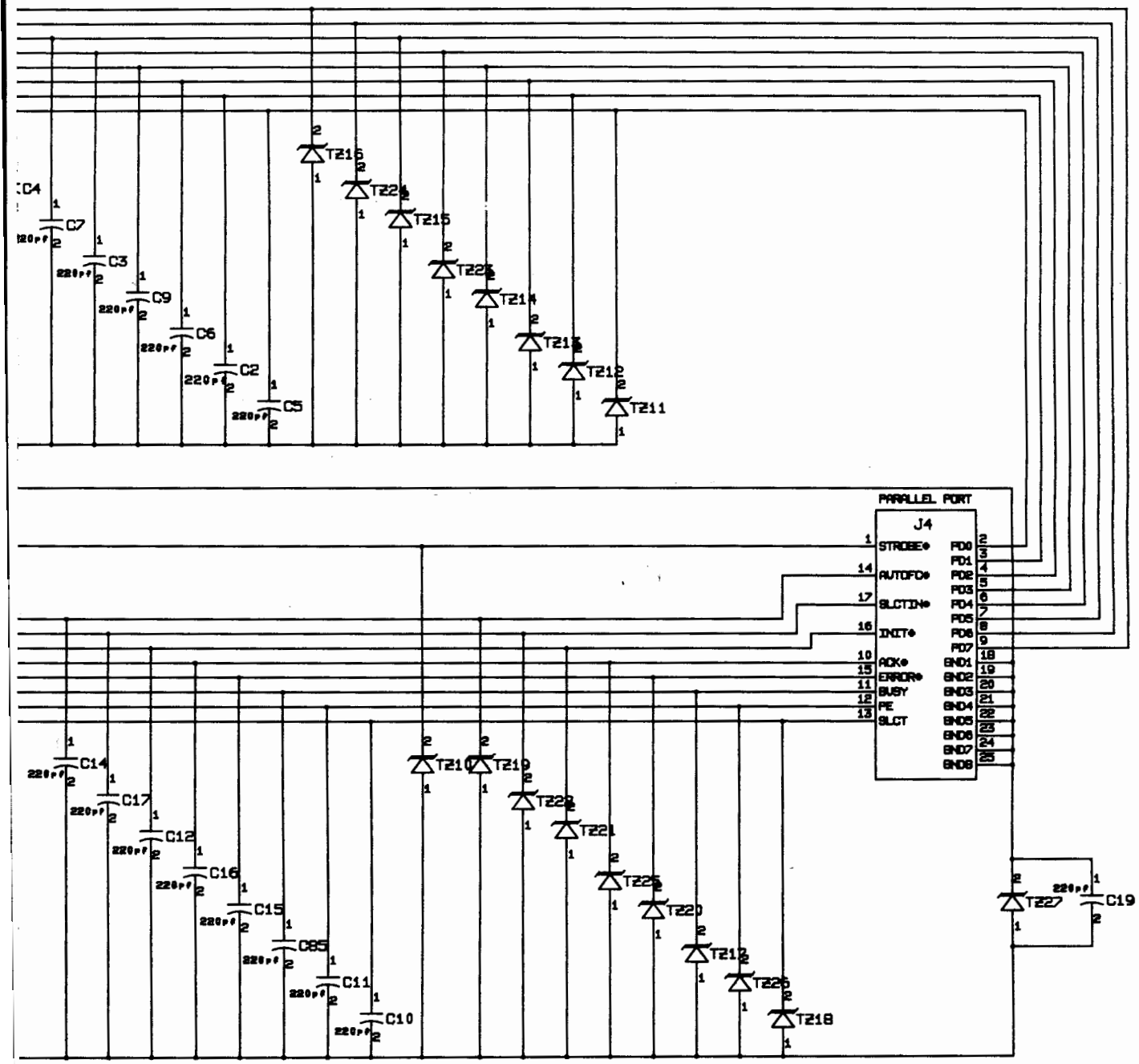
Figure B-37. Parallel/Serial Board (Serial Port 1 Interface)
Schematics B-75/B-76

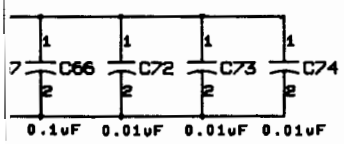

10 OHMS
RN3
2 15 DTR
5 12 RTS
4 13 TXD
7 10 DSR
3 14 CTS
6 11 RXD
1 16 RI
8 9 DCD
J5
10
GND 5
DTR 4
TXD 3
RXD 2
DCD 1
9 RI
8 CTS
7 RTS
6 DSR
11
SERIAL PORT 1
C27
C20
C21
C22
C23
C24
C26
C25
C32
C33
C34
C35
C18
T22
T28
T23
T29
T27
T24
T26
T25
T21
220pf
Logical Ground
Green Ground

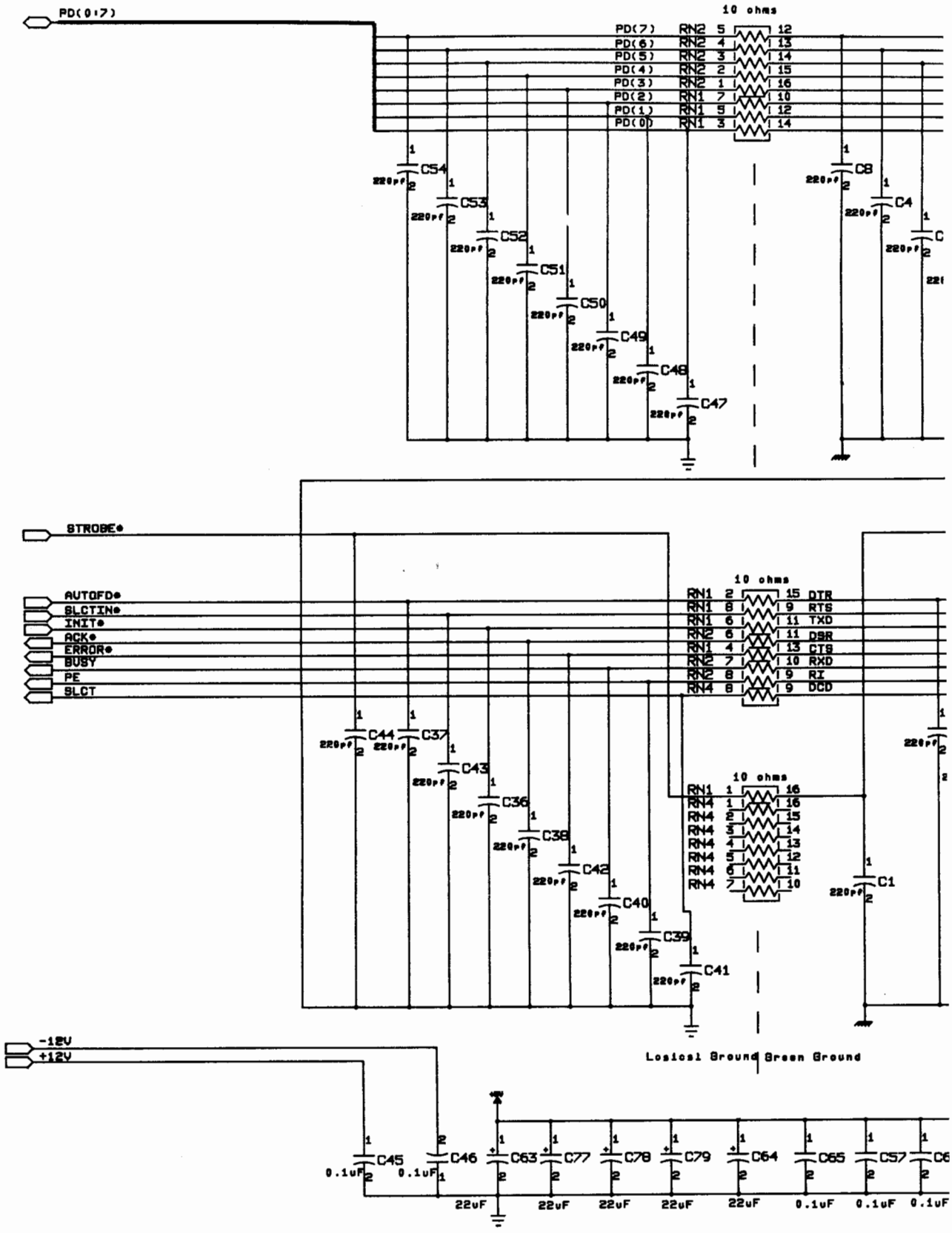


Logi

Figure B-38. Parallel/Serial Board (Parallel Port Interface)
Schematics B-77/B-78


C4
C7
C3
C9
C6
C2
C5
TZ16
TZ24
TZ15
TZ23
TZ14
TZ13
TZ12
TZ11
PARALLEL PORT
J4
STROBE*
AUTOFD*
SLCTIN*
INIT*
ACK*
ERROR*
BUSY
PE
SLCT
PD0
PD1
PD2
PD3
PD4
PD5
PD6
PD7
GND1
GND2
GND3
GND4
GND5
GND6
GND7
GND8
C14
C17
C12
C16
C15
C85
C11
C10
TZ10
TZ19
TZ22
TZ21
TZ25
TZ20
TZ17
TZ26
TZ18
TZ27
C19


C66
C72
C73
C74
0.1uF
0.01uF
0.01uF
0.01uF


PD(0:7)
10 ohms
PD(7) RN2 5 12
PD(6) RN2 4 13
PD(5) RN2 3 14
PD(4) RN2 2 15
PD(3) RN2 1 16
PD(2) RN1 7 10
PD(1) RN1 5 12
PD(0) RN1 3 14
C54 C53 C52 C51 C50 C49 C48 C47 C8 C4
220pF
STROBE*
AUTOFD*
SLCTIN*
INIT*
ACK*
ERROR*
BUSY
PE
SLCT
10 ohms
RN1 2 15 DTR
RN1 8 9 RTS
RN1 6 11 TXD
RN2 6 11 DSR
RN1 4 13 CTS
RN2 7 10 RXD
RN2 8 9 RI
RN4 8 9 DCD
C44 C37 C43 C36 C38 C42 C40 C39 C41 C1
10 ohms
RN1 1 16
RN4 1 16
RN4 2 15
RN4 3 14
RN4 4 13
RN4 5 12
RN4 6 11
RN4 7 10
-12V
+12V
Logical Ground
Green Ground
C45 0.1uF
C46 0.1uF
C63 22uF
C77 22uF
C78 22uF
C79 22uF
C64 22uF
C65 0.1uF
C57 0.1uF
0.1uF

Index

HEWLETT PACKARD

Printed in USA 03/91
Part Number 5960-0748

recycled paper