EK-11024-TM-003

PDP-11/24 System Technical Manual

digital™

EK-11024-TM-003

PDP-11/24 System Technical Manual

Prepared by Educational Services
of
Digital Equipment Corporation

1st Edition, June 1981
2nd Edition, February 1984
3rd Edition, January 1985

Printed in U.S.A.

This document was set on a DIGITAL DECset Integrated Publishing System.

CONTENTS

CONTENTS (Cont)

CONTENTS (Cont)

CONTENTS (Cont)

CONTENTS (Cont)

CONTENTS (Cont)

FIGURES

FIGURES (Cont)

FIGURES (Cont)

FIGURES (Cont)

TABLES

TABLES (Cont)

PREFACE

The PDP-11/24 is a multifunction minicomputer system that is available in a variety of configurations and is expandable by the user to conform to specific user requirements.

NOTICE: This equipment generates, uses, and may emit radio frequency energy. The equipment has been type tested and found to comply with the limits for a Class A computing device pursuant to Subpart J of Part 15 of FCC Rules, which are designed to provide reasonable protection against such radio-frequency interference when operated in a commercial environment. Operation of this equipment in a residential area may cause interference in which case the user (at his own expense) may be required to correct the interference.

This manual defines the PDP-11/24 system and provides the information required to unpack, install, and check out the system. This manual also contains a functional description of the central processor module and its options, as well as troubleshooting and diagnostic information.

Chapter 1, Introduction – Includes a general description of the PDP-11/24 system, its configurations, and specifications.

Chapter 2, Installation – Provides the information necessary to unpack, install, and check out the system. This chapter also contains jumper and switch information required to configure a PDP-11/24 system.

Chapter 3, PDP-11/24 Operation – Describes the front panel controls and indicators, the console command language, and register bit assignments.

Chapter 4, CPU Functional Description – Describes each of the major logic elements of the KDF11-UA processor module.

Chapter 5, Memory Management – Describes the function of memory management and gives a functional description of the memory management unit.

Chapter 6, KT24 UNIBUS Map Option – Provides a detailed functional description of the UNIBUS map, M9312 compatible bootstrap, and voltage monitoring logic contained on the KT24 UNIBUS map option.

Chapter 7, Floating-Point Processor Functional Description – Describes floating-point numbers and the function of the KEF11-A floating-point processor option.

Chapter 8, Commercial Instruction Set Processor – Describes the commercial instruction set and the function of the KEF11-B commercial instruction set processor option.

Chapter 9, Maintenance – Provides troubleshooting and diagnostic information.

CHAPTER 1
INTRODUCTION

1.1 MANUAL SCOPE

This manual provides an explanation of the PDP-11/24 system as well as system installation, checkout, and troubleshooting information. The manual primarily explains the central processing unit (CPU) operation. Although there is some component description, the manual does not detail the operation of all components in the system. For more details on individual components, the reader should refer to the appropriate manual(s) in the table of related documentation (Table 1-1).

The documents listed in Table 1-1 are available from the locations listed in the following paragraphs.

Table 1-1 Related Documentation

Title	Document Number
PDP-11 Processor Handbook	EB-19402-20
PDP-11 Peripherals Handbook	EB-18293-20
PDP-11 Terminals and Communication Handbook	EB-18251-20
PDP-11 Bus Handbook	EB-17525-20
M9312 Bootstrap/Terminator Module Technical Manual	EK-M9312-TM
MS11-L MOS Memory User's Guide	EK-MS11L-UG
MS11-P MOS Memory Technical Manual	EK-MS11P-TM
BA11-L Technical Manual	EK-BA11L-TM
BA11-A Technical Manual	EK-BA11A-TM
M7133-YA Users Guide	EK-M7133-UG

1.1.1 DIGITAL Personnel Ordering

Additional copies of *this* document and printed copies of the documents listed may be obtained from:

Digital Equipment Corporation
444 Whitney Street
Northboro, Massachusetts 01532
ATTN: Printing and Circulation Services (NR2/M15)
Customer Services Section

1.1.2 Customer Ordering Information

Purchase orders for supplies and accessories should be forwarded to:

Digital Equipment Corporation
P.O. Box CS2008
Nashua, New Hampshire 03061

Contact your local sales office or call DIGITAL Direct Catalog Sales toll-free 800-258-1710 from 8:30 a.m. to 5:00 p.m. eastern standard time (U.S. customers only). New Hampshire, Alaska, and Hawaii customers should dial (603)-884-6660. Terms and conditions include net 30 days and F.O.B. DIGITAL plant. Freight charges will be prepaid by Digital Equipment Corporation and added to the invoice. Minimum order is $35.00. Minimum does not apply when full payment is submitted with an order. Checks and money orders should be made out to Digital Equipment Corporation.

1.2 SYSTEM OVERVIEW

A block diagram of a typical PDP-11/24 system configuration is provided in Figure 1-1. Most of the components and peripherals communicate with each other on the UNIBUS bus. The UNIBUS bus is a high-speed bus which transfers addresses, data, and control information. An additional bus, Extended UNIBUS (EUB) bus, is provided between the CPU and memory. The EUB allows for the rapid transfer of 22-bit address information between the CPU and memory. Each device, including memory locations, processor registers, and peripheral device registers, is assigned an address on the UNIBUS bus. This allows all peripheral registers to be manipulated as easily as memory locations by the CPU.

1.3 PDP-11/24 CENTRAL PROCESSING UNIT

Figure 1-2 provides a block diagram of the PDP-11/24 central processing unit (KDF11-UA). The KDF11-UA unit is a hex-height, multilayer module [21.6 cm × 38.1 cm (8.5 in × 15.0 in)] which contains the following features:

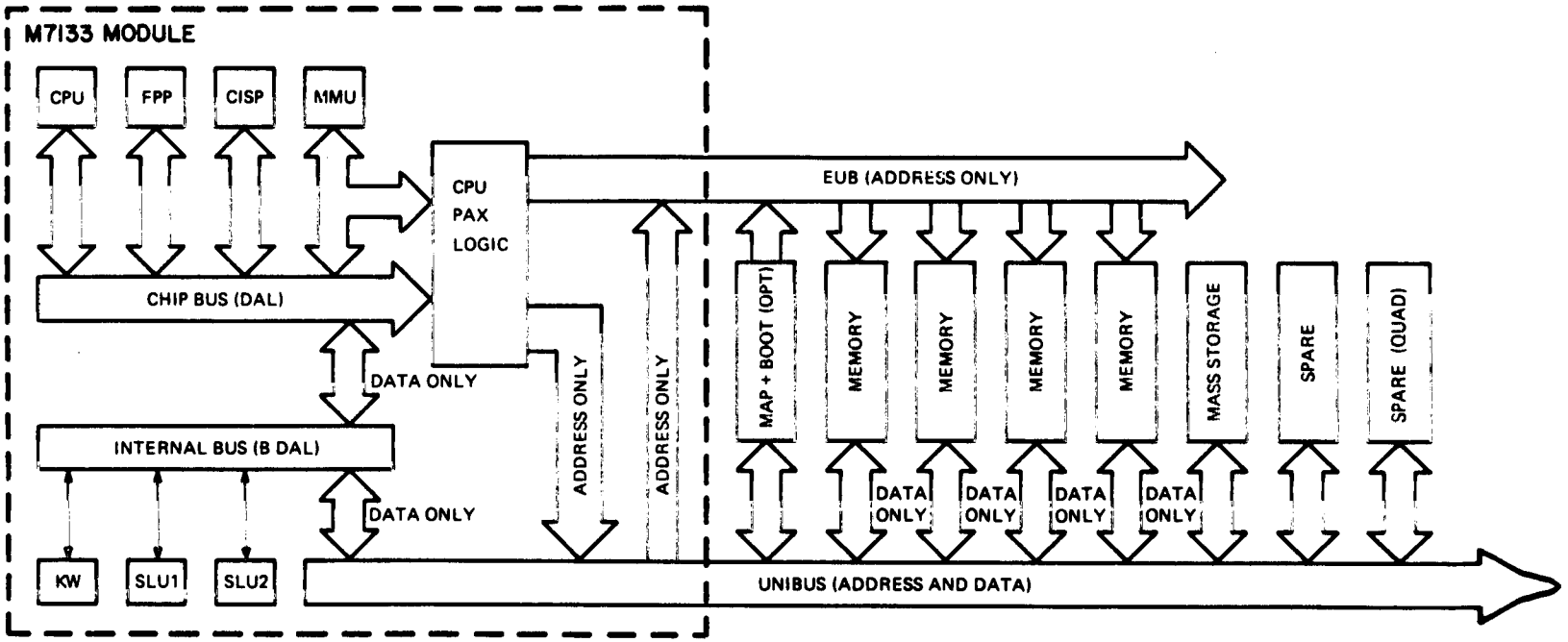
- Central Processor Base Machine
- Memory Management Unit (MMU)
- Floating-Point Processor (FPP) Option
- Commercial Instruction Set Processor (CISP) Option
- Two Serial Line Interfaces (SLU)
- Line Frequency Interrupt Real-Time Clock
- Extended UNIBUS Bus (EUB)

The central processor base machine contains the central processor base machine hybrid, data chip, and control chip mounted on one 40-pin package, as well as support logic contained on the module such as buffers and the physical address extension (PAX) latch. The central processor base machine hybrid contains the PDP-11 register file, the ALU, and microprogram storage. The MMU chip contains all the logic necessary for address relocation, as well as 16-bit working registers and 64-bit accumulators necessary for the execution of floating-point instructions. The remaining floating-point logic is contained on the floating-point processor control hybrid. An additional hybrid containing six chips is used for the execution of CIS instructions. The two major buses which provide for te transfer of information within the CPU are the microinstruction bus (MIB) and the data/address lines (DAL).

The MIB contains either control and status information or microinstructions from the control store. The DAL contains either data/address information or service information. The type of information present on the MIB and DAL is specified by the time state (phase time and phase bar time) the CPU is in. A more detailed explanation of the CPU time states and internal buses is provided in Chapter 4. In addition to the MIB and DAL, another information bus is included in the CPU: the buffered data/address lines (BDAL) is used for the transfer of data information. It provides a buffered data bus which transfers information between the DAL, line time clock, serial line units, and the UNIBUS bus. Each section of the CPU is explained in the following paragraphs. (Refer to Appendix D for the KDF11UA, M7133-YA version of unit assembly and circuit schematic.)

1.3.1 CPU Data and Control Chips

The CPU data chip contains the PDP-11 register file, the ALU, and conditional branching logic. The data chip performs all arithmetic and logical functions, handles all data/address transfers (except relocation) with the external buses, and operates most of the signals used for interchip communication and external system control. The control chip contains the microprogram and most of the microcontrol logic used in the PDP-11/24 processor. Both the CPU data chip and control chip are mounted on a single 40-pin, multilayer, ceramic hybrid IC.



MAJOR BUSES SHOWN:

1. DAL - COMMUNICATIONS BETWEEN CHIPS; COMMUNICATIONS TO EXTERNAL DEVICES
2. BDAL - BUFFERED CHIP BUS
3. EUB - EXTENDED UNIBUS - A DIRECT 22-BIT ADDRESS PATH TO MEMORY
4. UNIBUS - EXTERNAL BUS TO I/O DEVICES

TK-

Figure 1-1 PDP-11/24 System Block Diagram

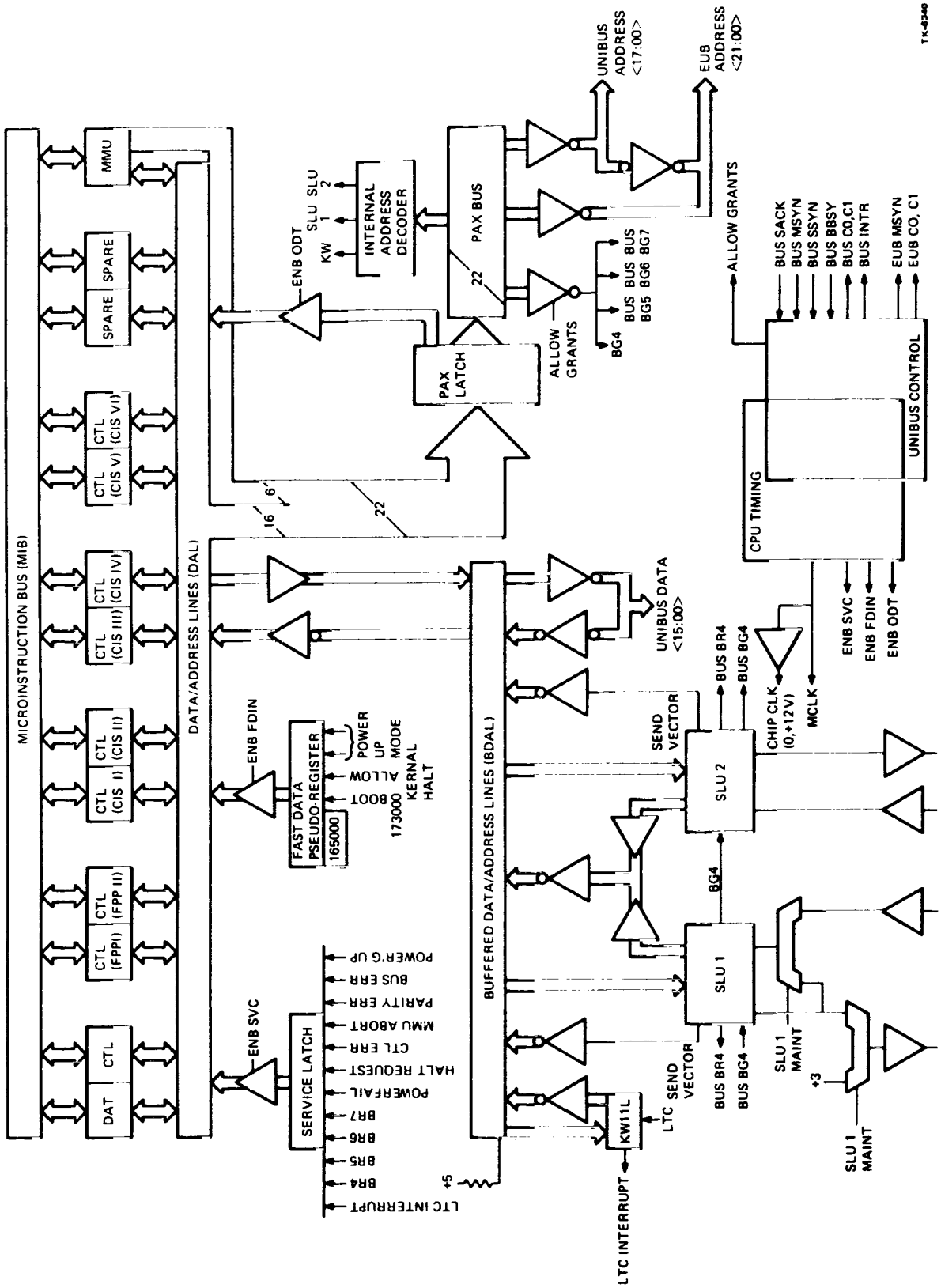


Figure 1-2 PDP-11/24 Central Processing Unit

1.3.2 Memory Management

The KDF11-UA CPU also contains a memory management unit (MMU) which maps 16-bit virtual addresses to the appropriate 18-bit or 22-bit physical addresses. The mapping scheme takes place automatically and is transparent to the user. The MMU contains the necessary status registers and active page registers (PAR/PDR register pairs) as well as access error detection capability. The MMU chip also contains the registers needed for storage of various information for floating-point operations.

The MMU is useful in a multiprogramming environment. The processor can operate in two modes: kernel and user. When in the kernel mode, the program has complete control and can execute all instructions. Monitors and supervisory programs are executed in this mode. The user mode allows memory protection in a multiuser operating system. This prevents the program from executing instructions which could modify the kernel program, halt the computer, or use memory space assigned to the kernel or to other users. Chapter 5 provides a complete description of the memory management system as implemented by the PDP-11/24 processor.

1.3.3 Operator's Console

The operator's console provides the communication link between the user and the processor. Although the PDP-11/24 processor does not have the traditional "lights and switches" console, octal debugging technique (ODT) exists as a portion of the microcode. This allows the processor to respond to commands and information entered via an ASCII terminal. The terminal emulates the capabilities found on "lights and switches" consoles. Communication between the user and the processor is via a stream of ASCII characters interpreted by the processor as console commands. Chapter 3 provides a detailed explanation of the console operations.

The actual PDP-11/24 console, however, does have a minimum number of switches and lights which perform basic functions and provide some system status information. Chapter 3 provides an explanation of the PDP-11/24 operator's console.

1.3.4 Serial Line Units

The serial line units (SLU1 and SLU2) provide two serial ASCII interfaces for the CPU. SLU1 is reserved for use by the system console. SLU2 may be used as a general purpose serial line unit. A typical application of this SLU is the connection of a TU58 DECtape II unit. The SLUs accept serial data from the console terminal and TU58 unit, convert it to parallel data, and transfer the data onto the BDAL. The SLUs also accept parallel data from the BDAL for serial conversion and transfer to the console terminal and TU58 unit. Both SLUs contain the registers necessary for data transfers. Paragraph 4.7 contains a detailed explanation of the SLUs and their associated logic.

1.3.5 Line Time Clock

The line time clock (LTC) is contained on the central processor module and provides the system with timing information at fixed intervals of 16.66 ms (60 Hz line) or 20.0 ms (50 Hz line). The PDP-11/24 line time clock is KW11-L compatible except that the "ready" bit is automatically cleared when the LTC interrupts. Accuracy is dependent on the accuracy of the line frequency. When the interrupt mode is enabled, an interrupt is generated for each cycle of the line frequency. When in the noninterrupt mode, a monitor bit in the status register is tested under program control. Each time the program notes that the bit has been set, it must clear the bit and note the passage of time. A description of the line time clock is provided in Paragraph 4.8.

1.3.6 Bootstrap

The PDP-11/24 processor contains either an M9312 bootstrap/terminator module or the optional UNIBUS map module (M7134) which contains equivalent logic. Either module contains 512 words of read-only memory (ROM) which can be used for diagnostic routines and bootstrap programs. Five sockets allow the user to interchange ROMs, enabling various peripheral devices to be bootstrapped. Diagnostics and the different bootstrap programs are selectable through the offset switchbank located on each module.

Bootstraps may be initiated in three ways:

1. By pressing the BOOT switch on the operator's console.
2. By system powerup.
3. By programmer's console "load address and start" sequence (whereby the programmer loads the starting address of a particular bootstrap routine).

Refer to the *M9312 Bootstrap/Terminator Technical Manual* for a complete description of the M9312 module. The UNIBUS map option (M7134) is described in Chapter 6 of this manual.

1.3.7 UNIBUS Termination

The UNIBUS bus is a transmission line and must be terminated at both ends. The CPU module provides the termination for one end of the UNIBUS bus and either the M9312 bootstrap/terminator module (in versions of the PDP-11/24 processor *without* the optional UNIBUS map module) or an M9302 terminator module (in versions of the PDP-11/24 processor *with* the UNIBUS map option) must be installed at the other end of the UNIBUS bus to complete the termination.

1.3.8 MS11-L MOS Memory System

The MS11-L metal-oxide semiconductor (MOS), random-access memory (RAM) is designed to be used with the PDP-11 UNIBUS bus or special buses with additional addressing capability such as the 22-bit EUB used in the PDP-11/24 processor. The MS11-L provides storage for 18-bit words (16 data bits and 2 parity bits). There are two versions of the MS11-L (Table 1-2), which differ only by the total memory capacity available on the module. The maximum memory capacity of the PDP-11/24 processor is determined by both the packaging of the system and the options installed (refer to Table 1-3).

The MS11-L consists of a single, hex-height module (M7891) that contains the UNIBUS interface, timing and control logic, refresh logic, and an MOS storage array. The module also contains logic to generate and check parity, and a control and status register (CSR).

A complete description of the MS11-L MOS memory is provided in the *MS11-L MOS Memory User's Guide*.

Table 1-2 MS11-L/P Versions

Option Designation	Module Designation	Storage Capacity
MS11-LB	M7891-BA	128K bytes (64K words × 18-bit)
MS11-LD	M7891-DA	256K bytes (128K words × 18-bit)
MS11-PB	M8743-BA	1024K bytes (512K words × 18-bit)

Table 1-3 Memory Capacity

Version of PDP-11/24 Processor	Maximum Capacity	Total Number of Memory Modules*
PDP-11/24* processor without KT24 option, mounted in BA11-L or BA11-A box.	256K bytes (128K words)	1
PDP-11/24* processor with KT24 option, mounted in BA11-L box.	768K bytes (384K words)	3†
PDP-11/24‡ processor with KT24 option, mounted in BA11-L box.	1024K bytes (512K words)	1
PDP-11/24* processor with KT24 option, mounted in BA11-A box.	1024K bytes (512K words)	4
PDP-11/24‡ processor with KT24 option, mounted in BA11-A box.	4096K bytes (2048K words)	4

* Based on MS11-LD.
† Three modules maximum in any combination of MS11-L memories.
‡ Based on MS11-PB.

1.3.9 MS11-P MOS Memory System

The MS11-P is a metal-oxide semiconductor (MOS), random-access memory (RAM) providing 512K words × 16-bit data storage. The MS11-PB is designed for use with the PDP-11 extended UNIBUS bus. Table 1-2 gives the total memory capacity of each module while Table 1-3 gives the memory capacity in packaged systems.

The MS11-P consists of a single-height M8743 module containing the extended UNIBUS interface, timing and control logic, error correction code (ECC) logic, and an MOS storage array. The module also contains circuits for ECC initialization, memory refresh, and a control and status register (CSR).

A complete description of the MS11-P memory is provided in the *MS11-P MOS Memory Technical Manual* (EK-MS11P-TM-001).

1.3.10 Floating-Point Processor

The KEF11-A floating point processor (FPP) is a microcode option for the KDF11-UA processor. Floating-point instructions supplement integer arithmetic instructions such as MUL and DIV which are contained in the standard PDP-11/24 instruction set. The FPP allows floating-point operations to be executed five to ten times faster than equivalent software routines and conserves memory space. Floating-point instructions provide for both single-precision (32-bit) and double-precision (64-bit) operands. The PDP-11/24 FPP implements the same floating-point instruction set found on the PDP-11/34, PDP-11/44, PDP-11/70, and other PDP-11 processors. A complete description of the FPP and floating-point instructions is provided in Chapter 7.

1.3.11 Commercial Instruction Set Processor

The KEF11-B commercial instruction set processor (CISP) is a microcode option for use with the PDP-11/24 processor. The CISP significantly extends the capability of the PDP-11/24 processor in the area of commercial data processing. The commercial instruction set (CIS) is a series of instructions for manipulating byte strings and performing decimal arithmetic. These instructions provide improved COBOL performance, text editing, and word processing capabilities. The KEF11-B processor is completely software compatible with the KE44-A CISP used with the PDP-11/44 processor. A complete description of the PDP-11/24 CISP is provided in Chapter 8.

1.4 GENERAL-PURPOSE REGISTERS

The data chip contains eight, 16-bit, general-purpose registers that provide for a variety of functions. These registers can serve as accumulators, index registers, autoincrement registers, autodecrement registers, or as stack pointers for temporary storage of data. Arithmetic operations can be freely mixed from general-purpose registers, memory locations, or device registers. Figure 1-3 identifies the eight, 16-bit general-purpose registers R0 through R7.

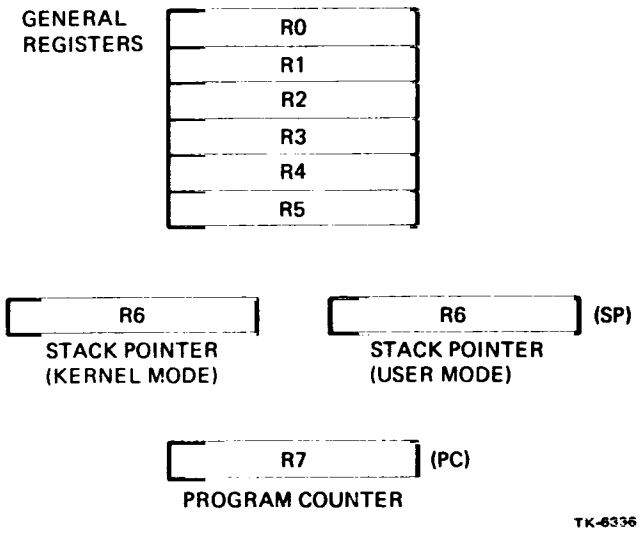


Figure 1-3 PDP-11/24 General Registers

Registers R6 and R7 are dedicated. R6 serves as the hardware stack pointer (SP) and contains the location (address) of the last entry on the stack. Two stack pointers actually exist; the selection of which stack pointer is to be used is determined by the processor operating mode. Thus, the kernel and user programs have independent hardware stacks. Register R7 serves as the processor's program counter (PC) and contains the address of the next instruction to be executed. It is normally used for addressing purposes only and not as an accumulator. Register operations are internal to the processor and do not require bus cycles (except for instruction fetch); all memory and peripheral device data transfers do require bus cycles and longer execution time. Thus, general registers used for processor operations result in faster execution times.

1.5 ADDRESSING MEMORY AND PERIPHERALS

The KDF11-UA processor uses 16-bit data paths throughput. These same data paths are also used to construct operand and instruction addresses. Octal notation is used to describe information on the data paths.

A word is divided into a high byte and a low byte as shown in Figure 1-4.

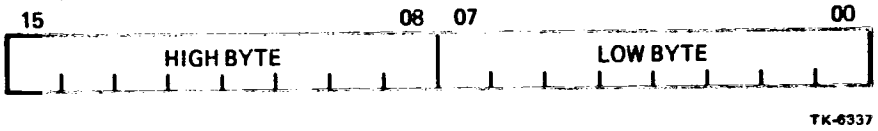


Figure 1-4 High and Low Bytes

Word addresses are always even numbered. Byte addresses can be either even or odd numbered. Low bytes are stored at even-numbered memory locations and high bytes at odd-numbered memory locations. Thus, it is convenient to view memory in either of the ways shown in Figure 1-5.

The full 16-bit data path allows a program to specify operand addresses (that is, virtual addresses) anywhere within a 64K byte range or 32K word range. This virtual address range is fixed by the instruction format and cannot be changed by the user.

For applications that require more than 32K words of physical address, such as multiprogramming and/or timesharing applications, the memory management unit (MMU) makes additional addressing bits available. These bits allow up to 1920K memory words to be physically addressed by the processor. This additional addressing capability is part of the standard memory management within the KDF11-UA architecture. Use of more than 128K words or memory in the PDP-11/24 processor requires the optional UNIBUS map module to be installed.

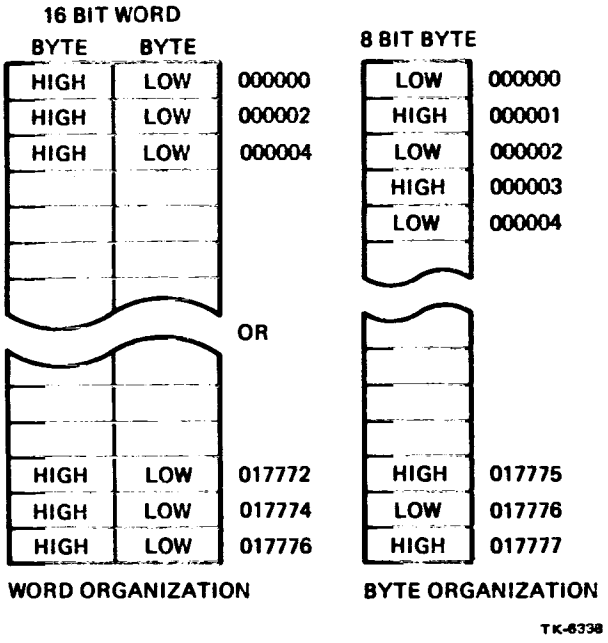


Figure 1-5 Word and Byte Addressing for the First 4K

1.6 ADDRESSING MODES

Much of the flexibility of the KDF11-UA processor is derived from its wide range of addressing capabilities. Addressing modes include sequential forward or backward addressing, address indexing, indirect addressing, absolute 16-bit word and 8-bit byte addressing, and stack addressing. Variable-length instruction formatting allows a minimum number of words to be used for each addressing mode. The result is efficient use of program storage space. For more details on addressing modes refer to the *PDP-11 Processor Handbook*.

1.7 INSTRUCTION SET

The KDF11-UA instruction set provides over 400 powerful instructions. As a comparison, consider that most other (that is, accumulator-oriented) 16-bit processors require three separate instructions to execute a common double-operand instruction (for example, ADD).

Conventional Approach:

LDA A	Load contents of memory location A into accumulator.
ADD B	Add contents of memory location B to accumulator.
STA B	Store result at location B.

By contrast, the KDF11-UA processor can fetch both operands, execute, and store the result in one instruction.

KDF11-UA Approach:

ADD A, B Add contents of location A to location B; store results at location B.

This greater efficiency not only saves memory space and time, but also improves processor speed since fewer instruction fetches are required.

Another major advantage to the KDF11-UA instruction set is the absence of special-purpose input/output instructions. Special I/O instructions are unnecessary since peripheral device registers are accessed in the same way as main memory locations. This approach to handling I/O devices allows the normal instruction set to be used to test and/or manipulate the various I/O device register bits. For example, a compare instruction can test status bits directly in the I/O device register without bringing them into memory or disturbing any of the general registers; control bits can be set, cleared, or shifted as is most convenient; and peripheral data can be arithmetically or logically altered when received at the device register and before being stored in memory. Refer to the *PDP-11 Processor Handbook* for a complete description of the instruction set and its utilization.

1.8 CONFIGURATION

The PDP-11/24 processor is contained in either the BA11-L [13.3 cm (5.24 in)] mounting box or the BA11-A [26.67 cm (10.50 in)] mounting box. Table 1-4 lists the basic processor configurations available. Table 1-5 lists the hardware options.

1.8.1 BA11-L Mounting Box

The BA11-L [13.3 cm (5.24 in)] mounting box contains the PDP-11/24 nine-slot backplane with the system modules installed, the H777-UA, -UB power supply, and fans to provide cooling for the modules and power supply. A front bezel is attached to the front of the mounting box and contains the operator's control panel and ventilating slots to allow the entry of air.

Table 1-4 Basic Processor Configurations

Designation	Description
PDP-11/24-AA, -AB*	Contains a CPU, 128K bytes of MS11-L MOS memory, M9312 bootstrap, and a BA11-L mounting box.
PDP-11/24-AC, -AD*	Contains a CPU, 256K bytes of MS11-L MOS memory, M9312 bootstrap, and a BA11-L mounting box.
PDP-11/24-CC, -CD*	Contains a CPU, 1M byte of MS11-P MOS memory, M9312 bootstrap, and a BA11-L mounting box.
PDP-11/24-BC, -BD*	Contains a CPU, 256K bytes of MS11-L MOS memory, M9312 bootstrap, and a BA11-A mounting box.
PDP-11/24-DC, -DD*	Contains a CPU, 1M byte of MS11-P MOS memory, M9312 bootstrap, and a BA11-A mounting box.

* -AA, -AC, -CC, -BC, and -DC designate 120 Vac versions.
-AB, -AD, -CD, -BD, and -DD designate 240 Vac versions.

Table 1-5 Hardware Options

Option	Description
KEF11-A	Floating-point processor (LSI chip package)
KEF11-B	Commercial instruction set processor (LSI chip package)
KT24	UNIBUS map/bootstrap module
M9302	UNIBUS terminator module
MS11-L	MOS memory
MS11-P	MOS memory

1.8.2 BA11-A Mounting Box

The BA11-A [26.67 cm (10.50 in)] mounting box contains the PDP-11/24 nine-slot backplane with the system modules installed, the H7140-AA, -AB power supply, and a fan assembly. The fan assembly contains three fans and provides cooling for the modules and power supply. A front bezel is attached to the front of the mounting box and contains the operator's control panel and ventilating slots to allow the entry of air. In addition to the PDP-11/24 backplane, the BA11-A mounting box has the space to mount an additional four system units of equipment. The system units may be backplanes for specific controllers (for example, DH11, RH11, or RK611) or general-purpose backplanes (for example, DD11-CK or DD11-DK).

1.8.3 System Configuration

Two hardware configurations are available as packaged systems (Figure 1-6). Each system is available with a variety of software packages.

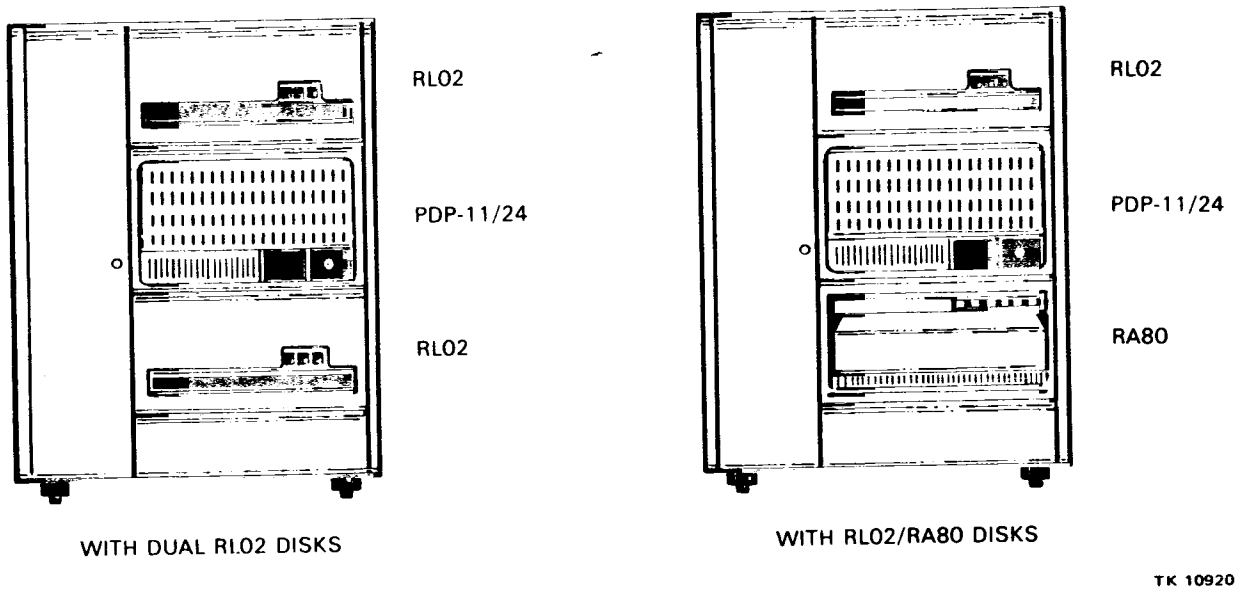


Figure 1-6 PDP-11/24 Packaged Systems in an H9645 Cabinet

RL02 Packaged System (SX-FXMMB-EA/EB)
This system contains a PDP-11/24 processor in a 26.67 cm (10.50 in) BA11-A mounting box, and two RL02 cartridge disk drives that together provide 20M bytes of removable hard-disk storage. This system is packaged in a 101 cm (39.76 in) high × 69.55 cm (27.38 in) wide × 76.2 cm (30 in) deep cabinet.

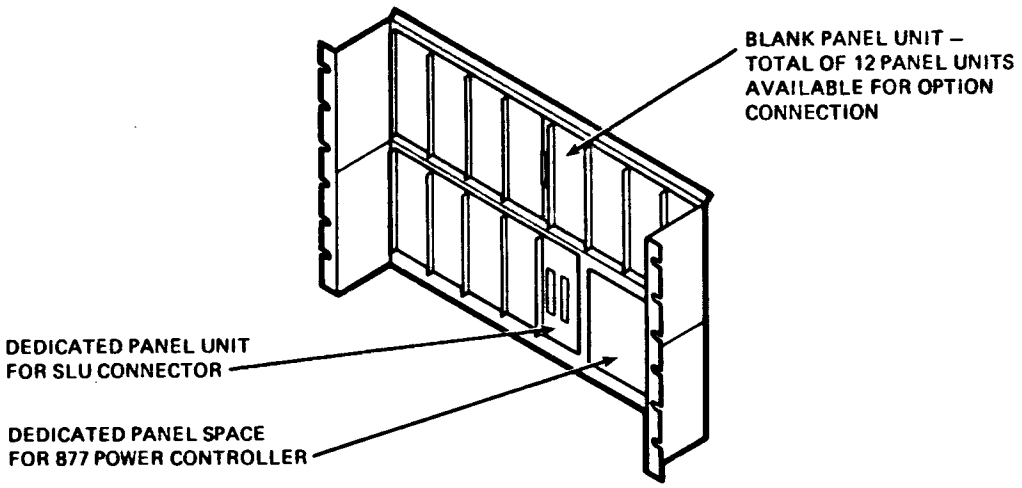
RA80/RL02 Packaged System (SX-FXGMB-EK/EN)
This system contains a PDP-11/24 processor in a 26.67 cm (10.50 in) BA11-A mounting box, one RA80 disk drive, and one RL02 disk drive. This configuration provides 121M bytes of fixed storage, 10M bytes of removable hard-disk storage, and a capability of incorporating 1M byte of main memory. This system is packaged in a 101 cm (39.76 in) high × 54.29 cm (21.38 in) wide × 76.2 cm (30 in) deep cabinet.

The I/O connector panels (Figure 1-7) for the single-bay and wide-body cabinets used for FCC-compliant PDP-11/24 systems, provide the transition between internal cabling and the external shielded cabling to peripheral devices. All cables that enter or exit cabinets must pass through the IOCPs.

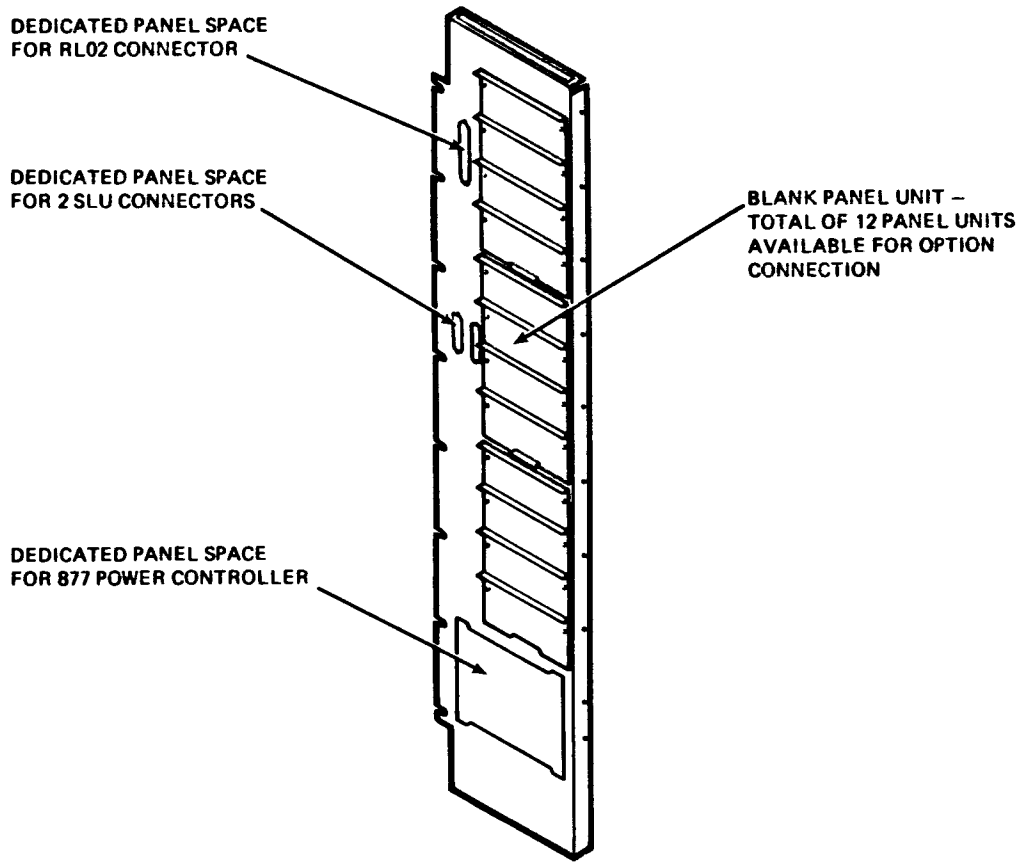
The I/O panel system is comprised of three components: an internal cable that originates at the option module or controller, a shielded external cable that attaches the I/O panel to the peripheral, and an I/O connector panel insert that mounts in the IOCP and joins the internal and external cables. With the IOCP, the connector insert provides the shielding and filtering necessary to contain potential interference within the cabinet.

The I/O panel is provided as part of the CPU system cabinet. The connector insert and internal cable are provided with the specific option.

The capacity of the I/O panel varies with the cabinet and CPU type. The type and style of inserts varies depending on the amount and type of connectors required by the option. Each I/O panel accepts multiple inserts. Unused I/O panel space is filled with blank inserts to maintain shielding continuity.



HORIZONTAL IOCP (STANDARD FOR H9642 CABINETS)



VERTICAL IOCP (STANDARD FOR H9645 CABINETS)

MKV84-0010

Figure 1-7 I/O Connector Panels (IOCPs) for H9642/H9645 Cabinets

1.9 PDP-11/24 BACKPLANE

The PDP-11/24 backplane provides the electrical connections between the modules in the system. The PDP-11/24 processor must use its own backplane. The DD11-CK, -DK, or -PK cannot be used to hold the PDP-11/24 processor module. The backplane consists of nine hex slots for module placement. The following guidelines must be followed for module placement (refer to Figure 2-25).

1. The CPU module (M7133) must be installed in slot 1.

2. M9312 bootstrap/terminator or UNIBUS cable must only be installed in slot 9.

3. NPR (DMA) options must be installed in slots 7 or 8. If installed in other slots, the NPG jumper wire from CA1 to CB1 must be removed.

4. Non-NPR options must be installed in slots 3, 4, 5, 6, or 9. If installed in other slots, a wire must be added to carry NPG. This wire should connect CA1 to CB1.

5. All empty slots, except slot 2, must have grant cards.

6. Slots 7 and 8 must have double-height grant cards (G7273) in rows C and D.

   Slots 3, 4, 5, 6, and 9 may use single-height grant cards (G727A or G7270) in row D or double-height grant cards in rows C and D.

1.10 SPECIFICATIONS

Table 1-6 lists the specifications for the PDP-11/24 processor. Tables 1-7 and 1-8 list the power (voltage and amps) supplied by the power supplies contained in the BA11-L and BA11-A mounting boxes. The voltages designated with a "B" are memory voltages and are backed up by the optional battery backup unit when it is installed in the system. Detailed specifications for peripheral devices supplied with these units are contained in the user's guide associated with the device.

Table 1-6 PDP-11/24 Specifications

Characteristic	Description
AC Power	
BA11-L Mounting Box	
PDP-11/24-AA, -AC, -CC	104–127 Vrms, 47–63 Hz, 1 phase power, 5 amps rms maximum at 120 Vac
PDP-11/24-AB, -AD, -CD	208–258 Vrms, 47–63 Hz, 1 phase power, 2.5 amps rms maximum at 240 Vac
BA11-A Mounting Box	
PDP-11/24-BC, -DC	90–128 Vrms, 47–63 Hz, 1 phase power, 16 amps rms maximum at 120 Vac
PDP-11/24-BD, -DD	180–256 Vrms, 47–63 Hz, 1 phase power, 9 amps rms maximum at 240 Vac
Mechanical	
Dimensions	
BA11-L	Box is 13.5 cm high × 42.21 cm wide × 69 cm deep (5.25 in × 16.62 in × 26 in). The BA11-L mounts in a standard 48 cm (19 in) rack.
BA11-A	Box is 26.3 cm high × 42.21 cm wide × 69 cm deep (10.35 in × 16.62 in × 25 in). The BA11-A mounts in a standard 48 cm (19 in) rack.
Weight	
BA11-L	20 kg (45 lb)
BA11-A	32 kg (70 lb)
Operating Environment	
The BA11-L and BA11-A mounting boxes with PDP-11/24 CPU have the same operating and nonoperating environment specifications.	
Temperature	5° C to 50° C (41° F to 122° F)
Humidity	10% to 95% with a maximum wet bulb of 32° C (89.9° F) and a minimum dew point of 2° C (36° F)
Altitude	to 2.4 km (8000 ft)
Nonoperating Environment	
Temperature	−40° C to 80° C (−40° F to 176° F)
Humidity	to 95% (noncondensing)
Altitude	to 9.1 km (30,000 ft)

Table 1-7 BA11-L DC Power

Regulator	Voltage	Amps
Main (70-12909)	+5 V	32 A
MOS (54-11601)	+5 B	4 A
	+15 B	2 A*
	−15 B	2 A*
COMM (54-13764)	+15 V	3 A
	−15 V	2 A

*The sum of these currents must not exceed 4 amps.

Table 1-8 BA11-A DC Power

Voltage	Amps
+5 V*	120 A
+15 V	2 A
−15 V	3 A
+5 B	10 A
+12 B	4 A
−12 B	1 A

*The main +5 V must be derated according to the +15 V and −15 V consumption:

$$I_{+5} = 120 - 5\,(I_{+15} - 1) - 5\,(I_{-15} - 1)$$

CHAPTER 2
INSTALLATION

2.1 GENERAL

This chapter provides the information necessary for site preparation, unpacking, inspection, and first-time start-up of the basic PDP-11/24 system.

2.2 SITE CONSIDERATIONS

The computer room environment should have an air distribution system that provides cool, well-filtered, humidified air. The room air pressure should be kept higher than that of adjacent areas to prevent dust infiltration.

Computer area environment can have a substantial effect upon the overall reliability of the system. Temperature cycling and thermal gradients induce temporary or permanent microscopic changes in materials that can affect performance or endurance. High temperatures tend to increase the rate of deterioration for nearly every material. High absolute humidity (dew point) causes moisture absorption that can result in dimensional and handling changes in paper and plastic media (line printer paper, cards, paper tape, magnetic tape, and so on).

Low humidity allows static electricity to build up, while lack of air cleanliness results in dust that reduces tape life and leads to excessive head wear and early data errors in all moving magnetic storage media (tapes and disks). This combination of static electricity and airborne dust is especially detrimental to magnetic tapes.

Vibration can also cause slow degradation of mechanical parts and, when severe, may cause errors on disks and tapes.

Hardware errors can also be caused by electromagnetic interference (EMI). EMI sources that have been known to cause failures include: radar installations, lightning strikes, power transmission lines, vehicle ignition systems, broadcast transmitters, arc welder, and so on.

2.2.1 Humidity and Temperature

The PDP-11/24 system is designed to operate in a temperature range of 5°C to 50°C (41°F to 122°F) at a relative humidity of 10 to 95 percent, as long as no condensation occurs. In system configurations that use magnetic tape units, card readers, disks, and so on, refer to the appropriate manual for their specifications. Nominal operating conditions for a system configuration are a temperature of 20°C (70°F) and a relative humidity of 45 percent.

2.2.2 Air-Conditioning

When used, computer room air-conditioning equipment should conform to the requirements of the Standard for the Installation of Air-Conditioning and Ventilating Systems (Non-Residential) N.F.P.A. No. 90A, as well as the requirements of the Standard for Electronic Computer Systems, N.F.P.A. No. 75.

2.2.3 Acoustical Damping

Some peripheral devices (such as line printers and magnetic tape transports) are quite noisy. In installations that use a group of high noise-level devices, an acoustically damped ceiling reduces the noise.

2.2.4 Lighting

If CRT peripheral devices are part of the system, the illumination surrounding these peripherals should be reduced to enable the operator to conveniently observe the display.

2.2.5 Special Mounting Conditions

If the system will be subjected to rolling, pitching, or vibration of the mounting surface (for example, aboard ship), the cabinetry should be anchored securely to the installation floor by mounting bolts. Since such installations require modifications to the cabinets, Digital Equipment Corporation must be notified when the order is placed so that the necessary modifications can be made.

2.2.6 Static Electricity

Static electricity can be an annoyance to operating personnel and can (in extreme cases) affect the operational characteristics of the PDP-11/24 system and related peripheral equipment. If carpeting is installed on the computer floor, it should be of a type designed to minimize the effects of static electricity. Flooring consisting of metal panels, or flooring with metal edges, should be adequately grounded.

2.3 ELECTRICAL REQUIREMENTS

The PDP-11/24 system can be operated from a 120/240 Vac, 47 to 63 Hz power source. The primary ac operational voltages should be within the defined tolerances. The primary power outlets at the installation site must be compatible with the PDP-11/24 primary power input connectors. Refer to the related BA11-L or BA11-A mounting box manual for details concerning power requirements.

The types of connectors used with the PDP-11/24 system depend on the mounting box (BA11-A, BA11-L) or cabinet configuration, and whether the system is configured for 120 Vac or 240 Vac operation. Figure 2-1 shows the connectors used with the BA11-A and BA11-L mounting boxes. Figure 2-2 shows the connectors used on the 874-D, -E power controller.

The PDP-11/24 three-prong power connector (when inserted into a properly wired receptacle) grounds the computer chassis. It is unsafe to operate the computer unless the case is grounded because normal current leakage from the power supply flows to the metal parts of the chassis. If the integrity of the ground circuit is questionable, the user is advised to measure the potential between the computer case and a known ground with a voltmeter, or to notify the appropriate Field Service representative.

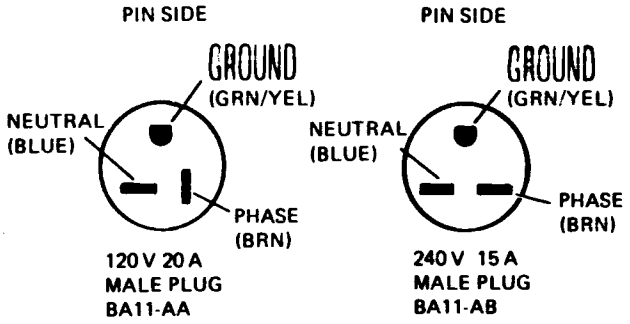
Computer systems are often sensitive to the interference present on some ac power lines. If the computer is to be installed in an electrically noisy environment, it is necessary to provide primary power to the computer on a separate power line from lighting or air-conditioning so that computer operation is not affected by voltage surges or fluctuations.

2.4 UNPACKING

The system equipment, associated devices, and cabinets are packaged and shipped in reinforced cartons and are protected internally by foam inserts and polyethylene bags. Accessories and supplies such as documentation, magnetic tape, or disks connecting cables and hardware are packaged in separate containers. Before unpackaging any carton, remove the packing list from the container and check to ensure that the items ordered are listed. When the items are unpackaged, use the list to check that all the items are contained in the package. The unpacking information for consoles, printers, disk drives, and magnetic tape is contained in the user's guide supplied with each device.

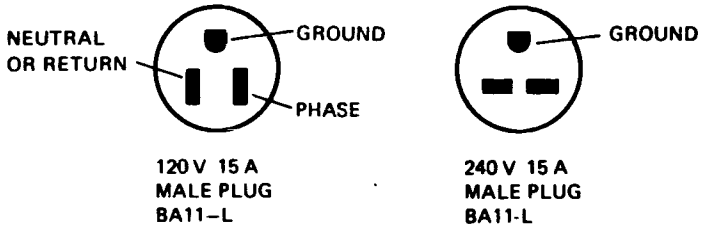
NOTE

Retain the packaging materials and shipping containers in the event that reshipping is required.



NEMA™ DESIGNATION	POWER RATING	DIGITAL PART NO.
5–20 P	120 V, 20 A	12–15183–00
5–20 R		12–12265–00 * *
6–15 P		90–08853–00
6–15 R	240 V, 15 A	12–11204–01 * *

*P = PLUG
R = RECEPTACLE
**DUAL RECEPTACLE OUTLET

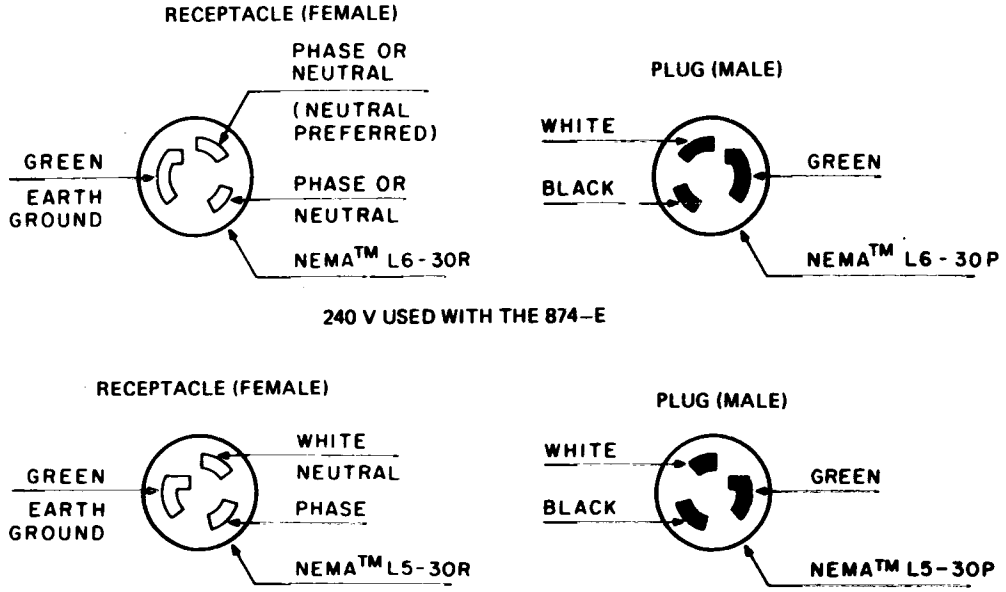


NEMA™* DESIGNATION	POWER RATING	DIGITAL PART NO.
5–15 P	120 V, 15 A	90–08938–00
5–15 R		12–05351–00 * *
6–15 P	240 V, 15 A	90–08853–00
6–15 R		12–11204–01 * *

*P = PLUG
R = RECEPTACLE
**DUAL RECEPTACLE OUTLET

TK-6452

Figure 2-1 Connector Specifications for BA11-A and BA11-L



CONNECTOR SPECIFICATIONS

MODEL NUMBER	POWER	RATING	PLUG NEMA CODE	RECEPTACLE (SUPPLIED BY CUSTOMER)	
				NEMA CODE	DEC PART NO.
874-D	120 V	30 A	L5-30P	L5-30R	12-11194
874-E	240 V	20 A	L6-30P	L6-30R	12-11191

TK-6461

Figure 2-2 874-D, -E Connector Specifications

2.4.1 PDP-11/24-AA, -AC, -BC Unit Removal

The units are packaged in reinforced cartons and are protected by foam inserts and by a polyethylene bag as shown in Figure 2-3. To remove the unit from the container perform the following procedure.

CAUTION

The PDP-11/24-BC unit weighs approximately 34 kg (75 lb). Use care when lifting the unit from the carton.

1. Open the leaves of the outer carton by cutting the tape at the seams.
2. Remove the two (2) top protectors.
3. Remove the unit from the box.
4. Remove the unit from the polyethylene bag.
5. Inspect the unit for visible damage and to ensure that the contents are complete.

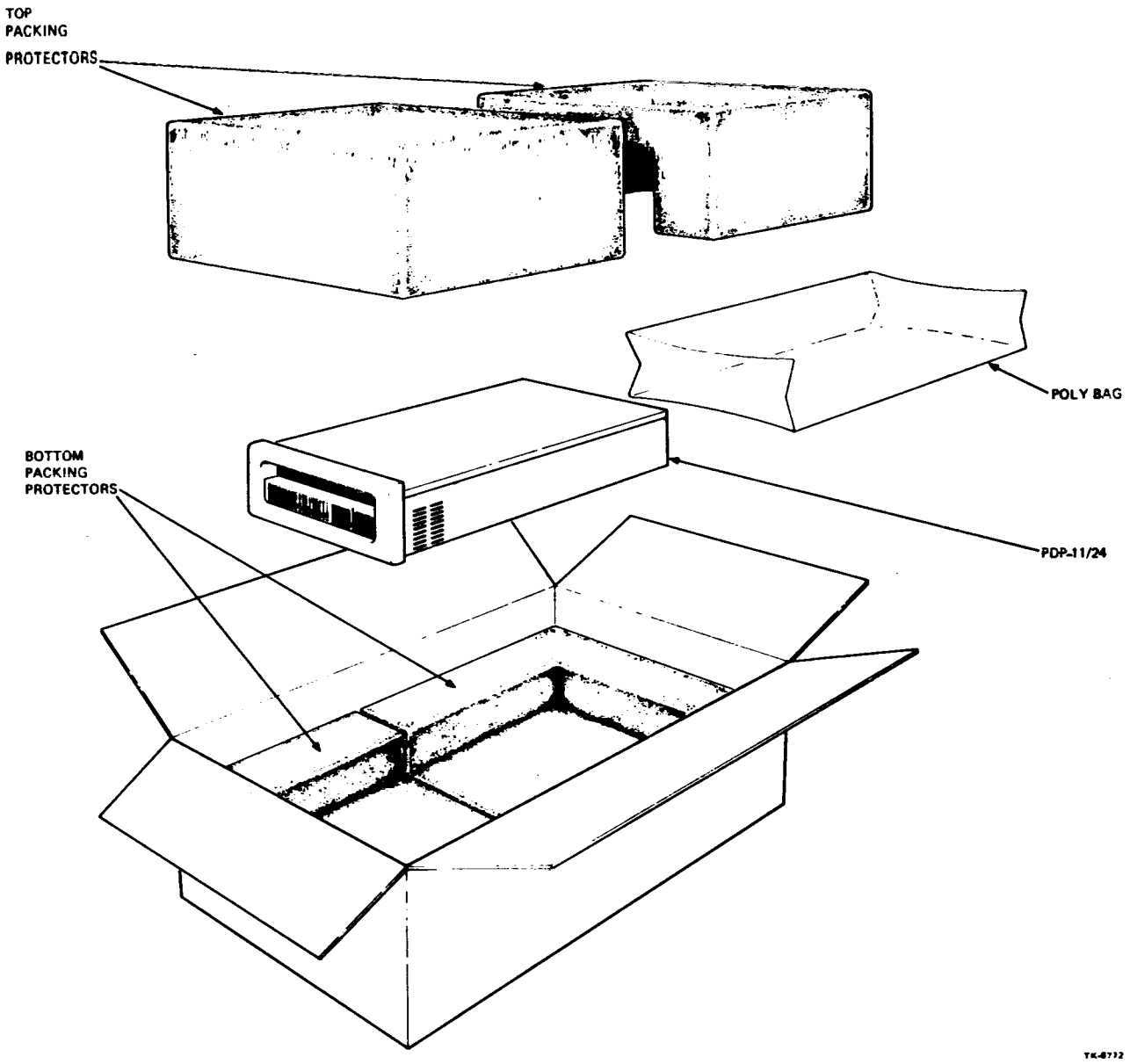


Figure 2-3 PDP-11/24-AA, -AC, -BC Unit Packaging

2.4.2 PDP-11/24 System Cabinet Removal

The PDP-11/24 system cabinets are attached to a wooden base, covered with a polyethylene bag, and enclosed by a carton as shown in Figure 2-4. To remove the unit perform the following procedure.

1. Cut the polyester straps used to secure the carton to the base.
2. Slide the carton up and away from the cabinet.
3. Remove the polyethylene bag from the cabinet.
4. Remove the bolts that hold the wooden base to the bottom of the cabinet.
5. Remove the cabinet from the wooden base and set the cabinet in its operating location.

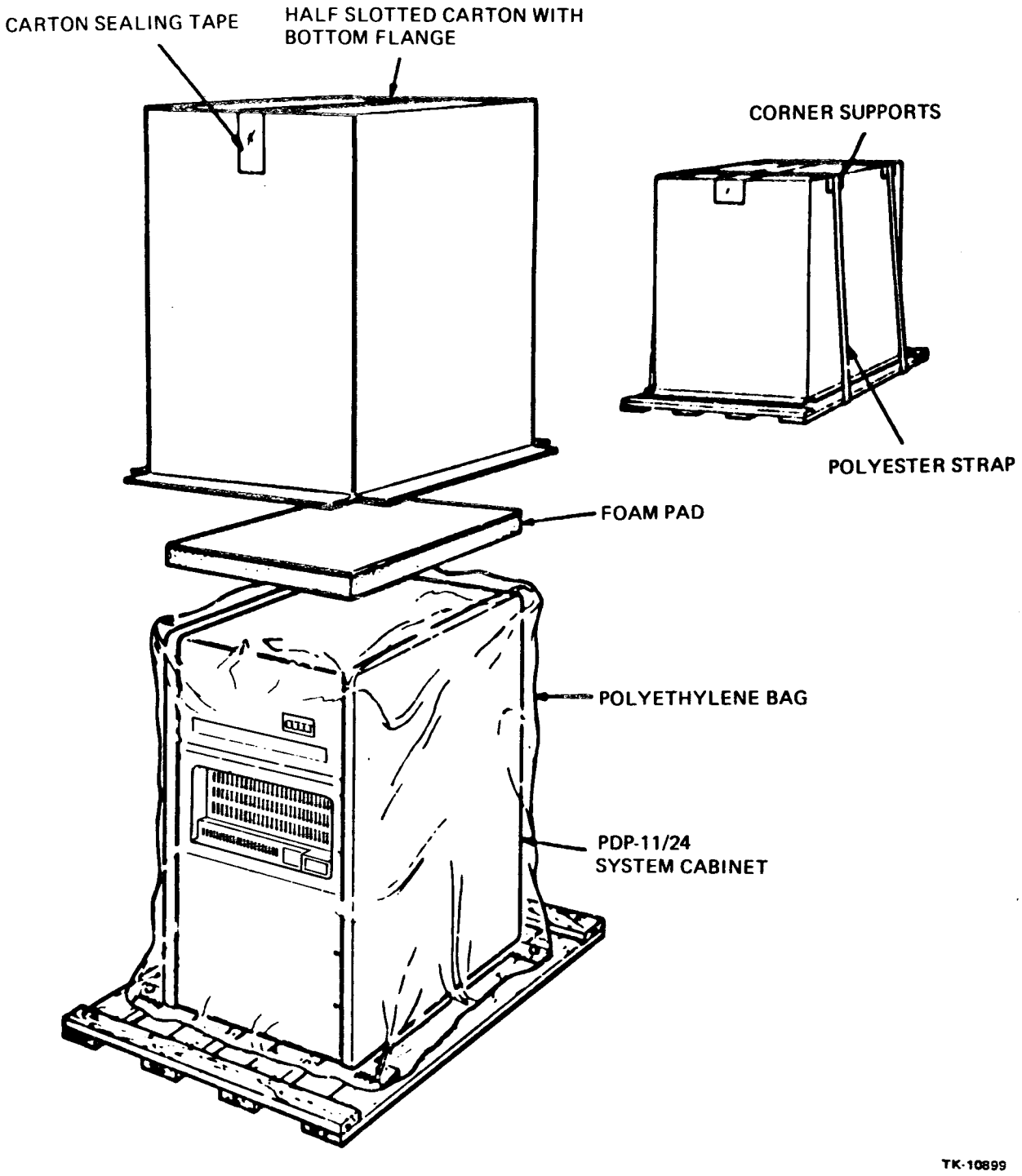


Figure 2-4 Unpacking a PDP-11/24 System

2.5 INITIAL INSPECTION

After unpacking the computer, extend the wire frame assembly containing the logic and power subassemblies for the BA11-L 13.3 cm (5.24 in) mounting box or remove the top and bottom covers on the BA11-A 26.67 cm (10.50 in) mounting box. Examine the following areas:

1. Check the overall appearance for scratches, dents, chipped paint, dust, and so on.
2. Check for loose or missing hardware (for example, screws and nuts).
3. Toggle the front panel switch and operate the keyswitch to make certain they operate freely and are unrestricted.
4. Examine the backplane for bent pins.
5. Check the power and console harness for proper connection to the power supply and front console.

2.6 INSTALLATION

2.6.1 BA11-L Installation

The BA11-L mounting box is designed to be installed in a standard 48.26 cm (19 in) rack or cabinet using its own integrated mounting envelope/slide assembly (Figure 2-5).

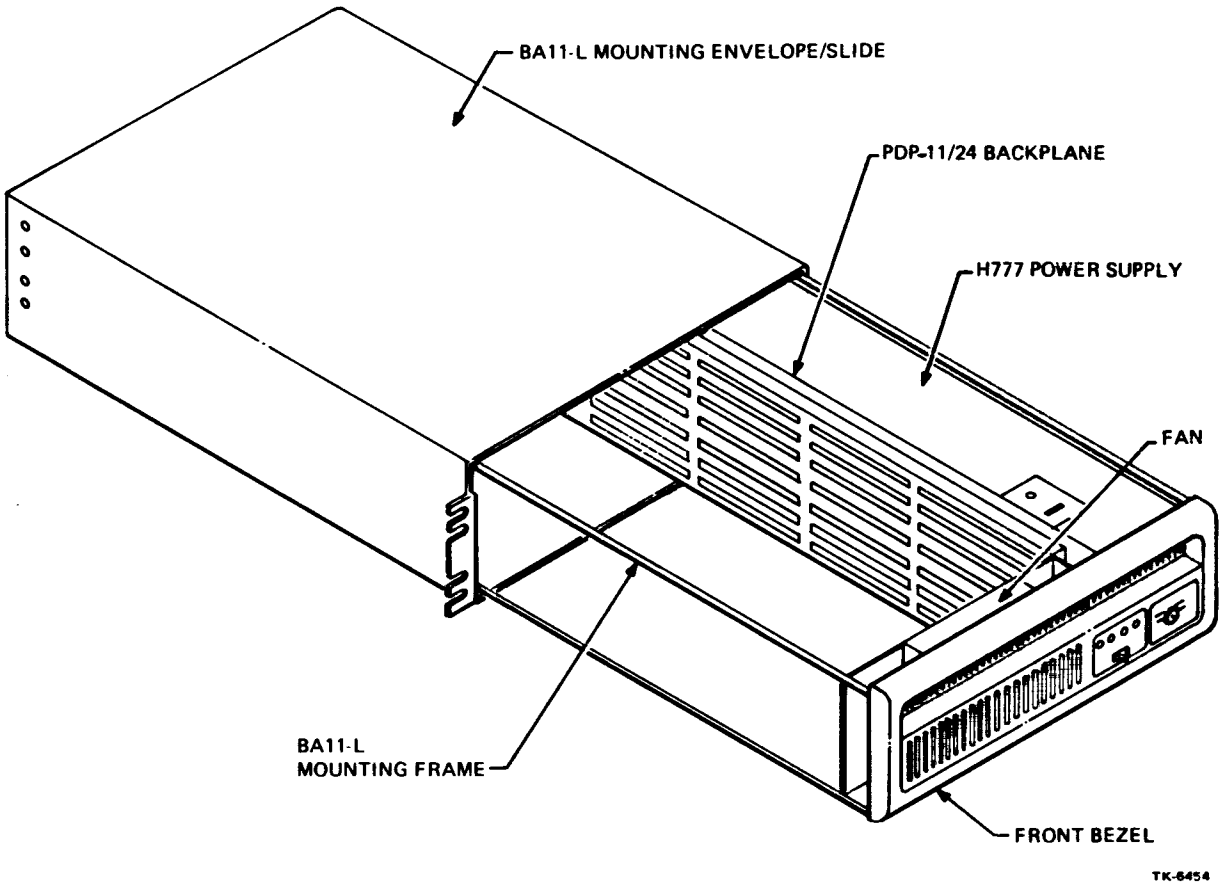


Figure 2-5 BA11-L Mounting Box

To install the BA11-L box perform the following steps.

1. Remove all shipping brackets.

2. Remove the mounting frame (Figure 2-5) from the mounting envelope. This is done by pulling the mounting frame forward until the safety catches are engaged. When the safety catches are engaged, press in on the locking tabs located on both sides of the mounting frame and remove the frame from the mounting envelope.

3. Locate and install the mounting envelope in the rack or cabinet as shown in Figure 2-6, using the 10-32 screws and KEP nuts supplied with the unit.

4. Slide the mounting frame into the mounting envelope. The safety catches must be pressed in so the wire frame can be inserted into the mounting envelope.

5. Attach the ground lead from the mounting frame to the rack or cabinet rail.

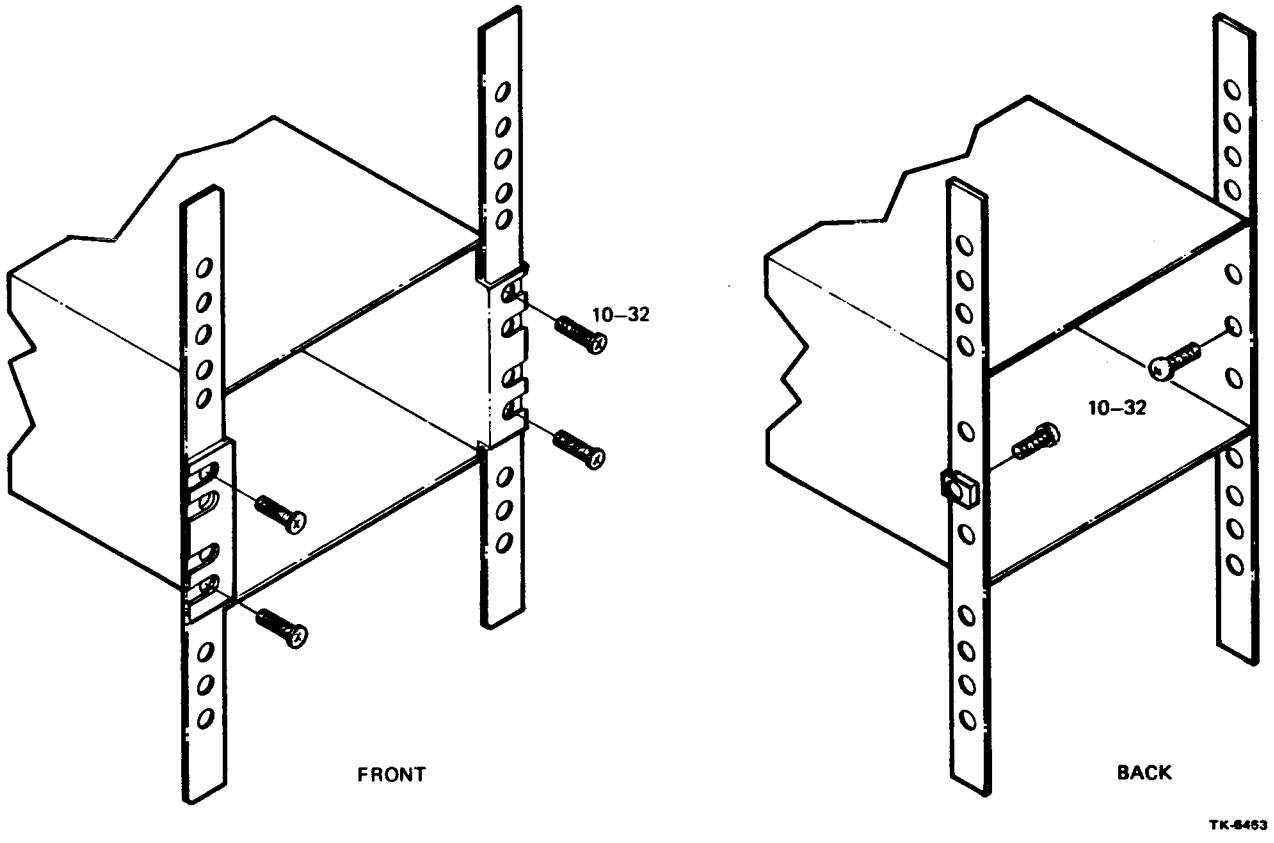


Figure 2-6 Mounting Envelope Installation

2.6.2 BA11-A Installation

The BA11-AA, -AB mounting box is designed to be installed within a standard 48.26 cm (19 in) rack or cabinet on slide-mounting assemblies (Figure 2-7).

A slide kit is available (part no. H9544-MG) and includes one each of the following items: left and right index plates and mounting hardware, and left and right slide assembly and mounting hardware.

The index plates supplied with the kit are to be mounted onto the sides of the BA11-AA, -AB mounting box and permit the box to be tilted on the slides for servicing.

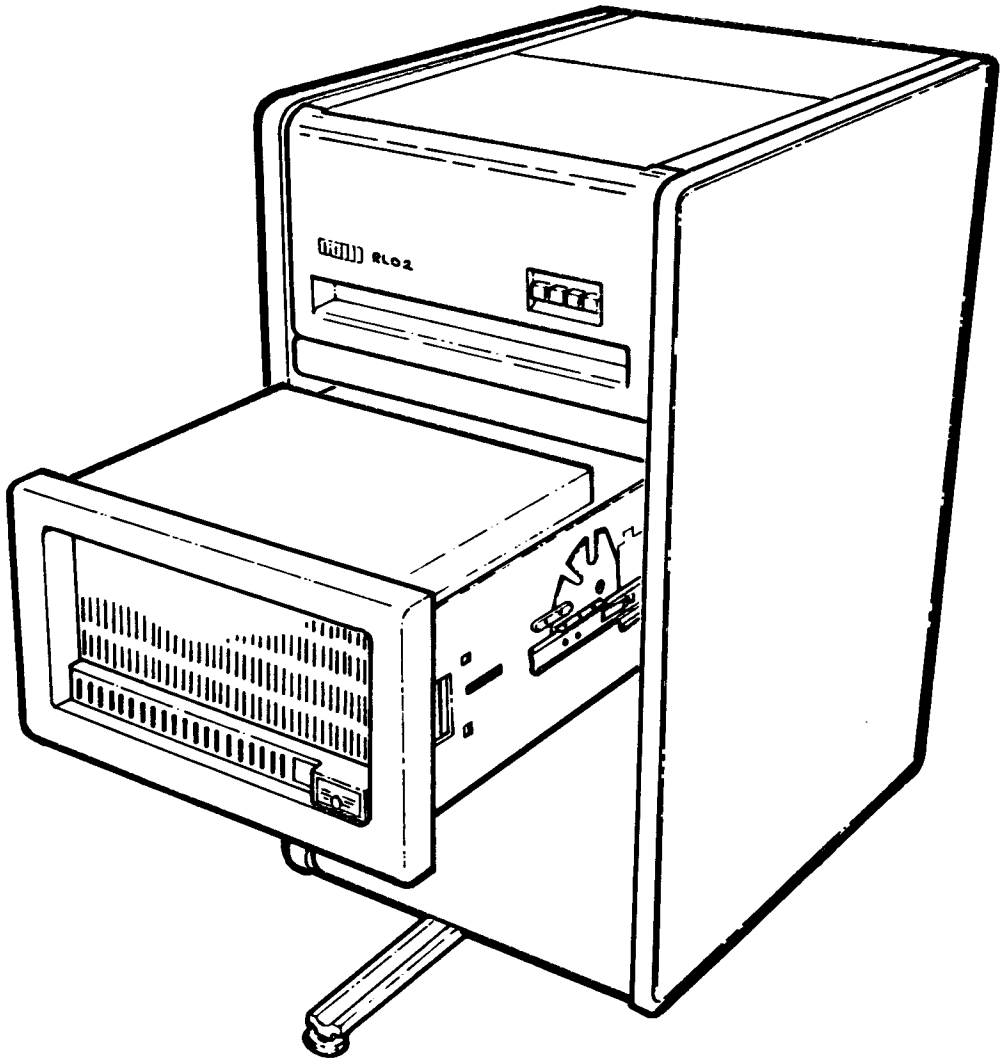


Figure 2-7 PDP-11/24 System – CPU Mounting Box Fully Extended

2.6.2.1 Index Plate Mounting – To install the index plates refer to Figure 2-8 and perform the following procedure.

1. Position the right index plate onto the pawl as shown. The index plate mounting tab protrudes away from the side of the box.
2. Insert the pivot screw and tighten with a screwdriver.
3. Ensure that the index plate rotates freely when the locking pawl is released.
4. Perform Steps 1 through 3 to install the left index plate.

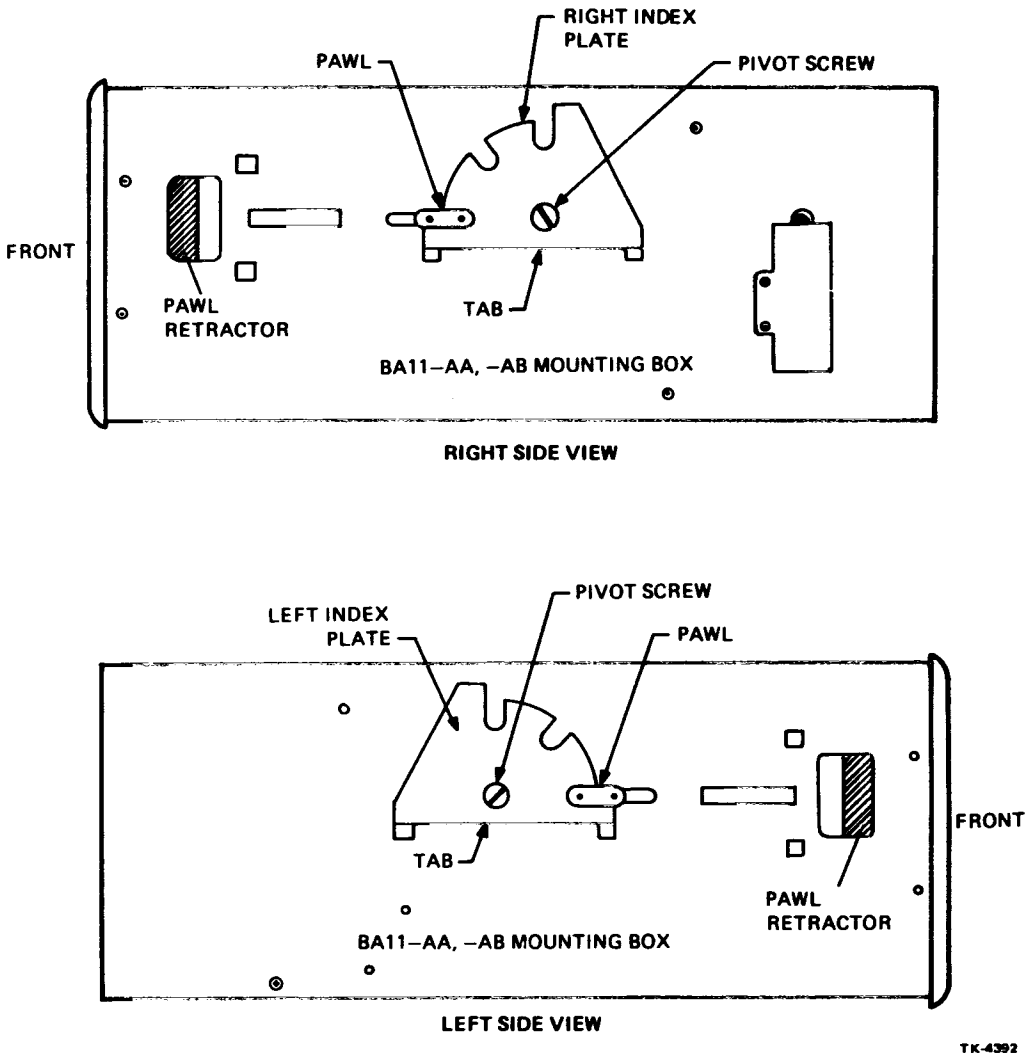


Figure 2-8 BA11-AA, -AB Mounting Box Index Plate Installation

2.6.2.2 Slide Assembly Mounting – Figure 2-9 shows a single-channel slide mounted to a BA11-A expander box that is fully extended from the cabinet. Figure 2-10 shows the mounting holes for the CPU of the two packaged systems using the PDP-11/24 system.

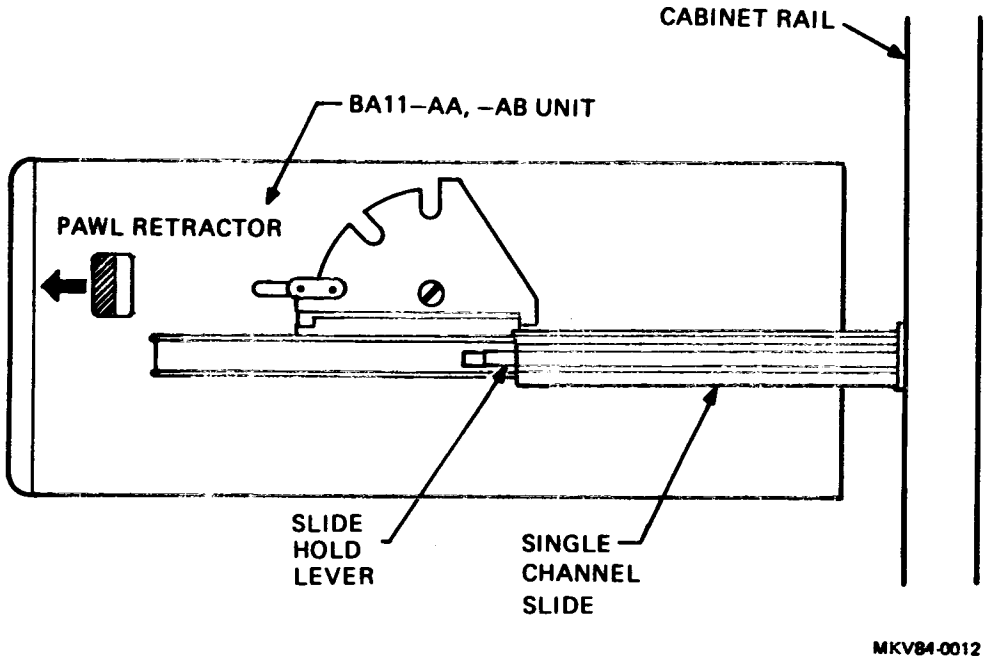


Figure 2-9 Single-Channel Slide Assembly

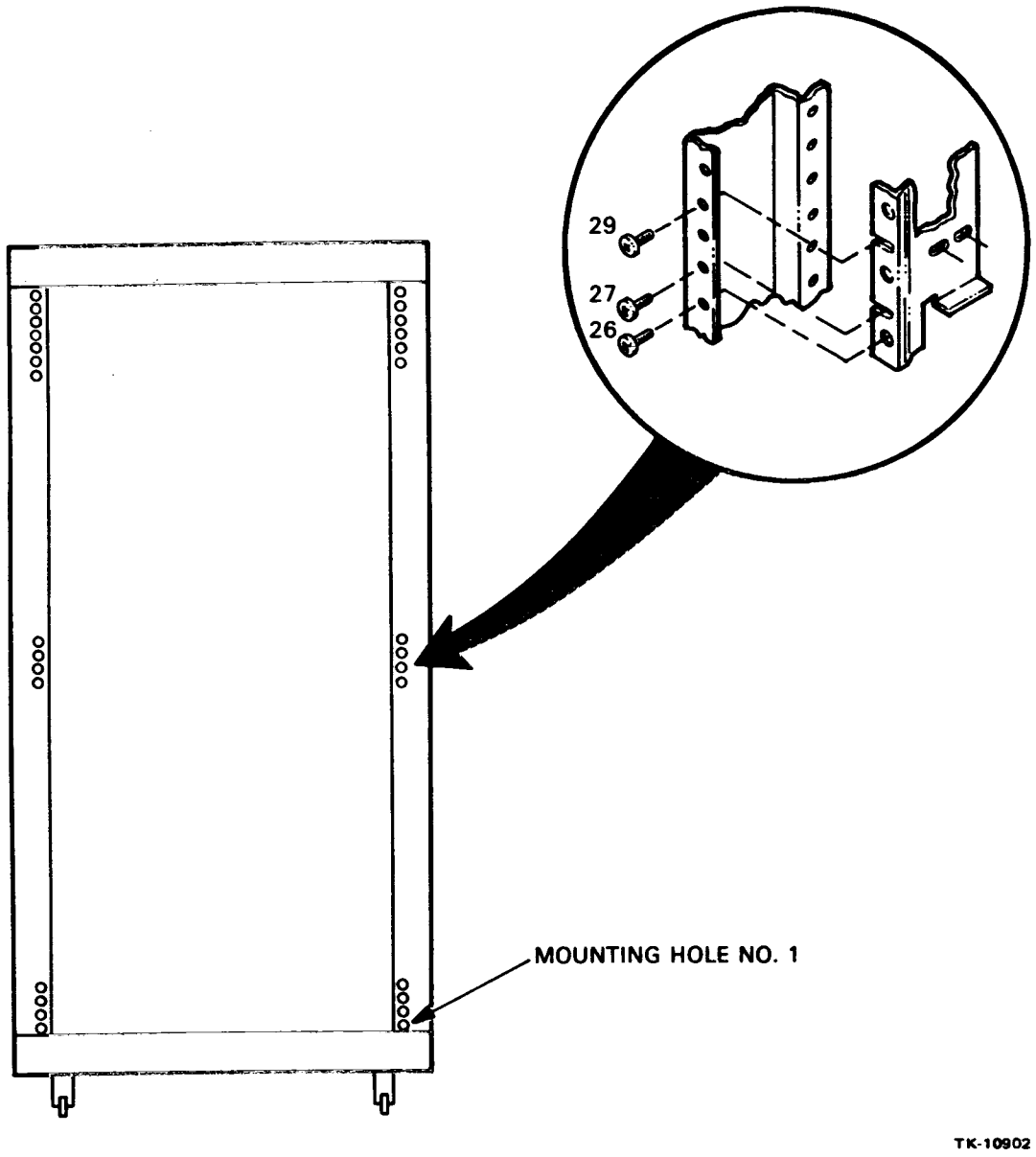


Figure 2-10 Designated Mounting Holes for Slide-Mounted PDP-11/24 CPUs Installed in H9642 (Single-Bay) or H9645 (Wide-Body) Cabinets

Figure 2-11 shows tne hardware and installation of a single-channel slide assembly. To install the assembly perform the following procedure.

1. Position the left slide against the left front and left rear cabinet rail as shown.
2. Insert one 10-32 screw and washer through the top hole in the slide bracket, through the hole in the front rail, and into the top threaded hole in the nut plate. Do not tighten.
3. Perform the same procedures in Step 1 and Step 2 at the left rear rail of the cabinet.
4. Insert one 10-32 screw and washer through the second hole from the bottom in the slide bracket, through the hole in the front rail, and into the nut plate. Tighten both screws in the front rail.
5. Perform the same procedure in Step 4 at the left rear rail of the cabinet.
6. Perform Steps 1 through 5 to install the remaining slide onto the right side of the cabinet.

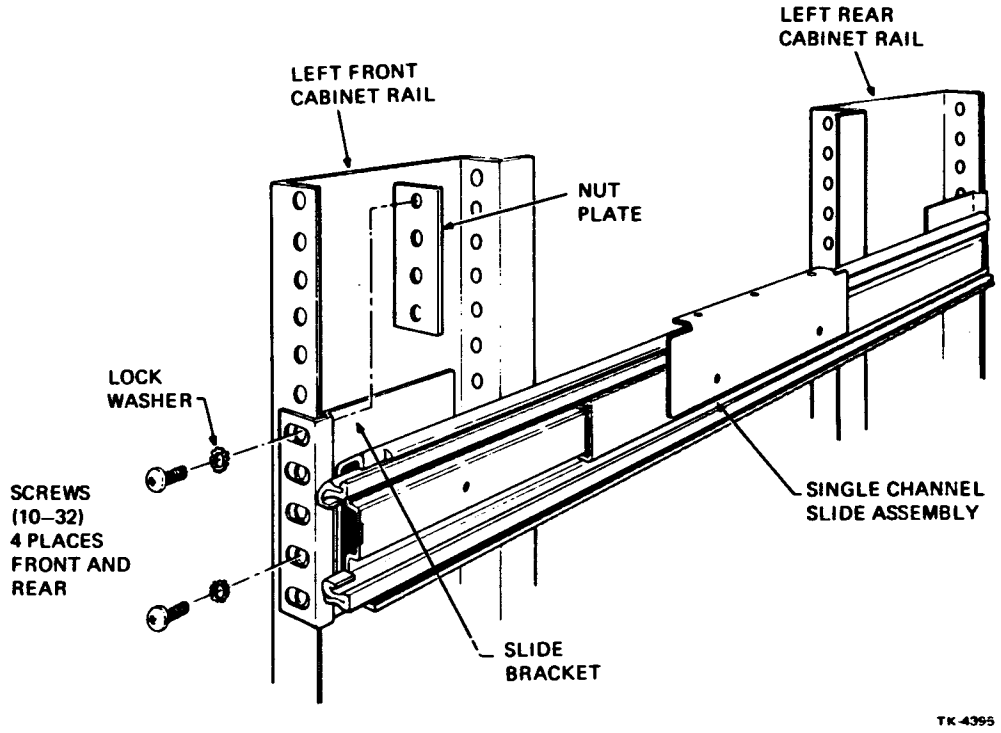


Figure 2-11 Cabinet Slide Installation

2.6.2.3 Mounting Box-to-Slide Installation – Figure 2-12 shows the method and hardware used to install the mounting box onto the slide mounting bracket. Perform the following procedure.

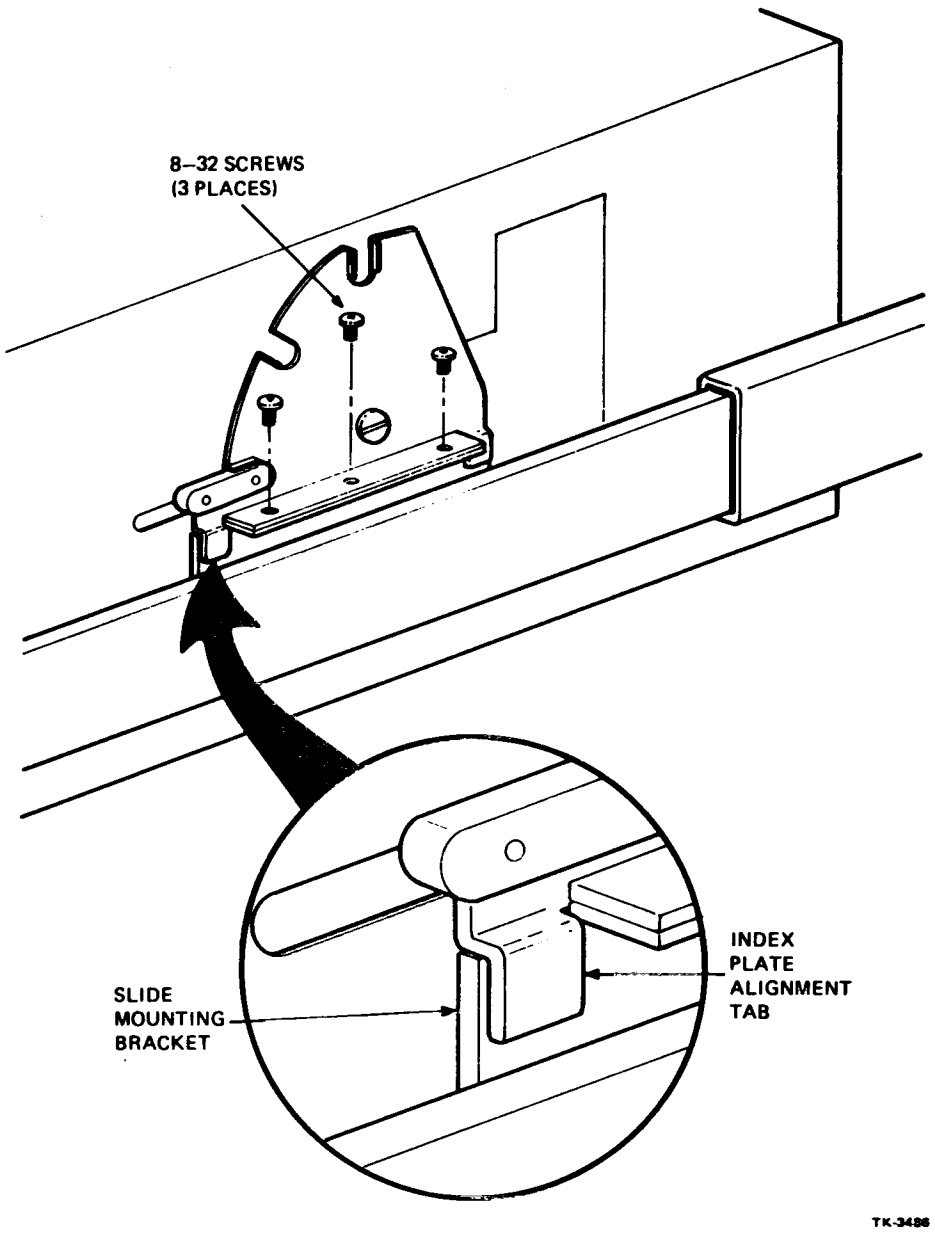


Figure 2-12 Mounting Box-to-Slide Installation

1. Extend the left and right slide channels to their maximum position at the front of the cabinet. When fully extended, the channels will be held in place by the slide hold lever shown in Figure 2-9.

2. Carefully lift the mounting box over and above the extended slides and set the index plate over the slide mounting bracket on each side of the box. The index plate alignment tabs will engage the sides of the slide mounting brackets.

NOTE

When the slides are fully extended, it may be necessary to force the ends of the slides inward toward the sides of the mounting box.

3. Insert the three 8-32 screws through the left index plate tab and into the threaded holes of the slide mounting bracket.

4. Perform the same procedure in Step 3 for the right index plate.

2.6.3 PDP-11/24 System Cabinet Installation

The PDP-11/24 system cabinet is supplied with four rollers on the bottom frame and four leveler feet. The cabinet can be positioned alone or attached to another H9640 series cabinet. To install the PDP-11/24 system cabinet perform the following steps.

1. Roll the system cabinet into position.

2. Open the rear door of the cabinet. Use a 4 mm (5/32 in) hex wrench to release the door fasteners (Figure 2-13).

3. Tilt the top of the door away from the cabinet and lift the door until the lower pins are removed from the holes in the lower brackets.

4. Remove the rear door.

5. Remove the shipping brackets supplied with the devices mounted in the cabinet. Shipping brackets are painted orange. If the system contains RL02 disk drives, make certain that the black bracket attached to the top unit is not removed.

NOTE

Retain all shipping brackets in the event reshipping is required.

6. Loosen the two 1/4-20 screws and washers that attach the right and left brackets to the cabinet frame.

7. Remove and retain the right and left brackets.

8. Remove the left and right side panels by grasping each panel by the ends at the front and back of the cabinet and lifting the panel approximately 2.54 cm (1 in) to disengage the panel. Pull the panel away from the cabinet to remove it.

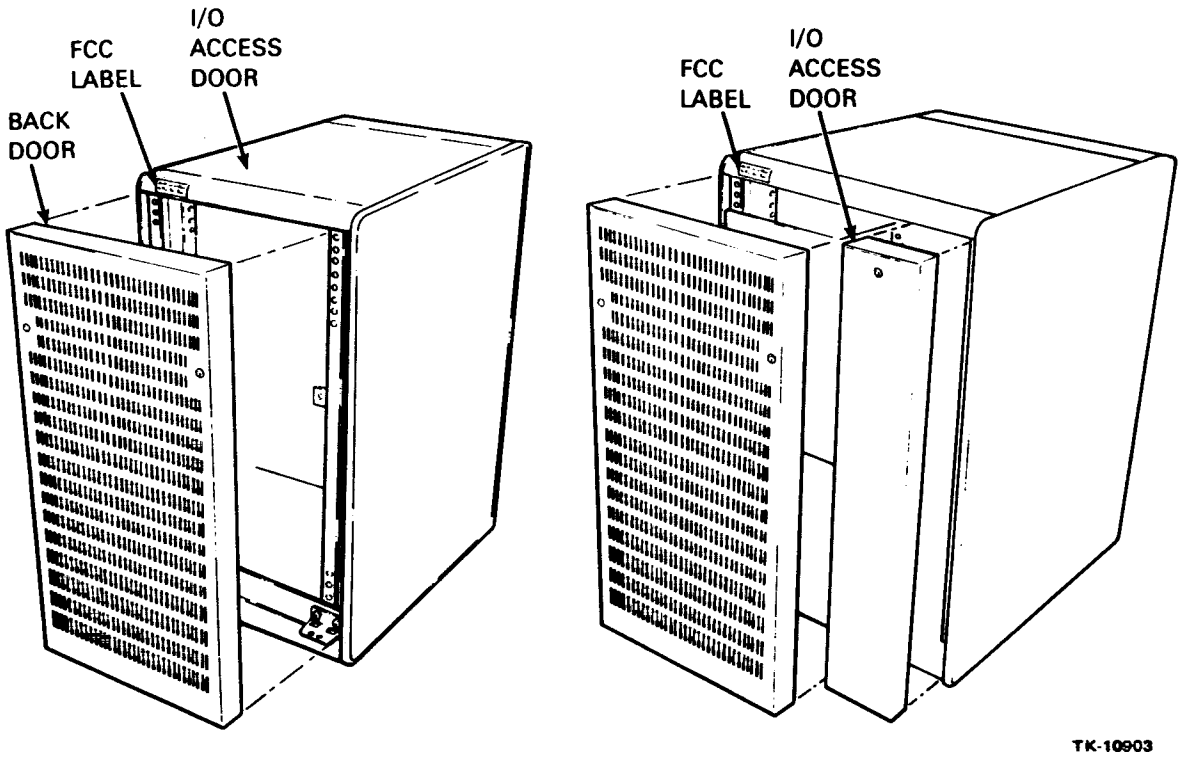


Figure 2-13 H9642 (Single-Bay) and H9645 (Wide-Body) Cabinets – Rear View

9. Install the leveler feet (four) supplied with the cabinet (Figure 2-14) by performing the following steps:

   a. Screw a locking nut into each leveler foot.

   b. Screw each leveler foot into a leveler nut.

   c. Slide each nut and leveler foot assembly into a slot located at each corner of the cabinet base.

10. Install the stabilizer bar supplied with the cabinet (Figure 2-15) by performing the following steps.

   a. Screw a leveler foot into the stabilizer bar.

   b. From the back of the cabinet, insert the stabilizer arm into the channel located on the bottom of the cabinet.

   c. Remove the retaining cable and screw from the back of the cabinet base and attach the retaining cable to the underside of the stabilizer arm.

NOTE

The retaining cable is used to limit the forward extension of the stabilizer arm.

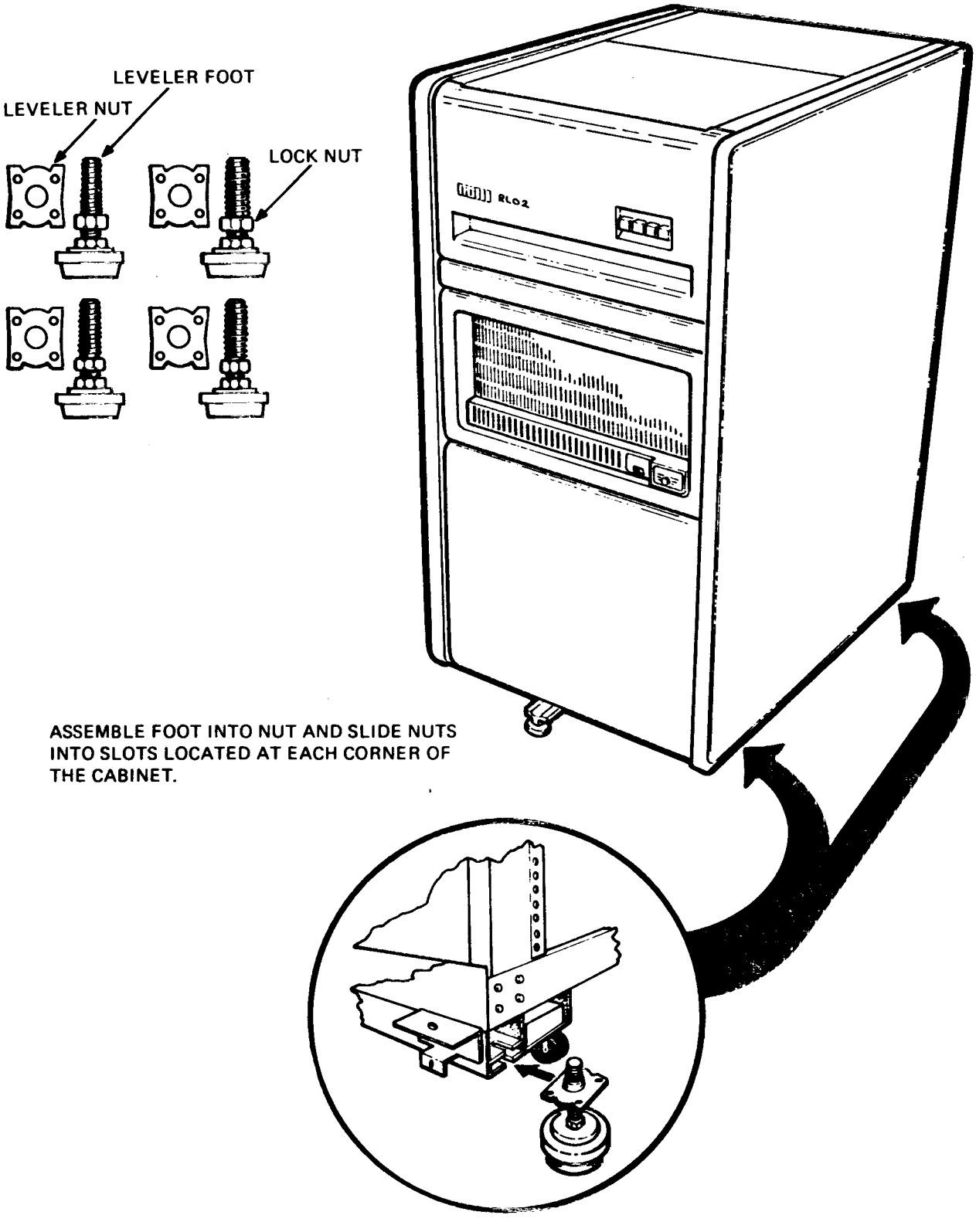


Figure 2-14 Installing Leveler Feet in the H9642 Cabinet

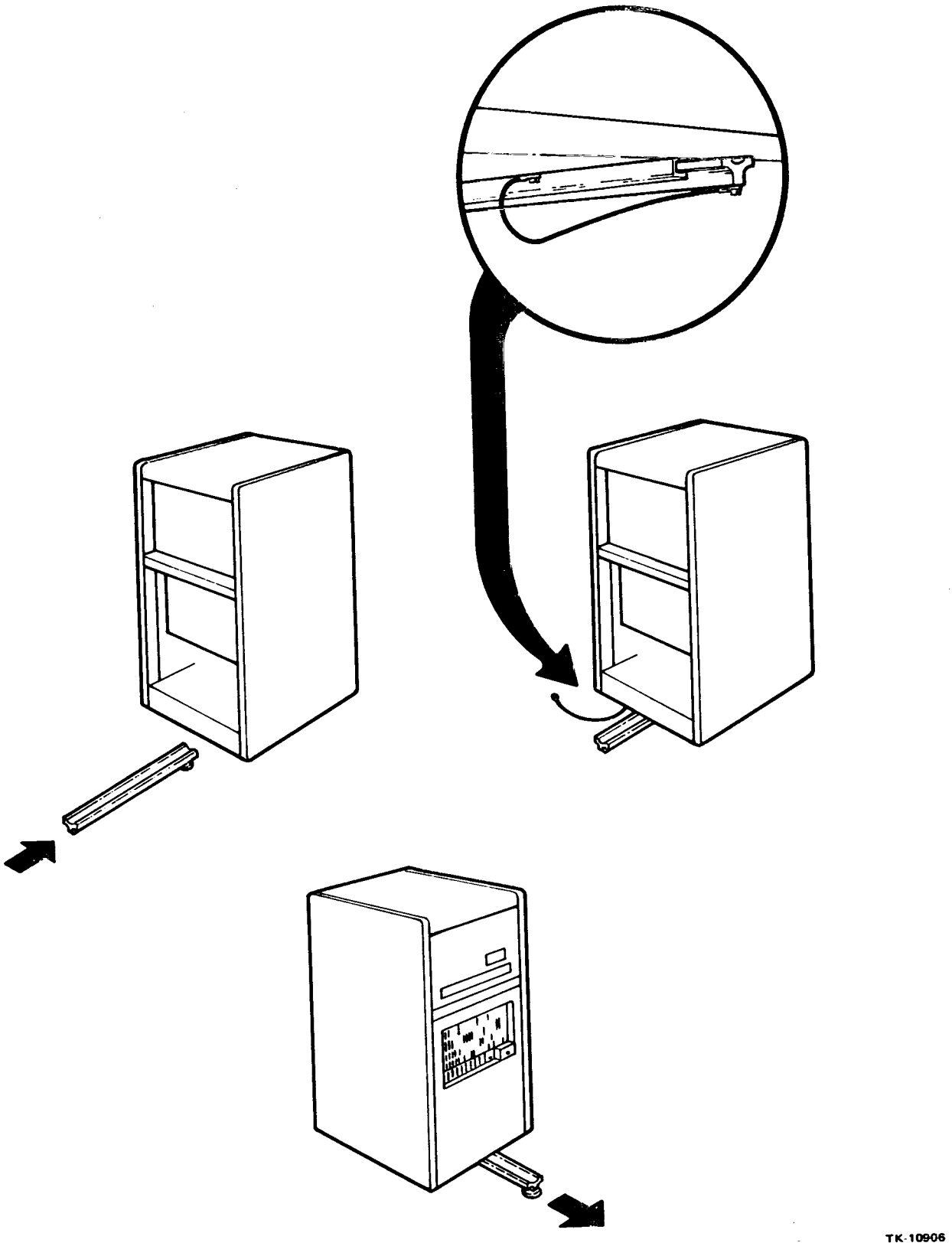


Figure 2-15 Stabilizer Arm Installation

11. If an expander cabinet is a part of the system, it should be attached to the system cabinet at this time. Refer to Paragraph 2.6.4 for instructions.

12. Replace the left and right side panels removed in Step 8.

13. Replace the brackets removed in Steps 6 and 7 and tighten the 1/4-20 screws.

14. Tilt the rear door and insert the lower pins into the lower brackets.

15. Move the top of the rear door towards its mounting position.

16. Close the cabinet rear door.

17. With the cabinet(s) in final position, adjust the leveler feet (Figure 2-16) on all cabinets for contact with the floor to prevent movement, and then tighten the lock nut on each foot. Adjust the leveler foot on each stabilizer arm for smooth sliding contact with the floor.

18. Installation of terminals and peripheral devices (disk drives, tape drives, printers, and so on) should be done at this time.

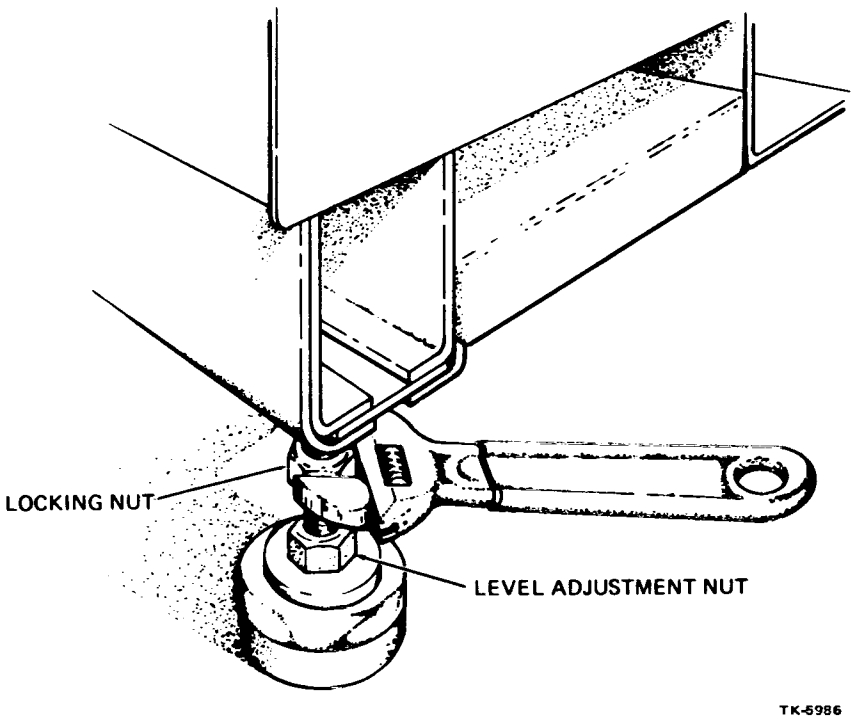


Figure 2-16 Leveler Feet Adjustment

2.6.4 Expander Cabinet Installation

The following procedure should be used when adding a shielded (that is, FCC-compliant) H9642 expander cabinet to a shielded H9642 (single-bay) or H9645 (wide-body) CPU cabinet. Perform the following steps.

1. Wheel the expansion cabinet into proximity with the CPU cabinet.

2. Carry out the following procedure on both cabinets.

   a. Remove the rear doors (using a 4 mm [5/32 in] hex wrench to release the two door fasteners of each main panel) and the single narrow vertical panel (H9645 wide-body only). Refer to Figure 2-13 for a right-rear view of the expansion cabinet.

   b. To remove each of the large rear doors, tilt the top of the door away from the cabinet while lifting the door. Lift until the two pins projecting below the bottom lip of each door are clear of their respective brackets.

   c. Set the two doors aside.

   d. On both cabinets, loosen the two 1/4-20 screws and washers that attach each door bracket to the cabinet frame.

   e. Remove and retain the two brackets of each cabinet.

   f. Remove the narrow vertical panel covering the I/O space at the rear of the CPU cabinet. To remove, unlatch the single fastener at the top of this panel by using the 4 mm (5/32 in) hex wrench.

   g. Lift the panel off its mounting buttons and set aside.

3. Prepare the *CPU cabinet* for mating with the H9642 expander cabinet by performing the following steps.

   a. Remove the top cover by:

      1) Removing the two screws located beneath the back of the top cover
      2) Disconnecting the top-cover ground strap
      3) Lifting the top cover off the cabinet frame
      4) Replacing the removed screws in their respective holes for safekeeping

   b. Remove the decorative end panel from the left side of the cabinet. To remove, grasp the panel at the front and back and lift approximately 2.54 cm (1 in). Pull the panel away from the cabinet and set aside.

      NOTE

      H9640-series shielded cabinets are designed for either right- or left-side add-on of expander cabinets. For PDP-11/24 systems, add-on is always to the left side of the CPU cabinet as seen from the front.

   c. If the CPU cabinet is resting on its wheels, skip over the next step and proceed to Step e.

d. Loosen the locking nuts on the four leveler feet located at the bottom corners of the cabinet (Figure 2-14) and raise the feet until the cabinet is resting on its wheels.

e. Remove the RFI shield panel from the left side of the CPU cabinet (Figure 2-17).

CAUTION

Exercise care when handling the RFI shields. Careless handling can cause damage to the RFI-gasket springs located on its front and back edges.

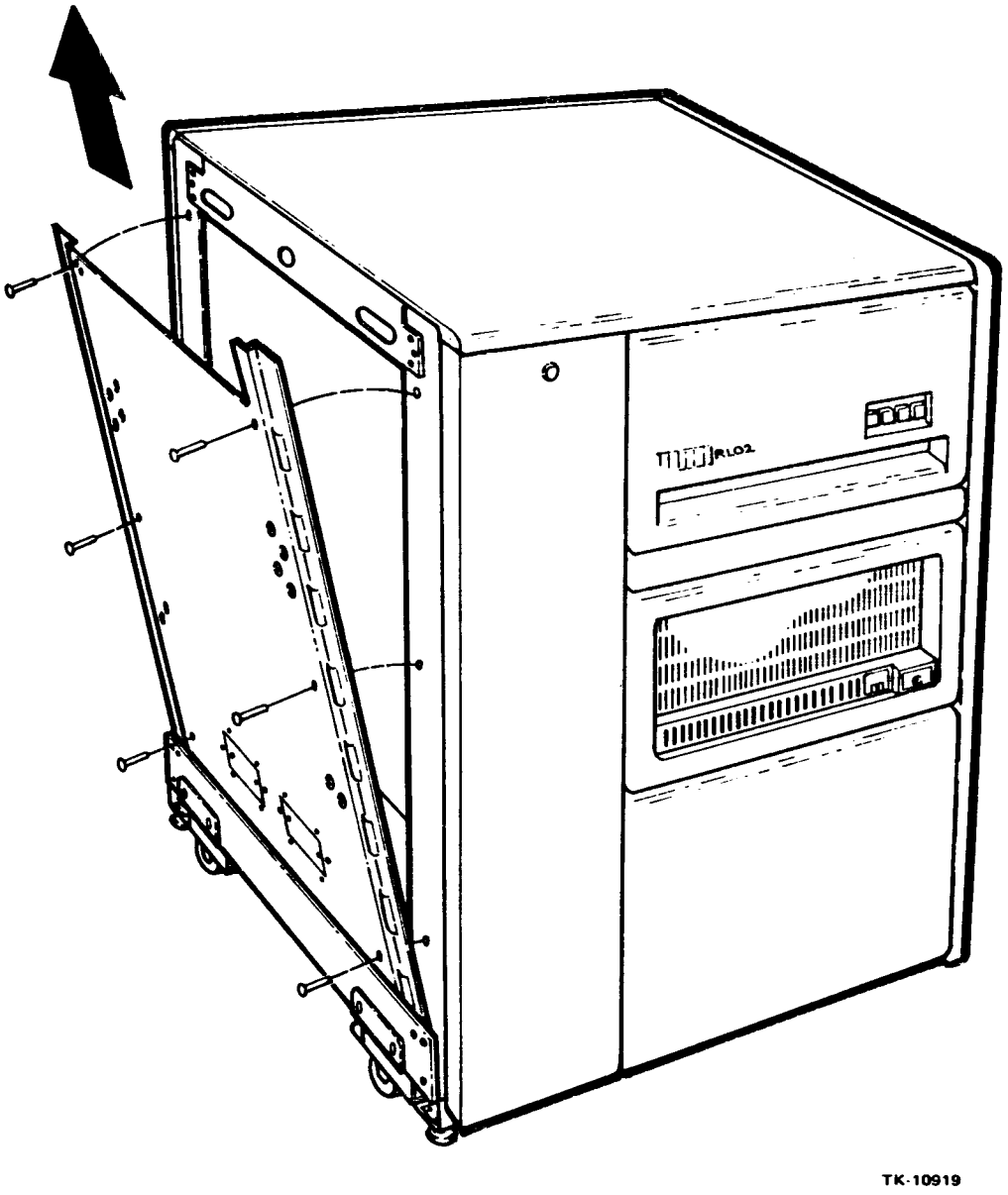


Figure 2-17 CPU Cabinet – Removing the RFI Shield Panel

To remove the RFI shield panel perform the following steps.

1) Remove and save the two round head screws with lock washers

2) Remove and save the four key button screws

3) Grasp the top of the RFI panel and pull away from the cabinet approximately 30.48 cm (12 in)

4) Lift the shield panel up and away until the projecting legs located at the bottom of the panel are clear of the structural cross member at the bottom of the cabinet

f. Remove the two knockouts from the RFI shield panel Figure 2-18 in the following sequence.

1) Place two wooden blocks (removed from the shipping skids) on the floor. Lay the shield panel on the blocks with the knockouts positioned over the blocks as shown in the illustration.

2) Use a hammer and a flat-blade screwdriver to break the upper edge of each knockout free of the shield panel.

3) Lift the shield panel off the blocks and push inward on the upper edge of each knockout until the lower edges break free.

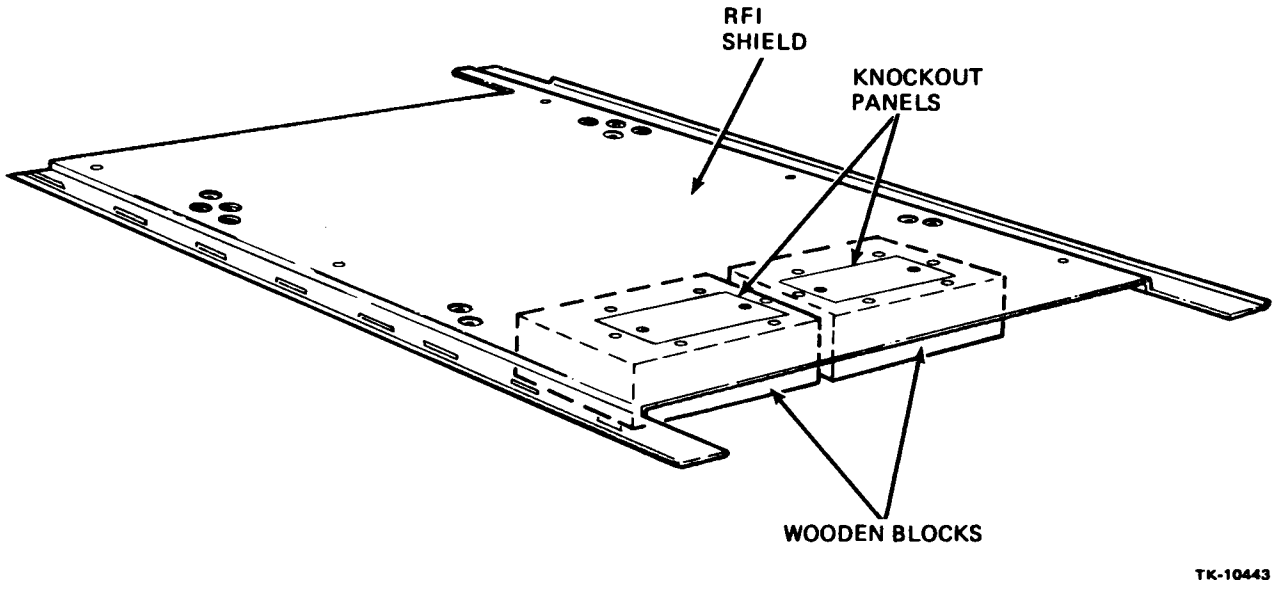


Figure 2-18 Removing Knockout Panels in the RFI Shield Panel

g. Install a cable-port flange in each of the two ports formerly filled by the knockouts of the RFI shield panel (Figure 2-19). Install as follows.

1) Place one of the flanges from the cabinet joiner-bar kit (P/N H9645-JE) over a knockout hole on the outside of the shield panel.

2) Align the screw holes in the flange with the screw holes in the shield panel. The lip of the flange should be pressed into the knockout hole.

3) Insert six 8/32 × 1/4-inch screws into the flange screw holes from the inside of the shield panel; tighten the screws.

4) Repeat Steps 1) through 3) to install the second flange section.

h. Carefully position the shield panel upright, leaning it against the left side of the CPU cabinet. DO NOT install the shield panel onto the CPU cabinet at this time.

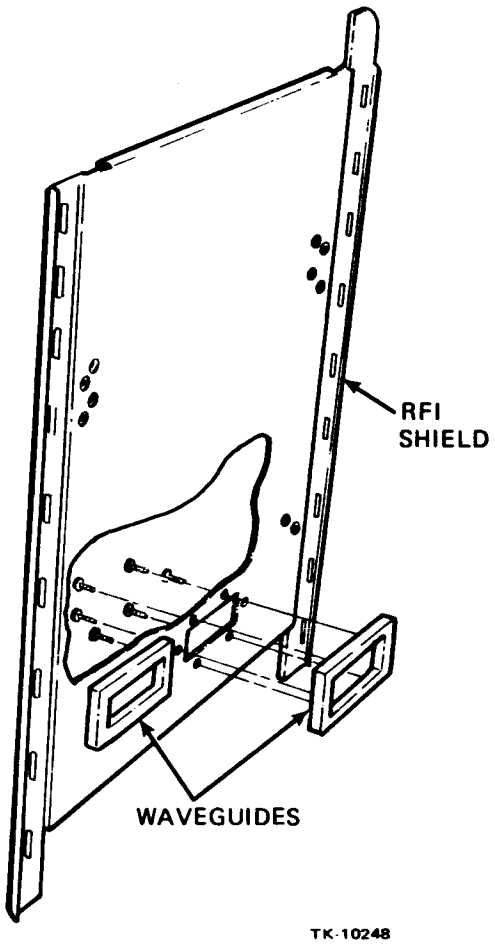


Figure 2-19 Installation of Cable-Port Flanges in the RFI Shield Panel

4. Prepare the *H9642 expander cabinet* for mating with the CPU cabinet by performing the following steps.

   a. Remove the expander panel attached to the right side of the expander cabinet. Grasp the panel at its front and back, then lift the panel up and away from the cabinet (Figure 2-20).

   b. Install the four leveler feet. Raise the feet to their upper position so that the cabinet is resting on its wheels (Figure 2-14).

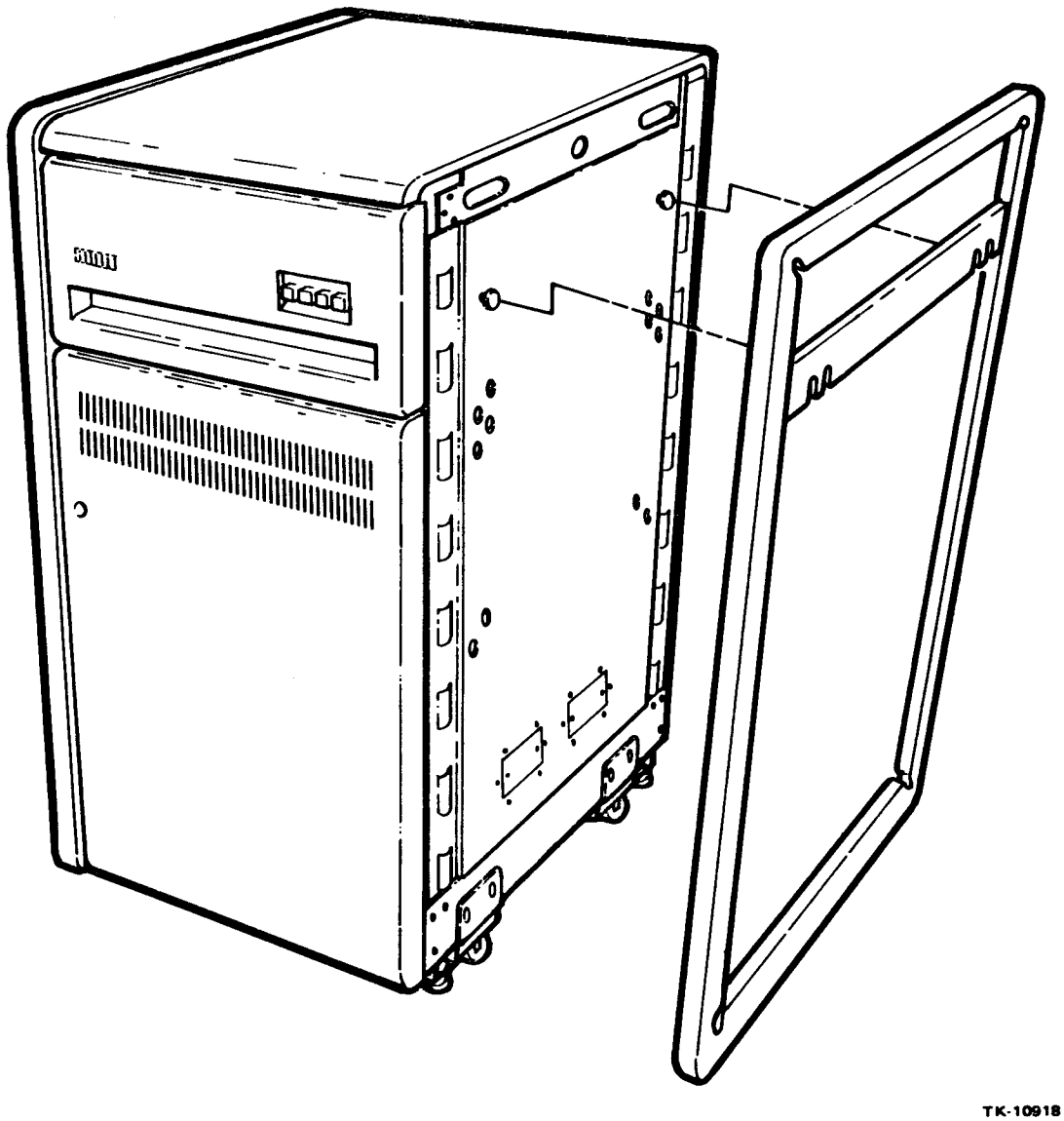


Figure 2-20 Removing the Expander Panel from the H9642 Expander Cabinet

c. Remove the front door of the expander cabinet (Figure 2-20).

d. Remove the RFI shield panel from the right side of the expander cabinet. Follow the procedure outlined in item 3.e of the previous section.

e. Remove the two knockouts from the RFI shield panel. Use the procedure outlined under item 3.f of the previous section.

f. Replace the RFI shield panel in its former position on the side of the expander cabinet. Reverse the order and sense of the procedure given in item 3.e.

CAUTION

Use care when inserting the projecting legs of the shield panel over the lower front and back of the cabinet frame. Careless insertion may damage the RFI gasket springs on the shield panel.

g. Install the expander panel on the right side of the expander cabinet (Figure 2-20). To install, reverse the procedure given in item 4.a of this section.

CAUTION

Be sure to use the expander panel shipped with the expander cabinet. This panel has a single (upper) locking bar. DO NOT use an expander panel that has both upper and lower locking bars.

NOTE

The longer set of key slots on the expander panel should be attached to the expander cabinet.

h. Move the expander cabinet so that its right side is approximately 0.9 m (3 ft) from the left side of the CPU cabinet. This separation permits easy access through either of the facing sides.

i. Locate, inside the expander cabinet, the ribbon cable(s) that must be routed to the CPU cabinet via the cable ports at the bottom of the cabinets (Figure 2-21).

j. Route the unconnected end of the expander cabinet's ribbon cable through the nearer of the two cable ports in the RFI shield panel, and through the two corresponding ports in the shield panel of the CPU.

k. Bring the cable up the inner left side of the CPU cabinet to the cable rack at the top of the cabinet. If necessary, move the expander cabinet closer to the CPU cabinet to adapt to the cable length.

l. Reinstall the CPU's RFI shield panel on the left side of the cabinet. Reverse the order and sense of the steps for shield panel removal (item 3.e of the previous section).

CAUTION

Use care when inserting the projecting legs of the shield panel over the lower front and back of the cabinet frame. Careless insertion may damage the RFI gasket springs on the shield panel.

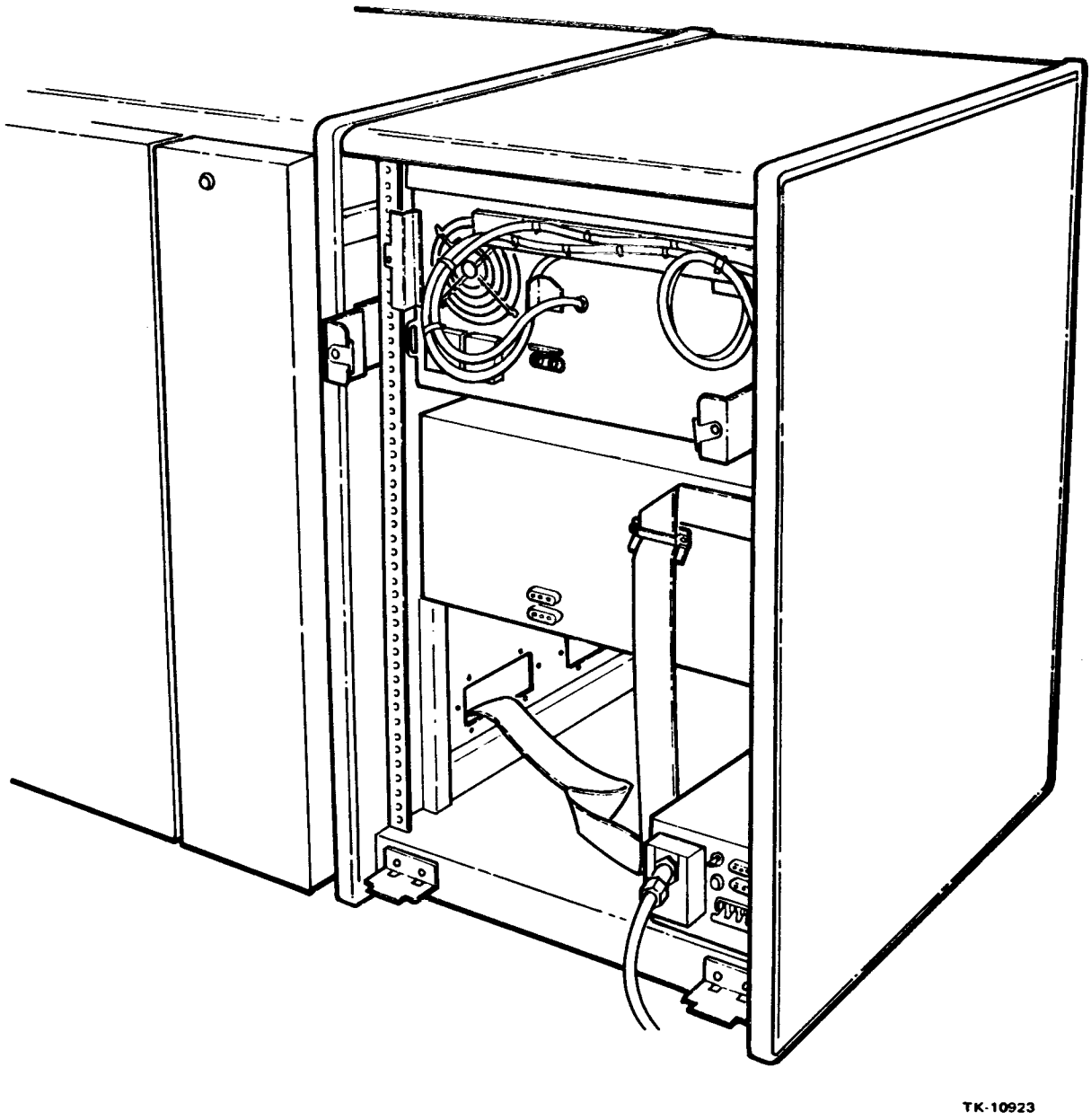


Figure 2-21 Interior of H9642 Expansion Cabinet – Rear View

5. Join the CPU and expander cabinets in the following sequence.

   a. Carefully move the expander cabinet until its right side just meets the left side of the CPU cabinet (Figure 2-22).

CAUTION
Make sure that the cable does not bunch up between cabinets.

   b. Raise the expander panel on the right side of the expander cabinet approximately 2.54 cm (1 in).

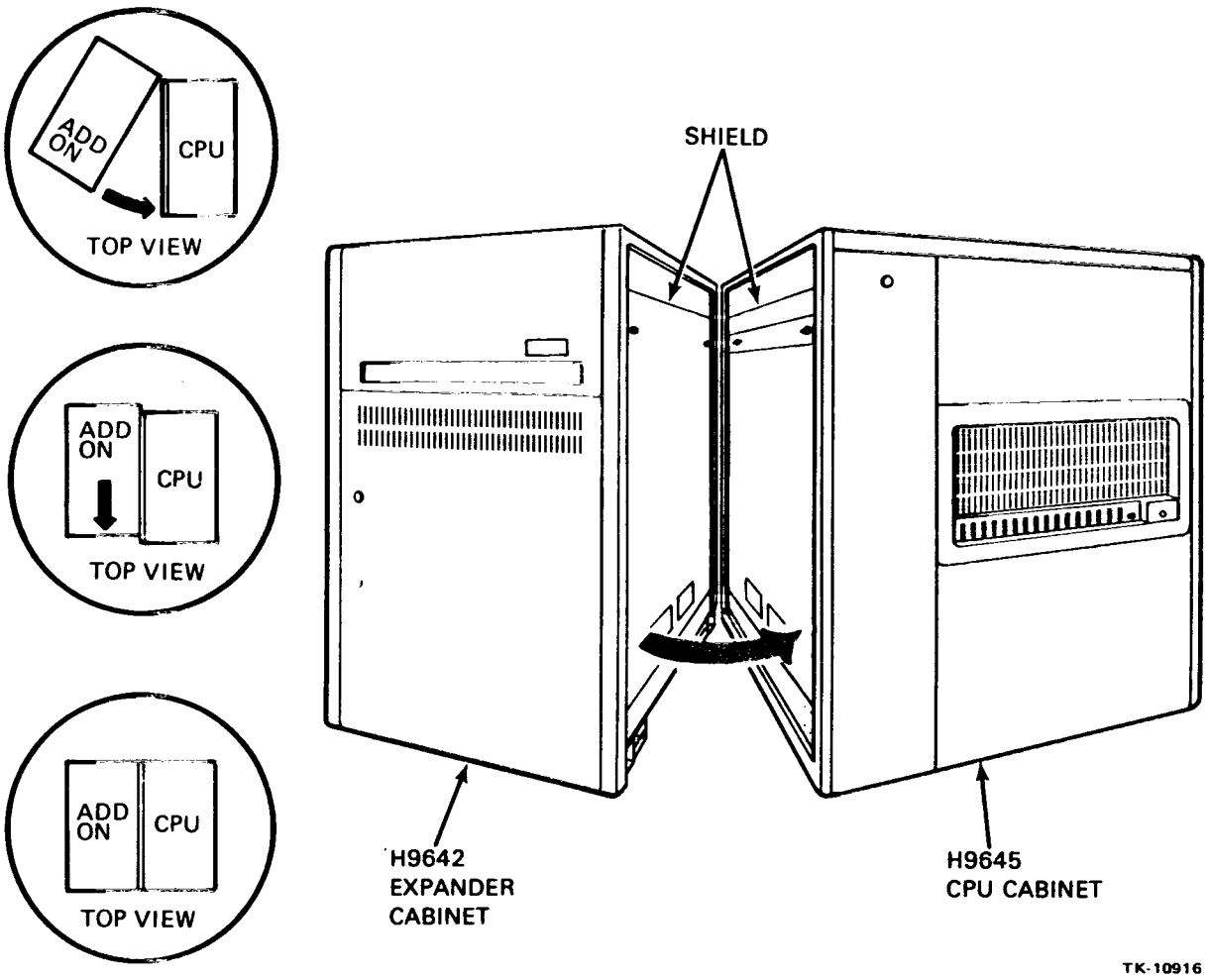


Figure 2-22 Joining the H9645 CPU Cabinet and the H9642 Expander Cabinet

c. While holding the cabinets together, push the expander panel down slightly until its key slots just begin to engage the upper key buttons at the front and rear of both cabinets.

d. From inside the expander cabinet:

1) Make sure that the lips on both cable-port flanges are inserted into the knock-out holes.

2) Make sure that the six screw holes on each cable-port flange are aligned with the screw holes in the RFI shield panel.

NOTE

Jockey the expander cabinet and/or adjust its leveler feet until correct alignment of cable-port flanges is achieved for both ports of both cabinets.

3) Secure each cable-port flange to its RFI shield panel with six 8/32 × 1/4-inch screws.

e. Push down firmly on the expander panel to securely lock the cabinets together.

f. Bolt the cabinets together at the front; use the front interconnecting bar provided (Figure 2-23).

g. Bolt the cabinets together at the back; use the rear interconnecting bar provided (Figure 2-24)

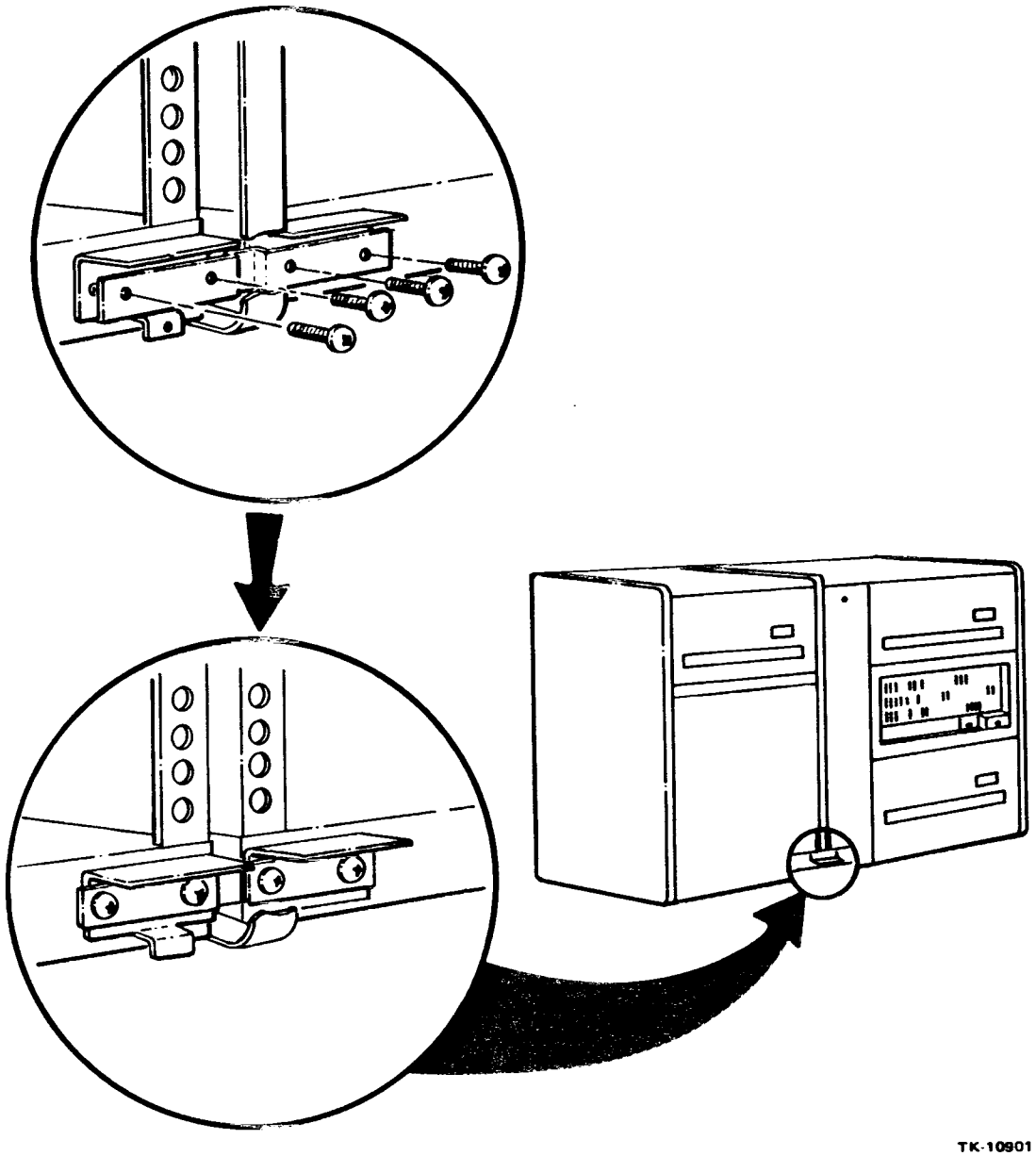


Figure 2-23 Detail of Front Interconnecting Bar Used in Joining the H9645 CPU Cabinet and the H9642 Expander Cabinet

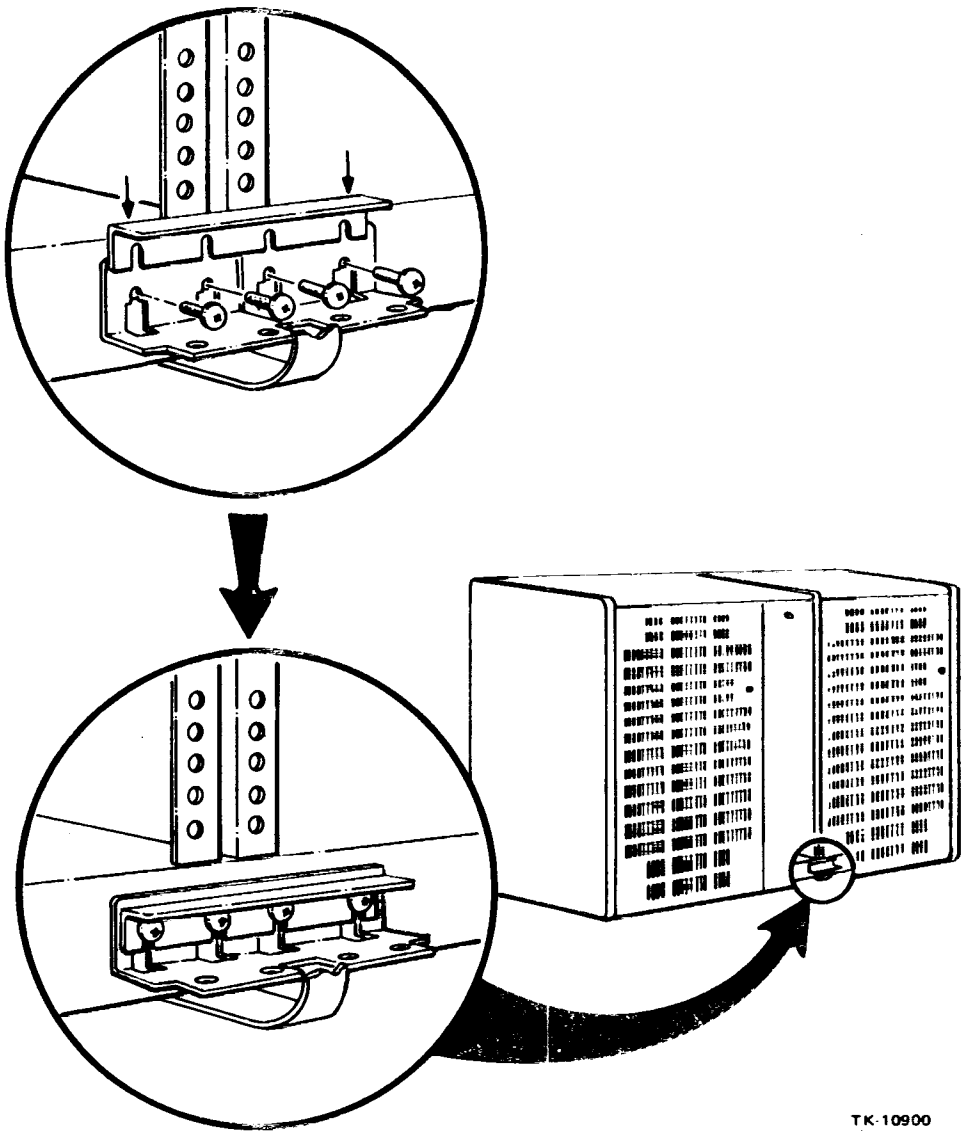


Figure 2-24 Detail of Back Interconnecting Bar Used in Joining the H9645 CPU Cabinet and the H9642 Expander Cabinet

2.7 MODULE UTILIZATION IN A TYPICAL SYSTEM

The PDP-11/24 processor module uses its own backplane (part no. 70-16905) and cannot use DD11-CK, DD11-DK, or DD11-PK backplanes. Figure 2-25 shows the module utilization of the PDP-11/24 processor backplane.

1. M9312 bootstrap/terminator module or UNIBUS cable must only be installed in slot 9.
2. NPR (DMA) options must only be installed in slots 7 or 8.

3. Non-NPR options must only be installed in slots 3, 4, 5, 6, or 9.

4. All empty slots, except slot 2, must have grant cards. Slots 7 and 8 must have a double-height grant card (G7273) in rows C and D.

   Slots 3, 4, 5, 6, and 9 may use a single-height grant card (G727A or G7270) in row D or a double-height grant card in rows C and D.

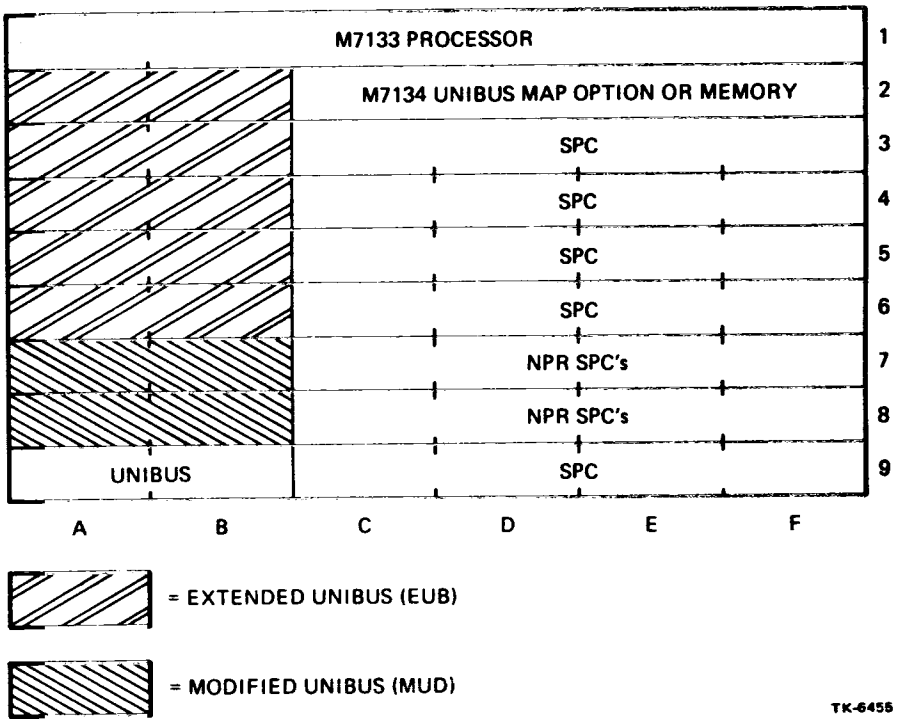


Figure 2-25 PDP-11/24 Module Utilization

2.8 MODULE JUMPER AND SWITCH CONFIGURATIONS

(See Appendix D, Section D.2, for M7133-YA CPU module jumper and switch locations.)

2.8.1 CPU Module (M7133)

The CPU module contains two switchpacks and 14 jumper leads. Figure 2-26 shows the location of the jumpers and switches contained on the CPU module. Table 2-1 lists the function of the jumper leads.

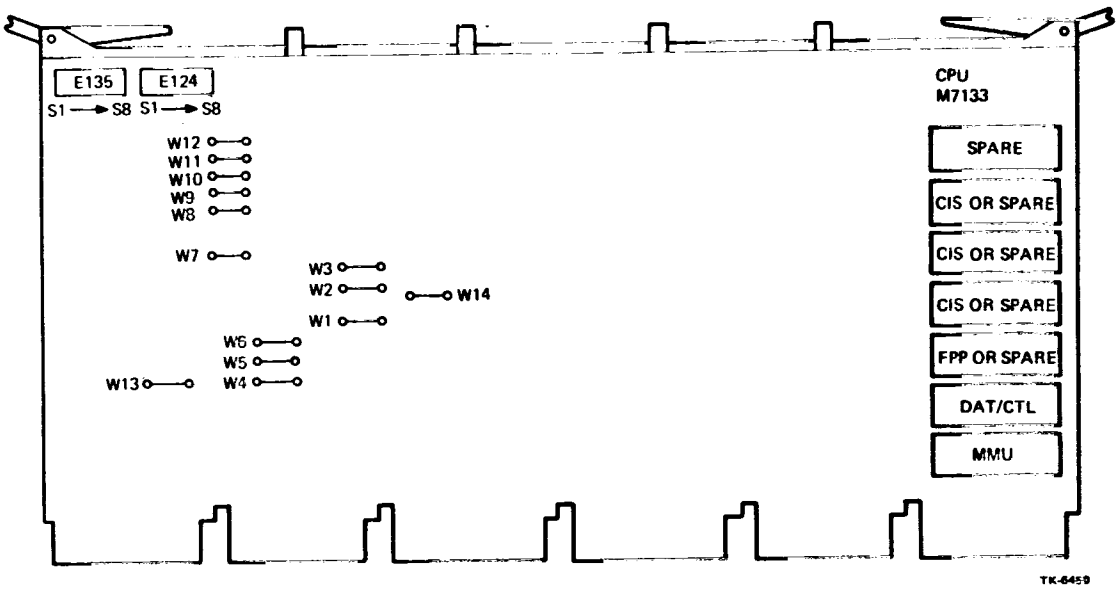


Figure 2-26 PDP-11/24 CPU Module, Jumper, and Switch Locations

With the exception of E124 S5 and S6, the two switchpacks are used to configure the serial line units. Switch S5 of E124 when ON enables the field service maintenance mode on SLU1. This allows the looping of the received data back to the transmitter via the parallel data bus. When S5 is OFF, normal SLU operation is enabled. Switch S6 of E124 is used to determine UNIBUS addressing: when ON, S6 allows only the top 128K addresses to go to the UNIBUS bus; when OFF, the lower 18 bits of every address go to the UNIBUS bus.

Serial Line Units (SLUs) – The PDP-11/24 CPU module contains two serial line units (SLUs). Serial line unit 1 (SLU1) is used by the console terminal. Serial line unit 2 (SLU2) is a general-purpose serial line unit that can be used for a second terminal, a serial line printer, a TU58 DECtape II cartridge tape drive, or other similar devices. The two switchpacks (E135, E124) and jumpers W1, W4, and W9-W13 are used to configure the SLUs.

Table 2-1 CPU Module Jumper Lead Selection

Jumper Lead	Function
CPU	
W2	Controls power fail power-up action. IN = Boot on powerup from powerfail. OUT = Power up via location 24_8 provided the optional battery backup unit is installed and functioning.
W3	Controls kernel halt enable. IN = In kernel mode allows halt instruction to be executed. In all other modes, a halt instruction traps to location 10_8. OUT = All halt instructions trap to location 10_8 (except during power-fail).
W14	Selects boot address when boot on powerup is enabled. IN = $165\ 000_8$ OUT = $173\ 000_8$
SLU	
W1	Determines number of stop bits in SLU1. IN = One stop bit OUT = Two stop bits For manufacturing tests, the number of stop bits defaults to one.
W4, W9–W13	Select SLU2 receiver and transmitter baud rate.
W5–W8	Determines parity of SLUs.

Switches S1-S8 of E135 are used to select the various baud rates generated by the two baud rate generators contained on the CPU module. Table 2-2 lists the baud rates and their switch settings.

Table 2-2 Baud Rate Selection

		Switch Pack E135			
Baud Rate 1		**5**	**6**	**7**	**8**
Baud Rate 2		**1**	**2**	**3**	**4**
Baud Rate*	50	ON	ON	ON	ON
	75	ON	ON	ON	OFF
	110	ON	ON	OFF	ON
	134.5	ON	ON	OFF	OFF
	150	ON	OFF	ON	ON
	200	ON	OFF	ON	OFF
	300	ON	OFF	OFF	ON
	600	ON	OFF	OFF	OFF
	1200	OFF	ON	ON	ON
	1800	OFF	ON	ON	OFF
	2000	OFF	ON	OFF	ON
	2400	OFF	ON	OFF	OFF
	3600	OFF	OFF	ON	ON
	4800	OFF	OFF	ON	OFF
	9600	OFF	OFF	OFF	ON
	19200	OFF	OFF	OFF	OFF

*For manufacturing tests, the baud rates default to 19200 baud.

SLU1 transmit and receive baud rates are selected by S1-S4 of E124. Table 2-3 lists the switch settings and baud rate selected.

Table 2-3 SLU1 Baud Rate Selection

	E124	
Transmit	**2**	**1**
Receive	**3**	**4**
Baud Rate 1	ON	OFF
Baud Rate 2	OFF	ON

SLU2 transmit and receive baud rates are selected by jumper leads W4, W9-W13. Table 2-4 lists the baud rates and their respective jumper configuration.

Table 2-4 SLU2 Baud Rate Selection

	Jumpers		
Transmit **Receive**	**W11** **W9**	**W10** **W12**	**W13** **W4**
Baud Rate 1	In	Out	Out
Baud Rate 2	Out	In	Out
19.2K Baud	Out	Out	In

Table 2-5 lists the jumper lead configuration for setting the number of stop bits on SLU1 and selecting and enabling parity in SLU1 and SLU2.

Table 2-5 Serial Line Unit Jumper Lead Selection

Jumper Lead		
SLU1	**SLU2**	**Function**
W1	None; Fixed at 1.	Selects the number of stop bits. IN = One stop bit OUT = Two stop bits For manufacturing tests, the number of stop bits defaults to one.
W5	W7	When IN, parity detection is enabled. When OUT, parity detection is disabled.
W6	W8	When IN, odd parity is generated. When OUT, even parity is generated.

It is recommended to set SLU1's transmit/receive baud rate to baud rate 1 and SLU2's transmit/receive baud rate to baud rate 2. Thus, baud rate 1 controls SLU1 and baud rate 2 controls SLU2. Other baud rate configurations can be used; for a more detailed explanation of the baud rate logic and setup, refer to Paragraph 4.7.3 of this manual.

2.8.2 M9312 Bootstrap/Terminator Module

The M9312 bootstrap/terminator module is configured for use in the PDP-11/24 processor by 12 jumper leads (W1-W12). Five sockets on the M9312 module are used to hold the ROMs which contain diagnostic routines and bootstrap programs used with the PDP-11/24 processor and its associated peripheral devices. Figure 2-27 shows the location of the jumpers and ROM sockets on the M9312 module. For a more detailed explanation of the M9312 module, refer to the *M9312 Bootstrap/Terminator Module Technical Manual* (EK-M9312-TM).

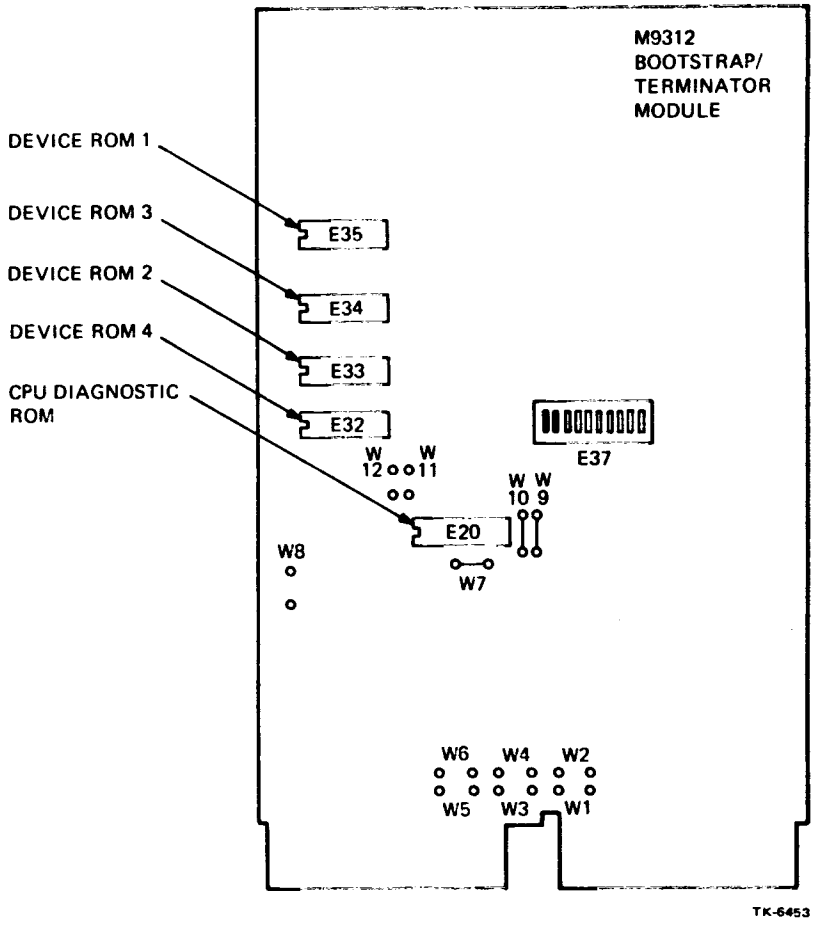


Figure 2-27 M9312 Jumper and Switch Locations

2.8.2.1 M9312 Jumper Configuration – For use in the PDP-11/24 processor, the jumpers on the M9312 module should be configured according to Table 2-6.

Table 2-6 M9312 Jumper Configuration in PDP-11/24 System

Jumper	Configuration
W1–W5	Out
W6	Out
W7	In
W8	Out
W9–10	In
W11–12	Out

2.8.2.2 M9312 ROMs – The M9312 module contains a CPU diagnostic ROM socket (E20), which accepts the PDP-11/24 CPU diagnostic ROM, and four device boot ROM sockets (E32, E33, E34, and E35) that accept device bootstrap ROMs. The CPU diagnostic ROM checks the performance of the central processor and main memory when power is initially applied to the system or when a device bootstrap is initiated.

The device bootstrap ROMs contain device-dependent programs that load programs from a selected peripheral device into main memory. The number and type of ROMs installed are dependent upon the devices in the system. A bootstrap program is initiated by the operator from the console terminal when selected by the BOOT position of the toggle switch on the front panel or via ODT. The processor executes the bootstrap program directly without the operator keying in the initial program.

Table 2-7 lists the location of the CPU diagnostic and bootstrap loader ROMs and the starting addresses associated with each. The selection of a starting address determines whether the CPU diagnostic is performed when the bootstrap program is initiated. The second devices listed in the table are for selecting a second device bootstrap program contained in the same ROM as the first device bootstrap program. Switches S1 through S10 at location E37 of the M9312 module select the starting address of the bootstrap program. Switch S1 determines the upper three digits of the starting address as follows:

S1 = ON (165XXX) (Boot to ODT [console mode])
S1 = OFF (173XXX) (Boot selected device)

Switch S2 on the M9312 module must be off for operation with the PDP-11/24 processor.

Switches S3 through S10 are bits <08:01> of the starting address.

Table 2-7 CPU Diagnostic and Bootstrap Loader ROM Addresses

		S1	S3–S10		
Boot	Diagnose		First Device All ROMs	Second Device 23-755A9 RP04/5/6 RM02/3	Second Device 23-756A9 TU55/56 23-760A9 TTY tape
ODT	No	On	004	XXX	XXX
	Yes		006	XXX	XXX
Device ROM 1	No	Off	004	050	034
	Yes		006	052	036
Device ROM 2	No	Off	204	250	234
	Yes		206	252	236
Device ROM 3	No	Off	404	450	434
	Yes		406	452	436
Device ROM 4	No	Off	604	650	634
	Yes		606	652	636

The position of the bootstrap ROMs on the module must be sequential, starting with BT1 and progressing to BT4 as listed in Table 2-8.

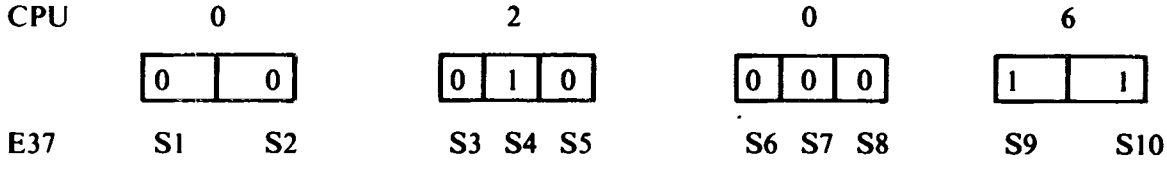
Table 2-8 Bootstrap ROM Locations

Bootstrap ROM	Location
Device 1	E35
Device 2	E33
Device 3	E34
Device 4	E32

For example:

To select an RL02 installed in the second ROM location (E33) and run the CPU diagnostic program, set the switches as follows.

RL01/02 ROM in location E33 (Device ROM 2)
Run diagnostic and then boot RL02



0 = OFF
1 = ON

Table 2-9 lists the part numbers for available bootstrap ROMs. Some of the ROMs contain more than one device program. The DDCMP bootstrap loaders for the communication devices DL11, DMC-11, DU11, and DUP-11 are supplied in three ROMs for each device.

Table 2-9 Device ROM Part Numbers

Device	Mnemonic	ROM Part Number
Paper Tape		
ASR 33	TT	23-760A9
PC05	PR	
DECnet Down-Line Load		
DDCMP(DI-11)	XL	23-926A9 23-927A9 23-928A9
DDCMP(DMC-11)	XM	23-862A9 23-863A9 23-864A9
DDCMP(DU-11)	XU	23-868A9 23-869A9 23-870A9

Table 2-9 Device ROM Part Numbers (Cont)

Device	Mnemonic	ROM Part Number
DDCMP(DUP-11)	XW	23-865A9 23-866A9 23-867A9
Disks		
RA80/81	DU	23-767A9
RK03/05	DK	23-756A9
TU55/56*	DT	
RK06/07	DM	23-752A9
RL01/02	DL	23-751A9
RP02/03	DP	23-755A9
RP04/05/06	DB	
RM02/03	DB	
RS03/04	DS	23-759A9
RX01	DX	23-753A9
RX02	DY	23-811A9
Tapes		
TS04	MS	23-764A9
TU10,TE10,TS03	MT	23-758A9
TU10/45/77,TE16	MM	23-757A9
TU58†	DD	23-765A9
TU60	CT	23-761A9

*DECtape I
†DECtape II

2.8.3 UNIBUS Map Module (M7134)

The UNIBUS map module is an option available with the PDP-11/24 processor that gives the PDP-11/24 processor 22-bit addressing capability through the use of a UNIBUS map. The module also contains an M9312 compatible bootstrap and a voltage monitor similar to that used in the PDP-11/44 processor. The UNIBUS map module shown in Figure 2-28 contains twelve jumper lead locations and two switchpacks located at E6 and E58.

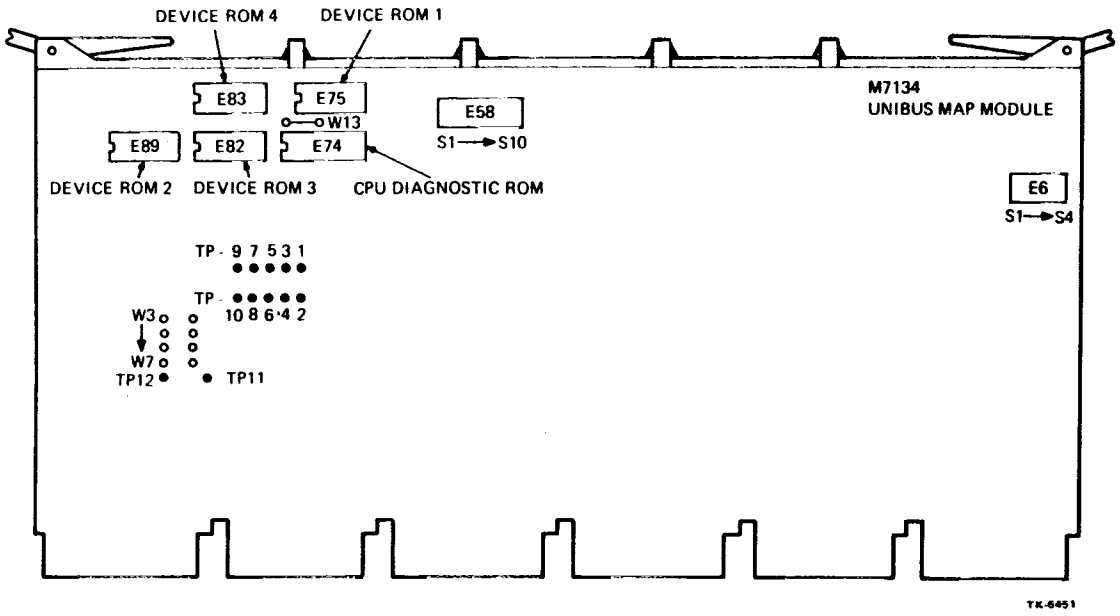


Figure 2-28 UNIBUS Map Module, Jumper, Switch, and ROM Locations

2.8.3.1 UNIBUS Map Module Jumper Leads and Memory Page Selection – Table 2-10 lists the function of the jumper leads on the UNIBUS map module.

Jumper leads TP1-TP10 and W3-W7 specify the lower and upper limit, respectively, of a set of UNIBUS addresses not mapped to main memory (asserted on the UNIBUS bus only). Except for the I/O page, every UNIBUS address not in this set is mapped to main memory (asserted on the memory bus) by the UNIBUS map. Devices on the UNIBUS bus should be addressed only in unmapped or I/O page address space.

When the upper and lower limits are set to the same octal bank (page) number, every non-I/O page address is mapped to main memory. Most DIGITAL supplied configurations conform to this case. However, systems with UNIBUS memory or I/O devices that are located in the UNIBUS memory region require changes to be made to the mapping jumpers.

Table 2-10 UNIBUS Map Module Jumper Lead Functions

Jumper Lead	Function
W3–W7	UNIBUS map page number, upper limit
TP1–TP10*	UNIBUS map page number, lower limit
TP11–TP12*	When OUT, diagnostic ROM enabled, (normally OUT)
W13	When IN, diagnostic ROM enabled, (normally IN)

* These jumpers are wire wrap posts. For example, a jumper IN consists of connecting TP1 to TP2 or TP5 to TP6 and so on. A

Table 2-11 lists the jumper leads selection for the lower limit of the set of unmapped addresses. UNIBUS addresses from zero to just below this limit are mapped to main memory. The octal bank number in this table is the first bank of the unmapped set, except when the lower and upper limits are set to the same bank number, in which case only the I/O page is unmapped.

Table 2-12 lists the jumper selections for the upper limit of the set of unmapped addresses. UNIBUS addresses above this limit and below 760 000 are mapped to main memory. The octal bank number in this table is the first mapped bank after the unmapped set, except when it is bank 37, the I/O page (always unmapped).

The following example describes the jumper selections for three pages (12K words) of UNIBUS device addresses immediately preceding the I/O page.

1. Page (bank) 37 is the I/O page; therefore, the three pages to be unmapped are 34, 35, and 36.

2. Read jumper settings for TP10 through TP1 from the lower limit (Table 2-11) at bank 34, the first unmapped bank; TP1-TP2 = out, TP3-TP4 = out, TP5-TP6 = out, TP7-TP8 = in, and TP9-TP10 = in.

3. Read jumper settings for W7 through W3 from the upper limit (Table 2-12) at bank 37, the next bank after the unmapped set desired (34, 35, and 36): W3 = out, W4 = out, W5 = out, W6 = out, and W7 = out.

Notice that the upper limit "Decimal K Words" amount minus the lower limit "Decimal K Words" amount is equal to 12K words. This is the size of the area desired.

2.8.3.2 Diagnostic and Bootstrap Loader ROMs – The UNIBUS map module contains a CPU diagnostic ROM and provides four sockets for the installation of device bootstrap ROMs. The CPU diagnostic ROM checks the performance of the central processor and main memory when power is initially applied to the system or when a device bootstrap program is initiated.

The device bootstrap ROMs contain device-dependent programs that load device programs from a selected peripheral device into main memory. The number and type of ROMs that are installed are dependent upon the devices included in the system. A bootstrap program is initiated by the operator from the console terminal when selected by the BOOT position of the toggle switch on the front panel or when requested via ODT. The processor executes the bootstrap program directly without the operator keying in the initial program.

Table 2-7 lists the location of the CPU diagnostic and bootstrap ROMs and the starting addresses associated with each. The selection of the first device starting address determines whether the CPU diagnostic is performed when the bootstrap program is enabled. The second devices listed in the table are for selecting a second device bootstrap program contained in the same ROM as the first device bootstrap program. Switches S1, and S3 through S10 at location E58 of the UNIBUS map module select the starting address of the bootstrap program. Switch S1 determines the upper three digits of the starting address as follows:

S1 = ON (165 XXX) (Boot to ODT [console mode])
S1 = OFF (173 XXX) (Boot a device)

Switch S2 enables or disables the reading of the boot ROMs on the UNIBUS map module. The switch settings are as follows:

S2 = ON (Boot ROMs readable)
S2 = OFF (UNIBUS map boot ROMs cannot be read)

Table 2-11 UNIBUS Map Jumper Leads, Lower Limit

Lowest Address in Unmapped Set	Decimal K Words	Octal Bank	Jumper Leads T1–TP2	TP3–TP4	TP5–TP6	TP7–TP8	TP9–TP10
None	124	37	Out	Out	Out	Out	Out
740 000	120	36	Out	Out	Out	Out	In
720 000	116	35	Out	Out	Out	In	Out
700 000	112	34	Out	Out	Out	In	In
660 000	108	33	Out	Out	In	Out	Out
640 000	104	32	Out	Out	In	Out	In
620 000	100	31	Out	Out	In	In	Out
600 000	96	30	Out	Out	In	In	In
560 000	92	27	Out	In	Out	Out	Out
540 000	88	26	Out	In	Out	Out	In
520 000	84	25	Out	In	Out	In	Out
500 000	80	24	Out	In	Out	In	In
460 000	76	23	Out	In	In	Out	Out
440 000	72	22	Out	In	In	Out	In
420 000	68	21	Out	In	In	In	Out
400 000	64	20	Out	In	In	In	In
360 000	60	17	In	Out	Out	Out	Out
340 000	56	16	In	Out	Out	Out	In
320 000	52	15	In	Out	Out	In	Out
300 000	48	14	In	Out	Out	In	In
260 000	44	13	In	Out	In	Out	Out
240 000	40	12	In	Out	In	Out	In
220 000	36	11	In	Out	In	In	Out
200 000	32	10	In	Out	In	In	In
160 000	28	07	In	In	Out	Out	Out
140 000	24	06	In	In	Out	Out	In
120 000	20	05	In	In	Out	In	Out
100 000	16	04	In	In	Out	In	In
060 000	12	03	In	In	In	Out	Out
040 000	08	02	In	In	In	Out	In
020 000	04	01	In	In	In	In	Out
000 000	00	00	In	In	In	In	In

Table 2-12 UNIBUS Map Jumper Leads, Upper Limit

Highest Address in Unmapped Set	Decimal K Words	Octal Bank	Jumper W3	W4	W5	W6	W7
757 777	124	37	Out	Out	Out	Out	Out
737 777	120	36	Out	Out	Out	Out	In
717 777	116	35	Out	Out	Out	In	Out
677 777	112	34	Out	Out	Out	In	In
657 777	108	33	Out	Out	In	Out	Out
637 777	104	32	Out	Out	In	Out	In
617 777	100	31	Out	Out	In	In	Out
577 777	96	30	Out	Out	In	In	In
557 777	92	27	Out	In	Out	Out	Out
537 777	88	26	Out	In	Out	Out	In
517 777	84	25	Out	In	Out	In	Out
477 777	80	24	Out	In	Out	In	In
457 777	76	23	Out	In	In	Out	Out
437 777	72	22	Out	In	In	Out	In
417 777	68	21	Out	In	In	In	Out
377 777	64	20	Out	In	In	In	In
357 777	60	17	In	Out	Out	Out	Out
337 777	56	16	In	Out	Out	Out	In
317 777	52	15	In	Out	Out	In	Out
277 777	48	14	In	Out	Out	In	In
257 777	44	13	In	Out	In	Out	Out
237 777	40	12	In	Out	In	Out	In
217 777	36	11	In	Out	In	In	Out
177 777	32	10	In	Out	In	In	In
157 777	28	07	In	In	Out	Out	Out
137 777	24	06	In	In	Out	Out	In
117 777	20	05	In	In	Out	In	Out
077 777	16	04	In	In	Out	In	In
057 777	12	03	In	In	In	Out	Out
037 777	08	02	In	In	In	Out	In
017 777	04	01	In	In	In	In	Out
None	00	00	In	In	In	In	In

Switches S3 through S10 are bits <08:01> of the starting address.

The position of the bootstrap ROMs on the module must be sequential, starting with BT1 and progressing to BT4 as listed on Table 2-13.

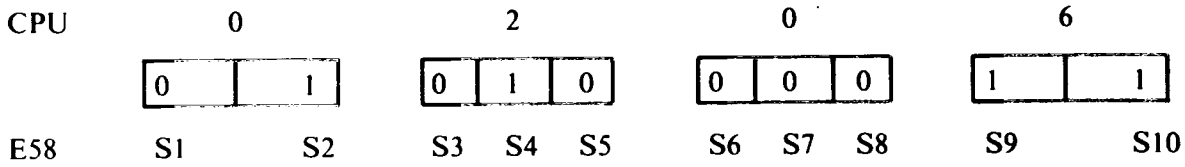
Table 2-13 Bootstrap ROM Locations

Bootstrap ROM	Location
Device 1	E75
Device 2	E89
Device 3	E82
Device 4	E83

For example:

To select an RL02 installed in the second ROM location (E89) and run the CPU diagnostic program, set the switches as described in the following example.

RL01/02 ROM in location E89 (Device ROM 2)
Run diagnostics and then boot RL02



0 = OFF
1 = ON

Table 2-9 lists the part numbers for the available bootstrap ROMs. Some of the ROMs contain more than one device program. For example, the RP02/03, RP04/5/6, and RM02/3 bootstrap programs are contained in one ROM. Other devices may require more than one ROM. The DDCMP bootstrap loaders for the communication devices DL11, DMC-11, DU11, and DUP-11 are supplied in three ROMs for each device.

2.8.3.3 Voltage Monitor Switch Selection – Table 2-14 lists the function of the switches associated with the voltage monitor circuit on the UNIBUS map module. Table 2-15 shows the normal switch configurations for the BA11-L and BA11-A mounting boxes.

2.9 INITIAL POWER-UP AND SYSTEM CHECKOUT

This section describes the procedure to be followed for the initial power-up of and operator introduction to the PDP-11/24 processor.

Table 2-14 Voltage Monitor Switch Selection

Switch (E6)	Function
S1	Selects +5 B voltage monitor output for diagnostic purposes. ON – Connected to monitor circuit OFF – Isolated from monitor circuit
S2	Selects −15 A voltage monitor output for diagnostic purposes. ON – Connected to monitor circuit OFF – Isolated from monitor circuit
S3	Selects the expected memory voltage. ON – ±15 B (BA11-L) OFF – ±12 B (BA11-A)
S4	Selects ±15 B or ±12 B (determined by S3) voltage monitor output for diagnostic purposes. ON – Connected to monitor circuit OFF – Isolated from monitor circuit

Table 2-15 Voltage Monitor Switch Settings

Switch	Mounting Box	
	BA11-L	BA11-A
S1	ON	ON
S2	ON	ON
S3	ON	OFF
S4	ON	ON

2.9.1 Power Check

Verify that the outlet(s) into which you will plug the PDP-11/24 processor is supplying the correct voltage that the computer system is configured for. If in doubt, consult the DIGITAL Field Service representative or a qualified electrician.

If the PDP-11/24 processor is part of a packaged system, place each of the power controller circuit breakers in the off (0) position (Figure 2-29). Insert each of the power cords into the appropriate outlets. The neon lamp(s) on each power controller should illuminate.

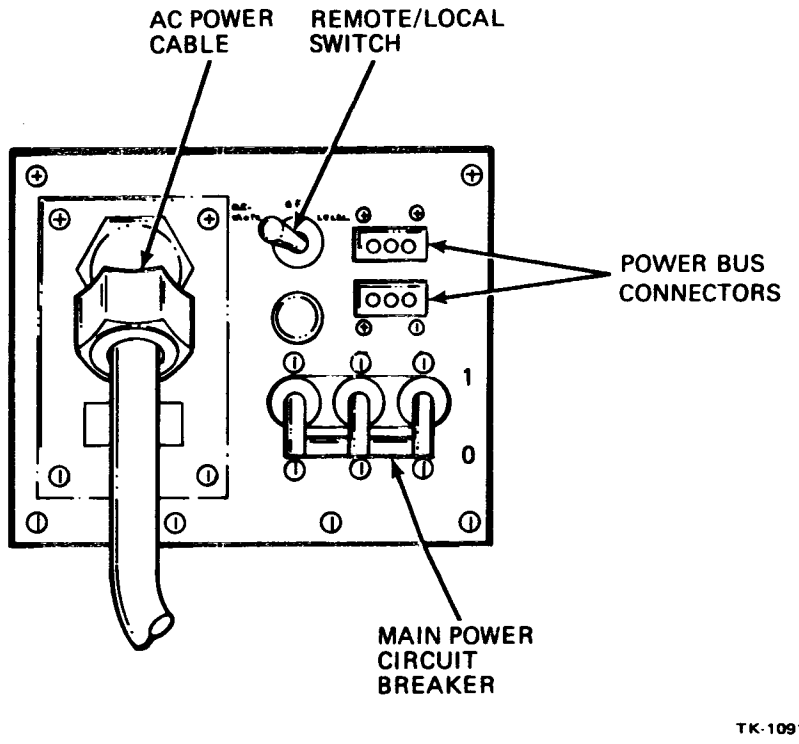


Figure 2-29 877 Power Controller – Front View

2.9.2 Power-up

Ensure that the toggle switch on each power control is in the REMOTE position. Ensure that the CPU keyswitch is in the DC OFF position and that each of the individual unit switches and/or circuit breakers is ON on all equipment in the system. The circuit breaker in the BA11-L mounting box is made accessible by pulling the CPU out of the wrap-around approximately 13 cm (5 in). The circuit breaker on the BA11-A mounting box is located at the back of the unit. Place each of the power controller circuit breakers in the on (1) position.

Place the CPU HALT/CONT/BOOT switch in the HALT position. Place the keyswitch in the LOCAL position. All units in the computer system should now power-up. If any units do not, verify that the power control bus is properly routed from the CPU to each power control in turn. Verify also that each power control is in the REMOTE position and that its circuit breaker is on (1). Verify that all unit switches and/or circuit breakers are on.

If the CPU has powered up successfully, the front-panel DC ON LED should be illuminated and the RUN light should have flashed. The CPU has now halted and the console terminal should display the ODT prompt:

```
nnnnnn
@
```

where nnnnnn is an octal number representing the updated program counter.

At this point, the user has complete control of the PDP-11/24 processor via console ODT. The following procedure is recommended, both to verify operation of the PDP-11/24 processor and to familiarize the user with its operation.

The PDP-11/24 processor contains diagnostics in the form of a ROM. These diagnostics are run whenever selected and the system is bootstrapped. They may be invoked in either of two ways:

1. Momentarily place the HALT/CONT/BOOT switch in the BOOT position, then return it to CONT.

2. Type 165000G at the console terminal.

Both of these actions cause the CPU to begin executing instructions starting at location 165000. These instructions verify correct operation of the CPU, the memory, and the console serial line (SLU1). When complete, the terminal prints out the system memory size. This should conform to the expected memory size. Table 2-16 provides the translation from the octal memory size to bytes and words.

Table 2-16 Memory Diagnostic Size Indication

Printout	Memory Size	
00400000	128K Bytes	64K Words
01000000	256K	128K
02000000	512K	256K
03000000	768K	384K
04000000	1024K	512K

The PDP-11/24 processor now attempts to boot the device selected by an internal switchpack. This is usually the main mass storage device on the system. Paragraphs 2.8.2 and 2.8.3 of this chapter contain information on the setup of this switchpack. The setting of the switchpack may be overridden by starting at other addresses. Refer to Paragraphs 2.8.2 and 2.8.3 for a description of this function.

At this point, the user should load the diagnostic media (typically RX02 floppy disk or RL02 cartridge disk) into drive 0 and take the necessary action to bring the media on-line. Write-protect the diagnostic media if possible. The selected device is bootstrapped as soon as it is ready and XXDP+, the diagnostic monitor, should present its introductory message.

The console terminal (typically) contains the following dialogue (text typed by the user is underlined):

```
165000
@165000G01000000

(The XXDP+ introductory message)
```

Diagnostics or chains of diagnostics may now be run. Specific information on the PDP-11/24 diagnostics is available in Chapter 9 of this manual. Information on XXDP+ is available in the *XXDP+ User's Guide* (AC-F348D-MC).

CHAPTER 3
PDP-11/24 OPERATION

3.1 CONSOLE OPERATION

PDP-11/24 system communication between the user and the system is provided by the operator's console (Figure 3-1) and the console terminal which emulates the functions found on the traditional "lights and switches" console. The PDP-11/24 operator's console has a minimum number of lights and switches which perform basic functions and provide system status information.

3.1.1 Operator's Console Switches

The operator's console contains two switches: power and HALT/CONT/BOOT. The function of each switch and its effect on system operation is explained in Table 3-1.

Table 3-1 PDP-11/24 Console Switches

Description	Function
Power (4-position rotary keyswitch)	
DC OFF	DC power is removed from the system; contents of MOS memory are lost and system cooling fans are off.
LOCAL*	Power is applied to the system; enables all functions and normal operation.
LOC DSBL	Power is present throughout the system. However, the HALT/CONT/BOOT switch is disabled and the "break" key on the terminal will not halt the CPU.
STDBY	DC power to most of the computer is off but dc power is applied to MOS memory to avoid data loss. **WARNING** **The DC OFF position does not remove ac power from the system. AC power is removed only by disconnecting the line cord.**
HALT/CONT/BOOT (3-position toggle switch)	
HALT	Halts the processor after all pending interrupts and aborts have been serviced.
CONT	Continue. The processor is enabled for normal operation; however, it will not begin executing instructions until commanded to from the console.
BOOT	Initializes the system. The processor will begin executing instructions contained in the hardware bootstrap.

* When the keyswitch is in the LOCAL position, a "BREAK" can halt the machine. Some terminals generate a "BREAK" without using the BREAK key (for example, on power-up, out of paper, going from on-line to off-line, and so on). To keep the machine from accidentally being halted, it is best to place the keyswitch in the LOC DISABLE position.

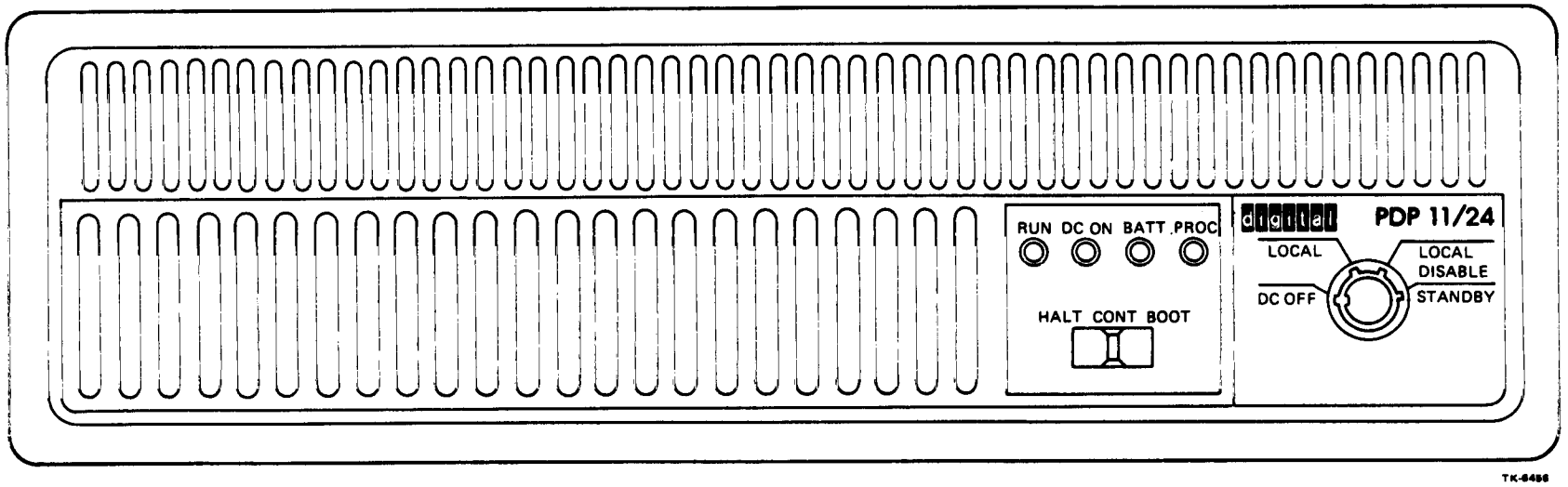


Figure 3-1 PDP-11/24 Operator's Console

When the BOOT switch is pressed or on powerup, the CPU begins executing instructions contained in the bootstrap ROMs. These instructions are diagnostics which test the CPU as well as other components in the system. On successful completion of the diagnostics, the instructions will bootstrap a device selected by the settings of the internal boot switches. If an error is detected by the diagnostics, the failing routine will cause a halt.

When the HALT/CONT/BOOT switch is in the HALT position, the system will boot up to ODT (Octal Debugging Technique). If the ODT prompt character (@) is not received on the console terminal, move the switch to the BOOT position, then to the HALT position, then back to the CONT position. Ordinarily, the switch should not be left in the HALT position.

3.1.2 Operator's Console Indicators

The four LEDs (RUN, DC ON, BATT and PROC) on the operator's console provide status information about the system (Table 3-2).

Table 3-2 PDP-11/24 Console LEDs

Description	Status	Function
RUN	On	Processor is fetching instructions.
	Off	Processor has halted or is waiting for an interrupt.
DC ON	On	Indicates that dc power is available but does not imply that the power is within the required levels unless the UNIBUS map option is installed.
	Flashing	The voltage monitor on the optional UNIBUS map module has detected a voltage out of tolerance.
	Off	Main +5 V is absent.
BATT	Off	Battery voltage is below the minimum level required to maintain the contents of MOS memory, battery backup is switched off, or the battery backup unit is not present in the system.
	Slow Flashing	Battery is charging and the voltage is above the minimum level required to maintain contents of MOS memory if power is removed. The amount of time that memory will be retained will depend on the amount of charge on the battery. The flash rate is fixed and does not vary with the charge rate of the battery.
	Fast Flashing	Indicates that primary power has been lost and the battery is discharging while maintaining MOS memory contents. The flash rate is fixed and does not indicate the charge level remaining on the battery.
	On Continuously	Battery is present and fully charged.
PROC	On	Processor is the current UNIBUS master.
	Off	Processor is not the current UNIBUS master.

3.2 CONSOLE TERMINAL

The console terminal provides the communication link between the user and the processor. Although the PDP-11/24 does not have a traditional "lights and switches" console, ODT provides an equivalent function. It allows the processor to respond to commands and information entered via an ASCII terminal. Communication between the user and the processor takes place via ASCII characters interpreted as console commands. The console terminal addresses lie between 777 560_8 and 777 566_8. These addresses are generated in microcode and cannot be changed. The hardware associated with the console terminal is described in Paragraph 4.7.

3.2.1 ODT Operation

The processor's microcode operates the serial line interface in half-duplex mode. Program I/O techniques are used rather than interrupts. When the ODT microcode is printing characters using the output side of the interface, the microcode is not monitoring the input side for incoming characters. Any characters coming in at this time may be lost. The UART chip may indicate overrun errors, but the microcode does not check any error bits in the interface, Therefore, users should not type ahead to ODT because those characters will not be recognized. In addition, if another processor is at the other end of the interface, it must obey half-duplex operation. No input characters should be sent until ODT has finished outputting.

Upon entry to ODT the RBUF register will be read using a DATI and the character present in the buffer will be ignored. This is done so that erroneous characters or user program characters are not interpreted by ODT as a command.

The input sequence for ODT is:

1. Test RCSR bit 7 (done flag) using a DATI, and if a zero, continue testing.

2. If RCSR bit 7 is a 1, read low byte of RBUF using a DATI.

The output sequence for ODT is:

1. Test XCSR bit 7 (done flag) using a DATI, and if a zero, continue testing.

2. If XCSR bit 7 is a 1, write XBUF using a DATO. The character is in the low byte and data in the high byte is undefined. (The high byte is ignored by the serial line unit.)

   If the interrupt enable (bit 6) in XCSR is set, an interrupt will be created to the software when the ODT command P is used. A G command will initialize the I/O system, clearing all the interrupt enables in peripherals and will not cause an interrupt.

3.2.2 ODT Entry Conditions

ODT mode may be entered by:

1. Executing a halt instruction in kernel mode, provided the enable kernel halt jumper is installed.

2. Placing the HALT/CONT/BOOT switch in the HALT position and the keyswitch in the LOCAL position.

3. Pressing the "break" key on the console terminal and placing the keyswitch in the LOCAL position.

Upon entry, ODT causes the following initialization:

1. Performs a DATI from RBUF (input data buffer at 777 562) and then ignores the character present in the buffer. This operation is done so that erroneous characters or user program characters are not interpreted by ODT as a command.

2. Prints a ⟨CR⟩ and ⟨LF⟩.

3. Prints the contents of the program counter (PC) in six digits.

4. Prints a ⟨CR⟩ and ⟨LF⟩.

5. Prints the prompt character, '@'.

6. Enters a wait loop for terminal input. The done flag (bit 7) in RCSR at 17 777 560_8 is constantly being tested via a DATI by the processor for a '1'. If it's a '0', the processor keeps testing.

3.3 ODT COMMAND SET

The ODT command set is described in the following paragraphs. The commands are a subset of ODT-11 and use the same command characters. Table 3-3 provides a list of the command set. ODT has 10 internal states. For each state, only specific characters are recognized as valid inputs; other inputs invoke a response of '?'. This is done to protect the user from accidentally destroying data via a syntax error. The ODT commands and states are described in Table 3-4.

Table 3-3 Console Command Language

ASCII Character	Octal Value	Function
/	057	A slash opens a bus address, processor register, or processor status word.
CR	015	Carriage Return closes an open location.
LF	012	Line Feed closes an open location and opens the next contigious location (bus addresses incremented by two, processor registers incremented by one).
$ or R	044 122	$ or R (Internal Register Designators), followed by a register number or S, opens that specific processor register (S represents the processor status word).
G	107	G (Go) starts program execution at the location entered immediately before G. Causes a bus initialization.
P	120	P (Proceed) resumes execution of a program without initializing the bus.
Control-S	023	Control-S (Binary Dump) displays a portion of memory more efficiently than a slash (/) or ⟨LF⟩ commands (not a normal user command).

The parity bit (bit 7) on all input characters is ignored (i.e., not stripped) by ODT, and if the input character is echoed, the state of the parity bit is copied to the output buffer (XBUF). Output characters internally generated (e.g., ⟨CR⟩) by ODT have the parity bit equal to zero. All commands are echoed except for ⟨LF⟩.

The following description of the ODT command set will sometimes refer to a command that has not been previously defined when defining another command. For this reason it is recommended that this section first be scanned for familiarization with the commands and then re-read for detail. The word "location," as used in this section, refers to a bus address, processor register, or processor status word (PSW). Bus addresses must be supplied as 18-bit addresses (i.e., the PSW, if accessed by its address, must be accessed via location 777 776 and not 177 776). When reading this section it should be noted that the user's entry will be shown underlined and the processor response will not be underlined.

/(ASCII 057_8) Slash

This command is used to open an address, processor register, or processor status word and is normally preceded by other characters which specify a location. In response to /, ODT will print the contents of the location (i.e., six characters) and then a space (ASCII 40). After printing is complete, ODT will wait for either new data for that location or a valid close command. The space character is issued to ensure the location's contents and possible new contents entered by the user are legible on the terminal.

Example: @001000/012525 ⟨SPACE⟩

where:

@ = ODT prompt character

001000 = Octal location in the bus address space desired by the user (leading zeros are not required)

/ = Command to open and print contents of location

012525 = Contents of octal location 1000

⟨SPACE⟩ = Space character generated by ODT

The / command can be used without a location specifier to verify the data just entered into a previously opened location. The / will be recognized only if entered immediately after a prompt character. A / issued immediately after the processor enters ODT mode will cause a ?⟨CR⟩, ⟨LF⟩, to be printed because a location has not been opened.

Example: @1000/012525 ⟨SPACE⟩ 1234 ⟨CR⟩⟨CR⟩⟨LF⟩
@/001234 ⟨SPACE⟩

where:

First line = New data of 1234 entered into location 1000 and location closed with ⟨CR⟩

Second line = A / was entered without a location specifier. The previous location was reopened to reveal that the new contents were correctly entered into memory.

CR (ASCII 015_8) Carriage Return
This command is used to close an open location. If a location's contents are to be changed, the user should precede the ⟨CR⟩ with the new data. If no change is desired, ⟨CR⟩ will close the location without altering its contents.

Example: @R2/004321 ⟨SPACE⟩ ⟨CR⟩ ⟨CR⟩ ⟨LF⟩
@

Processor register R2 was opened and no change was desired, so the user issued ⟨CR⟩. In response to the ⟨CR⟩, ODT echoed ⟨CR⟩ and printed ⟨CR⟩, ⟨LF⟩ and @.

Example: @R2/004321 ⟨SPACE⟩ 1234 ⟨CR⟩ ⟨CR⟩ ⟨LF⟩
@

In this case, the user desired to change R2. The new data, 1234, was entered before issuing the ⟨CR⟩. ODT deposited the new data in the open location and closed the location. ODT echoed the ⟨CR⟩ entered by the user and then printed ⟨CR⟩, ⟨LF⟩, and @.

LF (ASCII 012_8) Line Feed
This command is used to close an open location and then open the next contiguous location. Bus addresses and processor registers will be incremented by two and one, respectively. If the PS is open when a ⟨LF⟩ is issued, it will be closed and a ⟨CR⟩, ⟨LF⟩, @ will be printed; no new location will be opened. If the open location's contents are to be changed, the new data should precede the ⟨LF⟩. If no data is entered, the location is closed without being altered.

Example: @R2/123456 ⟨SPACE⟩ ⟨LF⟩ ⟨CR⟩ ⟨LF⟩
R3/054321 ⟨SPACE⟩

In this case, the user entered ⟨LF⟩ with no data preceding it. In response, ODT closed R2 and then opened R3. When a user has the last register, R7, open, and issues ⟨LF⟩, ODT will "rollover" to the beginning register, R0. When the user has the last address of a 32KW segment open and issues ⟨LF⟩, ODT will open the first location of that same segment. If the user wishes to cross the 32KW boundary, he must re-enter the address for the desired 32KW segment. (i.e., ODT is modulo 32KW). This operation is the same as that found on other PDP-11 consoles.

Example of register rollover: @R7/000000 ⟨SPACE⟩ ⟨LF⟩ ⟨CR⟩ ⟨LF⟩
R0/123456 ⟨SPACE⟩

Example of memory rollover: @577776/000001 ⟨SPACE⟩ ⟨LF⟩ ⟨CR⟩ ⟨LF⟩
400000/125252 ⟨SPACE⟩

Unlike other commands, ODT will not echo the ⟨LF⟩ (ASCII 012_8). Instead it will print ⟨CR⟩, then ⟨LF⟩ so that teletype printers operate properly. In order to make this easier for ODT to decode, ODT will also not echo ASCII 000_8, 002_8 or 010_8. It responds to these three characters with ? ⟨CR⟩, ⟨LF⟩, @.

$(ASCII 044_8) or R (ASCII 122_8) Internal Register Designator
Either character when followed by a register number, 0 to 7, or the PSW designator, S, will open that specific processor register.

The $ character is recognized to be compatible with ODT-11. The R character was introduced for the convenience of one keystroke and being representative of what it does.

Example: @$0/000123 ⟨SPACE⟩ ⟨CR⟩

Example: @R0/000123 ⟨SPACE⟩ ⟨LF⟩
@R1/054321 ⟨SPACE⟩

If more than one character (digit or S) is after the R or $, ODT will use the last character as the register designator. An exception occurs if the last register designator equals 077. Then, ODT will open the PSW rather than R7.

Stack Pointer
R6, the stack pointer, deserves special attention. Two stack pointers actually exist: one for the kernel mode and one for the user mode. The selection of which stack pointer is to be used is based on the current mode bits of the PSW (bits ⟨15:14⟩).

Example: @RS/ 140000 ⟨SP⟩⟨CR⟩⟨LF⟩ (The machine is in the user mode.)
@R6/ 123456 ⟨SP⟩⟨CR⟩⟨LF⟩ (The user stack pointer is examined.)
@RS/ 140000 ⟨SP⟩0⟨CR⟩⟨LF⟩ (The machine is forced into the kernel mode.)
@R6/ 001000 ⟨SP⟩⟨CR⟩⟨LF⟩ (The kernel stack pointer is examined.)
@RS/ 000000 ⟨SP⟩140000⟨CR⟩⟨LF⟩ (The machine is returned to the user mode.)

S (ASCII 123_8) Processor Status Word
This designator is for opening the processor status word (PSW) and can only be used after the user has entered an R or $ register designator.

Example: @RS/100377 ⟨SPACE⟩ 0 ⟨CR⟩ ⟨CR⟩ ⟨LF⟩
@/000020 ⟨SPACE⟩

Note the trace bit (bit 4) of the PSW cannot be modified by the user. This is done to ensure the PDP-11 program debug utilities (e.g., ODT-11), which use the T-bit for single stepping, will not be accidentally harmed by the user.

If the user issues a ⟨LF⟩ while the PSW is open, the PSW will be closed and ODT will print ⟨CR⟩, ⟨LF⟩ @. No new location is opened in this case.

G (ASCII 107_8) Go
This command is used to start program execution at a location entered immediately before the G. This function is equivalent to the LOAD ADDRESS and START switch sequence on other PDP-11 consoles.

Example: @200G ⟨NULL⟩ ⟨NULL⟩

The ODT sequence for a G command is as follows.

1. Print two nulls (ASCII 000_8) so that the bus initialize which follows will not flush the G character from the double-buffered UART chip in the serial line interface.

2. Load R7 (PC) with the entered data. If no data is entered, 0 is used. (In the example, R7 will equal 000 200, which is where program execution will begin.)

3. The PSW (and floating-point status register if the MMU is present) will be cleared to 0.

4. The bus will be initialized. Logic on the CPU module asserts BUS INIT for approximately 100 ms.

5. The processor will enter the service state. If there are any service conditions present, they will be processed. If the BUS HALT L signal is asserted, the processor will re-enter the ODT state. This feature can be practically used to initialize a system without starting a program (R7 is altered). To single step a program issue a G command and then successive P commands, all done with the HALT/CONT/BOOT switch in the HALT position.

P (ASCII 120_8) Proceed
This command is used to resume execution of a program and corresponds to the CONTINUE switch on other PDP-11 consoles. No programmer-visible machine state is altered using this command.

Example: @P

Program execution resumes at the place pointed to by R7. After the P is echoed, the ODT state is left and the processor immediately enters the state to fetch the next microinstruction. If the HALT signal is asserted, it will be recognized at the end of the instruction (during the service state) and the processor will enter the ODT state. Upon entry, the contents of the PC (R7) will be printed. In this fashion, a user can single-instruction step through a program and get a PC "trace" displayed on the terminal. It should be noted that if a reset instruction is executed during the time that P is being transmitted, the echo of the P may be lost or corrupted.

H (ASCII 110_8) Toggle Halt Command
The CPU may be single stepped by a second method. Instead of placing the HALT/CONT/BOOT switch in the HALT position, the user can type an H on the console terminal. This action has the effect of toggling the halt flip-flop located in the CPU. This action is the same as that done by setting the HALT/CONT/BOOT switch to the HALT position.

The CPU may now be single stepped by using the G and P commands, previously discussed. When exiting the single-step mode, type H and the halt flip-flop will be cleared.

To prevent accidental toggling of the halt flip-flop, the H that is typed must be uppercase (shift H). All other ODT commands may be typed in either upper or lowercase.

NOTE
No indication is made as to the state of the halt flip-flop. If the CPU does not seem to run, the halt flip-flop may be set or the HALT/CONT/BOOT switch may be in the HALT position.

Control-S (ASCII 023_8) Binary Dump
This command is used for manufacturing test purposes and is not a normal user command. It is described here to explain the machine's response if accidentally invoked. It is intended to more efficiently display a portion of memory, compared to using the slash (/) and ⟨LF⟩ commands. The protocol is as follows.

1. After a prompt character, ODT receives a Control-S command and echoes it.

2. The computer at the other end of the serial line must send two 8-bit bytes, which ODT interprets as a starting address. These two bytes are not echoed.

   The first byte specifies starting address ⟨15:08⟩ and the second byte specifies starting address ⟨07:00⟩. Bus address bits ⟨17:16⟩ are always forced to be zero; the dump command is restricted to the first 32K words of address space.

3. After the second address byte has been received, ODT will output to the serial line 10 bytes, starting at the address specified previously. When the output is finished, ODT will print ⟨CR⟩, ⟨LF⟩, @.

   If a user accidentally enters this command, it is recommended, in order to exit from the command, that two "@" characters (ASCII 100) be entered as a starting address. After the binary dump, the user will get a prompt character, @.

3.3.1 Address Specification

All I/O addresses (124K to 128K) must be entered as 18-bit addresses, regardless of whether the MMU is present or not. For example, if it is desired to open the RCSR of the SLU1, enter 777 560_8, not 177 560_8. Similarly, 18-bit addresses must be used to access memory greater than 32KW. Addresses 760 000_8–777 776_8 will map to the I/O page; memory at physical addresses 00 760 000_8–16 777 776_8 cannot be modified or examined via ODT.

3.3.2 Processor I/O Addresses

Certain processor and MMU registers have I/O addresses assigned to them for programming purposes. If address 777 776_8 is referenced via ODT, the PSW will respond. Processor registers R0 through R7 will not respond (i.e., time-out will occur) to bus addresses 777 700_8 through 777 707_8, whether using ODT or a program.

The MMU contains status registers and PAR/PDR pairs. These registers can be accessed from ODT by entering their bus address.

Example: @777572/000001 ⟨SPACE⟩

In this case, memory management status register 0 was opened and the memory management enable is seen to be set.

The FP11 accumulators, which are also in the MMU chip, cannot be accessed from ODT. Only FP11 instructions can access these registers.

3.3.3 Entering of Octal Digits

In general, when specifying an address or data, ODT will use the last six octal digits if more than six have been entered. Leading zeros do not need to be entered for either address or data; ODT forces zeros as the default.

3.3.4 ODT Time-Out

When a nonexistent address is specified, ODT will respond to the bus time-out by printing ?, ⟨CR⟩, ⟨LF⟩, @.

3.3.5 Memory Parity Errors

If the trap enable bit has been set in the memory parity CSR that controls the bank of memory being examined, ODT will respond to a location containing bad parity by typing ?⟨CR⟩⟨LF⟩@. The actual contents of the memory location may be read by specifically clearing the appropriate trap enable bit or by initializing the PDP-11/24 processor. The memory word may then be read with no indication of its bad parity.

3.3.6 Invalid Characters

Any character which ODT does not recognize during a particular sequence will be echoed (with the exception of ASCII 0, 2, 10, or 12 as noted earlier), and ODT will print a ?, ⟨CR⟩, ⟨LF⟩, @. ODT has 10 internal states. When in a state only specific characters are recognized. In this way, an attempt has been made to lower the probability of a user unintentionally destroying a program by pressing the wrong key. Table 3-4 defines the states and valid input characters.

Table 3-4 ODT States and Valid Input Characters

State	Example of Terminal Output	Valid Input	Comment
1	@	0–7 R, $ G P CTL-S /	Slash legal only if a location was previously opened.
2	@R OR @$	0–7 S	The user has indicated that a register will be selected.
3	@Ri OR @$i	0–7	A register has been selected.
4	@R1/123456	0–7 CR LF	The selected register has been opened.
5	@R1/123456 1000	0–7 CR LF	New data has been supplied as the register's contents.
6	@1000	0–7 / G	The user has indicated that a bus address is being selected.
7	@1000/123456	0–7 CR LF	The selected address has been opened.
8	@1000/123456 1000	0–7 CR LF	New data has been supplied as the location's contents.
9	@⟨CTL-S⟩	2 binary bytes	

3.3.7 Correcting Type-In Errors

ODT does not recognize the ⟨RUBOUT⟩ or ⟨DELETE⟩ character. If at any time the user realizes that a number has been incorrectly typed, the error may be corrected by typing the six correct digits. ODT recognizes only the last six digits typed. For example:

@0010 772340/ 000000 ⟨SPACE⟩(the user began to open location 001000 to open memory management PAR 0 instead).

3.4 PDP-11/24 REGISTERS

CPU and I/O device registers contained in the PDP-11/24 are located in the upper 4KW (I/O page) of the physical address space. Table 3-5 lists some of these registers and their addresses.

NOTE
The general-purpose registers in the PDP-11/24 cannot be accessed via a UNIBUS address.

Table 3-5 PDP-11/24 CPU and I/O Device Register Address

Address	Register
17 777 776	Processor Status Word
17 777 656 – 17 777 640	User Mode Page Address Registers
17 777 616 – 17 777 600	User Mode Page Descriptor Register
17 777 576	Memory Management Status Register 2 (SR2)
17 777 574	Memory Management Status Register 1 (SR1)
17 777 572	Memory Management Status Register 0 (SR0)
17 777 570	Display Register
17 777 566	SLU1 XBUF
17 777 564	SLU1 XCSR
17 777 562	SLU1 RBUF
17 777 560	SLU1 RCSR
17 777 546	LTC Status Register
17 776 506	SLU2 XBUF
17 776 504	SLU2 XCSR
17 776 502	SLU2 RBUF
17 776 500	SLU2 RCSR
17 772 516	Memory Management Status Register 3 (SR3)
17 772 356 – 17 772 340	Kernel Mode Page Address Registers
17 772 316 – 17 772 300	Kernel Page Descriptor Registers

Note: The general purpose registers in the PDP-11/24 cannot be accessed via a UNIBUS address.

3.4.1 Processor Status Word

The processor status word (PSW) contains information on the current status of the PDP-11/24. This information includes: the current processor priority, current and previous operational modes, the condition codes describing the results of the last instruction, an indicator for detecting the execution of an instruction to be trapped during program debugging (trap bit), and a bit indicating the temporary suspension of a CIS instruction. Figure 3-2 shows the format of the PSW, and Table 3-6 describes the functions of each bit.

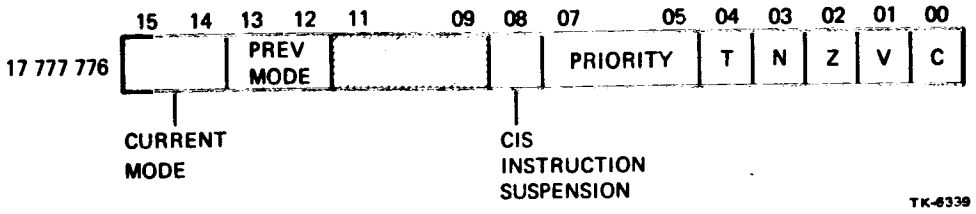


Figure 3-2 Processor Status Word

Table 3-6 Processor Status Word (PSW) Bit Descriptions

Bit	Description	Comment
15:14	Current mode	Contains current mode code; used to determine current operational mode. PSW⟨15:14⟩ = :00 Kernel Mode PSW⟨15:14⟩ = :01 Unused PSW⟨15:14⟩ = :10 Unused PSW⟨15:14⟩ = :11 User Mode
13:12	Previous mode	Contains previous mode code; used to determine last operational mode. PSW⟨13:13⟩ = :00 Kernel Mode PSW⟨13:12⟩ = :01 Unused PSW⟨13:12⟩ = :10 Unused PSW⟨13:12⟩ = :11 User Mode
11:09	Unused	Unused PSW bits will read as zeros and cannot be written into.
08	CIS instruction suspension	A CIS instruction is temporarily suspended while an interrupt is serviced.
07:05	Priority	Contains interrupt priority level; the CPU will not service interrupts at or below its priority level.
04	T-Bit	Contains trace bit; after execution of the next instruction, the processor will trap through the vector at 14_8. This is useful for tracing program flow.
03:00	Condition codes	The N, Z, V, and C condition codes are updated at the end of each instruction. These flags can be tested by the conditional branch instructions to control the flow of the program.

3.4.2 Serial Line Unit 1 Registers

Serial line unit 1 (SLU1) is used by the console terminal of the PDP-11/24. The following paragraphs describe the functions of the various registers associated with SLU1.

3.4.2.1 SLU1 Receiver Control/Status Register (RCSR) – Figure 3-3 shows the format of the SLU1 receiver control/status register (RCSR), and Table 3-7 lists and describes the functions of the bits.

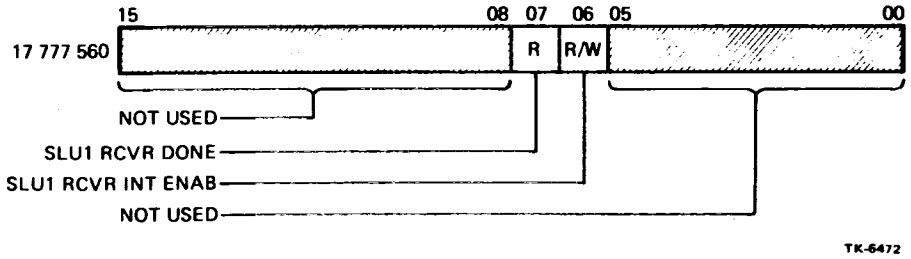


Figure 3-3 SLU1 RCSR Format

Table 3-7 SLU1 RCSR Bit Descriptions

Bit	Description
15:08	Not used.
07	SLU1 Receiver Done – A read-only bit that is set to 1 when a complete character is contained in the SLU1 RBUF. Cleared when the RBUF is addressed or when an initialize operation occurs.
06	SLU1 Receiver Interrupt Enable – A read/write bit set to 1 to allow the interrupt sequence to be initiated when the RCVR DONE bit is set.
05:00	Not used.

3.4.2.2 SLU1 Receiver Data Buffer (RBUF) – Figure 3-4 shows the format of the SLU1 receiver data buffer register, and Table 3-8 lists and describes the functions of the bits.

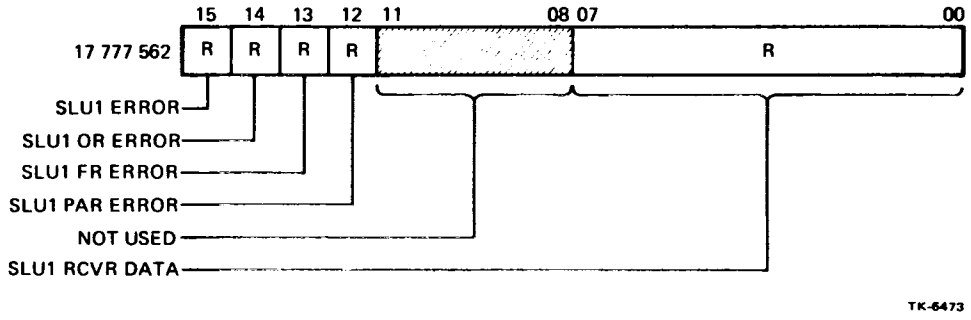


Figure 3-4 SLU1 RBUF Format

Table 3-8 SLU1 RBUF Bit Descriptions

Bit	Description
15	SLU1 Error – A read-only bit that is set to 1 when the SLU1 OR ERROR (bit 14), the SLU1 FR ERROR (bit 13), or the SLU1 PAR ERROR (bit 12) is set to 1. Cleared when the specific error bit(s) is (are) cleared.
14	SLU1 Overrun Error – A read-only bit that is set to 1 if the character in the RBUF has not been read before another character is received. Cleared by an initialize operation or when the RBUF is emptied.
13	SLU1 Framing Error – A read-only bit that is set to 1 when the character read does not include a valid stop bit(s). Cleared when a valid character is received. This bit may indicate an error in transmission or the reception of a "break" character.
12	SLU1 Parity Error – A read-only bit that is set to 1 when the parity of the data in the RBUF is incorrect relative to the parity mode selected. This indicates an error in transmission. Cleared when the parity of the next character is validated.
11:08	Not used.
07:00	SLU1 Receiver Data – Read-only bits that are the data character that was read from the terminal.

3.4.2.3 SLU1 Transmitter Control/Status Register (XCSR) – Figure 3-5 shows the format of the SLU1 transmitter control and status register (XCSR), and Table 3-9 lists the functions of the bits.

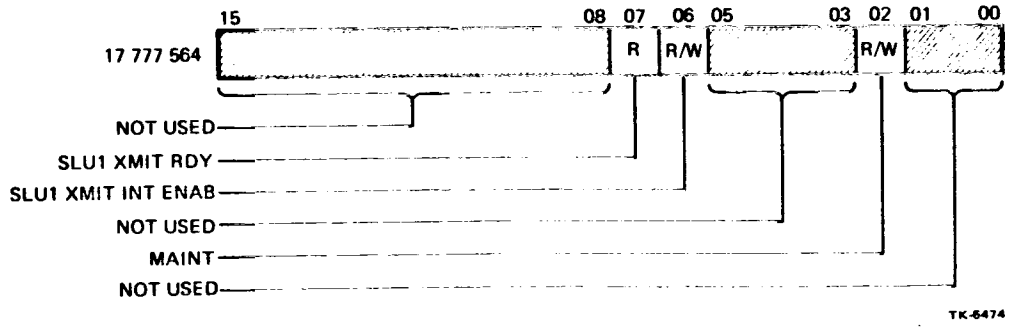


Figure 3-5 SLU1 XCSR Format

Table 3-9 SLU1 XCSR Bit Descriptions

Bit	Description
15:08	Not used.
07	Transmitter Ready – A read-only bit that is set to 1 when the console terminal XBUF register is ready to accept a character or when an initialize operation occurs. It initiates the interrupt sequence if the SLU1 XMIT INT ENB (bit 06) is set to 1. Cleared when the XBUF receives a character.
06	Transmitter Interrupt Enable – A read/write bit that is set to 1 by the program to enable the interrupt sequence to be initiated if the SLU1 XMIT RDY (bit 07) is set to 1. Cleared by program or by the initialize sequence.
05:03	Not used.
02	Maintenance – A read/write bit which, when set to 1 by the program, will cause a closed loop test of the SLU1 UART. The serial output of the XBUF will be returned to serial input of the RBUF. The data transfer rate will be 19200 baud. Cleared by an initialize operation or by the program.
01:00	Not used.

3.4.2.4 SLU1 Transmitter Buffer Register (XBUF) – Figure 3-6 shows the format of the transmitter buffer register (XBUF), and Table 3-10 lists the functions of the bits.

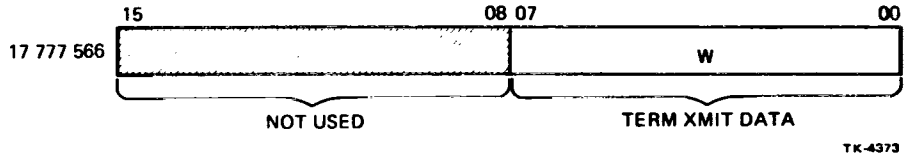


Figure 3-6 SLU1 XBUF Format

Table 3-10 SLU1 XBUF Bit Descriptions

Bit	Description
15:08	Not used.
07:00	Terminal Transmitter Data – These are write-only bits which form the data character to be transferred to the console terminal.

3.4.3 Serial Line Unit 2 Registers

Serial line unit 2 (SLU2) is a general-purpose serial line unit which may be used for a second terminal, a serial line printer, a TU58 DECtape II cartridge tape drive, or other similar devices. The following paragraphs describe the functions of the various registers associated with SLU2.

3.4.3.1 SLU2 Receiver Control/Status Register (RCSR) – Figure 3-7 shows the format of the SLU2 receiver control/status register (RCSR), and Table 3-11 lists the functions of the bits.

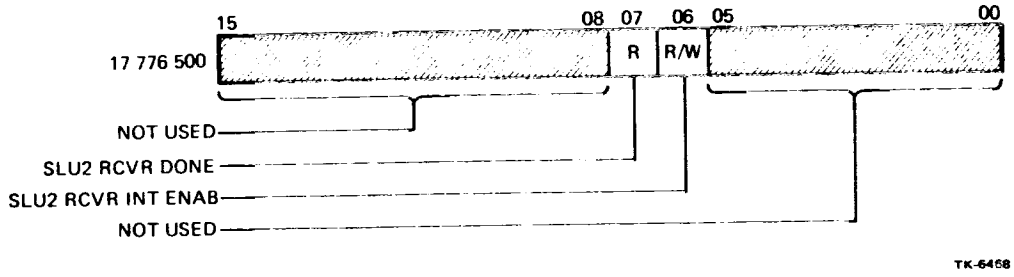


Figure 3-7 SLU2 RCSR Format

Table 3-11 SLU2 RCSR Bit Descriptions

Bit	Description
15:08	Not used.
07	SLU2 Receiver Done – A read-only bit that is set to 1 when a complete character is contained in the SLU2 RBUF. Cleared when the SLU2 RBUF is addressed or when an initialize operation occurs. Initiates the interrupt sequence when the SLU2 RCVR INT ENAB bit (06) is set to 1.
06	SLU2 Receiver Interrupt Enable – A read/write bit which is set to 1 by the program to allow the interrupt sequence to be initiated by the SLU2 RCVR DONE bit (07).
05:00	Not used.

3.4.3.2 SLU2 Receiver Buffer Register (RBUF) – Figure 3-8 shows the format of the SLU2 receiver buffer register (RBUF), and Table 3-12 lists the functions of the bits.

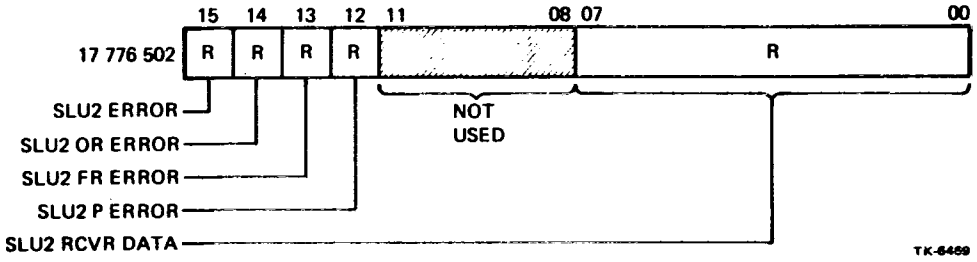


Figure 3-8 SLU2 RBUF Format

Table 3-12 SLU2 RBUF Bit Descriptions

Bit	Description
15	SLU2 Error – A read-only bit that is set to 1 when the SLU2 OR ERROR (bit 14) FR ERROR (bit 13), or the SLU2 PAR ERROR (bit 12) is set to 1. Cleared when the specific error bit(s) is (are) cleared.
14	SLU2 Overrun Error – A read-only bit that is set to 1 if the character in the RBUF has not been read before another character is received. Cleared by an initialize operation or when the RBUF is emptied.
13	SLU2 Framing Error – A read-only bit that is set to 1 when the character read in the RBUF does not have a valid stop bit. Cleared when a valid character is received. This bit may indicate an error in transmission or the reception of a "break" character.
12	SLU2 Parity Error – A read-only bit that is set to 1 when the parity of the character read in the RBUF is incorrect relative to the parity mode selected. Cleared when the parity of the next character is validated.
11:08	Not used.
07:00	SLU2 Received Data – These are read-only bits that form the data character received.

3.4.3.3 SLU2 Transmitter Control/Status Register (XCSR) – Figure 3-9 shows the format of the SLU2 transmitter control/status register (XCSR), and Table 3-13 lists the functions of the bits.

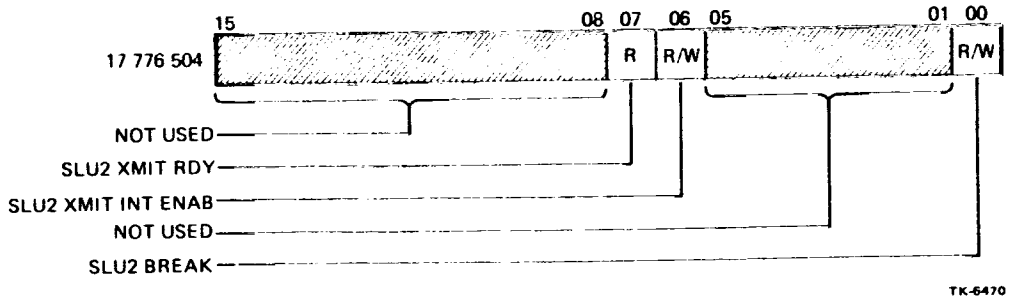


Figure 3-9 SLU2 XCSR Format

Table 3-13 SLU2 XCSR Bit Descriptions

Bit	Description
15:08	Not used.
07	SLU2 Transmitter Ready – A read-only bit that is set to 1 when the SLU2 XBUF is ready to accept a character or when an initialize operation occurs. Setting the bit initiates an interrupt sequence if the SLU2 XMIT ENAB (bit 06) is set to 1. Cleared when a character is written into the XBUF.
06	SLU2 Transmitter Interrupt Enable – A read/write bit that is set to 1 by the program to enable the interrupt sequence if the SLU2 XMIT RDY (bit 07) is set to 1. Cleared by the program or by the initialize sequence.
05:01	Not used.
00	SLU2 Send Break – A read/write bit that is set to 1 by the program and that causes the serial output to be forced to the "spacing" condition. If this condition is held long enough, it will be recognized as a break. Cleared by the program or by an initialize sequence.

3.4.3.4 SLU2 Transmitter Data Buffer Register (XBUF) – Figure 3-10 shows the format of the SLU2 transmitter buffer register (XBUF), and Table 3-14 lists the functions of the bits.

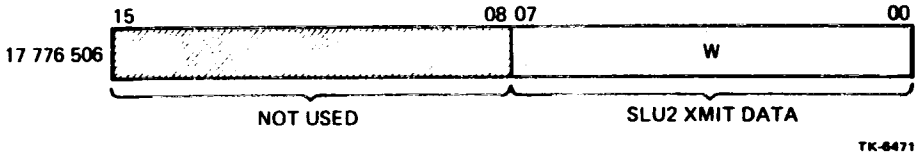


Figure 3-10 SLU2 XBUF Format

Table 3-14 SLU2 XBUF Bit Descriptions

Bit	Description
15:08	Not used.
07:00	SLU2 Transmitter Data – These are write-only bits that form the data character to be transferred.

3.4.4 Line Time Clock Control/Status Register (LTC)
Figure 3-11 shows the format of the line time clock control status register (LTC), and Table 3-15 lists the functions of the bits.

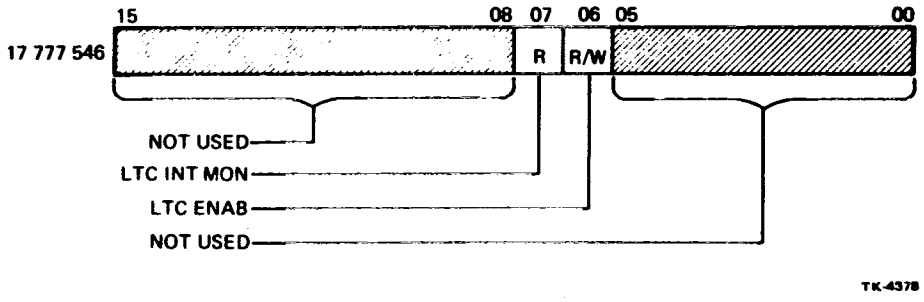


Figure 3-11 Line Time Clock Format

Table 3-15 Line Time Clock Status Register Bit Descriptions

Bit	Description
15:08	Not used.
07	Line Time Clock Monitor – A read-only bit set to 1 for each cycle of the ac voltage and cleared by the program. Provides timing information at an interval of 16.66 ms on a 60 Hz power-line and 20 ms on a 50 Hz power-line. Also set during the initialize sequence. Cleared automatically after an LTC interrupt.
06	Line Time Clock Interrupt Enable – A read-write bit set to 1 by the program to allow the interrupt sequence to be initiated when the LTC MON (bit 07) is set.
05:00	Not used.

3.4.5 Memory Management Registers

The 16-bit virtual address is translated to a 22-bit physical address by the memory management function. Four status registers, 16 page address registers (PARs), and 16 page descriptor registers (PDRs) are associated with memory management.

3.4.5.1 Status Register 0 (SR0) – Memory management status register 0 (SR0) contains error flags, the page number whose reference caused the abort, and various status flags. The format of SR0 is shown in Figure 3-12, and bit descriptions are listed in Table 3-16.

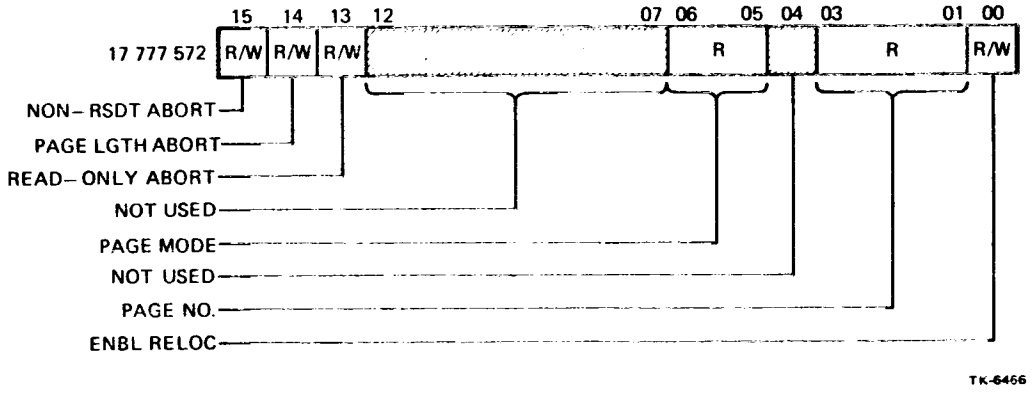


Figure 3-12 Memory Management SR0 Format

Table 3-16 SR0 Bit Descriptions

Bit	Description
15:13	Error Flags – These error bits are prioritized; i.e., flags to the right are less significant and are to be ignored if a flag to the left is present. For example, a nonresident fault service routine would ignore page length and access control faults.
15	Nonresident Abort – This bit is set to 1 when an attempt to access a page with an access control field (ACF) key is equal to 0. It is also set if there is an attempt to use memory relocation with a processor mode of 2 or 1.
14	Page Length Abort – This bit is set to 1 if there is an attempt to access a location in a page with a block number (virtual address bits 12:06) that is outside the area authorized by the page length field (PLF) of the page descriptor register (PDR) for that page. It is also set if there is an attempt to use memory relocation with a processor mode of 2 or 1. Bits 15 and 14 can be set simultaneously by the same access attempt.
13	Read-Only Abort – This bit is set if there is an attempt to write in a read-only page. Read-only pages have an access key of 1.
12:07	Not used.
06:05	Page Mode – These bits indicate the processor mode (kernel/user) associated with the page causing the abort: kernel = 00, illegal mode = 01, illegal mode = 10, user = 11. If an illegal mode is specified, bits 15 and 14 will be set.
04	Not used.
03:01	Page Number – These bits contain the page number of the reference causing a memory management fault.
00	Enable Relocation – When this bit is set, all addresses are relocated. When this bit is clear, the memory management facility is inoperative and addresses are not relocated or protected.

3.4.5.2 Status Register 1 (SR1) – This register is not used by the PDP-11/24. If an attempt to read this register is made, the register is read as 000 000. This register cannot be written into. This register is included to avoid system bus errors if SR1 is referenced.

3.4.5.3 Status Register 2 (SR2) – SR2 is loaded with the 16-bit PC received from the data chip. This occurs on all address cycles when the control information indicates that the current address on the bus is that of an instruction. The contents of SR2 are frozen when an abort flag is set. SR2 is a read-only status register. Figure 3-13 shows the format of SR2.

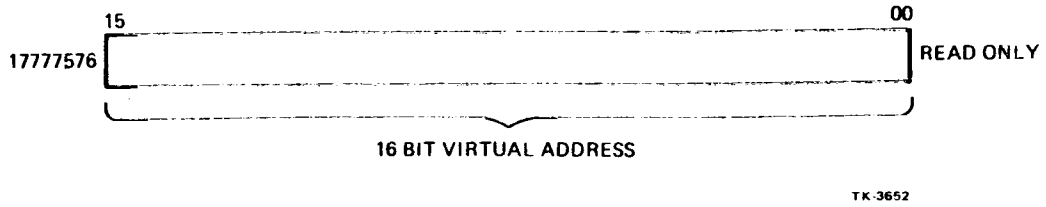


Figure 3-13 Memory Management SR2 Format

3.4.5.4 Status Register 3 (SR3) – The PDP-11/24 only uses two bits of this register, bits 5 and 4. When cleared, bit ⟨05⟩ disables the optional UNIBUS mapping logic and when set, enables the UNIBUS map. When cleared, bit ⟨04⟩ selects 18-bit mapping mode within the MMU and when set, selects 22-bit mapping mode in the MMU. This bit is used internally by the MMU. All other bits in the register are read as 0s. Figure 3-14 shows the format of SR3.

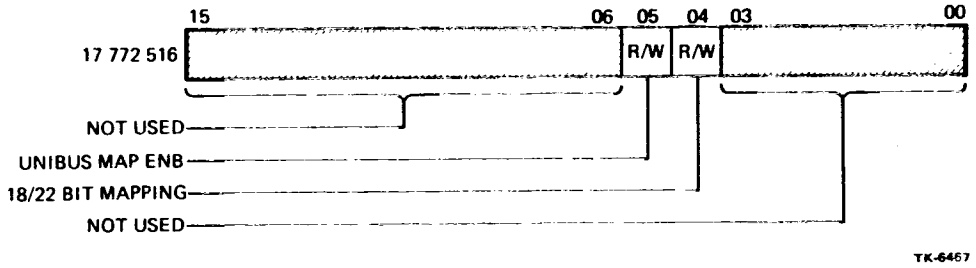


Figure 3-14 Memory Management SR3 Format

3.4.6 Display Register

The display register is a write-only register located at address 17 777 570. Any attempt to read from this register will result in a trap through the vector at location 4. This means that instructions which perform a DATIP/DATO sequence (INC, DEC, and CLR) may not be used to access the display register. When accessing the display register, an MOV instruction is the preferred instruction. This type of access is in keeping with the operation of the PDP-11/70. This register uses only bits 1 and 0 (Figure 3-15) which, when set, light two LEDs located on the CPU Module (Figure 3-16). The display register is used by the power-up diagnostics as they run. Three different sections of the PDP-11/24 are tested by these diagnostics, and the display register LEDs are used to indicate the condition of each test. Table 3-17 describes the functions of the LEDs. (See Appendix D, Section D.2, Figure D-1 for CPU module M7133-YA LED locations.)

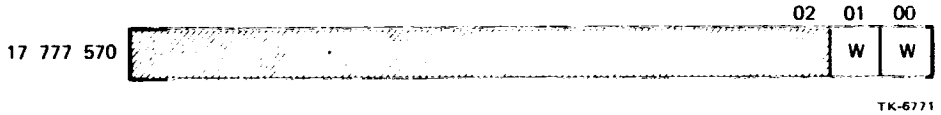


Figure 3-15 Display Register

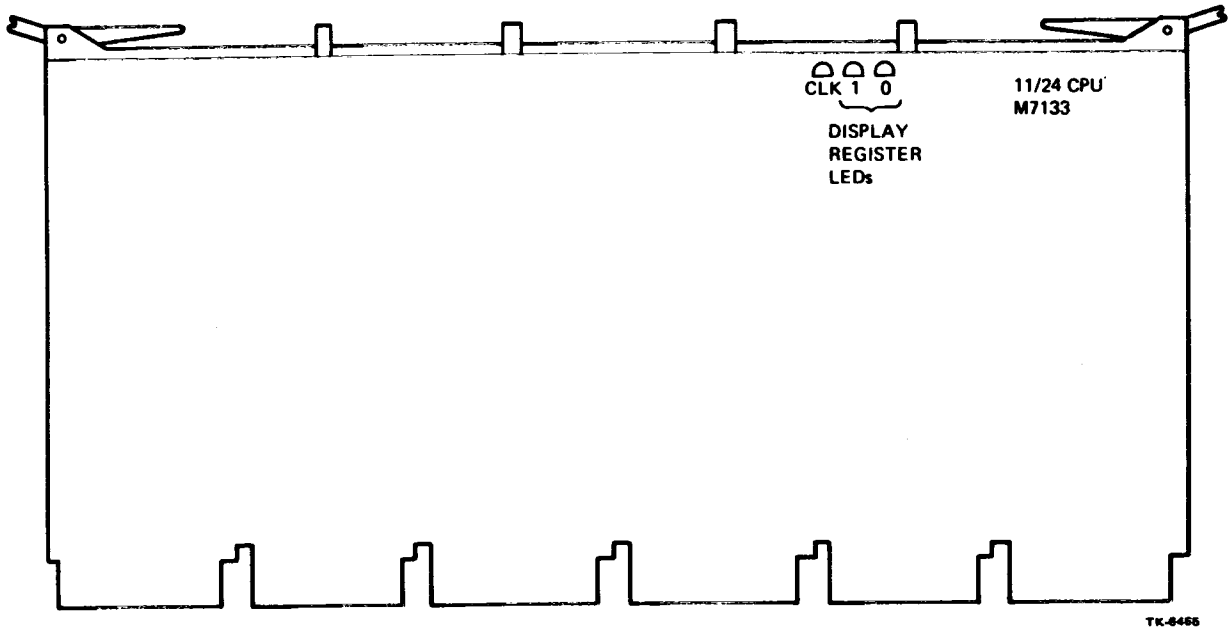


Figure 3-16 Display Register LED Location

Table 3-17 Display Register LED Description

LED Condition 1	LED Condition 0	Description
OFF	OFF	The diagnostic has not begun.
ON	ON	The processor test has begun.
ON	OFF	The processor test has successfully completed. Memory and memory management are now being tested.
OFF	ON	The memory test has successfully completed and memory has been cleared to 0s. SLU1 is now being tested.
OFF	OFF	All tests successfully completed. The console terminal will now print the memory size.

CHAPTER 4
CPU FUNCTIONAL DESCRIPTION

4.1 INTRODUCTION

This chapter provides a functional description of the PDP-11/24 central processing unit and associated logic contained on the KDF11-UA (M7133) module. Simplified block diagrams are used to illustrate various sections of the logic. The print set will be referenced throughout the discussion.

4.1.1 Central Processing Unit

The central processing unit in the PDP-11/24 is contained on two silicon chips. These two chips, the data chip and the control chip, contain the data paths, microstore, and microsequencer. The two chips are mounted on a multilayer, ceramic, 40-pin hybrid. This 40-pin hybrid, called the DAT/CTL hybrid, is then plugged into the PDP-11/24 CPU module (M7133).

All the basic PDP-11 processor functions are contained on this 40-pin hybrid assembly. The data chip contains the PDP-11 register set, ALU, shifter, literal generation, conditional microbranching logic, and other functions associated with the data paths of a PDP-11 processor. The control chip contains the microprogram and the necessary microsequencing logic.

The DAT/CTL hybrid contains the necessary microcode and logic to execute:

1. The basic PDP-11 instruction set
2. The extended instruction set (EIS)
3. The ODT console emulation

The DAT/CTL hybrid, along with its associated timing and interface logic, makes up the base machine of the PDP-11/24 (Figure 4-1).

4.1.2 Processor Options

The PDP-11/24 base machine can be expanded by the addition of options which extend the microcode, the data paths or a combination of both.

4.1.2.1 Memory Management Unit – The memory management unit (MMU) is an extension of the data paths contained in the base machine (Figure 4-2). The MMU allows regions of memory to be relocated and/or protected. The MMU is contained on one silicon chip mounted in a conventional 40-pin IC.

The MMU is installed in every PDP-11/24 and is considered optional because it is not needed for the base machine to function.

The MMU contains the registers, adders, and comparitors needed to generate a 22-bit physical address (PA) from the base machine's 16-bit virtual address (VA). The 22-bit address is transferred on the 16-bit wide data/address lines (DAL) plus 6-bit wide extension of the DAL (Figure 4-2).

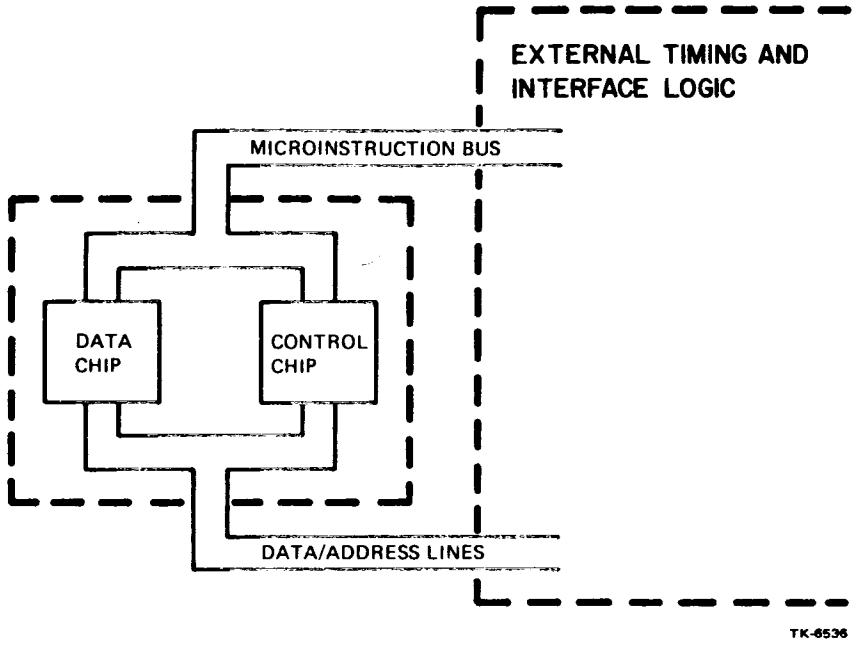


Figure 4-1 Base Machine

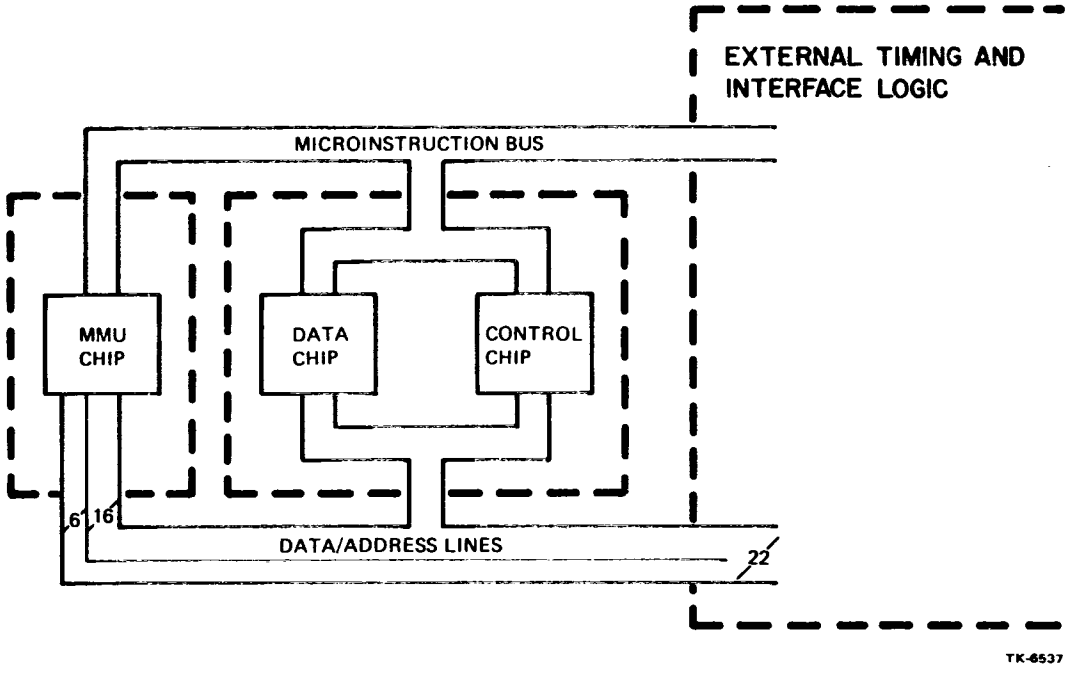


Figure 4-2 MMU and Base Machine

For a detailed explanation of how the MMU functions refer to Chapter 5, Memory Management, of this manual.

In addition to the memory management function of the MMU, the MMU contains the additional registers required by the floating-point processor.

4.1.2.2 Floating-Point Processor – The floating-point processor (FPP) is an extension of the data paths and microcode of the basic PDP-11/24 processor (Figure 4-3). The FPP requires additional registers, which are contained in the MMU. The additional microcode and microsequencing logic required by the FPP is contained on two control chips mounted on a 40-pin ceramic hybrid IC. FPP instructions are executed by the FPP microcode using the DATA chip and additional storage registers contained in the MMU. In normal operation the function of the data chip is controlled by the microinstructions contained in the control chip of the DAT/CTL hybrid in the base machine. The base machine control chip will assert its chip select line (CSEL 1 L) to indicate that it is selected and in control of the machine. When a floating-point instruction is decoded, the base machine control chip issues a jump microinstruction to FPP control chip 1 and releases the chip select line. FPP control chip 1 recognizes its address in the jump microinstruction and becomes active. The FPP control chip indicates it is active by asserting its chip select line. Processing of the floating-point instruction continues under the control of the first floating-point control chip. If necessary, control will be passed to the second FPP control chip. When the instruction is complete, control is passed back to the base machine control chip. Only one control chip is in use at any time.

When the FPP is not installed or is not functioning properly, the address in the jump microinstruction will not be recognized and the chip select line will not be asserted. When the chip select line is not asserted, the external timing logic recognizes this as an error condition and will force a chip reset. The action forces the base machine control chip to take control of the machine. The base machine will then examine the service register and, noting that a CTL error existed, force a trap through the vector at 10_8. This trap notifies the user's program that a FPP instruction cannot be executed.

4.1.2.3 Commercial Instruction Set – The commercial instruction set (CIS) is a microcode-only extension of the PDP-11/24's functions (Figure 4-4). CIS does not require the presence of the memory management chip. The CIS provides decimal arithmetic and byte-manipulation instructions. Six additional control chips contain the additional microcode used in the execution of the CIS. The CIS control chips are mounted on a double-width, 40-pin, ceramic hybrid. Control is passed between the CIS control chips in the same way as it is in the FPP.

4.2 TIMING

The PDP-11/24 timing provides the enabling lines for various control signals to the chip set as well as other logic on the module. These timing signals are also used to indicate what type of information will be present on the microinstruction bus (MIB) or the data/address lines (DAL). The PDP-11/24 has the following timing cycles: short cycle, address cycle, address relocation cycle, data in cycle and data out cycle. (See Appendix D, Section D.3 for changes in signal names on the circuit schematic for M7133-YA CPU version.)

K1 OSC H (75 ns) clocks a counter which generates K1 MCLK H. K1 MCLK H is divided into two time states, phase time (PT1 through PT4) and phase bar time (PBT1 through PBT4). Phase time is asserted when K1 MCLK H is high, and phase bar time is asserted when K1 MCLK H is low. Table 4-1 describes the various time states and Figure 4-5 illustrates the timing flow for PDP-11/24 operations.

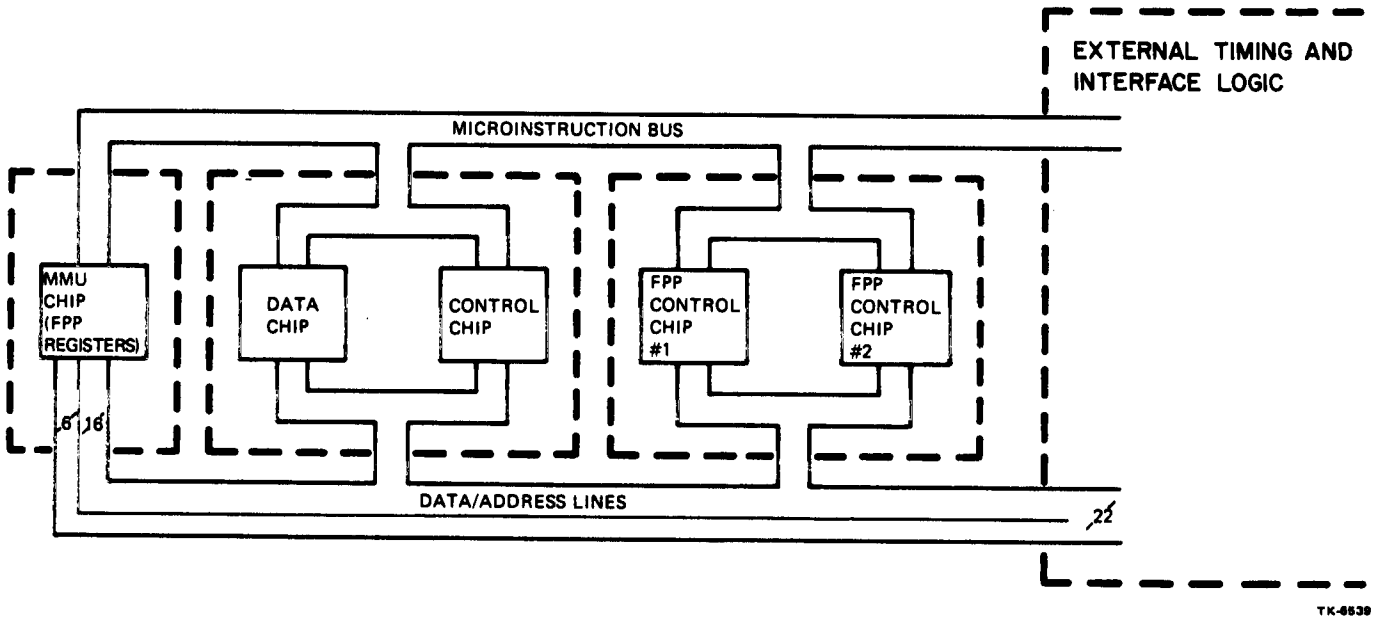


Figure 4-3 FPP and Base Machine

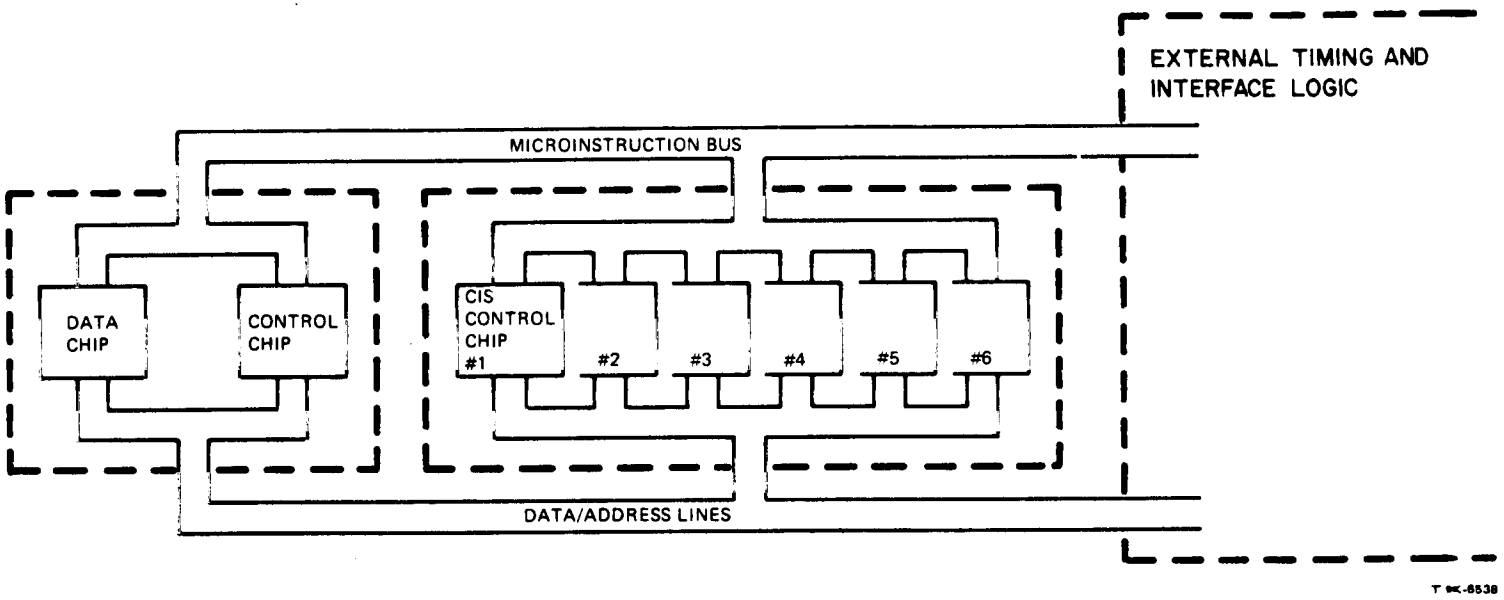


Figure 4-4 CIS and Base Machine

Table 4-1 PDP-11/24 Timing States

Timing State	Description
PT1 (Phase Time 1)	This is the beginning of the microinstruction execution. PT2 immediately follows.
PT2 (Phase Time 2)	This is the second state for microinstruction execution. If the microinstruction does not specify an address operation or external data operation, this is the final execute microinstruction state. If this is an address microcycle, execution continues in PT3. If an external data operation is to be performed and the data cycle requires the UNIBUS or EUB, the processor will wait in PT2 until the UNIBUS or EUB is available. Execution will then continue in PT3.
PT3 (Phase Time 3)	This state is used to wait for slave-sync or allow additional time for the assertion of a virtual address. It is entered only if an address microinstruction or data-type microinstruction is being executed. If the microinstruction is a DATI (data-in), PT4 is asserted next. Any other microinstruction causes PBT1 to be asserted after PT3. This state will be stretched to wait for slave-sync from a device if this is a data-type microinstruction.
PT4 (Phase Time 4)	This state is used to provide UNIBUS data deskew. It is at the end of this time state that data from the UNIBUS is latched into the processor.
PBT1 (Phase Bar Time 1)	This is the beginning of a microinstruction fetch. Also, during this time state the MMU may be translating a virtual address into the appropriate physical address. PBT2 immediately follows.
PBT2 (Phase Bar Time 2)	This is the second state for microinstruction fetch. If no address relocation is taking place PT1 immediately follows. If an address relocation cycle is enabled PBT3 immediately follows. A microjump to a nonexistent control chip will result in PBT3 being entered.
PBT3 (Phase Bar Time 3)	The translated physical address from the MMU is available on the DAL. PBT4 immediately follows.
PBT4 (Phase Bar Time 4)	PBT4 is asserted during a relocation cycle to allow time for the translated physical address from the MMU to be read. PBT4 is also asserted when an abort or reset condition occurs. When an abort or reset condition occurs, PBT4 provides the time needed to re-establish normal micromachine operation. PT1 immediately follows.

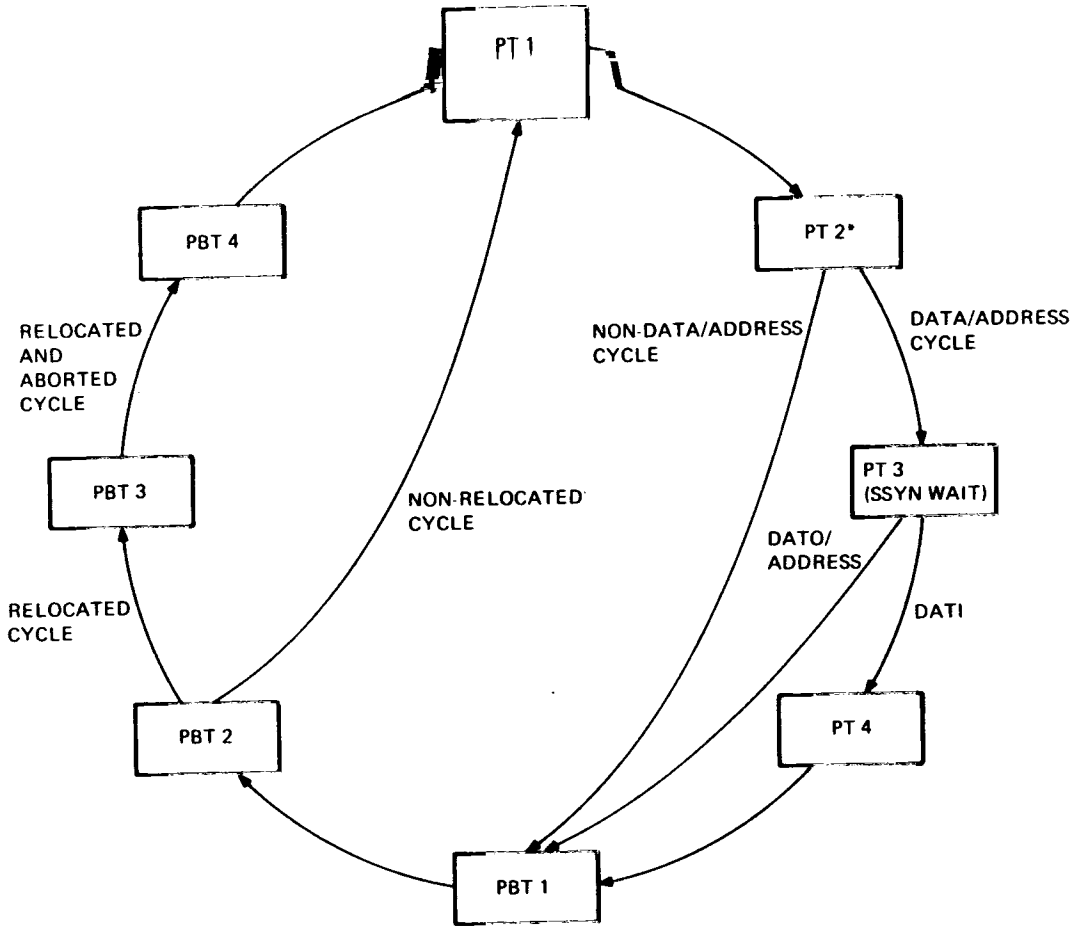


Figure 4-5 PDP-11/24 Timing Flow

4.2.1 Timing Cycles

The various timing cycles used in the PDP-11/24 are explained in the following paragraphs.

4.2.1.1 Short Cycle – The basic timing cycle used in the PDP-11/24 is the short cycle. A short cycle is 300 ns long and consists of PT1, PT2, PBT1 and PBT2. A short cycle is executed when there is no need to stop the base machine (i.e., no external I/O, no error conditions, etc.). Figure 4-6 illustrates a short cycle. The execution of microinstructions is pipelined; therefore, two short cycles are illustrated.

The execution of a short cycle starts during PBT of the previous cycle when the microinstruction is read from the control store and is asserted on the microinstruction bus (MIB). The microinstruction is then latched into the microinstruction register (MIR) on the transition from the previous cycle's PBT to the current cycle's PT. After the microinstruction is latched, it is replaced on the MIB by control information generated by the microinstruction. The control information is then used to control the internal and external logic in the processing of the data on the data/address lines (DAL). At the end of PT2 the data on the DAL is latched into the data chip. During PBT1 and PBT2 the DAL is supplied with service information and fetches the next microinstruction. The microcode can then use the information on the DAL to modify the microaddress used to fetch the next microinstruction.

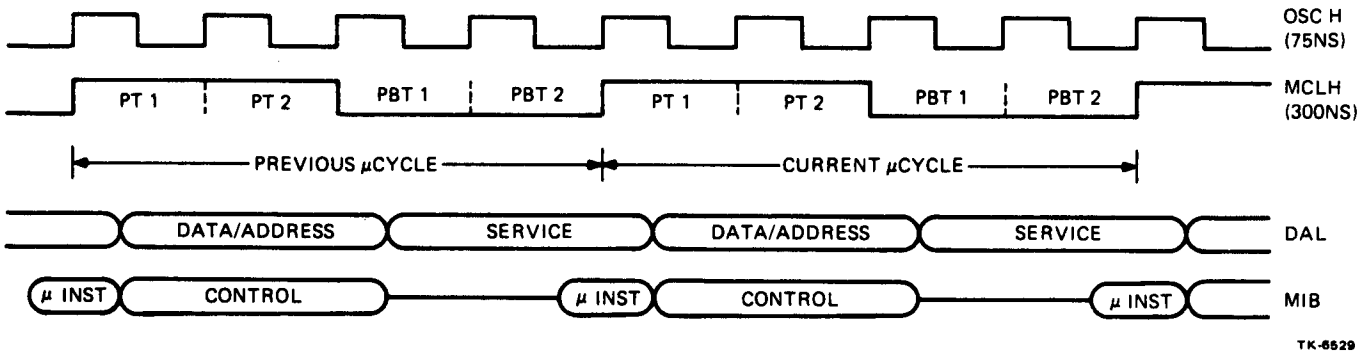


Figure 4-6 Pipelined Flow of Short Cycle

4.2.1.2 Input/Output Cycles – Timing cycles that have input or output (I/O) to the system bus require additional time to complete the transfer process. Two types of I/O cycles exist:

1. Address Cycle – This cycle specifies the address and control information.
2. Data Cycle – This cycle performs the actual transfer of data.

The address cycle will be described first and then the data cycle.

Address Cycle – The address cycle specifies the address and type of data cycle to be performed. There are two types of address cycles, nonrelocated and relocated. The cycle to be used is determined by whether the memory management unit (MMU) is enabled or disabled. When the MMU is enabled it takes the 16-bit virtual address (VA) and converts it to a 22-bit physical address (PA). This function uses a relocated address cycle. A nonrelocated address cycle uses the 16-bit VA as the PA.

A nonrelocated address cycle (Figure 4-7) is similar to a short cycle. The address generated by the data chip is used as the PA. The PA along with control signals C0 and C1 are strobed into the PAX latch at the end of PT3. C0 and C1 are used to determine the type of data transfer to be performed (Table 4-2). The remainder of the cycle is the same as a short cycle.

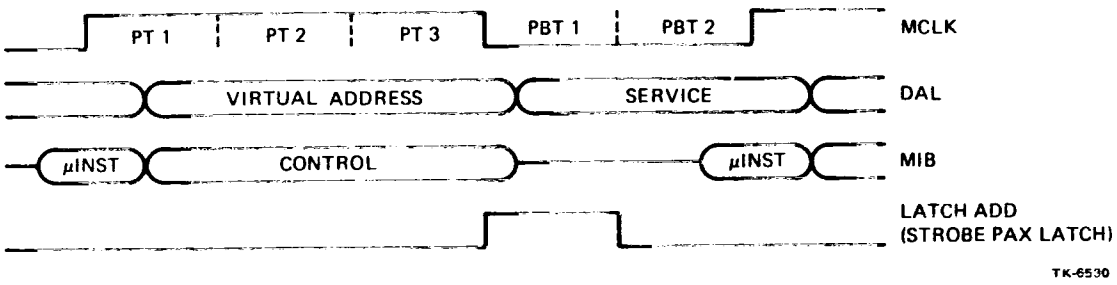


Figure 4-7 Nonrelocated Address Cycle

Table 4-2 Data Cycle Types

Type of Cycle	C0	C1
DATI (Read)	Unasserted	Unasserted
DATIP (Read-modify-write)	Asserted	Unasserted
DATO (Write)	*	Asserted
DATOB (Write byte)	*	Asserted

*Not established until the data microcycle.

A relocated address cycle (Figure 4-8) requires additional time to allow for the generation of the relocated PA. The VA is strobed into the PAX latch and latched into the MMU at the end of PT3. The MMU begins the address relocation at this time. Since the MMU requires the use of the DAL during address relocation, the service information that is usually on the DAL during PBT is inhibited. Approximately 180 ns into PBT, the relocated address is present on the DAL. The relocated PA is then strobed into the PAX latch at the end of PBT3. If the address relocation is successful (i.e., legal), the address cycle is complete. If there is an MMU error or illegal access error, the MMU will assert K13 ABORT L (Figure 4-9) and force control back to the DAT/CTL hybrid. The base machine will then read the service register and, recognizing an MMU error, will cause the execution of a trap through the vector at 250_8. This trap notifies the user's program of an MMU abort.

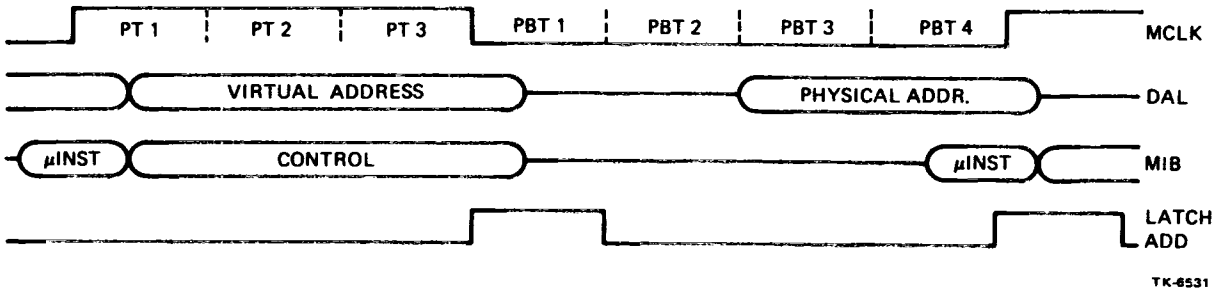


Figure 4-8 Relocated Address Cycle

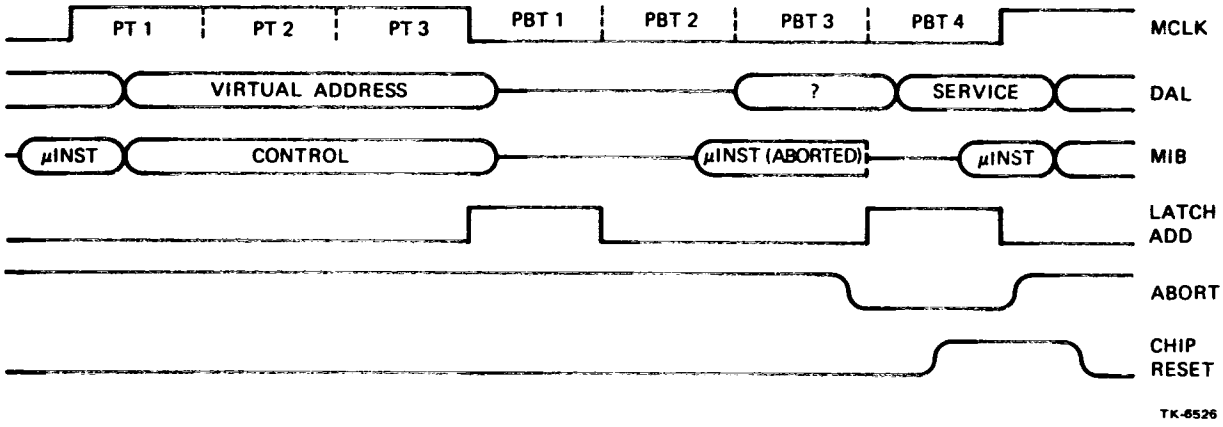


Figure 4-9 Relocated Address Cycle – Aborted

Data Cycles – Data cycles are used to transfer data from the processor to memory or a peripheral device. Data cycles normally follow address cycles. A data cycle is generated using a short cycle that is altered to meet the transfer requirements of the memory bus (EUB) or UNIBUS. The processor is synchronous, and the EUB and UNIBUS are asynchronous; therefore, it is necessary to stop the processor clock and wait for the EUB or UNIBUS to complete the data transfer. Figure 4-10 shows the timing for the various data cycles.

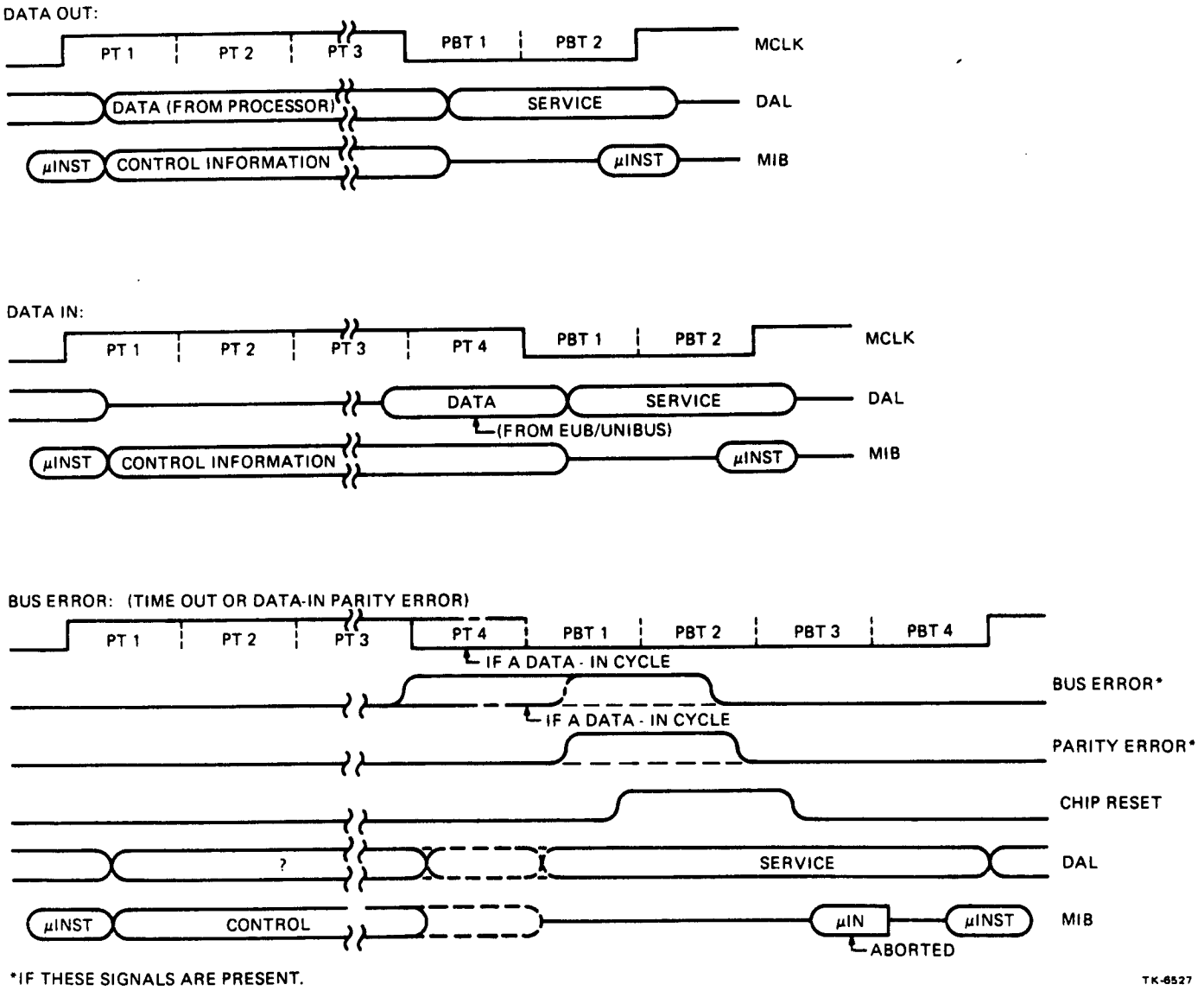


Figure 4-10 Data Cycles

A data cycle begins with the latching of the microinstruction on the transition from the previous cycle's PBT to the current cycle's PT. Data cycle execution begins in PT1, which is immediately followed by PT2 as in a short cycle. The processor then arbitrates for use of the bus with PT2 being extended in 75 ns increments until the bus is free. PT3 is then entered and the processor clock is stopped until the data transfer is completed and BSSYN H is asserted. The processor clock is stopped during PT3 because BSSYN H may not be asserted during the 75 ns of PT3. If a data-out operation was being performed, the data transfer is complete at the assertion of BSSYN H and PBT1 is entered. If a data-in operation was being performed, it is necessary to allow an additional 75 ns deskew time before latching the data into the processor. Therefore PT4 is entered to allow for the deskew time. The data is latched into the processor on the transition of PT4 to PBT1. Other aspects of UNIBUS timing are discussed in Paragraph 4.5.2.

If a bus time-out error or bus parity error occurred during the data transfer, CHIP RESET will be asserted and PBT3 and PBT4 will be entered to force control back to the base machine control chip. The base machine control chip will execute the appropriate trap (114_8 or 4_8) to inform the user's program of the error.

4.2.2 Timing Logic

Phase time and phase bar time are generated by the logic illustrated in Figure 4-11. Most processor timing is controlled by a programmed logic array (PLA) which monitors the control information on the MIB. The PLA does not directly generate the time states but controls the timing circuit shown in Figure 4-12 via the signal CHG CLK L. CHG CLK L asserted during phase time (MCLK H) initiates phase bar time, or, when asserted during phase bar time (MCLK L), initiates phase time. CHG CLK L will be asserted at various times according to the operation to be performed. The PLA timing equations in the print set specifies when CHG CLK is generated. The various inputs to the timing PLA and the corresponding outputs are illustrated in Figure 4-11. (See Appendix D, Section D.3, Figures D-2 and D-3 for corresponding M7133-YA CPU logic diagrams).

When a short cycle microinstruction is executed (no address/data transfer to take place), PBT1 must be asserted after PT2. To assert PBT1, CHG CLK L must be asserted. In this case, the timing PLA "looks" only at the following inputs: MCLK H, CT1 and MIB ⟨12,8,9⟩. According to the timing PLA equation, CHG CLK L will be generated, in this case, if MCLK H is asserted (indicating phase time); CT1 H (from the time state counter) is asserted; and MIB ⟨12,8,9⟩ are set indicating a non-I/O operation. If these conditions are met, CHG CLK L will be asserted clearing MCLK H and initiating phase bar time.

When any of the other timing cycles are being executed, the PLA will assert CHG CLK L at the required times. The timing PLA also controls the I/O control signals (C), C1, LAT ADD, etc).

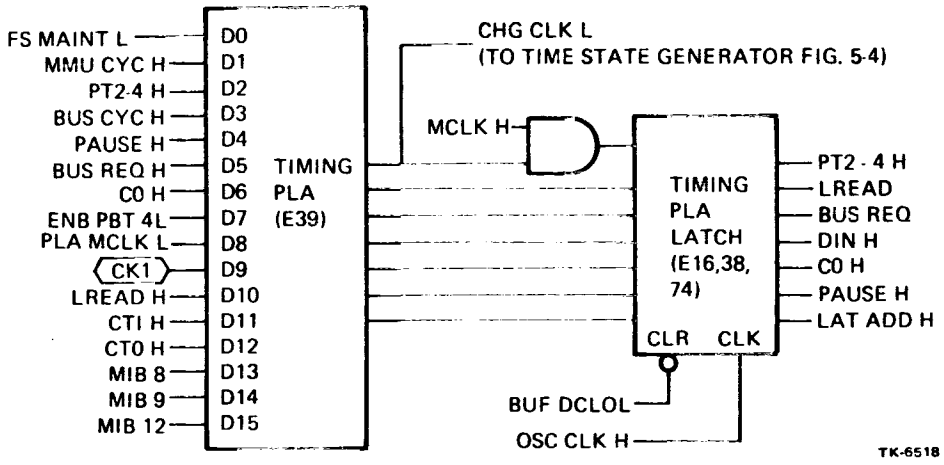


Figure 4-11 PLA Control Signals

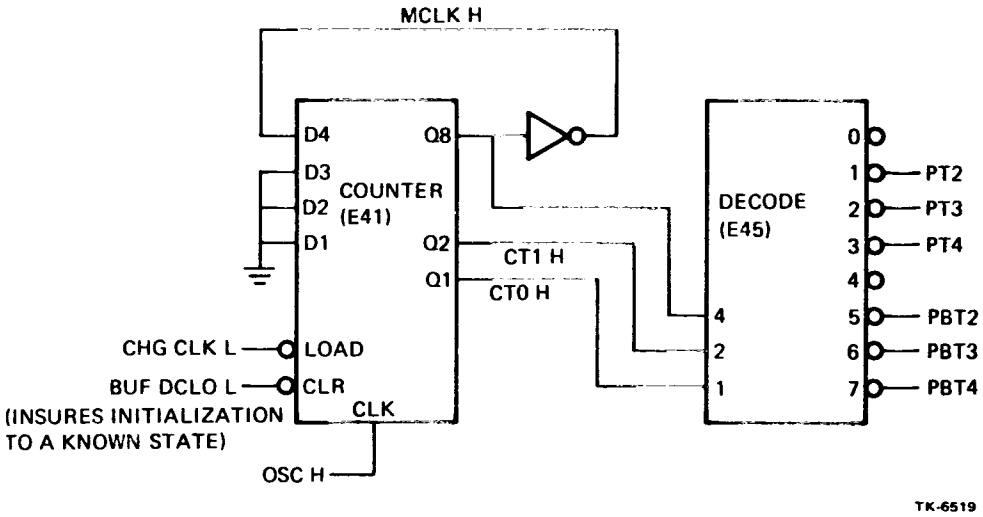


Figure 4-12 Phase Time and Phase Bar Time Generation

4.3 PDP-11/24 CENTRAL PROCESSING UNIT

The PDP-11/24 central processing unit is composed of a data chip and a control chip mounted on one 40-pin hybrid package. A detailed description of each chip follows.

4.3.1 Control Path

The control path allows for the transfer of microinstructions from the control chip to the other chips in the system via the MIB⟨15:00⟩. Figure 4-13 is a simplified block diagram of the control path. The control store is divided into four quadrants. Quadrant 1 is a PLA and quadrants 2, 3, and 4 are conventional ROMs.

The PLA input register (PIR) stores external data or service information from the DAL which is used to access locations within the control store. The control store contains microinstructions which reside within the address range 000000_8 through 000777_8. Each location contains a 25-bit word composed of a 16-bit microinstruction and the 9-bit next address field (NAF) needed for the next microinstruction fetch. The format of the microword is illustrated in Figure 4-14.

PDP-11 macroinstructions are loaded directly from the DAL into the PIR. These macroinstructions are used by the control store to generate the corresponding microinstructions for various operations. Initial power-up will assert CHIP RESET. The assertion of CHIP RESET will return control of the processor to the base machine control chip. The processor then enters the service state, microaddress 0, and service information is read into the PIR. The service information consists of service request data. Service information is placed on the DAL <15:00> during phase bar time (Figure 4-15) and comes from various sources within the PDP-11/24 processor. The upper three bits (DAL <15:13>), however, are ignored by the processor base machine chip and are replaced by service information internal to the control chip. Table 4-3 describes the DAL service information, and Table 4-4 gives the priority level. As seen in Figure 4-16, ENB SVC L causes service information to be placed on the DAL. ENV SVC L is asserted every phase bar time during PBT2 if there is no MMU cycle (MMU CYC L cleared) in progress. During MMU cycles, service information is not asserted and is replaced by the relocated physical address from the MMU chip. In addition to initial power-up, the processor will be reset and forced to the service state as a result of any of the following: a control error (reserved instruction trap), an MMU error, a parity error, a bus error, or DLY DCLO H.

If the processor has entered the service state as a result of initial powerup, the microcode must properly initialize the total processor. The power-up action to be taken is determined by start-up information read from logic external to the processor base machine.

System start-up information from the fast data-in register is placed on the DAL ⟨15:00⟩ during phase time. ENB FDIN L allows the start-up information to be placed on the DAL. This signal is generated after PT1 (during PT1 through PT4) if the general-purpose output (GPO) code on the MIB is 10, indicating system powerup. It is also generated during a halt instruction to check if kernel-halt is allowed. Figure 4-17 and Table 4-5 describe the information in the fast data-in register.

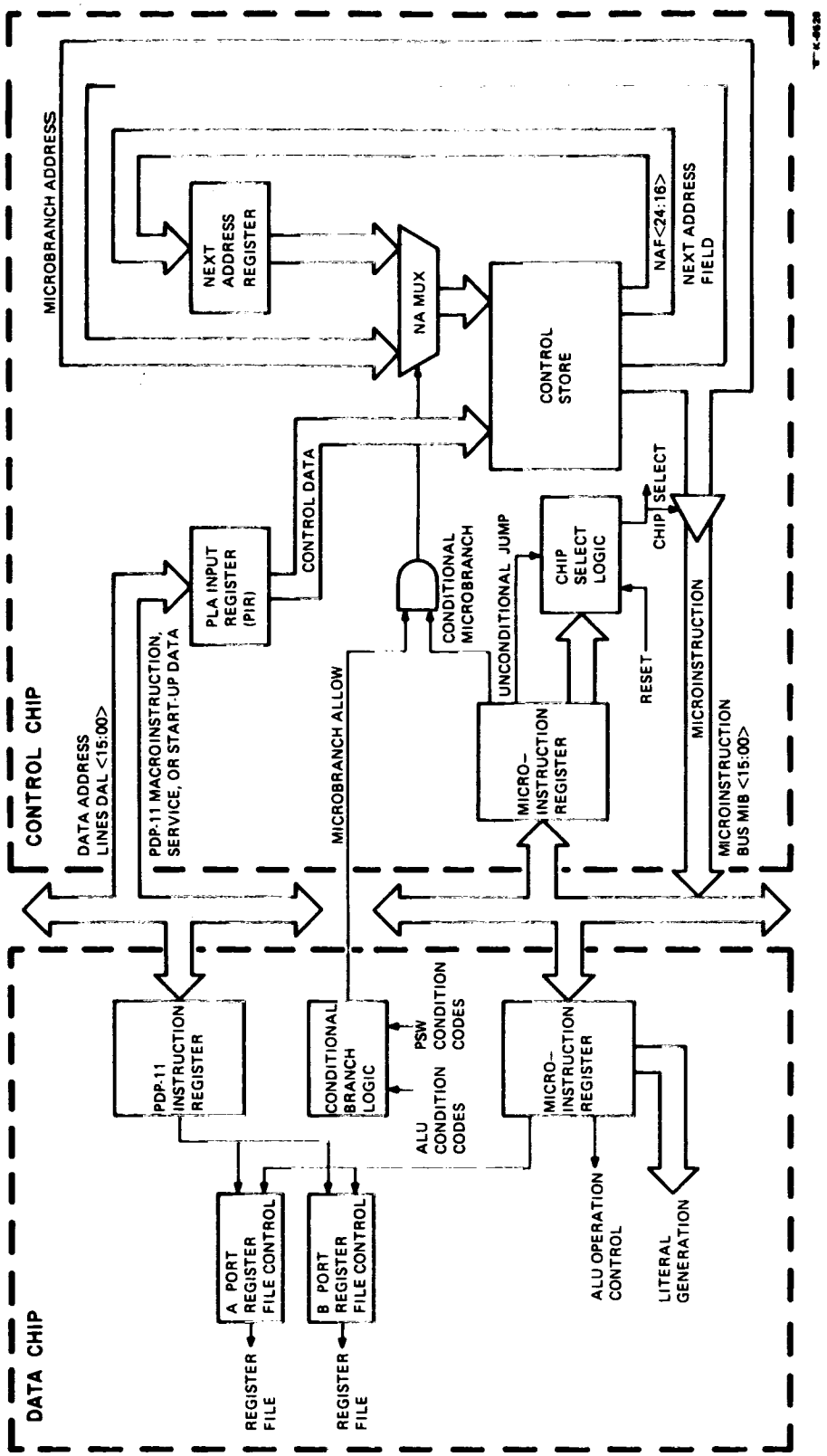


Figure 4-13 Control Path

Figure 4-14 Microinstruction Format

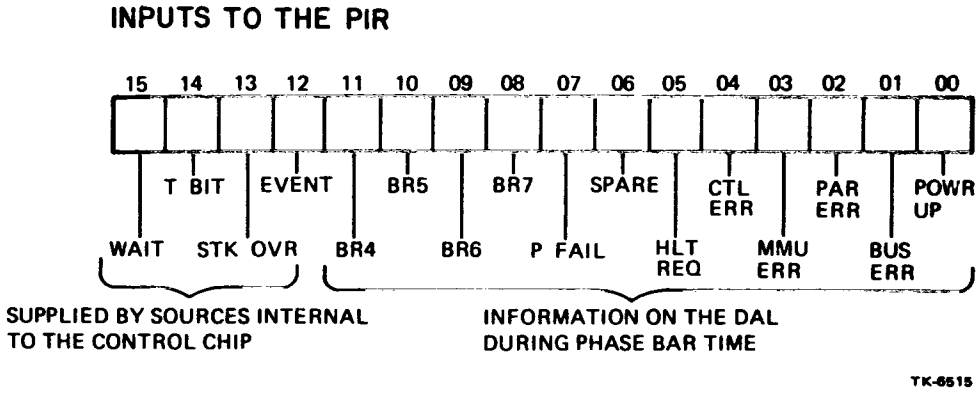


Figure 4-15 Service Information

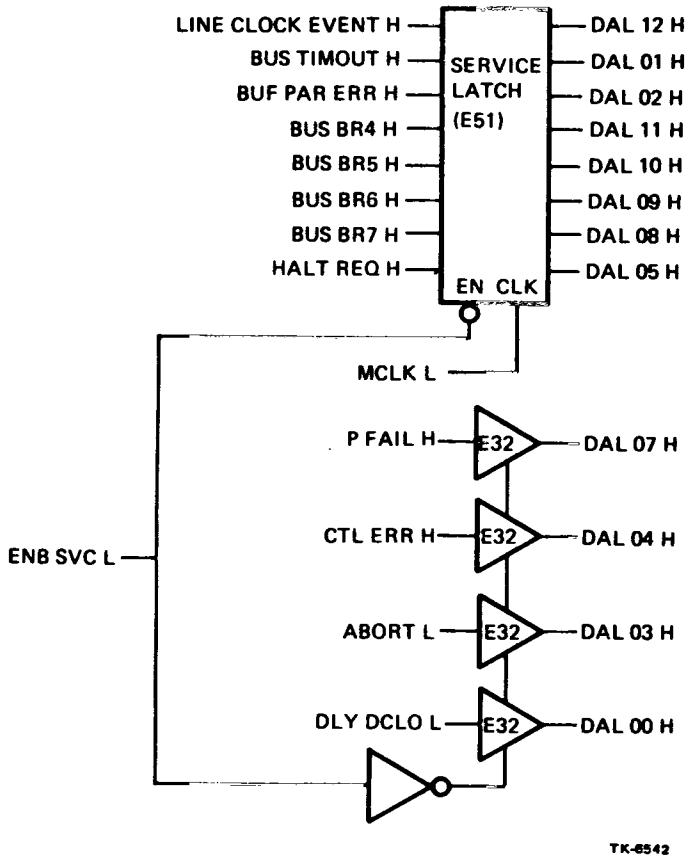


Figure 4-16 Service Information Logic

Table 4-3 DAL Service Information

DAL Bit	Asserted	Description
⟨15⟩ WAIT	Internal	Wait flip-flop (internal)
⟨14⟩ T-BIT	Internal	Trap bit (internal)
⟨13⟩ STK OVR	Internal	Stack overflow (internal)
⟨12⟩ EVENT	H	Event line (interrupt request level 6)
⟨11⟩ BR4	H	Interrupt request (level 4)
⟨10⟩ BR5	H	Interrupt request (level 5)
⟨09⟩ BR6	H	Interrupt request (level 6)
⟨08⟩ BR7	H	Interrupt request (level 7)
⟨07⟩ PWRF	H	Power-fail indication
⟨06⟩ Spare	H	Reserved
⟨05⟩ HALT REQ	H	Halt request
⟨04⟩ CTL ERR	L	Control error (asynchronous)
⟨03⟩ MMU ERR	L	Memory management error (abort, asynchronous)
⟨02⟩ PAR ERR	H	Parity error (asynchronous)
⟨01⟩ BUS ERR	H	System bus time-out error (asynchronous)
⟨00⟩ DCLO	H	Power-up indication (asynchronous)

Note: The unasserted state of all signals results in an instruction fetch.

Table 4-4 DAL Service Priority

Priority	Service
1	DCLO (Power-Up)
2	CTL ERR
3	MMU ERR
4	BUS ERR
5	PAR ERR
6	Spare
7	T-BIT
8	STK OVR
9	PWRF
10	BR7
11	EVENT
12	BR6
13	BR5
14	BR4
15	HALT REQ
16	WAIT
17	Inst Fetch

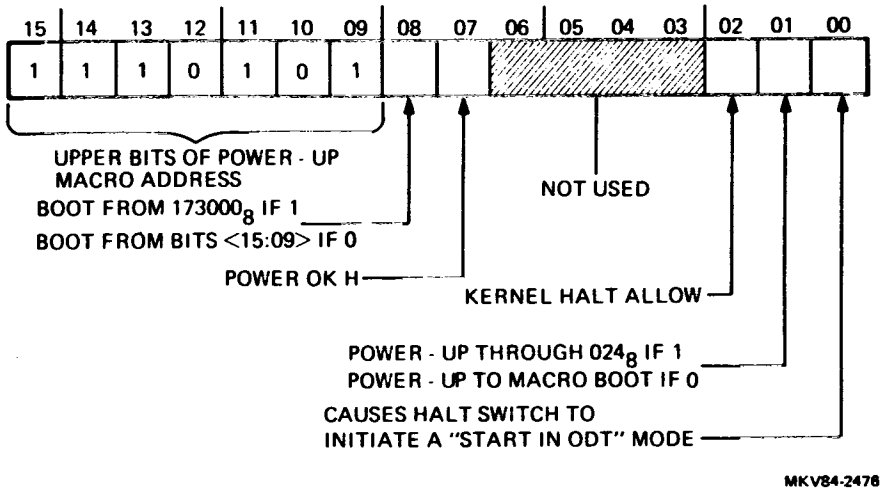


Figure 4-17 Fast Data-In Register

Table 4-5 Fast Data-In Register

Bit	Description
DAL⟨15:09⟩	Upper bits of power-up macroaddress.
DAL⟨08⟩	Forces boot to address 173000_8 if asserted (1). When cleared (0) the boot address is read from bits ⟨15:09⟩.
DAL⟨07⟩	When asserted (1) system power level is OK; when cleared (0) system power is bad.
DAL⟨06:03⟩	Not used.
DAL⟨02⟩	Halt option. When jumper W3 is in place, kernel halt mode is allowed. When jumper W3 is removed, trap to 10, unless a power-fail is in progress. This allows the processor to stop at the end of the power-fail routine but not at any other time.
DAL⟨01⟩	Power-up option. When jumper W2 is out, power up through vector 24. When jumper W2 is in, power up to user boot in macrocode, using the address selected by bits ⟨15:09, 08⟩.
DAL⟨00⟩	Causes HALT switch to initiate a "Start in ODT" mode.

4.3.2 Microinstruction Bus (MIB⟨15:00⟩)

The microinstruction bus (MIB) contains the next microinstruction during phase bar time and various control information during phase time. Figure 4-18 illustrates the bit assignments of the MIB during phase time and phase bar time. Part of this control information consists of the address in/out (AIO) bits, X codes, and general-purpose output (GPO) code bits which are encoded to specify particular functions. The remaining control bits each relate to an operation or status indication. A description of MIB control information is provided in Table 4-6.

The AIO codes are used by the timing PLA shown in Figure 4-11 to generate the proper control signals. The timing PLA equations in the print set show the necessary AIO codes for particular outputs from the PLA. The GPO codes are used by different portions of logic external to the chip set. The X control bits are used internally by the chip set.

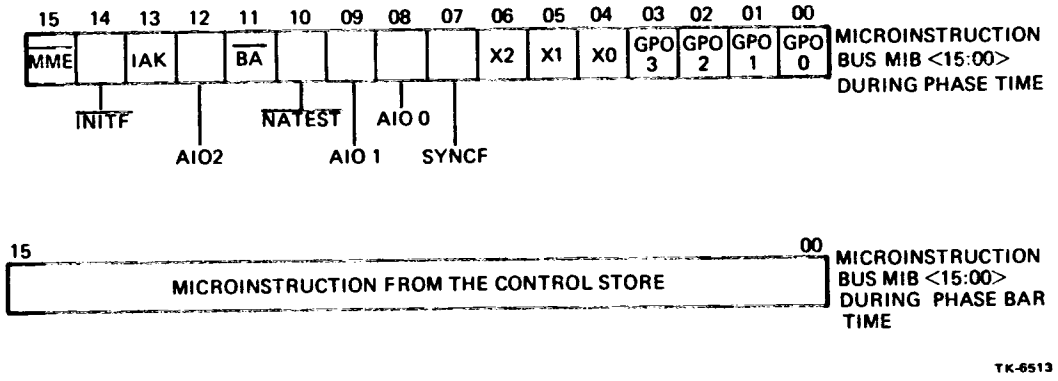


Figure 4-18 Microinstruction Bus

Table 4-6 Microinstruction Bus Control Information

MIB Bits	Description
MIB⟨15⟩ MME	Memory Management Enable. This active-low signal indicates to the data chip that the virtual address from the data chip will be relocated during the second-half of the microinstruction cycle (PBT). The two sources of this signal are the MMU chip or logic external to the chip set which detects an ODT cycle and simulates the MMU.
MIB⟨14⟩ INIT F	Initialization. This active-low signal triggers a 100 ms one-shot (print K2) which generates the UNIBUS initialization signal BUS INIT L.

Table 4-6 Microinstruction Bus Control Information (Cont)

MIB Bits	Description
MIB⟨13⟩ IAK	Interrupt Acknowledge. This signal is asserted high during an input vector microinstruction. It is used to indicate to external logic that a vector-in bus operation is occurring.
MIB⟨12,9,8⟩ AIO⟨02:00⟩	Address In/Out. These three encoded signal lines tell external logic whether or not a bus cycle is being performed and the type of cycle. The signals are encoded as follows:

AIO2	AIO1	AIO0	Operation	
0	0	0	AWO	Address cycle (write)
0	0	1	ARW	Address cycle (read/modify/write)
0	1	0	Unused	
0	1	1	ARO	Address cycle (read)
1	0	0	DOUTB	Data out byte
1	0	1	DOUT	Data out word
1	1	0	DIN	Data in word
1	1	1	NOP	No operation

MIB⟨11⟩ BA	Branch Allow L. This asserted low signal is generated by the data chip to indicate to the control chips that the requested microbranch condition is true. This allows the conditional microbranch to occur.
MIB⟨10⟩ NATest	Next Address Field Test. This asserted low signal is used during control chip testing. The control chip will output its next address field rather than output the microinstruction.
MIB⟨07⟩ Sync F	This output is asserted high at the beginning of an external bus cycle (address microinstruction) and is reset at the conclusion of the external bus cycle. The sync bit indicates an on-going bus cycle.

Table 4-6 Microinstruction Bus Control Information (Cont)

MIB Bits — **Description**

MIB⟨06:04⟩
X⟨02:00⟩

The X code bits provide control information from the data chip to control chips. These signal lines are encoded as follows:

X2	X1	X0	Operation
0	0	0	Load control chip with PIR subroutine return information.
0	0	1	Used to update the PIR in the control chips or the IR in the MMU chip.
0	1	0	Sets the stack overflow flip-flop in the control chip.
0	1	1	Indicates to the MMU chip user mode address relocation.
1	0	0	Load PSW priority, trap bits into control chips.
1	0	1	Load PSW priority bits into the control chips.
1	1	0	Load PSW trap bit into the control chips.
1	1	1	No operation.

MIB⟨03:00⟩
GPO⟨03:00⟩

General-Purpose Output. These general-purpose output lines provide various external control information much like the X codes provide internal control information and are encoded as follows:

GPO3	GPO2	GPO1	GPO0	Operation
0	0	0	0	No GPO operation requested.
0	0	0	1	Trigger the RUN light. This only occurs on the instruction fetch and when an input character is echoed in micro-ODT (K1).
0	0	1	0	Not used.
0	0	1	1	Toggle the halt flip-flop (K6).
0	1	0	0	Not used.
0	1	0	1	Clear the event flip-flop in the line clock (K6).
0	1	1	0	Clear the power-fail flip-flop (K2).
0	1	1	1	Latch address bits ⟨17:16⟩ for micro-ODT.
1	X	X	X	Read the fast data-in register (K1).

4.3.3 PDP-11/24 Data Path

A block diagram of the data path is provided in Figure 4-19. Data flow through the data path is controlled by the microcode. Each microinstruction from the control store generates a unique set of outputs on the MIB that control the data path elements and determine the ALU functions to be performed. Sequences of these microinstructions are combined into microroutines that perform various PDP-11 instruction operations. The arithmetic logic unit (ALU) performs the arithmetic, logical, and shifting operations in the data chip. The ALU has four status flags associated with its operation. These status flags are updated at the conclusion of an ALU operation, and are conditionally written under microprogram control into the PSW condition codes (in the data chip) at the end of a cycle. These ALU status flags are summarized as follows.

PDP-11/24	ALU Condition Codes
AN	ALU result is negative
AV	ALU operation resulted in an arithmetic overflow
AZ	ALU result is 0
AC	A non-subtract ALU operation resulted in a carry (Subtract type operations caused a borrow)

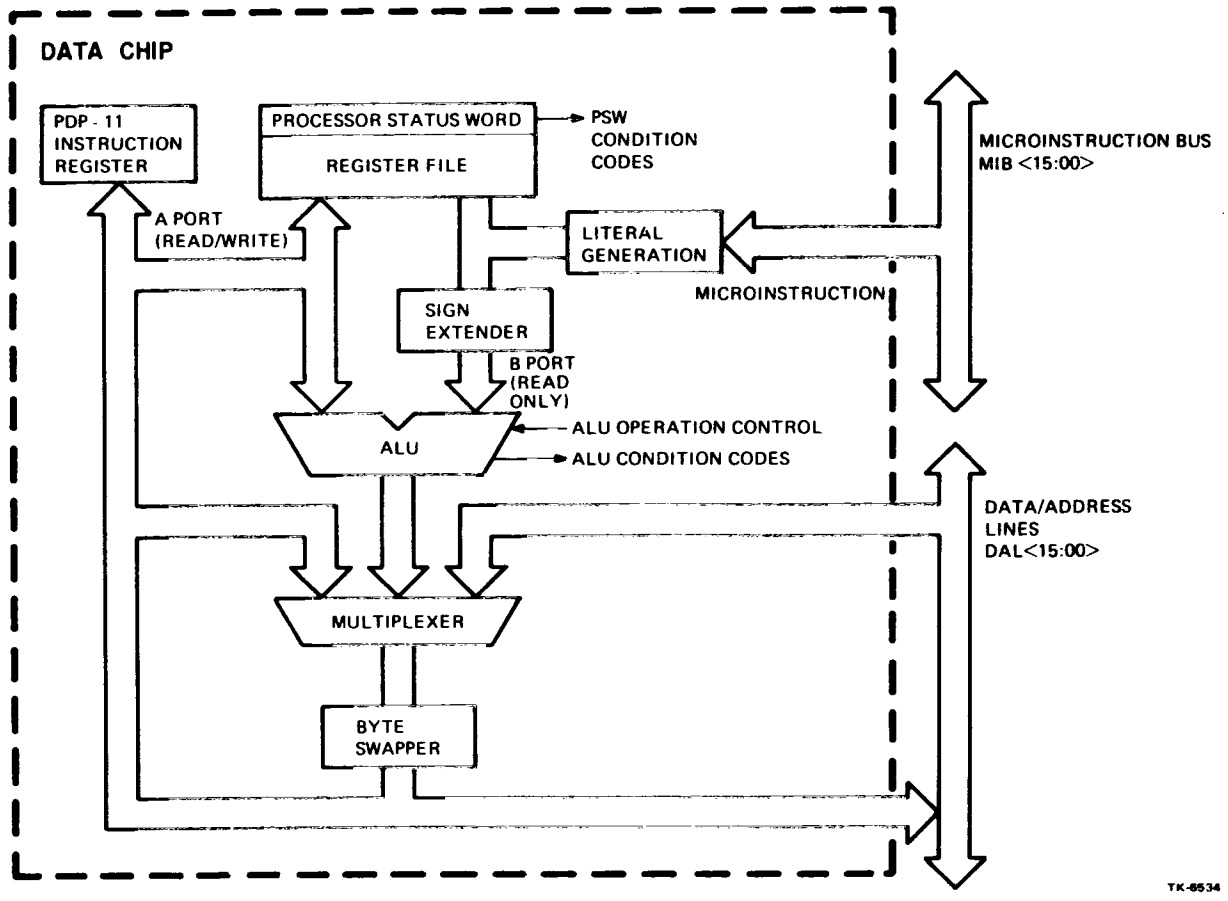


Figure 4-19 Data Path

The data path multiplexer selects one of three inputs for transfer to the byte swapper. These inputs are: the DAL⟨15:00⟩, the internal A bus (A⟨15:00⟩), and the output of the ALU. Selection of these inputs depends on whether the PSW address mode is active or not, and the input microinstruction. The byte swapper either passes the output of the multiplexer through unaltered or interchanges the bytes of the word. Byte swapping occurs with the execution of SWAB instruction or when inputting or modifying a byte during an external bus cycle that references an odd address.

4.3.4 Data Address Lines (DAL⟨15:00⟩)

The 16-bit DAL output transfers addresses and data from the data chip to the UNIBUS and the MMU chip as well as the BDAL and the control chip. Information is output onto the DAL for all output microinstruction cycles during phase time. Service information is placed on the DAL during phase bar time. The 16-bit DAL input receives information from the UNIBUS, the MMU chip, or other sources during phase time. All data is connected to the register file via the data path multiplexer and byte swapper. PDP-11 instructions are loaded directly into the instruction register. The DAL also contains information for use in the PSW explicit address logic. Explicit references directly access the PSW. This information is saved and used during phase bar time when the register file is written and updated. The DAL input to the data chip is active during phase bar time of relocation operations to load the new address from the MMU chip into the PSW explicit address logic.

4.4 PDP-11/24 CHIP SET CONTROL LOGIC

The chip set control logic shown in Figure 4-20 generates two signals: CHIP RESET H, which goes to each control chip in the chip set; and CTL ERR H, which is placed on DAL ⟨04⟩ at PBT2. CHIP RESET H is used to reset the chip set when any of the following conditions occur. (See Appendix D, Section D.3, Figure D-4 for corresponding M7133-YA CPU logic diagram.)

- ABORT L is asserted, indicating a relocation operation error in the MMU.
- BUS ERR H is asserted, indicating a UNIBUS error.
- BUF PAR ERR H is asserted, indicating a parity error on the UNIBUS.
- DLY DCLO H is asserted, indicating that an unacceptable power level is available to the system.
- CSEL 1 L or CSEL 2 L is not asserted, indicating that no control chip acknowledged receipt of a jump microinstruction.

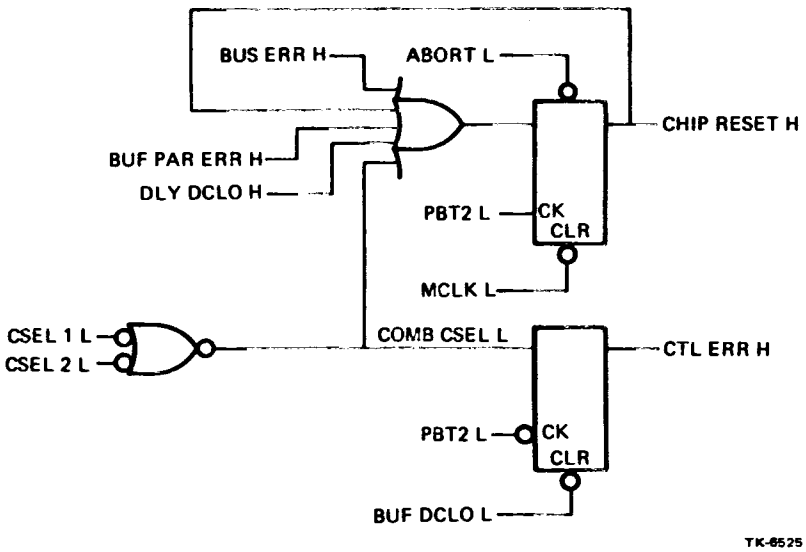


Figure 4-20 PDP-11/24 Chip Set Control Logic

CHIP RESET H will be asserted at the end of PBT2 and will be cleared at the beginning of PT1. CHIP RESET H causes a service cycle to occur. If CHIP RESET H is asserted because no control chip asserted its chip select line, CTL ERR H will be asserted. CTL ERR H indicates that either the selected control chip is not present in the system or it did not assert its chip select line. During the service cycle the CTL ERR bit (bits ⟨04⟩) of the service register will indicate the error. The service cycle will determine which condition caused the chip reset. BUF DCLO L clears CTL ERR H to ensure proper system initialization.

4.5 SYSTEM BUS LOGIC

The two external busses used by the PDP-11/24 are the UNIBUS and the 22-bit extended UNIBUS (EUB). the UNIBUS carries all external data transfers and UNIBUS addressing performed by the PDP-11/24. The EUB is used to address memory only; memory data is transferred on the UNIBUS data lines. (See Appendix D, Section D.3, for M7133-YA CPU variation in signal names on circuit schematic.)

4.5.1 Address and Data

4.5.1.1 Address – Addresses generated by the processor for use in I/O cycles are latched from the DAL into the PAX latch (K7). The PAX latch then drives the internal PAX bus (see Figure 4-21). Logic in the processor then determines the destination of the address that the processor has generated and routes it to the proper bus. There are three types of addresses:

1. Internal address
2. Main memory address
3. UNIBUS address

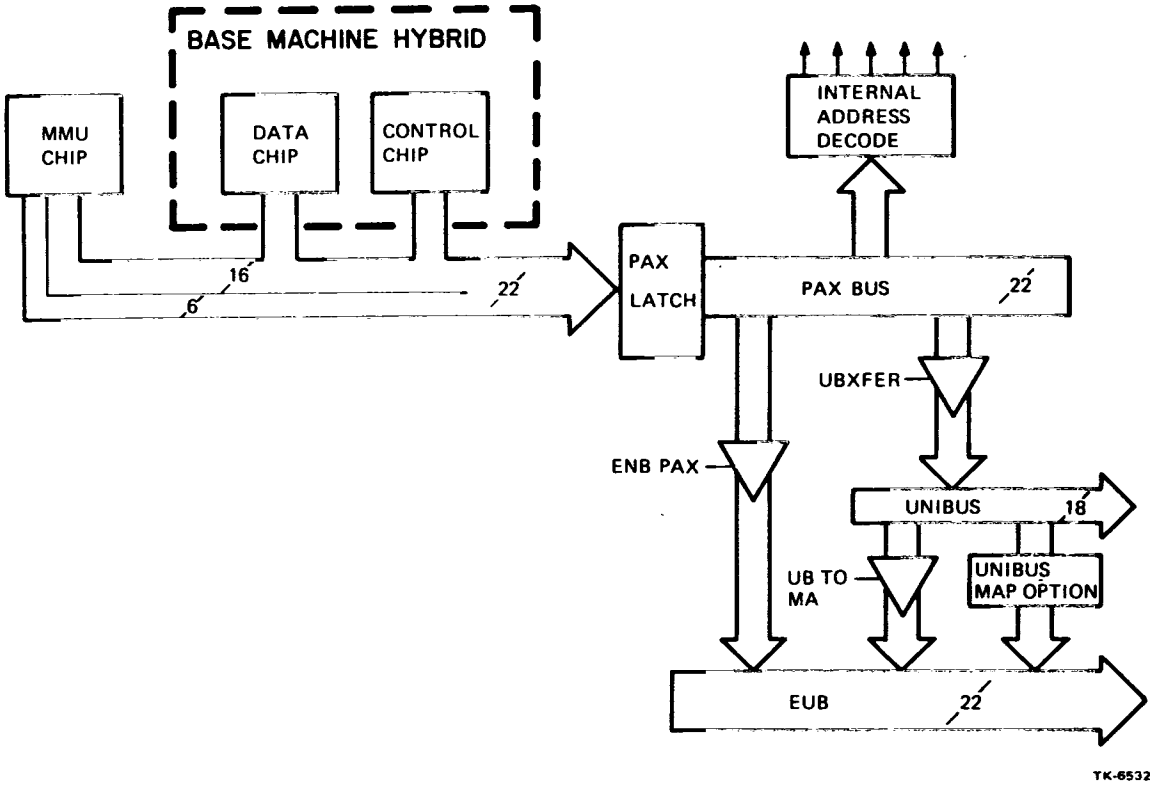


Figure 4-21 Address Paths

Internal addresses, i.e., serial line unit registers, are decoded by a PLA and its associated logic (K9). There is no external bus cycle performed for an internal address.

Main memory addressing requires a 22-bit address be passed to main memory using the EUB (K11). I/O page addresses are also passed to main memory so the memory parity control and status registers can be accessed. The signal K11 ENB PAX L enables the 22-bit address and control information onto the EUB.

UNIBUS addressing requires the 22-bit physical address to be truncated to 18-bits by dropping bits (21:18) of the physical address. The 18-bit address is passed to the UNIBUS (K11) when K7 UB-XFER L is asserted. It should be noted that control bits C0 and C1 are also transmitted with the address.

4.5.1.2 Data Paths – Data is transferred to and from the processor via a set of buses different from that used by address information. Figure 4-22 shows the path used by the data. When performing a data out, the data is passed from the DAT chip to the DAL and is then gated onto the BDAL (K4 and K5) by K4 DIR IN L not being asserted. If the data is to be used external to the processor (i.e., memory or UNIBUS), the data is gated onto the UNIBUS data lines by the assertion of K1 ENB DAT L.

NOTE

Main memory uses the UNIBUS data lines for data transfers. The EUB is used for address and control information only.

When performing a data in, the data transferred from main memory or a UNIBUS device is put on the UNIBUS data lines. The data is then gated from the UNIBUS to the BDAL by the assertion of K4 UBUS TO B DAL L. The data is then gated from the BDAL to the DAL and is passed to the processor by the assertion of K4 DIR IN L.

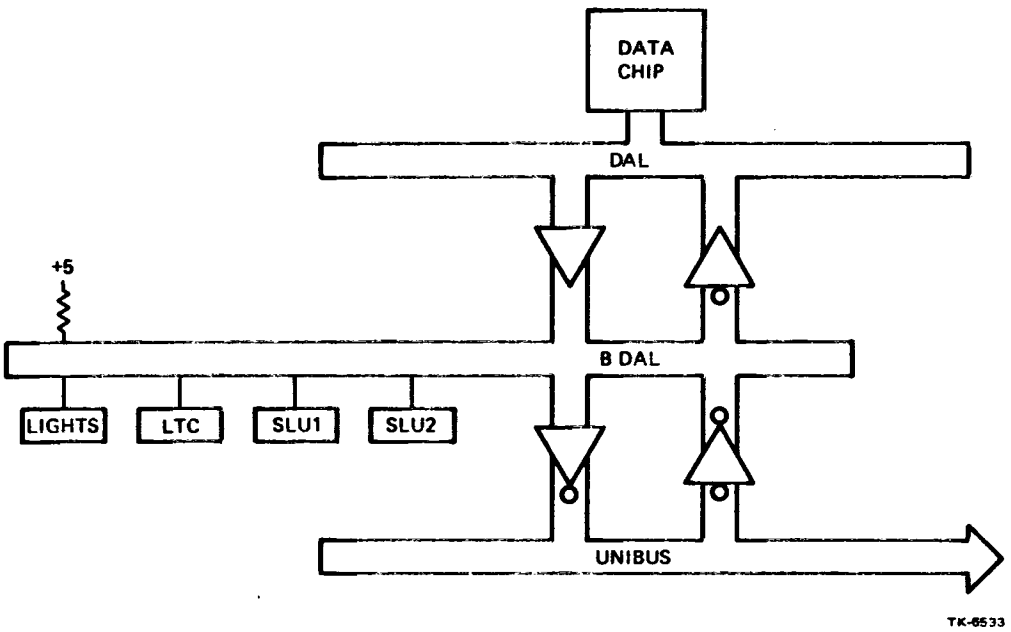


Figure 4-22 I/O Data Paths

4.5.1.3 Direct Memory Access – A direct memory access (DMA) occurs when a UNIBUS device directly accesses main memory without CPU intervention. This action in the PDP-11/24 requires that the UNIBUS address be passed to the EUB. The PDP-11/24 accomplishes this in one of two ways:

1. The UNIBUS address can be passed to the EUB using logic contained on the PDP-11/24 module (K10). This logic examines UNIBUS address bits ⟨17:13⟩ and ANDs them together. The output of the AND gate, K11 I/O PAGE H, is then passed to EUB ⟨21:18⟩. This action forces UNIBUS I/O page address to go to the EUB I/O page. UNIBUS address bits ⟨17:00⟩ and control bits C0 and C1 are passed to the EUB unchanged.

2. The optional UNIBUS map module (M7134), if installed, is responsible for passing the UNIBUS address to the EUB. The UNIBUS map may pass the address unchanged or relocate it as requested by the user's program. For a more detailed explanation as to how the UNIBUS map option functions, refer to Chapter 6 of this manual.

Address and control information can be passed from the UNIBUS to the EUB whenever the CPU is not using either of the buses. Because data going to and from main memory is transferred on the UNIBUS, there is no need to alter it in any way.

When performing a DMA, the internal registers contained on the CPU (LTC, SLU1, SLU2, and the display register) cannot be accessed. This is because there is no way for either the UNIBUS or EUB to access the PAX bus, thereby making the internal address decoder inaccessible to DMA activity.

4.5.2 System Bus Timing

When the CPU is performing a data transfer using the UNIBUS it becomes bus master and therefore must control the UNIBUS timing.

UNIBUS timing by the PDP-11/24 processor is generated by the logic on K1 and K2. When the CPU requires use of the UNIBUS or EUB bus (memory data is transfered on the UNIBUS data lines), the CPU asserts K1 PRE BUS REQ H. Logic on K2 (Figure 4-23), which monitors UNIBUS activity, will assert K2 TAKE BUS H when the bus is free. K2 TAKE BUS H and K1 PRE BUS REQ H will cause the bus master flip-flop (E65) to be set on the next high-to-low transition of the K1 PRE OSC L. The bus master flip-flop (Figure 4-24) then asserts K1 PROC MAST H. The bus master flip-flop being set indicates that the processor is now the bus master.

UNIBUS timing requires that BBSY (bus busy) be asserted. This action tells all other devices on the UNIBUS that the bus is currently in use. BBSY is asserted by the processor when K1 PROC MAST H is asserted, which asserts K1 ENB A+C H, which asserts K2 BUS BBSY L. The assertion of K1 ENB A+C H also gates address and control information onto the UNIBUS. If this is a read cycle, the processor waits for the requested data. If this is a write cycle, K1 ENB DATA L is asserted and data is gated onto the UNIBUS.

The processor must now allow 150 ns for front end deskew; 75 ns for differential skews in the UNIBUS and 75 ns for the UNIBUS devices to decode the address and control information. The 150 ns of front end deskew is created by delay line E100. This is accomplished by the delay line not allowing K2 BUS MSYN L to be asserted for 150 ns after K1 PROC MAST H is asserted. Note that if a main memory transfer is taking place the EUB receives K1 MEM MSYN H only 120 ns after K1 PROC MAST H is asserted. This is possible because of the shorter length of the memory bus as compared to the length of the UNIBUS.

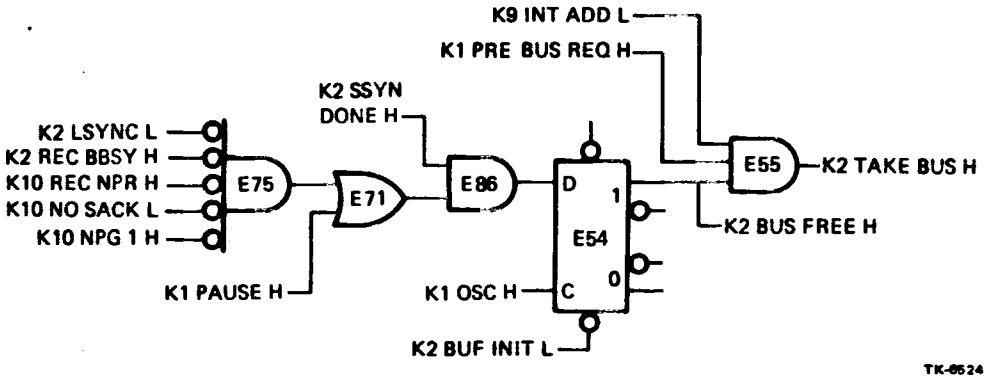


Figure 4-23 Bus Monitor Logic

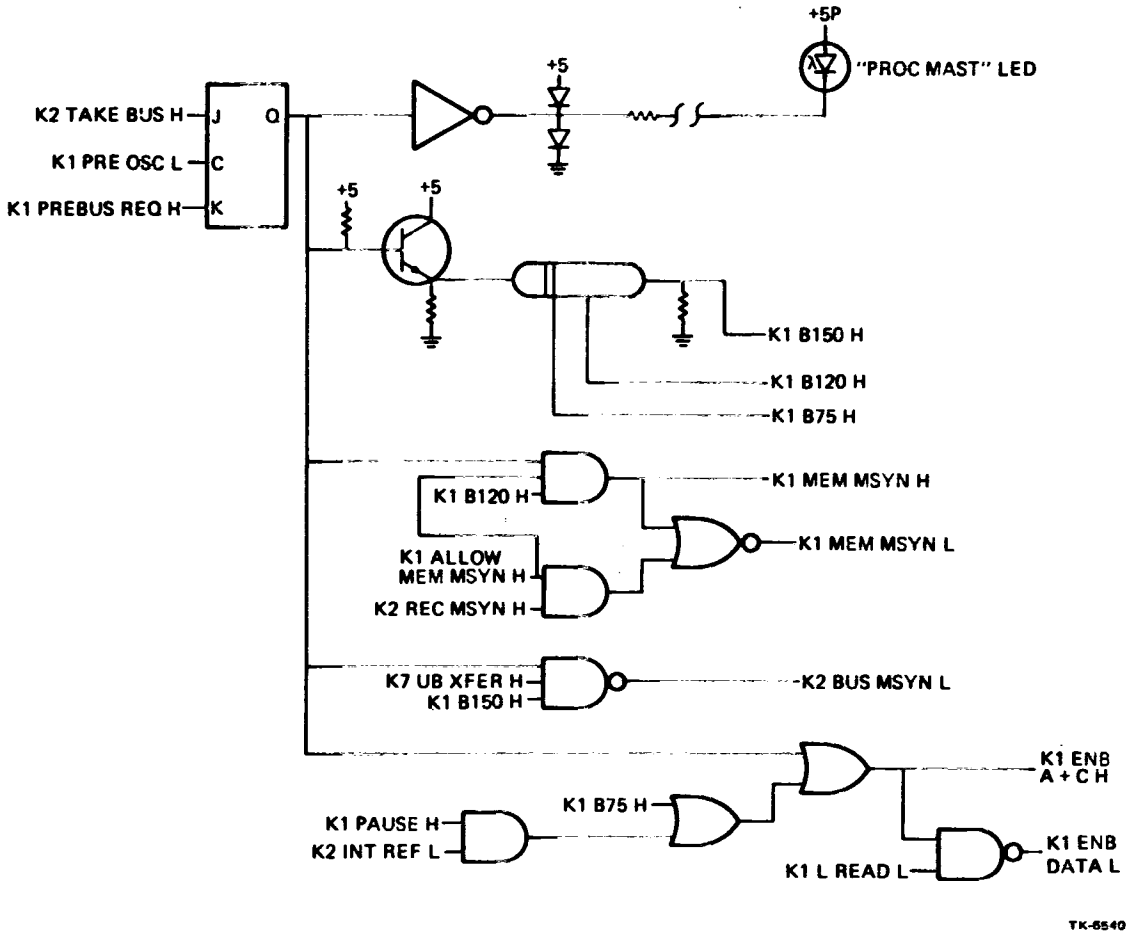


Figure 4-24 UNIBUS Timing Logic

During front end deskew the processor enters time state PT3, which stops the processor clock, and the processor waits for K2 BUS SSYN L to be asserted by the accessed device. The assertion of both K2 BUS SSYN L and K1 PROC MAST H will assert K2 BSSYN H, which will restart the processor clock, which will negate K1 PRE BUS REQ H. On the next high-to-low transition of K1 OSC L, the bus master flip-flop (E65) will be cleared, thus negating K1 PROC MAST H and K2 BUS MSYN L. UNIBUS timing requires that valid address and control information be maintained for 75 ns after the removal of MSYN. This is accomplished by delay line E100 delaying the negation of K1 ENB A + C H for 75 ns. This 75 ns time period is called tail end deskew.

When the processor is bus master, the setting of the bus master flip-flop (E65) causes a LED, labeled PROC, on the front panel to be illuminated. This allows the user to monitor the amount of the processor's bus activity.

The following steps summarize system bus timing:

1. Assert BBSY, address and control information, and data (if a write cycle).
2. Wait 150 ns – front end deskew.
3. Assert MSYN.
4. Wait for SSYN.
5. Wait 75 ns (PT4) if a read cycle – data deskew.
6. Remove MSYN.
7. Wait 75 ns – tail end deskew.
8. Remove BBSY, address and control information, and data (if a write cycle).

4.5.3 System Bus Errors

Two error conditions can occur during a UNIBUS cycle. One results from an attempt to address a nonexistent or defective memory location or I/O device (time-out). The second error is a result of reading from a memory location with bad parity (parity error).

4.5.3.1 Time-Out Error – When the processor initiates a UNIBUS or EUB cycle, it asserts MSYN. After the processor has asserted MSYN, the processor (as bus master) must receive an SSYN signal from the addressed device within 25 microseconds. Logic on K1 (Figure 4-25) is used to monitor the return of BSSYN. The processor waits for BSSYN during time state PT3. Entering time state PT3 stops the processor clock and holds K1 MCLK H asserted. This action, along with the assertion of K1 CPU MAST H and K2 LSYNC H, causes K1 VALID MAST L to be asserted and triggers the one-shot E62. If BSSYN is not returned within 25 microseconds, one-shot E62 times out and sets flip-flop E67. This action asserts K1 BUS TIMEOUT H to signal a bus timeout error.

The second instance of a timeout occurring is during an interrupt cycle. During an interrupt cycle a device gains control of the bus via a BUS REQUEST. If a higher priority device is not waiting for the bus, a BUS GRANT comes back to the device to indicate that the device is now the bus master. The device then issues a BUS INTR L signal and sends out an interrupt vector to the processor. The device waits for BSSYN from the processor in time state PT3. Entering time state PT3 stops the processor clock and holds K1 MCLK H asserted. This action causes K1 VALID MAST L to be asserted and triggers one-shot E62. If BSSYN is not returned within 25 microseconds, one-shot E62 times out and sets flip-flop E67. This action asserts K1 BUS TIMEOUT to signal a bus timeout error.

When a timeout error occurs, the current operation is aborted and a trap is executed through the vector at location 4. An exception to this action is when the timeout occurs while an interrupt vector is being read, in which case no action is taken. If BSSYN is returned within 25 microseconds (thus signaling a successful transfer), the processor clock is restarted and one-shot E62 and flip-flop E67 are cleared when MCLK goes on the transition from PT to PBT.

4.5.3.2 **Parity Error** – Modern memory systems employ methods of error detection and, in some cases, error correction. If an uncorrectable error is detected when reading from memory, the memory subsystem notifies the processor by asserting K2 PB L and not asserting K2 PA L. This action causes K2 PAR ERR H to be asserted and the processor to abort the current operation. When the operation is aborted the processor will trap through the vector at location 114_8. The exception to this is when the console is in the console ODT mode, in which case an error is indicated by ODT printing ?⟨CR⟩⟨LF⟩@.

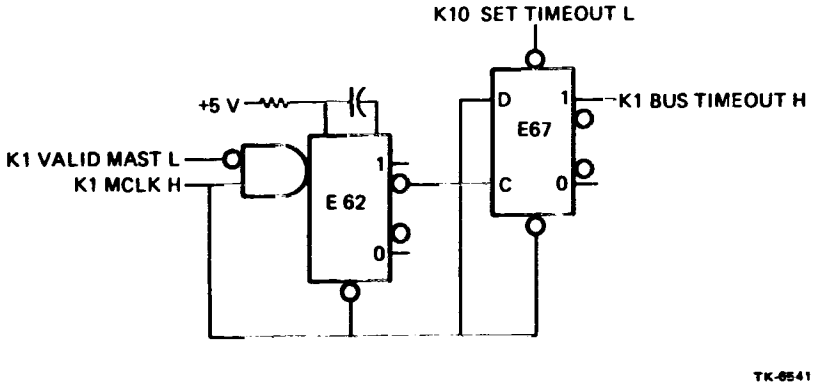


Figure 4-25 Bus Time-Out Logic

4.6 INTERNAL ADDRESS DECODE

The internal address decode logic shown in Figure 4-26 provides I/O units contained on the processor module with the necessary control information. The address decode PLA monitors the address lines PAX⟨12:00⟩, control line C0, L READ, FS MAINT from the field service maintenance switch, and L BSIO, which indicates a reference to the I/O page. The print set contains the PLA truth table for the internal address decode logic. The truth table specifies the necessary inputs to the address decode PLA to generate the control information for the SLUs, the line time clock, and the internal display register (two LEDs located on the module). Two decoders are used to generate the actual control signals for the I/O units. One decoder generates the control signals that enable the reading of registers; the other generates the control signals that enable the writing of the I/O registers. The read decoder is enabled when:

- The address decode PLA has the address of a valid internal register.
- DIR IN H is asserted, indicating that the data transfer direction is inward as a result of an input microinstruction.
- The output of the SLU timing shift register is low.

Table 4-7 describes the outputs of the read decoder.

The write decoder is enabled when:

- The address decode PLA has decoded a data out to an address of a valid internal register.
- BUS REQ L is asserted, indicating a data transfer or MAINT L, RDONE L is asserted.
- The output of the SLU timing shift register is low.

Table 4-8 describes the outputs of the write decodes.

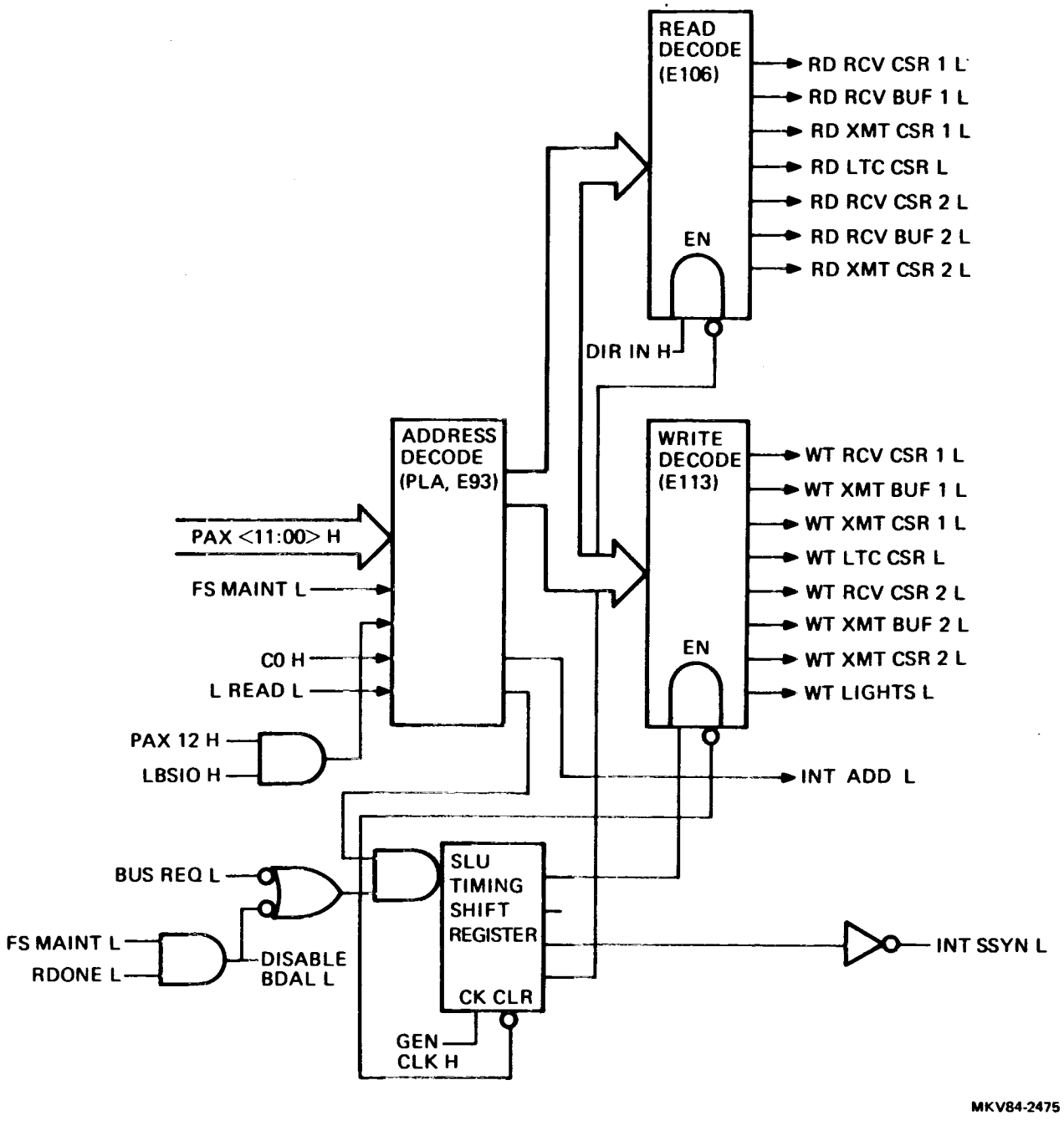


Figure 4-26 Internal Address Decode

Table 4-7 Read Decoder Outputs

Decoded Output	Description
RD RCV CSR 1 L	Reads the contents of the SLU1 receiver control/status register of SLU1 onto the BDAL.
RD RCV BUF 1 L	Reads the contents of the SLU1 receiver data buffer onto the BDAL.
RD XMT CSR 1 L	Reads the contents of the SLU1 transmitter control/status register onto the BDAL.
RD LTC CSR L	Reads the contents of the line time clock control/status register onto the BDAL.
RD RCV CSR 2 L	Reads the contents of the SLU2 receiver control/status register onto the BDAL.
RD RCV BUF 2 L	Reads the contents of the SLU2 receiver data buffer onto the BDAL.
RD XMT CSR 2 L	Reads the contents of the SLU2 transmitter control/status register onto the BDAL.

Table 4-8 Write Decoder Outputs

Decoded Output	Description
WT RCV CSR 1 L	Writes data from the BDAL to the SLU1 receiver control/status register.
WT XMT BUF 1 L	Writes data from the BDAL to the SLU1 transmitter data buffer.
WT XMT CSR 1 L	Writes data from the BDAL to the SLU1 transmitter control/status register.
WT RCV CSR 2 L	Writes data from the BDAL to the SLU2 receiver control/status register.
WT XMT BUF 2 L	Writes data from the BDAL to the SLU2 transmitter data buffer.
WT XMT CSR 2 L	Writes data from the BDAL to the SLU2 transmitter control/status register.
WT LIGHTS L	Writes data from the BDAL to the internal display register on the CPU module.
WT LTC CSR L	Writes data from the BDAL to the line time clock control/status register.

4.7 SERIAL LINE UNITS

4.7.1 Console Terminal SLU

The console terminal SLU (SLU1) is the I/O port for an LA36 or equivalent EIA serial console. The console terminal operates in two modes: program I/O mode (as the system terminal) or console mode (as the programmer's console terminal).

In the console mode the terminal is used to functionally replace the switch register and light display of the traditional control panel. In this mode all characters input on the terminal are interpreted as console commands. Chapter 3 provides a description of the console commands. In the program I/O mode the terminal provides the user with an information path to and from a running program. SLU1 enables the transfer of data between the processor (parallel data) and the external terminal (serial data). The universal asynchronous receiver/transmitter (UART) (Figure 4-27) is the major functional area of the interface. The UART operation is described in the following paragraphs. (See Appendix D, Section D.3, Figure D-5 for corresponding M7133-YA CPU logic diagram.)

4.7.1.1 Transmitter Operation - Parallel data to be transferred from the central processor to the terminal is input to the UART on the BDAL (BDAL(07:00)). When the UART is in the idle state, the serial output (SERIAL OUT 1 H) is high. The parallel data is strobed into the UART by WT XMIT BUF 1 H. When the data is transferred to the transmitter shift register in the UART, XMT RDY 1 H is asserted, indicating that the data is being transferred to the terminal, and new data may be loaded into the transmitter data buffer register. The format of the serial character is determined by control inputs to the UART. Three jumpers provide the necessary control signals for the format of serial data. A description of these jumpers is provided in Table 4-9. The rate at which the serial data is transmitted is switch selectable and is outlined in Paragraph 4.7.3. (See Appendix D, Section D.2, Table D-1, for M7133-YA CPU switch/jumper locations.)

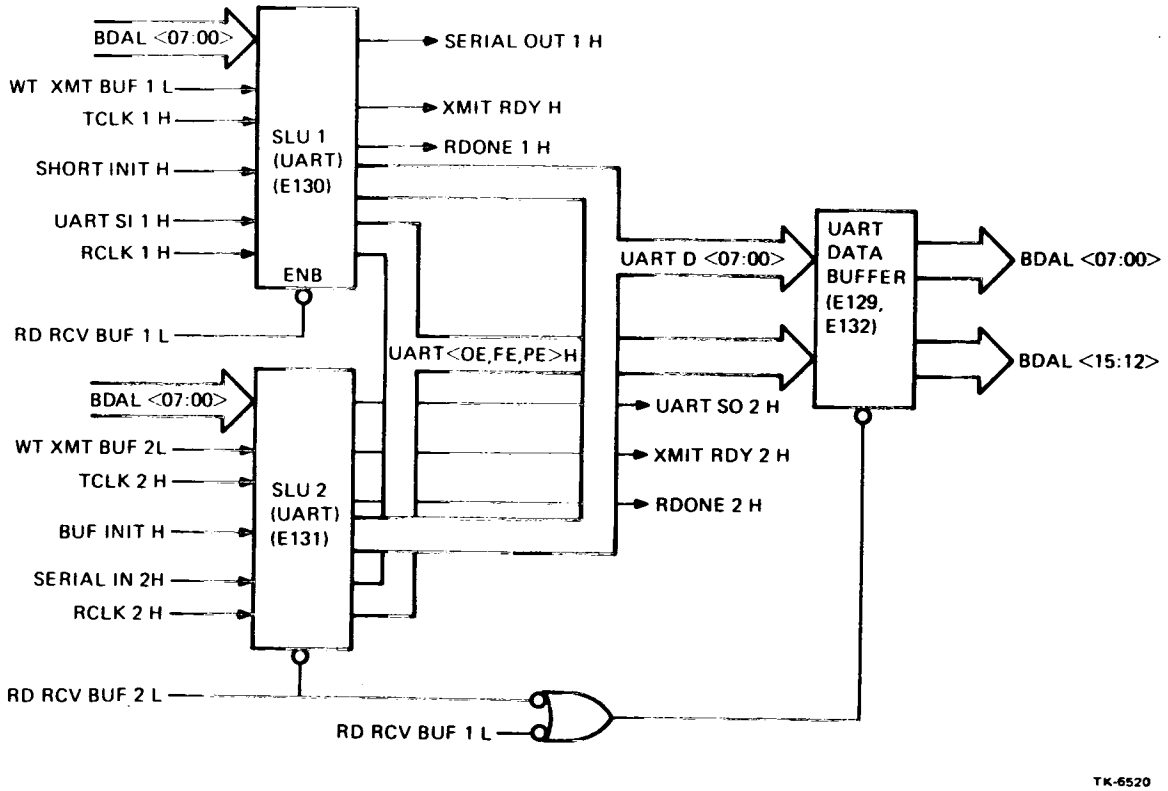


Figure 4-27 Serial Line Interface

Table 4-9 SLU1 Data Format Jumpers

Jumper	Description
W5	When IN, the SLU allows parity generation and checking; when OUT, the SLU inhibits parity generation, checking, and forces the parity error bit in the receiver/data buffer register to be cleared.
W6	When IN, the SLU generates and checks ODD parity; when OUT, the SLU generates and checks EVEN parity.
W1	Stop bit select When IN, one stop bit is generated (above 110 baud). When OUT, two stop bits are generated (110 baud and below).

4.7.1.2 Receiver Operation – The receiver section of the UART accepts a serial character from the terminal for conversion to parallel data. The parallel data is transferred on the BDAL(07:00). The receiver samples the serial line input (UART SI 1 H) at selected edges of the receiver clock (RCLK 1 H). The source of the data for the serial line input is a multiplexer. During normal operations, the multiplexer selects serial data from the terminal (SERIAL IN 1 H). In the maintenance mode the multiplexer selects the serial output of the UART (SERIAL OUT 1 H). This enables a closed loop test of the receiver and transmitter.

The receiver becomes active when it detects a mark-to-space (high-to-low) transition of the serial line input. It then shifts in the required number of data bits, the parity bit (if enabled), and the stop bit(s). The contents of the shift register is then transferred into the receiver data holding register and the data available flag (R DONE) is set.

If the receiver parity detection has been enabled (jumper W7 IN), the receiver checks the parity of the data bits plus the parity bit following the data bits. The receiver compares the parity of the received data with the parity select line. Even or odd parity check is determined by jumper W9. If the parity of the received character differs from the parity of the UART control logic, the parity error line (UART PE H) is asserted. This causes bit 12 (parity error) as well as bit 15 (error) of the receiver data buffer register to be set.

The receiver samples the first stop bit which occurs after the parity bit or after the data bits if no parity check is selected. If the stop bit(s) is (are) valid (logical 1), it indicates that the entire character has been correctly received. A low on the first stop bit sampled by the receiver indicates an invalid stop code. The UART will then generate a framing error signal (UART FE H) which sets bit 13 (framing error) and bit 15 (error) in the terminal receiver data buffer register.

A framing error can be caused by pressing the "break" key on the console terminal. If the console keyswitch is in the LOCAL position, the framing error will pass to the halt request line and stop the processor. This allows ODT mode to be entered without forcing the user to leave the console terminal.

In addition to the parity error and framing error conditions, a third error condition, overrun, is associated with receiver operation. The overrun condition indicates that a new character has been loaded into the UART holding register before the previous character was removed. This destroys the character already there. The UART will assert the overrun error line (UART OE H) when this occurs. Bit 14 (overrun error) and bit 15 (error) of the receiver data buffer register will also be set.

4.7.1.3 SLU1 Maintenance Configuration – The console terminal interface has two maintenance features which allow the UART as well as other portions of the interface to be checked. One feature is the maintenance bit (bit 02) of the transmitter control/status register which allows the looping of the transmitter's serial output back into the serial input of the receiver (Figure 4-28). A diagnostic routine contained in the M9312 diagnostic ROM sets this bit. Upon successful completion of the diagnostic routine the maintenance bit will be cleared. Other diagnostics use this bit in a similar fashion.

NOTE

The maintenance bit is set under program control. It should never be set from the console. This will effectively remove the console from the system, in which case a system initialization is necessary to allow the console terminal to be used again.

The other feature is the maintenance switch on the module (S2-5) which allows the looping of the received data back to the transmitter via the BDAL (Figure 4-29). The terminal is effectively put in local when the maintenance switch is set. This tests the receiver, transmitter, baud rates, and part of the internal data bus. The operator sets the switch and then types a ⟨U⟩ or an ⟨*⟩. The ⟨U⟩ should be received back as an ⟨*⟩, and the ⟨*⟩ should be received back as a ⟨U⟩. This will indicate proper operation of the terminal, UART, and a portion of the BDAL. This function is performed entirely without CPU intervention; however, it will destroy the program's state. The SLU shift register (K9) times operation of SLU1 while in field service maintenance mode.

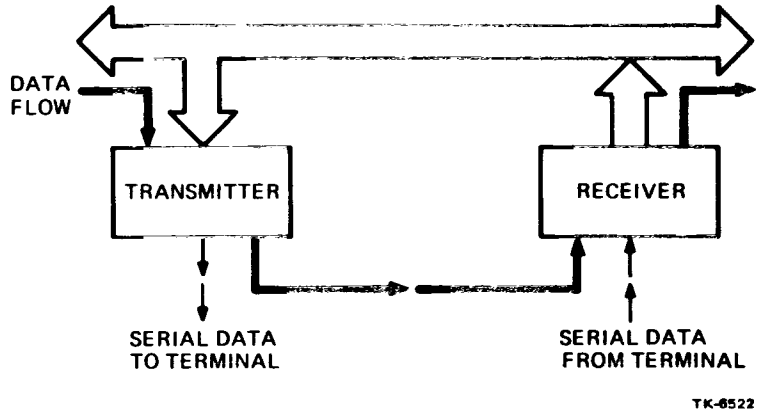


Figure 4-28 Maintenance Bit (XCSR) Maintenance Configuration

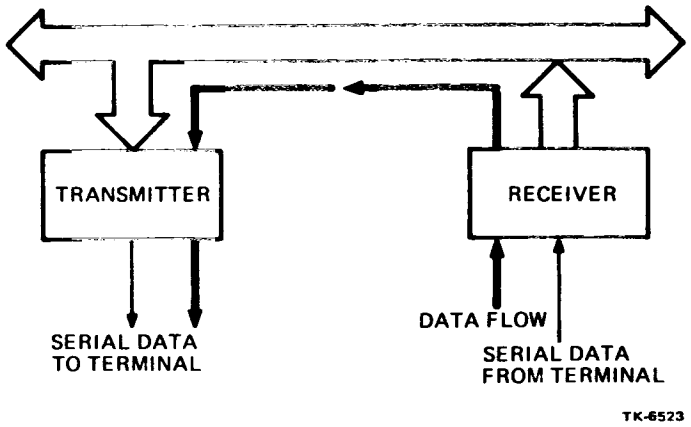


Figure 4-29 Maintenance Switch (S2-5) Maintenance Configuration

4.7.2 Serial Line Unit 2

Serial line unit 2 (SLU2) provides a general-purpose I/O port for the user. Operation of SLU2 is similar to that of SLU1. Serial data transferred to and from the I/O port is converted to parallel data for use by the central processor. As in SLU1 the UART is the major functional area of the interface.

4.7.2.1 Transmitter Operation (SLU2) - Parallel data on the BDAL (BDAL(07:00)) is loaded into the UART when the processor performs a DATO(B) to the SLU2 UART transmitter data buffer. The address decode logic generates WT XMT BUF 2 L, which loads the data on the BDAL into the UART transmitter data buffer.

The parallel data is converted to serial data by the UART and transmitted via UART SO 2 H. The I/O device receives the UART data via SERIAL OUT 2 H. The format of the serial data is determined by control inputs to the SLU2 UART. These control inputs are jumper selectable and are described in Table 4-10. (Refer to Appendix D, Section D.2, Table D-1 for M7133-YA CPU switch/jumper locations.)

Table 4-10 SLU2 Data Format Jumpers

Jumpers	Description
W8	When IN, the SLU generates and checks ODD parity; when OUT, the SLU generates and checks EVEN parity.
W7	When IN, the SLU allows parity generation and checking; when OUT, the SLU inhibits parity generation, checking, and forces the parity error bit in the receiver data buffer register to be cleared.

The SLU2 transmitter functions are similar to those of SLU1. SLU2 has the additional capability of transmitting a "break," which will set a framing error at the user's receiver. This is done by setting bit 0 in the SLU2 XMITR CSR (K8). The serial output of SLU2 will generate spaces as long as bit 0 is set. Bit 0 must remain set until the time a stop bit would have been set. The easiest way to time this is by transmitting characters on SLU2. The characters will not be sent since the line is held spacing, but the done bit is operational and properly timed. To overcome the double buffering of the UART, at least two characters (NULLS) should be sent; then bit 0 may be cleared.

4.7.2.2 SLU2 Receiver Operation - The receiver accepts serial data from user's device. The SLU2 UART does not have the diagnostic feature which allows the serial output of the UART to be fed back into the serial input. In order to perform a diagnostic check, a loopback connector (H325 or equivalent) and the appropriate diagnostics are required. All other receiver operations are similar to SLU1.

4.7.3 Baud Rate Logic

The baud rates for the console terminal receiver and transmitter are derived from a 2.7648 MHz crystal oscillator. This crystal controlled clock is then fed to:

1. A dual rate baud generator which develops two independent rates (50 to 19200 baud).
2. A fixed divider which develops 19200 baud.

The dual baud rate generator has switch selectable inputs which select its two output frequencies and the corresponding baud rates. Table 2-2 lists the switch settings and resulting baud rates.

The two switch selectable outputs (E135 S1-4 and S5-8) from the dual baud rate generator are known as K8 BR1 H and K8 BR2 H. The fixed 19.2K baud rate signal is called K8 19.2K BAUD. These three signals are then fed through switch pack E124 S1 through S4 and jumpers W4, W9–W13 to determine the SLU baud rates.

The SLU1 transmitter baud rate is selected by S1 and S2 of E124. The receiver baud rate is selected by S3 and S4 of E124. This arrangement allows the transmitter and receiver to be set to the same baud rate (baud rate 1 or 2), or to be set to different baud rates. When setting the transmitter and receiver to different baud rates the transmitter can be set to baud rate 1 and the receiver set to baud rate 2 or the transmitter set to baud rate 2 and the receiver set to baud rate 1. Table 2-3 lists the switch settings for SLU1 baud rates. (See Appendix D, Section D.2, Table D-1 for M7133-YA CPU baud rate switch settings.)

The SLU2 transmitter and receiver baud rates are selected by jumpers W4, W9–W13. The baud rates may be selected in a manner similar to SLU1 with the addition of the fixed 19.2K baud rate.

4.8 LINE TIME CLOCK (LTC)

The line time clock (Figure 4-30) provides the system with timing information at fixed intervals of 16.7 ms (60 Hz line frequency) or 20 ms (50 Hz line frequency). The LTC is KW11-L compatible with the exception of the monitor bit, which is automatically cleared after each interrupt. While in the interrupt mode an interrupt is generated for each cycle of the line frequency. When the interrupt mode is disabled, the monitor bit may be tested and cleared under program control without causing an interrupt. The line clock status register (LKS) contains two bits associated with the LTC operation: a monitor bit to provide noninterrupt mode timing information and an interrupt enable bit to allow the LTC to initiate interrupt sequences. The address of the LKS is fixed at 17 777 546$_8$. The monitor bit (bit ⟨07⟩) provides the software with a means of measuring time in a noninterrupt mode. This bit of the register is placed on BDAL⟨07⟩ when read. It is set once for each cycle of the ac power or after a system initialization. The program clears the monitor bit after noting that it was set. The monitor bit is cleared when either of the following conditions occur:

- GPO code = 5, which results in the signal DGP05 L being asserted. The microcode issues GPO 5 after honoring an LTC interrupt.
- Program writes a zero into the bit (the program cannot set the bit).

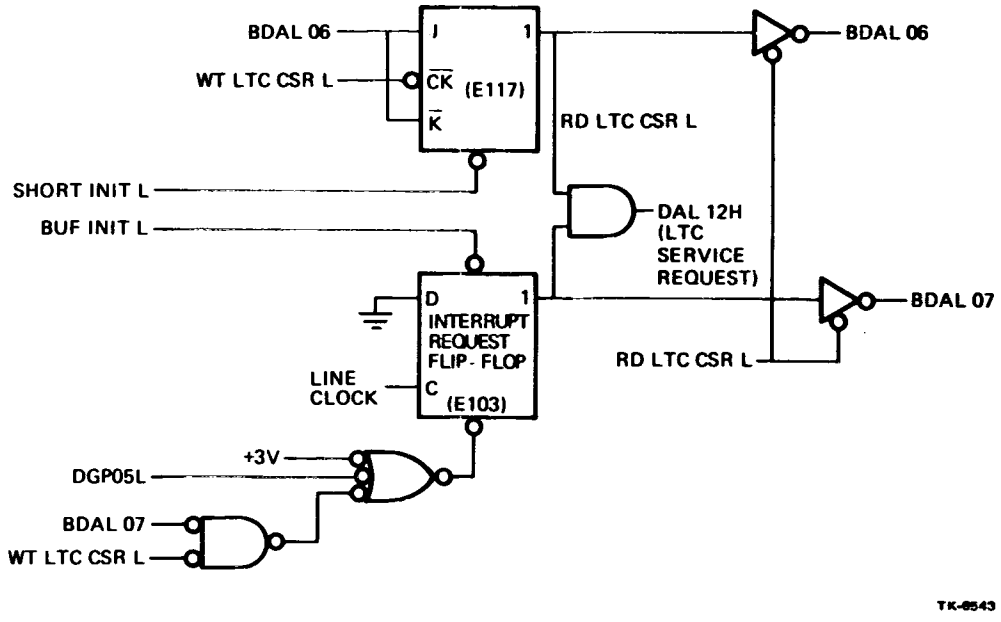


Figure 4-30 Line Time Clock

The LTC interrupt enable bit (bit ⟨06⟩) is a read/write bit which allows the LTC to generate periodic interrupt sequences. This bit is written when WT LTC CSR L is asserted. System initialization (BUF INIT L) clears this bit. This bit of the register is placed on BDAL ⟨06⟩ when read. Both bits in the LKS are read when RD LTC CSR L is asserted. The LTC interrupt request logic consists of the ready flip-flop and an AND gate. The flip-flop is set on the trailing edge of LINE CLOCK. The interrupt request is then presented at service time directly to the chip set on DAL ⟨12⟩. The hardware normally required to arbitrate the interrupt and generate a vector has been replaced by microcode in the chip set. The interrupt vector is fixed at 100_8 by the microcode. (See Appendix D, Section D.5, Figure D-6 for corresponding M7133-YA logic diagram.)

4.9 DISPLAY REGISTER

Two LEDs contained on the CPU module are used to implement a display register. The register is located at address 17 777 570 and is write-only. Any attempt to read this register will result in a time-out.

The display register consists of two flip-flops (E34) located on K12. The flip-flops receive BDAL ⟨01:00⟩ and are clocked by the signal K9 WT LIGHTS L from the internal address decoder. When set, the flip-flops directly drive the LEDs which contain internal current limiting resistors. (See Appendix D, Section D.3, Figure D-10 for M7133-YA signal names and logic diagram.)

4.10 INTERRUPT REQUEST LOGIC

The interrupt logic enables SLU1, and SLU2 and the line time clock (LTC) to interrupt the processor and initiate a service routine. The interrupt request logic for SLU1 is shown in Figure 4-31; the interrupt request logic for SLU2 is shown in Figure 4-32; and the LTC interrupt logic has been described in Paragraph 4.8. Each of these devices will direct the processor to the correct routine by generating the appropriate vector. Table 4-11 lists the interrupt vectors which correspond to each device interrupt. (See Appendix D, Section D.3, Figures D-7 and D-8 for corresponding M7133-YA CPU logic diagrams.)

The LTC requests an interrupt on a private request line which is equivalent to the highest BR6 request. The SLUs request an interrupt on the public BR4 line.

When the processor has arbitrated an interrupt other than the LTC, ALLOW GRANTS H will be asserted. Meanwhile, the CPU has encoded a grant signal (BG4–BG7) on the PAX lines and the grant has been latched into the PAX latch. For the purpose of this example, assume SLU1 is being serviced. The processor will assert BG4 (bus grant) by asserting PAX ⟨08⟩ (remember, the grants are encoded onto the PAX bits at service time). This indicates that the bus has been granted to at priority 4. Since SLU1 requested the interrupt, it will use the BG to initiate its interrupt sequence. If SLU1 did not require an interrupt, K10 INT PASS BG4 L would be asserted, passing the grant to SLU2. SLU2 will generate K12 INT PASS BG4 L if it does not require an interrupt. This daisy chain effect establishes interrupt priorities for devices at the same BR level. Whatever device accepts the grant will then generate BUS SACK acknowledging receipt of the BUS GRANT. BBSY and MSYN will eventually be cleared by the previous bus master indicating that the bus is free (from the previous bus cycle). SLU1 will now assert BBSY indicating that it is the new bus master.

SEND VECT 1 H will be asserted since SLU1 requested the interrupt. SEND VECT 2 H will be asserted if SLU2 requested the interrupt. This places the appropriate interrupt vector on the BDAL. XMT VECT 1 H (SLU1) or XMT VECT 2 H (SLU2) will be asserted if the respective transmitter caused the interrupt. XMT VECT 1 H (SLU1) or XMT VECT 2 H (SLU2) will not be asserted if the receiver caused the interrupt. SSYN will then be asserted by the CPU after it has read the interrupt vector. The interrupt vector and BUS BUSY are then cleared by the device and the processor clears SSYN. This indicates that the interrupt sequence has been performed satisfactorily.

4.11 INTERRUPT ERRORS

Two errors may occur during the processing of a interrupt. The errors are SACK time-out and vector time-out.

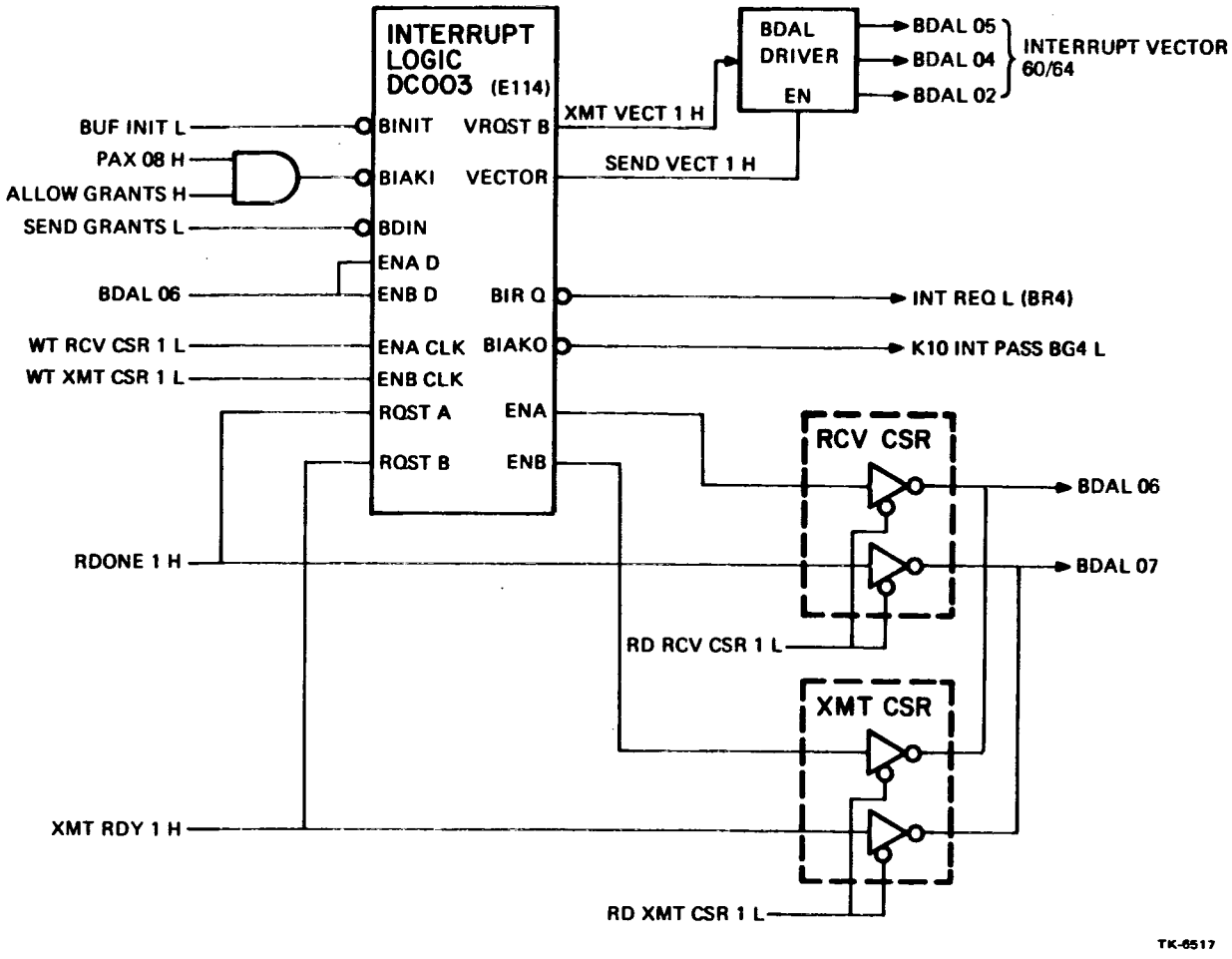


Figure 4-31 SLU1 Interrupt Logic

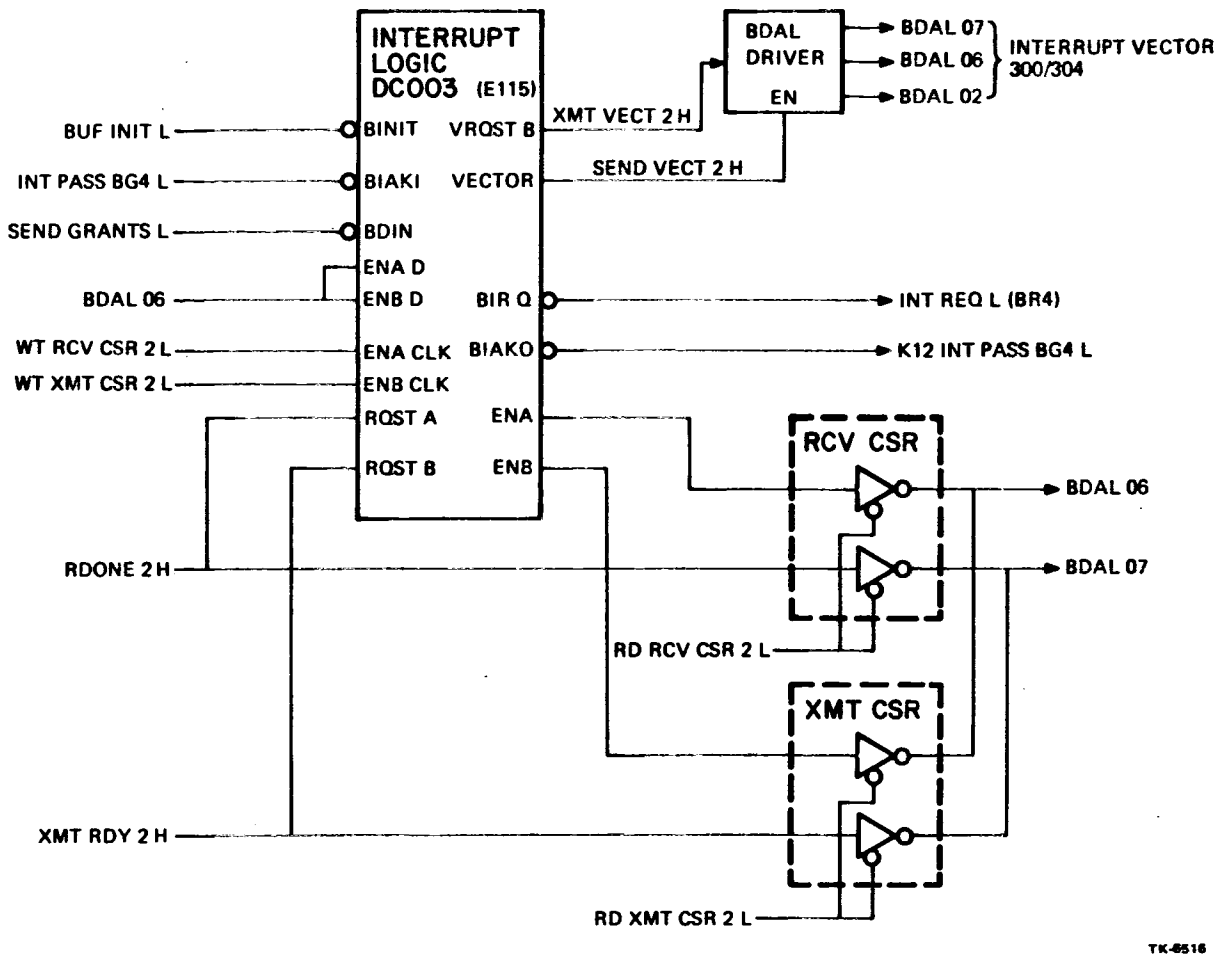


Figure 4-32 SLU2 Interrupt Logic

Table 4-11 Interrupt Vectors

Device	Interrupt Vector	Priority
SLU1 (Console Terminal) Receiver	060_8	4
SLU1 (Console Terminal) Transmitter	064_8	4
SLU2 Receiver	300_8	4
SLU2 Transmitter	304_8	4
Line Clock	100_8	6

4.11.1 SACK Time-Out

SACK (selection acknowledge) informs the CPU that a device has accepted the bus grant issued by the CPU. After issuing a grant the CPU must receive SACK before it can continue. Two conditions may cause a failure of SACK being sent:

1. No real device requested the interrupt. The request could have been caused by noise or a glitch on the BR line.

2. A real device had requested an interrupt but removed its request before the bus grant was sent.

When this condition is present the NO SACK time-out one shot, E99 (K10), times out after 25 μs and issues K10 BUS SACK L. The CPU now attempts to get an interrupt vector. This action will also fail.

4.11.2 Vector Time-Out

A device signals the availability of its interrupt vector by asserting BUS INTR L. If the device requesting an interrupt is defective or a nonexistent interrupt is being serviced, no vector is given to the CPU. When this happens the CPU waits for the BUS INTR L, which indicates that a vector has been asserted. Since no vector is asserted the CPU could wait forever. However, because the vector is not asserted, SSYN is not generated and the SSYN one shot, E62 (K1), times out (25 μs) and resets the base machine.

The resulting bus error from the vector time-out does not trap through 4_8 because the microcode sets an internal flag prior to attempting to fetch the vector. When this flag is set and a bus error occurs, no action is taken, and the CPU continues as if the interrupt had never been processed.

CHAPTER 5
MEMORY MANAGEMENT

5.1 INTRODUCTION

The PDP-11/24, like other PDP-11 processors, generates a 16-bit address. The 16-bit address is able to specify 64K byte addresses (32KW). This address differs from the 22-bit address (2048KW) used by the PDP-11/24 to address main memory. Thus, the 16-bit processor-generated addresses are called virtual addresses (VA), and the 22-bit memory addresses are called physical addresses (PA). It is the function of the memory management unit (MMU) to convert a 16-bit VA to a 22-bit PA. This is called relocation.

The 22-bit addresses (PA) used by the PDP-11/24 allow a possible address space of 2048KW. This address space is called physical address space. Figure 5-1 shows how physical address space, defined as follows, is divided.

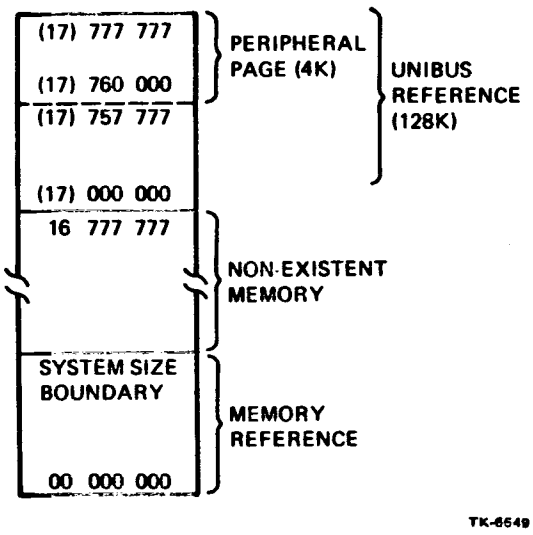


Figure 5-1 Physical Address Space

1. UNIBUS References – UNIBUS references access the upper 128KW of physical address space, 17 000 000$_8$ – 17 777 777$_8$, which correspond to UNIBUS addresses 000 000$_8$ – 777 777$_8$. UNIBUS references include the following:

   a. The peripheral page, which is reserved for UNIBUS device registers and consists of the upper 4KW of UNIBUS references.

   b. The remaining 124KW of UNIBUS space may be used by UNIBUS devices to access memory.

2. Memory References – Memory references include PAs from 00 000 000$_8$ through the system size boundary, which is the highest address in the system main memory.

3. Nonexistent Memory – Nonexistent memory includes the PAs from the system size boundary plus one through 16 777 777$_8$.

The main function of the MMU is address relocation and protection. The MMU performs address relocation by dividing the 32KW virtual address space into eight pages of 4KW each. Each page may then permit access to all or only a portion of its 4KW physical address space. In addition, each page may be protected by allowing it to be read only. Another form of protection is that which does not allow the processor to access physical addresses that are not mapped in the processor's current mode of operation, kernel or user. These features provide protection by allowing each user in a multiprogramming environment access to only those pages assigned to that user.

It is often desirable to load a program into one area of physical memory and execute it as if it was located in another area of memory, e.g., when several user programs are simultaneously stored in memory. When any one program is running, it must be accessed by the processor as if it was located in the set of addresses beginning at 000 000$_8$. This is possible via relocation. When the processor accesses virtual bus address 000 000$_8$, a base address is added to the virtual address, and the physical address produced points to the location of the program. Typically, this base address is added to all references while the program is running. An appropriate base address is used for each of the programs in memory.

Memory management specifies relocation on a page basis, which allows a large program to be loaded into nonadjacent physical pages in memory. This ability eliminates the need to shuffle programs to accommodate a new one. It also minimizes unusable memory fragments, thus allowing more users to be loaded into a specific memory size.

A program and its data can occupy as many as eight pages in memory. The size of each page may vary and can be any multiple of 32 words up to 4096 words in length. This feature allows small areas of memory to be protected, e.g., stacks or buffers, and also allows the last page of program, which exceeds 4K words, to be of adequate length to protect and relocate the remainder of the program. As a result, the memory fragmentation problem inherent with fixed-length pages is eliminated. The base address of each page can be any multiple of 32 words in the physical address space, thus ensuring efficient use of main memory. The variable page length also allows the pages to be dynamically changed at run time.

Memory management provides two separate sets of pages: one set for use by the processor while it is in kernel mode and another for use in the user mode. These sets of pages increase system protection by physically isolating user programs from the kernel program. The separate relocation register sets also greatly reduce the time necessary to switch context between kernel and user mapping. The two sets of registers also aid the user in designing an operating system that has clearly defined communications, is modular, and is more easily debugged and maintained.

5.2 RELOCATION

Relocation may be performed in one of the three modes. While relocation is off, the processor is said to be operating in the 16-bit mode. The virtual address is passed directly to the physical address bus and then main memory with the exception of virtual addresses in the range 160 000 through 177 777. These addresses are passed to the peripherals page located at physical addresses 17 760 000 through 17 777 777 (Figure 5-2).

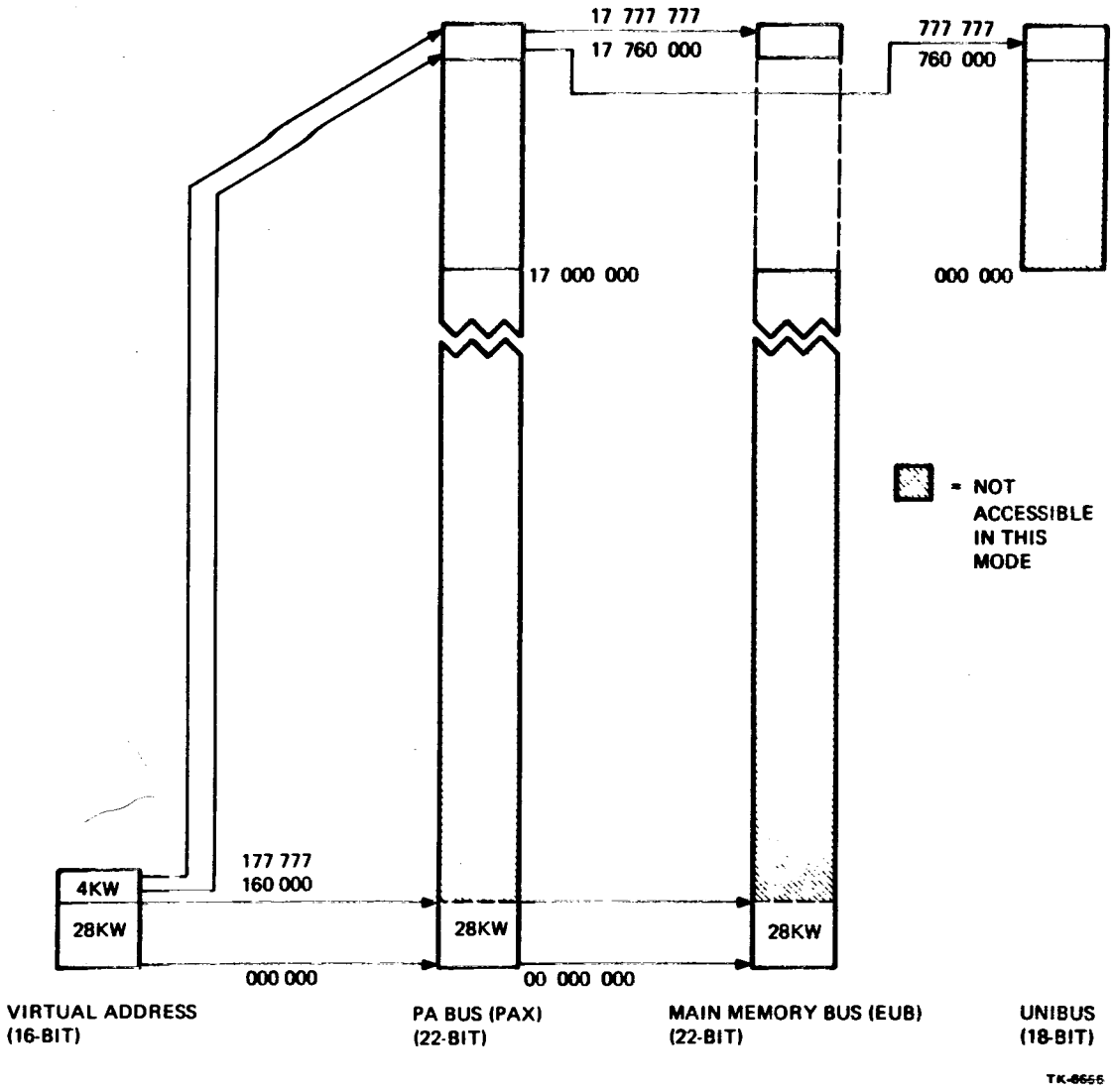


Figure 5-2 16-Bit Mode

The processor operates in the 16-bit mode whenever the MMU chip is absent or disabled, or immediately after the processor is initialized by a console GO command or by a reset instruction.

The processor may also operate in either the 18-bit or 22-bit relocation modes. In the 22-bit mode relocation is fully enabled and the MMU can map virtual addresses anywhere in physical address space. In the 18-bit mode the MMU can only map the first 124KW of main memory and the peripherals page. In the 18-bit mode the PDP-11/24 memory management function is compatible with earlier PDP-11 processors (e.g., PDP-11/34, 11/40 and 11/60) employing memory management. Figures 5-3 and 5-4 illustrate the 18- and 22-bit modes of operation.

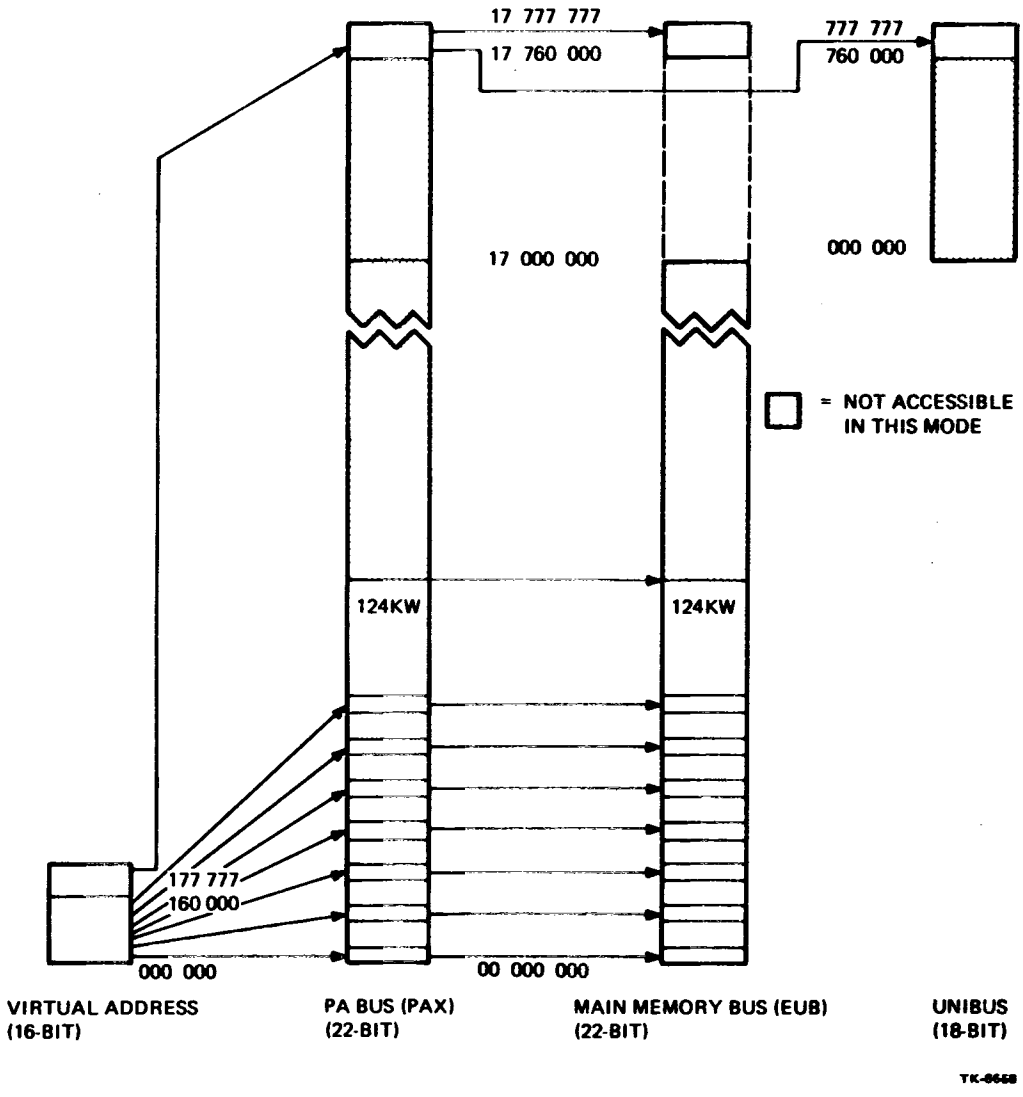


Figure 5-3 18-Bit Mode (Typical Case)

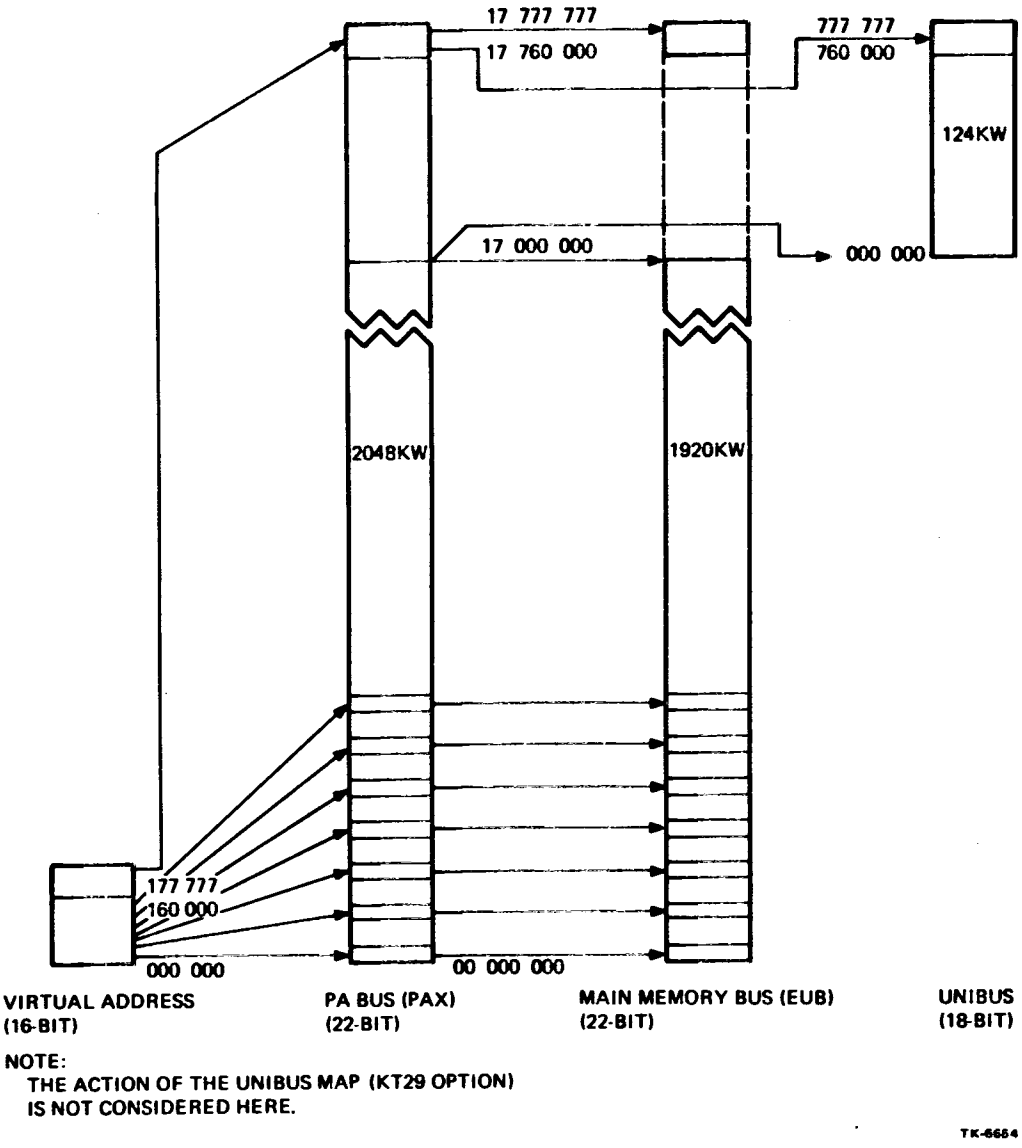


Figure 5-4 22-Bit Mode (Typical Case)

A switch-selectable submode of 16-bits and 18-bits mapping exists. Certain user programs written for earlier processors require physical addresses in the range 000 000_8 through 757 777_8 to be passed to the UNIBUS, rather than to only the main memory bus. Typical applications involve multiported UNIBUS memory, UNIBUS windows, and some graphics processors. To allow these earlier programs to run on the PDP-11/24, E124 S6 should be in the OFF position, thereby forcing all physical addresses to the UNIBUS as well as the memory bus. When this action occurs the 22-bit PA is truncated to 18-bits for use on the UNIBUS. Figures 5-5 and 5-6 illustrate this action. (See Appendix D, Section D.2 for M7133-YA CPU switch location.)

NOTE

Use of this switch is not recommended if the processor is operating in the 22-bit mode. Severe double addressing problems can occur. It is recommended to use direct programmed access to the UNIBUS when in the 22-bit mode.

Switch S6 should be on for normal operation of the PDP-11/24.

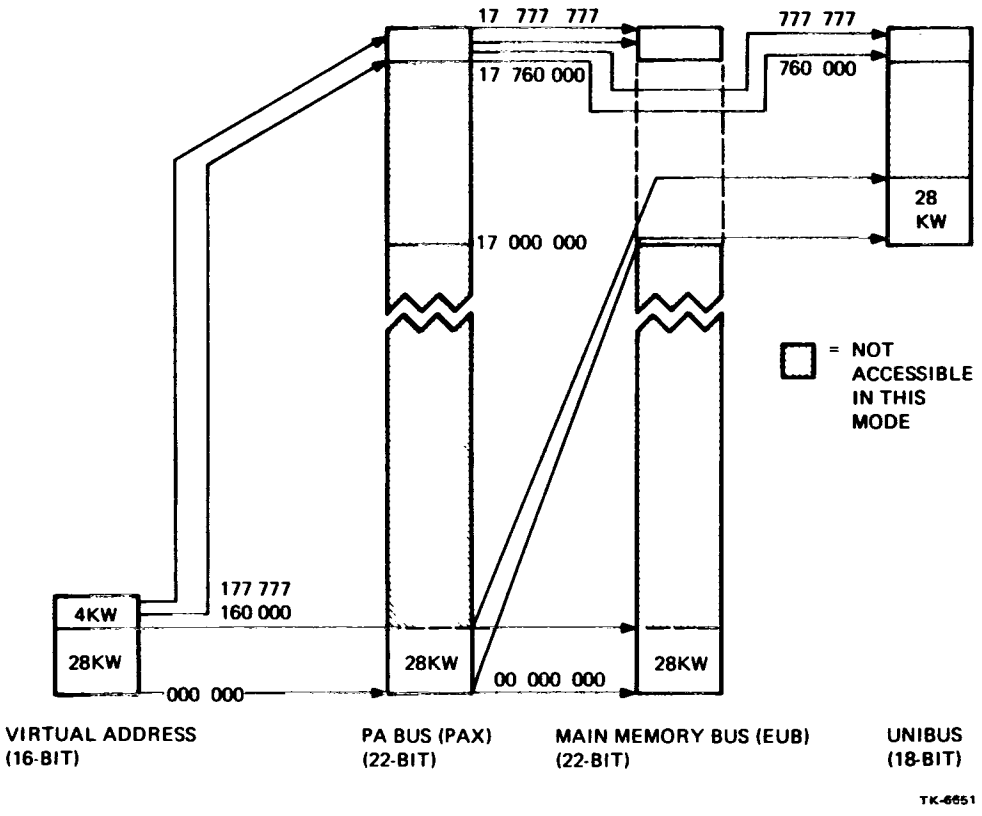


Figure 5-5 16-Bit Mode, E124, S6 OFF (Special Mode)

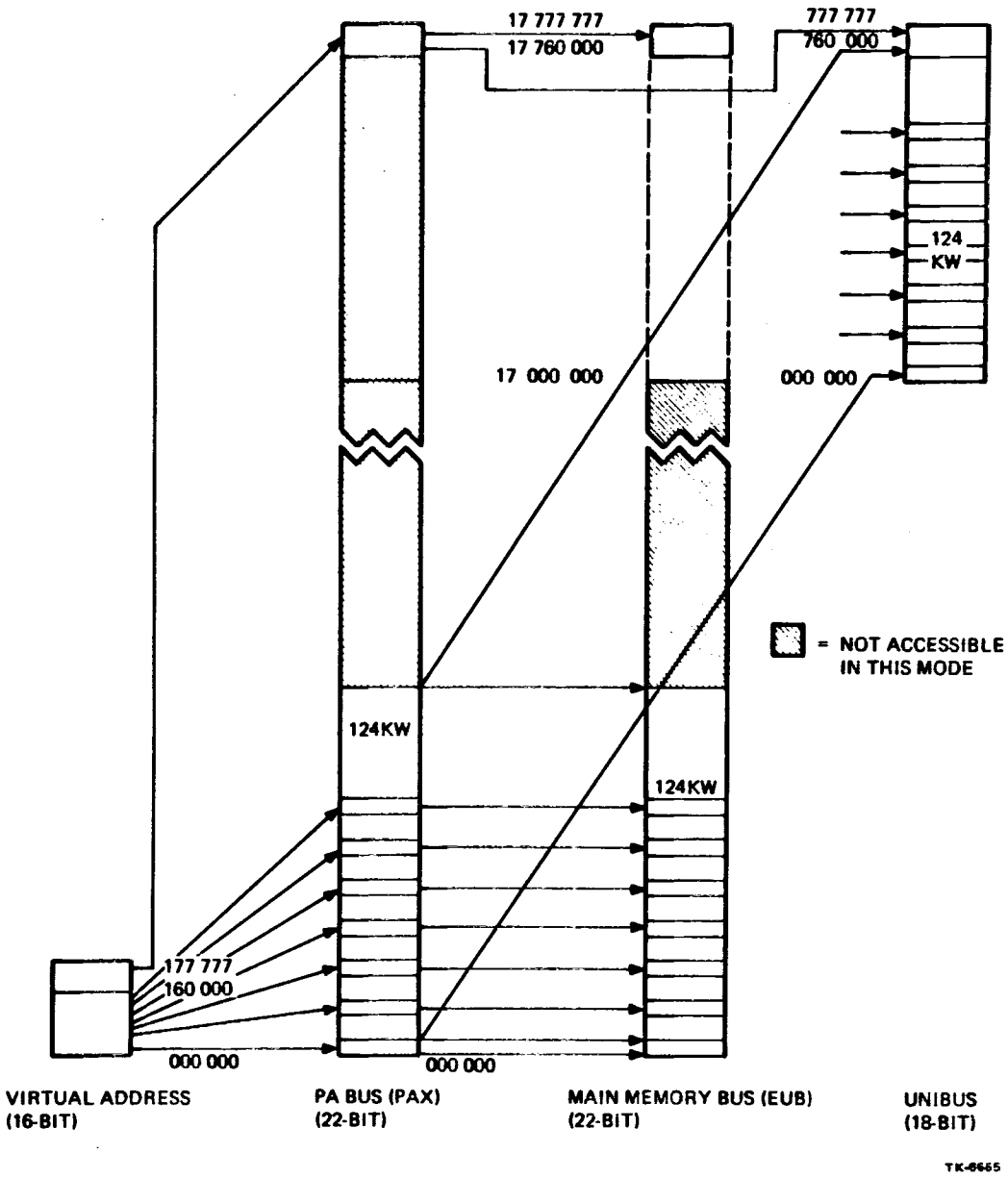


Figure 5-6 18-Bit Mode, E124, S6 OFF (Special Mode)
(Typical Case)

5.2.1 Address Mapping

The basic information needed for the construction of a PA comes from the VA, which is illustrated in Figure 5-7, and the appropriate APR set.

The virtual address consists of:

1. The Active Page Field (APF) – This three-bit field is used to determine which of the eight pages this VA belongs to, and thus, which of the eight PARs will be used to form the PA.

2. The Displacement Field (DF) – This 13-bit field contains an address relative to the beginning of a page. This permits page lengths of up to 4K words. The DF is further subdivided into two fields as shown in Figure 5-8.

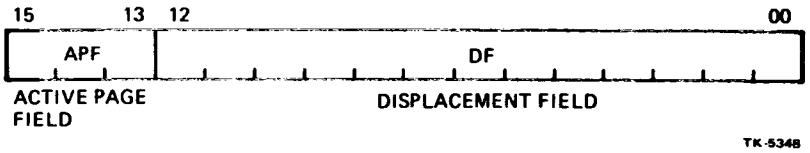


Figure 5-7 Interpretation of VA

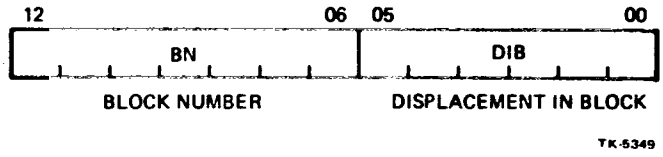


Figure 5-8 Displacement Field

The displacement field consists of:

1. The Block Number (BN) – This seven-bit field is interpreted as the block number within the current page.

2. The Displacement in Block (DIB) – This six-bit field contains the displacement within the block referred to by the block number (BN).

The remainder of the information needed to construct the PA comes from the 16-bit page address register and specifies the starting address of the memory page. The PAR is actually a block number in physical memory, e.g., PAR=3 indicates a page starting at the 96th (3 × 32) word in physical memory.

The formation of the PA is illustrated in Figure 5-9.

The logical sequence involved in constructing a PA is as follows.

1. The active page field of the VA is used to select a PAR (PAR0–PAR7).

2. The selected PAR contains the starting address of the currently active page as a block number in physical memory.

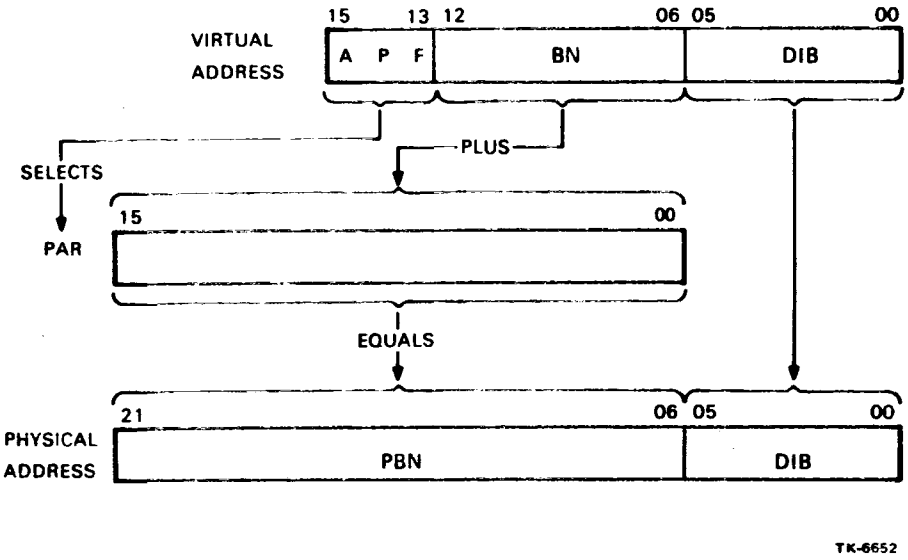


Figure 5-9 Construction of PA

3. The block number (BN) from the VA is added to the PAR to yield the number of the physical block in (PBN) memory which will contain the PA being constructed.

4. The displacement in block (DIB) of the VA is joined to the physical block number to yield a 22-bit PA.

5.2.2 Address Translation

The memory management unit (MMU) is able to relocate 16-bit addresses in one of two modes:

1. 18-bit mode – This mode of operation generally provides program compatibility with smaller PDP-11 processors, i.e., the PDP-11/34, 35, 40, 45, 55, 60.

2. 22-bit mode – This mode of operation generally provides compatibility with larger PDP-11 processors, i.e., PDP-11/44 and PDP-11/70.

Figure 5-10 is a block diagram of the memory management section of the MMU.

5.2.2.1 18-Bit Mapping – Refer to Figure 5-3. In 18-bit mapping the VA is added to bits ⟨11:00⟩ of the selected PAR to generate the PA. This address mode has a range of 128KW. The physical address space consists of 124KW ($00\ 000\ 000_8$ – $00\ 757\ 777_8$) and the 4KW peripheral page ($17\ 760\ 000_8$ – $17\ 777\ 777_8$).

Eighteen-bit mapping is enabled when bit 4 of SR3 is clear (18-bit mapping) and bit 0 of SR0 is set (relocation enabled). The MMU uses bits ⟨15:13⟩ of the VA to select the appropriate PAR-PDR pair.

After selecting the PAR the MMU adds bits ⟨12:06⟩ of the VA to bits ⟨11:00⟩ of the PAR (bits ⟨15:12⟩ are not used in 18-bit mapping and are masked off prior to the adder in the MMU). The sum of PAR ⟨11:00⟩ and VA ⟨12:06⟩ create PA ⟨21:06⟩ which are driven, along with bits ⟨05:00⟩ of the VA, onto the DAL to create the 22-bit physical address. If bits ⟨17:13⟩ of the PA are all 1s, a reference to the peripheral page is intended by the program. When this condition exists the MMU asserts –BSIO which forces logic on the CPU module (K7) to force bits ⟨21:18⟩ of the PA to 1s. If bits ⟨17:13⟩ of the PA are not all 1s, bits ⟨21:18⟩ of the PA are set to 0.

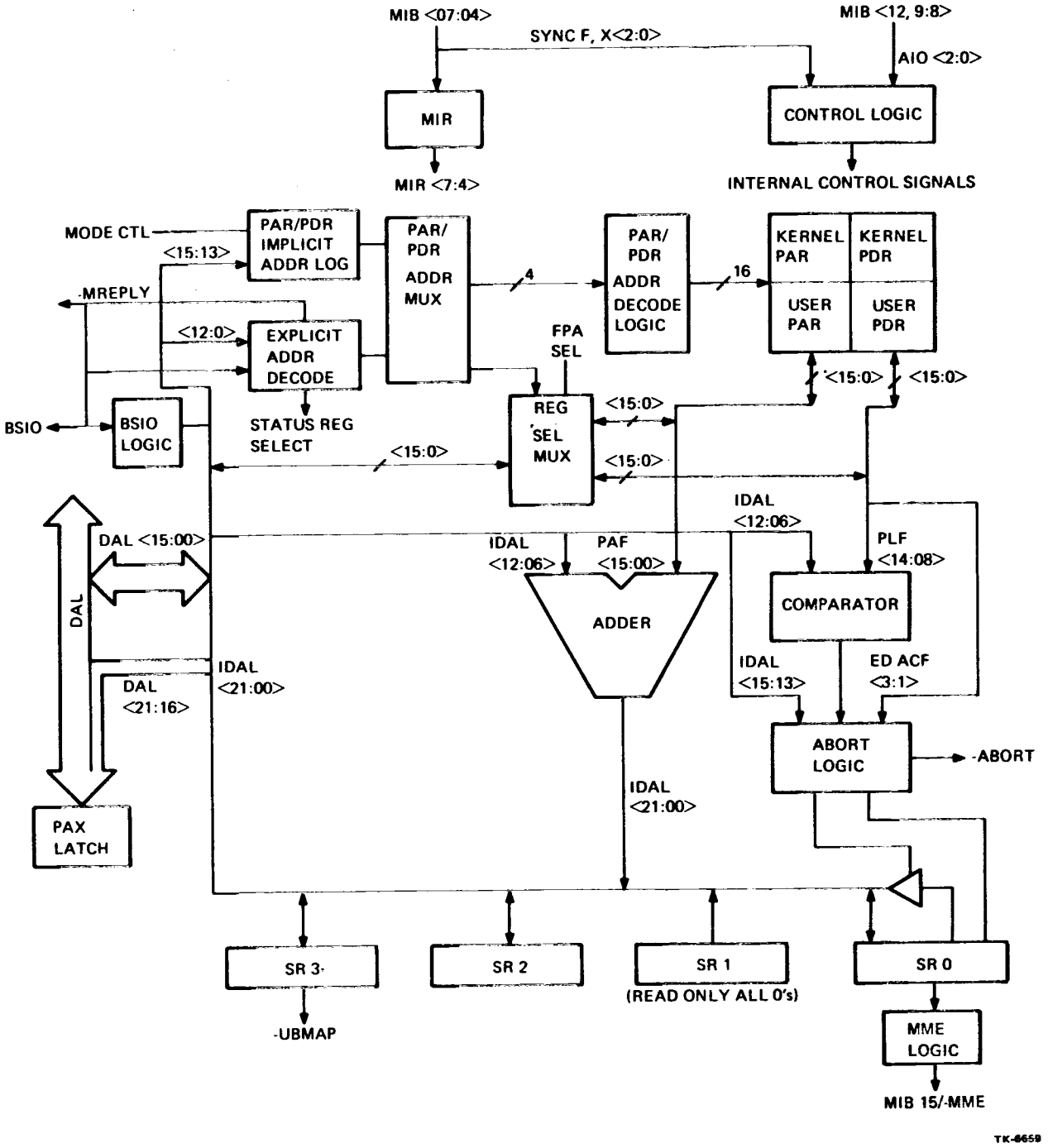


Figure 5-10 MMU Block Diagram

Figure 5-11 shows the case of an 18-bit PA that is not a UNIBUS reference, i.e., bits ⟨17:13⟩ are not all 1s. In this case bits ⟨21:18⟩ of the PA are set to 0s, which causes a memory reference.

Figure 5-12 shows the case of an 18-bit relocated address that is a UNIBUS I/O page reference, i.e., bits ⟨17:13⟩ are all 1s. In this case bits ⟨21:18⟩ of the PA are changed to 1s, which causes a UNIBUS reference.

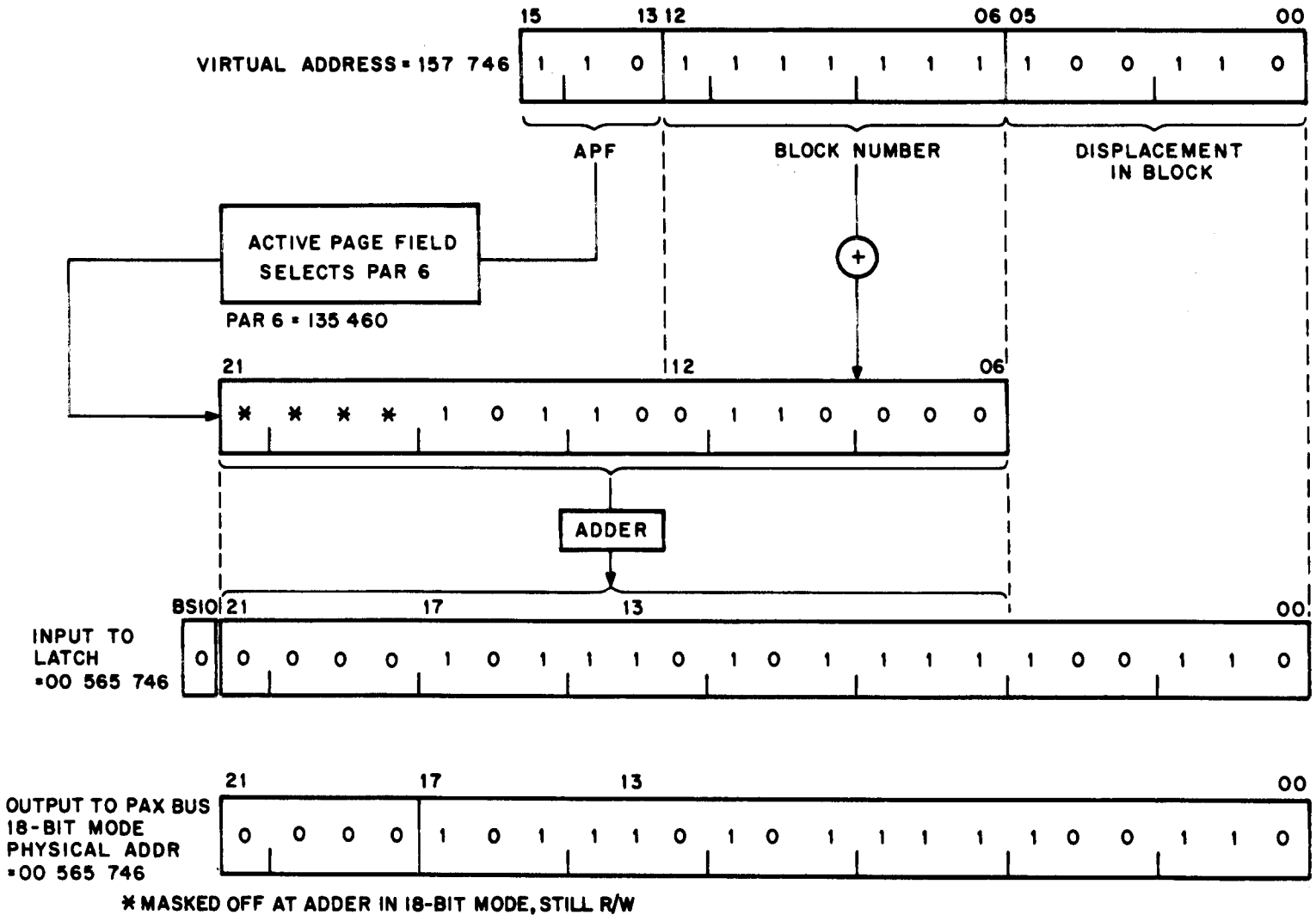


Figure 5-11 18-Bit Mapping: Memory Address

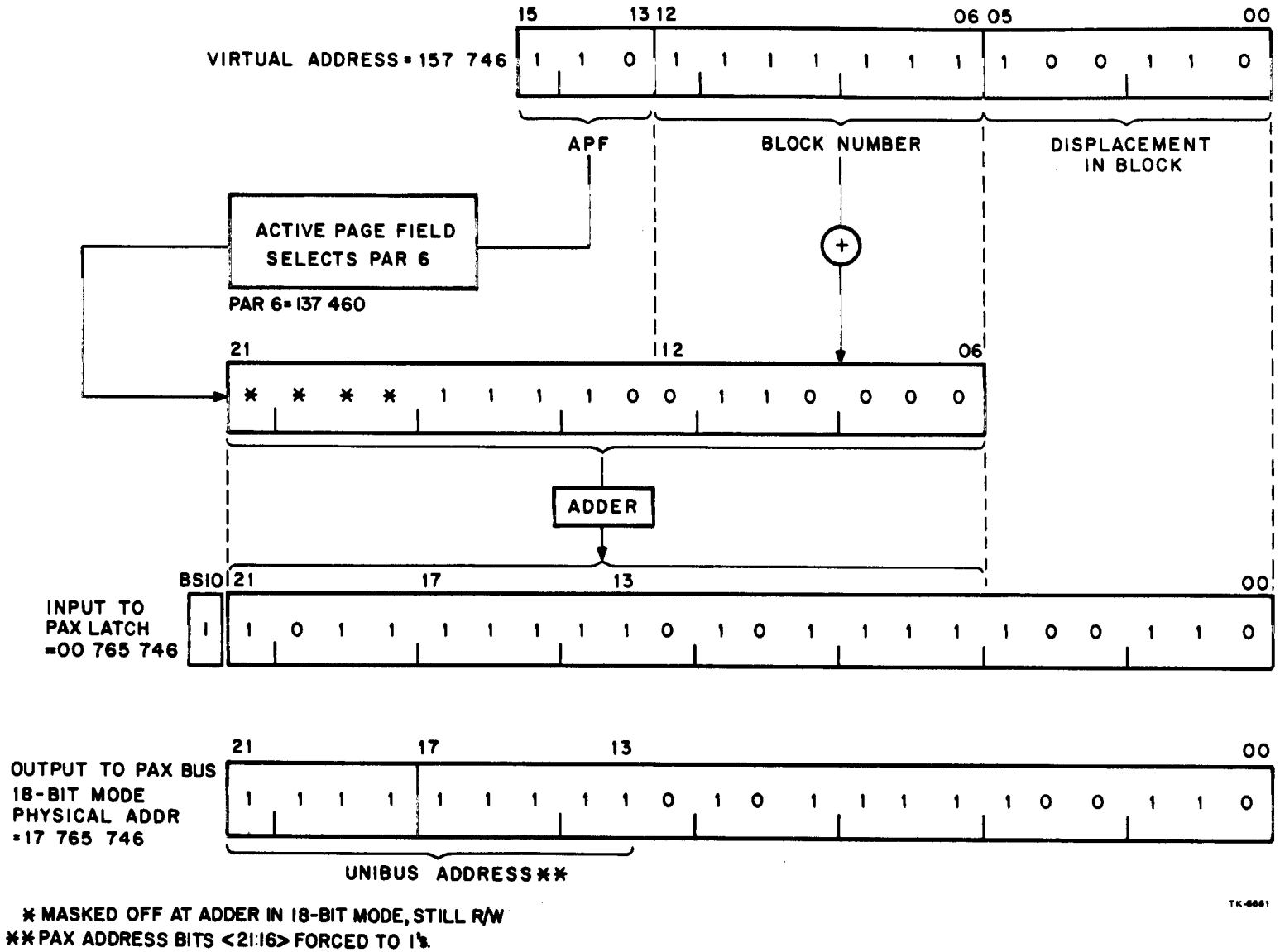


Figure 5-12 18-Bit Mapping: UNIBUS Address

5.2.2.2 22-Bit Mapping – Refer to Figure 5-4 and 5-13. In 22-bit mapping the VA is relocated in the same manner as 18-bit mapping, but the full 16-bits of the PAR are passed to the adder in the MMU. Thus, all addresses from 00 000 000_8 – 17 777 777_8 can be generated. In the 22-bit mode no special consideration of I/O page references is taken since the required physical address range (17 760 000_8 – 17 777 777_8) can be generated directly.

Twenty-two-bit mapping is enabled when bit 4 of SR3 is set (22-bit relocation) and bit 0 of SR0 is set (relocation enabled). As with 18-bit mapping bits ⟨05:00⟩ of the VA are passed through the MMU unaltered to become bits ⟨05:00⟩ of the PA. VA ⟨15:13⟩ are used to select a PAR. The contents of the PAR are added to bits ⟨12:06⟩ of the VA to create bits ⟨21:06⟩ of the PA.

5.2.2.3 Physical Addressing – After the 22-bit address is generated, steering logic in the CPU controls its routing (see Figure 5-14). Two modes of addressing are possible in the PDP-11/24:

1. Normal – 22-bit addressing mode E124 S6 closed
2. Special – 18-bit compatability mode E124 S6 open (22-bit addressing generally impossible)

In the normal mode the CPU generates a 22-bit physical address on the physical address extended (PAX) bus.

The processor will then make two decisions about the physical address:

1. Is it in the upper 128KW of address space?
2. Is it in the upper 4KW of address space?

Addresses below the upper 128KW are sent only to the memory bus.

Addresses in the upper 128KW of address space (Figure 5-10) are sent to the UNIBUS. When this happens PAX bits ⟨21:18⟩ are dropped because the largest address the UNIBUS can use is 18-bits. Addresses in the lower 124KW of UNIBUS space are not sent to main memory. Therefore, the maximum amount of memory that can be accessed is 2048KW – 128KW = 1920KW.

Addresses in the upper 4KW of memory are I/O page addresses. These addresses are sent to the UNIBUS and main memory. I/O devices are not normally located on the memory bus, but because the memory parity CSRs are located in this address range, they can be accessed by sending I/O page addresses on the memory bus.

In this special mode – 18-bit compatibility – the CPU forces all references in the physical address space to also be sent to the UNIBUS (Figure 5-15). This mode is used to simulate the operation of the PDP-11/34. (See Appendix D, Section D.2, for M7133-YA CPU switch locations.)

NOTE

Care should be taken when using the PDP-11/24 in the 18-bit compatibility mode. Dual addressing problems will occur if an address above 777 777_8 is generated. For example, address 01 777 651_8 will access UNIBUS address 777 651_8 as well as main memory location 01 777 651_8.

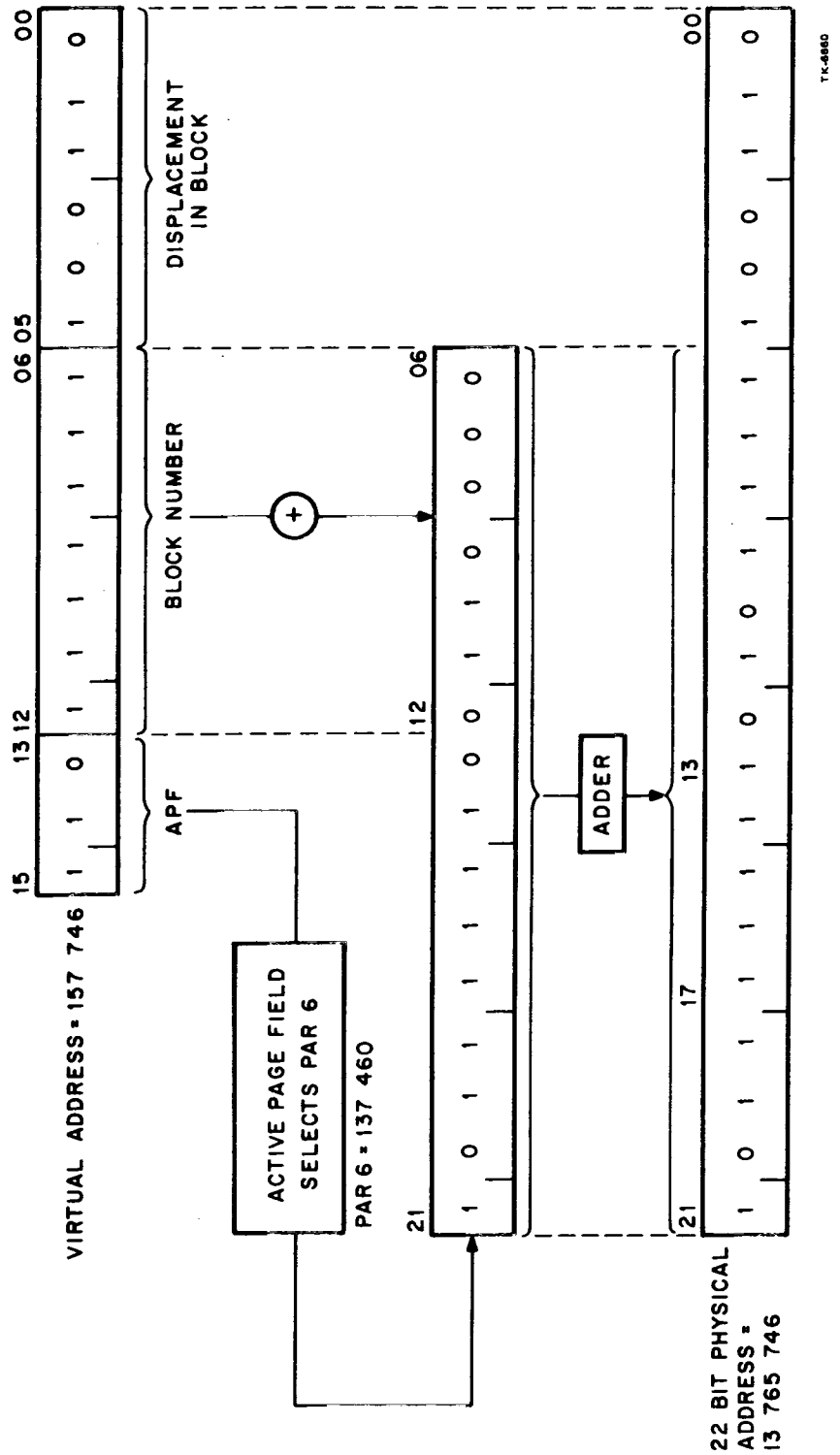


Figure 5-13 22-Bit Address Mapping

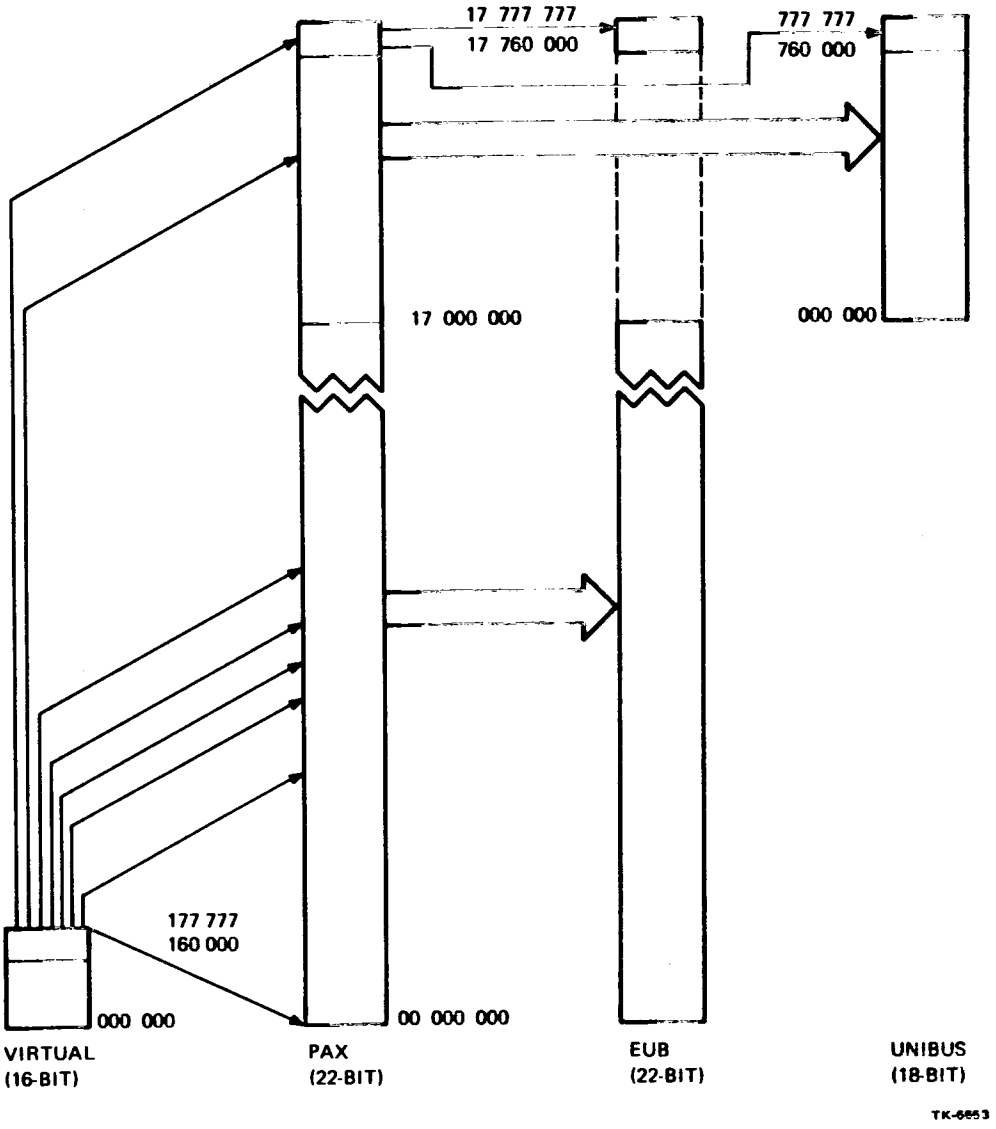


Figure 5-14 PDP-11 Address Space

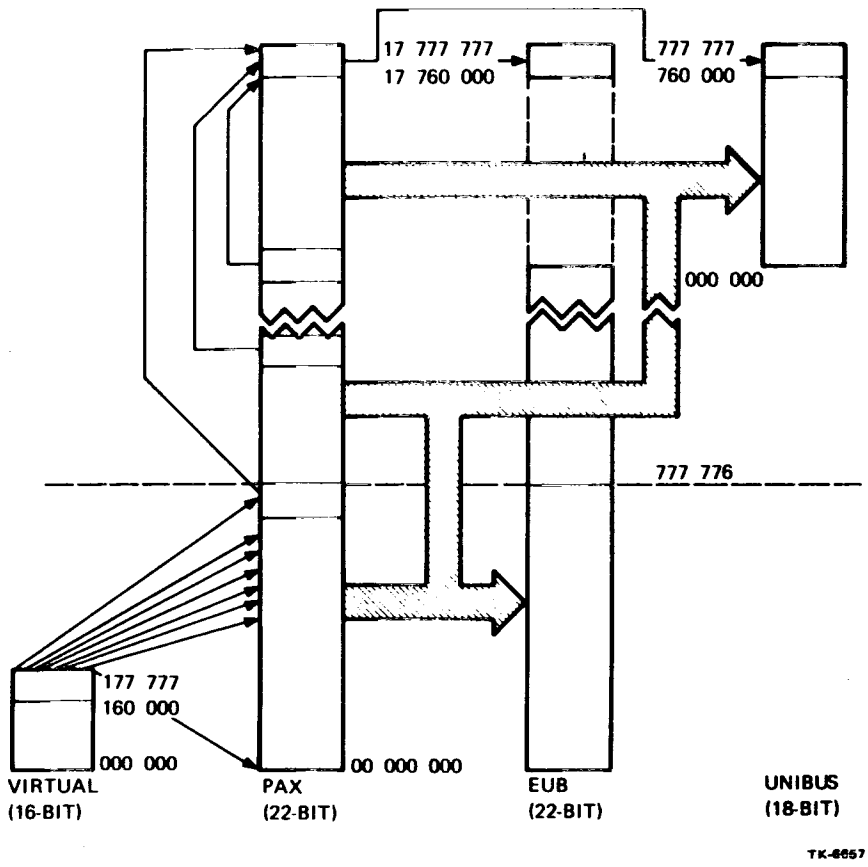


Figure 5-15 PDP-11 Address Space, CPU References with 18-Bit Switch OFF

5.3 MEMORY MANAGEMENT REGISTERS

The memory management chip provides two sets of active page registers (APRs). Each APR consists of a page address register (PAR) and a page descriptor register (PDR). These registers are always used as a pair and contain all the information required to locate and describe the currently active pages for each mode of operation (kernel and user). One PAR/PDR set is used in the kernel mode, the other in the User mode. The current mode bits or the previous mode bits of the processor status word determine which set will be referenced for each memory access. A program working in one mode can be prevented from using the register set of the other mode to access memory. A specific address in the I/O page is assigned to each PAR and PDR.

A relocation register reference to any APR with memory management enabled actually causes two words (32 bits) to be accessed at a time. This allows the page address register and the page descriptor register to be available together for relocation cycles. When directly reading or writing a PAR/PDR register, only the register that was specifically addressed by its UNIBUS address is connected to the data/address lines (DAL).

The PAR/PDR registers can be accessed in two ways:

1. When directly referenced for external data transfers by their UNIBUS addresses.
2. When referenced by a virtual address during a relocation cycle.

Each PAR/PDR register has a unique memory address in the I/O page (Table 5-1). PAR/PDR direct UNIBUS addressing is triggered at the beginning of a system bus operation if the physical address on the DAL during the address microinstruction cycle is the address of one of these registers and BSIO H (Bank Select I/O) is true, indicating an address in the I/O page. This physical address is derived from one of three possible sources:

1. The virtual address received from the data chip during the first half of the address microcycle.

2. A relocated virtual address that is generated internally (by the MMU chip) during the second half of the cycle.

3. An externally generated physical address (console ODT mode) that is received during the second half of the address microcycle.

Table 5-1 PAR/PDR Address Assignments

Kernel Active Page Registers			User Active Page Registers		
No.	**PAR**	**PDR**	**No.**	**PAR**	**PDR**
0	17772340	17772300	0	17777640	17777600
1	17772342	17772302	1	17777642	17777602
2	17772344	17772304	2	17777644	17777604
3	17772346	17772306	3	17777646	17777606
4	17772350	17772310	4	17777650	17777610
5	17772352	17772312	5	17777652	17777612
6	17772354	17772314	6	17777654	17777614
7	17772356	17772316	7	17777656	17777616

The physical address pointer is used to control register selection during all subsequent I/O transfers until the system bus operation ends. During this time no indirect addressing (relocation) can take place. When in the direct addressing mode, output microinstruction cycles cause the contents of the DAL to be written into the addressed word or byte of an MMU register. However, the W-bit (written into) of the PDR is only modified indirectly. It is cleared when either register of the APR set is written into and is set when the bank of virtual memory under control of the APR is written into. Direct UNIBUS references are the only way in which other PAR/PDR register contents can be altered.

Active page registers are addressed as PAR/PDR register pairs during address relocation cycles. The register pair is selected by bits (15:13) of the virtual address and the selected mode (user/kernel) sent by the data chip. PAR/PDR register pairs cannot be altered by indirect references (except for the W-bit).

Table 5-1 is a list of address assignments.

5.3.1 Page Address Register (PAR)

Figure 5-16 illustrates the page address register which contains the 16-bit base address of the page. This 16-bit displacement is added to bits ⟨12:06⟩ of the virtual address received from the data chip to create PAX⟨21:06⟩ of the relocated physical address (Figure 5-9). PAX⟨05:00⟩ are the same as bits ⟨05:00⟩ of the virtual address.

The page address register may be alternatively thought of as a relocation constant or as a base register containing a base address. Either interpretation indicates the basic function of the PAR in the relocation scheme.

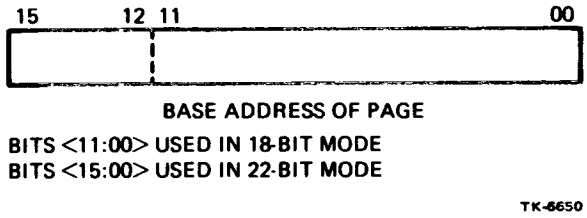


Figure 5-16 Page Address Register (PAR)

5.3.2 Page Descriptor Registers (PDR)

Figure 5-17 illustrates the page descriptor register which contains information relative to page expansion, page length and access control. Table 5-2 explains the function of the various bits in this register.

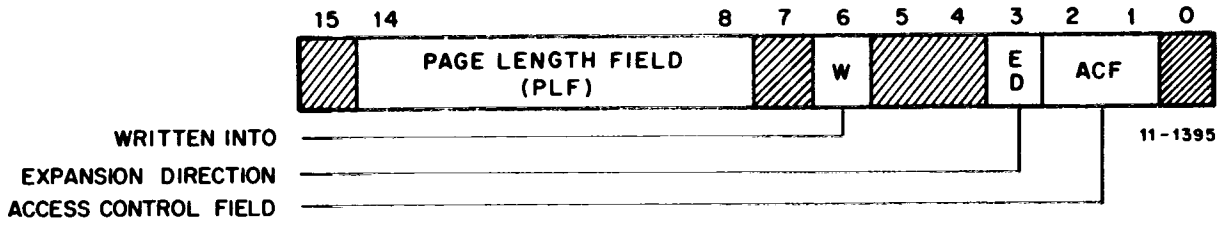


Figure 5-17 Page Descriptor Register (PDR)

Table 5-2 Page Descriptor Register (PDR)

Bit	Description
⟨14:08⟩ Page Length Field (PLF)	The seven-bit page length field specifies the number of addressable blocks (block = 32 words) in the accessed page. Bits ⟨12:06⟩ of the virtual address (desired block number) are compared with the PLF in the comparator to detect illegal references outside the program defined page boundary.
⟨06⟩ Written Into (W)	This bit is set if any location in the accessed page is written into after the APR set is initially accessed (unless the relocated destination is one of the internal memory management registers). Note that it can only be set during internally controlled relocation operations (CPU accesses), not NPRs (DMA). It is cleared when the PDR or PAR of that page is directly written.
⟨03⟩ Expansion Direction (ED)	This bit controls the expansion direction of the page boundary and is used in checking for page length violations. ED = 0: Upward expansion (most normal code) from block number 0 to include blocks with higher addresses. ED = 1: Downward expansion (stacks) from block number 177_8 to include blocks with lower address.
⟨02:01⟩ Access Control Field (ACF)	The two-bit access control field describes the access rights to the page. Any attempt to perform an operation not allowed by the ACF causes an MMU abort. The ACF codes are expanded below:

ACF	⟨02:01⟩	Function
0	0	Abort any access to this page.
0	1	Allow read accesses, abort any write access to this page.
1	0	Abort any access to this page.
1	1	Allow read or write.

Note: All unused PDR bits are read as zeros and cannot be written into.

5.4 MEMORY MANAGEMENT STATUS REGISTERS

The memory management chip contains the four status registers associated with the memory management operations. The PDP-11/24, however, does not use all four registers provided. The status registers can be referenced by their UNIBUS addresses (Table 5-3). Status Register 0 (SR0) contains information regarding memory management abort conditions as well as the enable relocation bit. A memory management abort condition is caused during internal address relocation by one of the following:

1. An attempt is made to access a nonresident page. This is detected by examining the PDR access control field.

2. A write protection violation has occurred. This is detected by examining the PDR access control field during write operations.

3. A page length error is detected (an access outside the allowable boundary of the page).

When an abort condition occurs the ABORT L output of the MMU chip is asserted during the second half of the cycle in which the relocated address is generated. The assertion of ABORT L resets the control chips (forcing the service microstate) and generates the MMU ERR service signal, which indicates an error in the memory management cycle has occurred. The asserton of MMU ERR will cause the processor to execute a trap through the vector at 250_8, thus notifying the user's program of the MMU abort.

The abort line is blocked when DMMUS L (disable MMU slave) is asserted. DMMUS L disables the MMU's ability to drive DAL⟨21:00⟩ and the BSIO signal as well. It informs the memory management enable logic that the ensuing relocation operation is being externally controlled and is used to block the MMU from using the chip bus.

Table 5-3 Memory Management Status Register UNIBUS I/O Addresses

Register	Address	Contents
17 777 572	Status Register 0 (SR0)	Abort information, status information, relocation enable.
17 777 574	Status Register 1 (SR1)	All 0's (not used in the PDP-11/24).
17 777 576	Status Register 2 (SR2)	Last PC fetched.
17 772 516	Status Register 3 (SR3)	22-/18-bit mapping mode selection, UNIBUS map enable.

5.4.1 Status Register 0 (SR0)

Status register 0 contains the relocation enable control bit as well as information regarding any memory management abort condition. Figure 5-18 illustrates the bit assignments of SR0 and Table 5-4 provides a description of these bits. SR0 bits ⟨06:05,03:01⟩ are updated during every relocation cycle. When a fault occurs SR0 bits ⟨15:13⟩ are set (depending on the error), and the mode, page, and all bits in status register 2 (SR2) are frozen in order to capture the state of the aborted relocation attempt. Subsequent relocation attempts or aborts will not alter the information present at the first fault. SR0 remains frozen until SR0 bits ⟨15:13⟩ are cleared by direct UNIBUS access or an initialization signal.

Direct setting of any of the abort flags will also cause the SR0 bits to be frozen but will not activate the abort output. SR0 is cleared by control signals from the data chip.

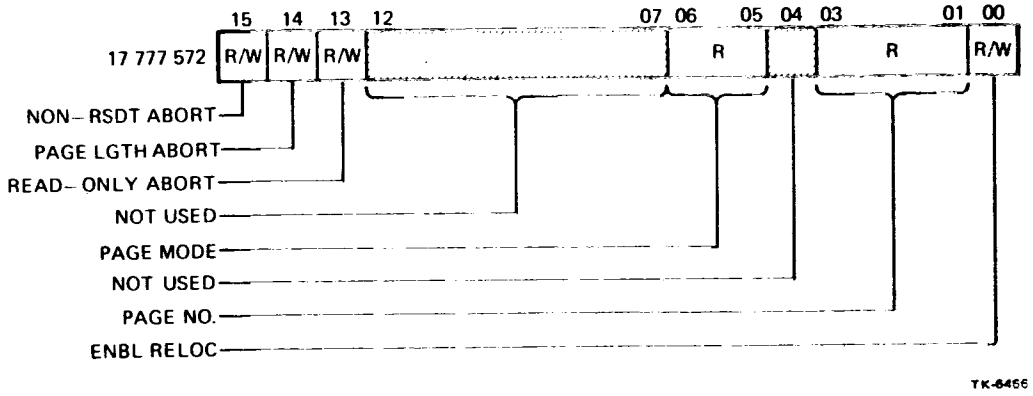


Figure 5-18 Memory Management Status Register 0 (17 777 572)

Table 5-4 Memory Management Status Register 0 (SR0) Bit Description

Bit	Description
⟨15⟩ Non-Resident Abort (ANR)	This bit is set when an attempt is made to access a page with an access control field in the PDR of 00 or 10. The ANR bit can be read or written.
⟨14⟩ Page Length Abort (APR)	This bit is set when the page length boundary is exceeded. The APR bit can be read or written.
⟨13⟩ Read-Only Abort (ARO)	This bit is set when an attempt is made to write a page with an access control field in the PDR of 01. The ARO bit can be read or written.
⟨12:07⟩	These bits are read as 0 and cannot be written into.
⟨06:05⟩ Mode	These bits are set to 11 if a relocation takes place while in user mode and set to 00 if a relocation takes place while in kernel mode. The mode bits are read only.
⟨04⟩	This bit is read as 0 and cannot be written into.
⟨03:01⟩ Page Number	These bits contain the page number of the virtual address. The page number bits are read only.
⟨00⟩ Enable	This bit enables the memory management function when set. The enable bit can be read or written.

5.4.2 Status Register 1 (SR1)
This register is not used in the PDP-11/24. When addressed by its UNIBUS address, all the bits in this register are read as 0 and cannot be written into. This register is included to avoid system bus errors if the SR1 is referenced.

5.4.3 Status Register 2 (SR2)
SR2 is loaded during a relocation cycle with the 16-bit virtual PC received from the data chip. This occurs on all address cycles when the control information indicates that the current address on the bus is that of an instruction. The contents of SR2 are frozen when an abort flag is set. SR2 is read only.

5.4.4 Status Register 3 (SR3)
The PDP-11/24 processor only uses two bits of this register, bits <05:04>. When cleared, bit <05> disables the optional UNIBUS mapping logic; when set, bit <05> enables the UNIBUS mapping logic. When cleared, bit <04> selects the 18-bit mapping mode within the MMU; when set, bit <04> selects the 22-bit mapping mode in the MMU. Bit <00> of SR0 then determines whether or not to enable the selected relocation mode. If relocation is not enabled, 16-bit mapping results. This bit is used internally by the MMU. All other bits in the register are read as 0s. SR3 is cleared by the same control signals that clear SR0.

5.5 MICROINSTRUCTION REGISTER (MIR)
At the beginning of every cycle, MIB ⟨07:04⟩ are loaded into the microinstruction register. During address cycles the microinstruction register is used to control the relocation operation. The relocation function is temporarily disabled, regardless of the state of the relocation enable bit (SR0⟨00⟩), when bit 7 of the microinstruction is set. This will also cause an initialization condition clearing status registers SR0 and SR3 if issued during an output status microinstruction. When bit 5 of the microinstruction is set, status register SR2 is loaded with the incoming instruction address.

5.6 MEMORY MANAGEMENT ENABLE (MME) LOGIC
The MME logic generates the MME L control output (MIB ⟨15⟩) during internally controlled relocation cycles. It also receives and interprets the disable MMU slave (DMMUS L) control input to detect externally controlled relocation cycles.

MME L is generated during the first half of the address microinstruction cycle (while other control information is on the MIB) if a relocation operation is to take place. MME L asserted causes external timing to delay MCLK until the relocation operation is complete and informs the data chip during address cycles that it must update its PSW explicit addressing logic with the new address information. MME L can be driven by external logic. If DMMUS L is not asserted and memory management is not internally enabled, the virtual address from the data chip is not relocated. When memory management is internally enabled, then the virtual address is relocated for internal and external use and MME L is asserted. Note that memory management cycles can be disabled by either bit ⟨00⟩ of status register 0 (SR0) or under microprogram control by MIR ⟨07⟩. Assertion of DMMUS L during an address cycle informs the MME logic that the following relocation operation is externally controlled. Any internally generated abort conditions are ignored and the DAL receive the new address for explicit address decoding.

BSIO H (bank select I/O) will be asserted when an internally relocated address accesses the I/O page of memory address space. The BSIO output is enabled during the second half of the relocation cycle unless an abort condition exists or the DMMUS L input is asserted.

5.7 MEMORY MANAGEMENT TIMING
A single clock MCLK controls all LSI chip functions including the MMU chip. An explanation of the MMU timing is contained in Paragraph 4.2.1.2 of this manual.

5.8 MEMORY MANAGEMENT CONTROL SIGNALS

Three groups of control inputs are used by the MMU: address I/O control (AIO ⟨02:00⟩), sync flip-flop (SYNCF), and X-control (X ⟨02:00⟩). The three-bit address I/O control inputs (MIB ⟨12,09:08⟩) indicate the operations if any for the present microinstruction cycle. Paragraph 4.3.2 provides a description of the AIO control bits. The MMU chip uses the address cycle information to initiate relocation operations and the data cycle information to control accesses to its internal registers.

The sync flip-flop (SYNCF) control input (MIB ⟨07⟩) signifies that a system bus cycle is in progress. This signal is set by the data chip when the bus cycle starts and is kept active for the duration of the cycle. It is used to maintain the state of the internal sync flip-flop and to control the PAR/PDR/SR0-3 UNIBUS addressing logic.

The three-bit X-control inputs (MIB ⟨06:04⟩) encode control information from the data chip. Paragraph 4.3.2 provides an explanation of these three bits.

CHAPTER 6
KT24 UNIBUS MAP OPTION

6.1 INTRODUCTION

The KT24 UNIBUS map option is a hex-height, multilayer module that contains the logic necessary to implement a UNIBUS map, M9312 compatible bootstrap and a voltage monitor for the PDP-11/24. The KT24 UNIBUS map option is installed in the second slot of the PDP-11/24 processor backplane. Figure 6-1 shows a functional block diagram of the KT24 UNIBUS map option. The following paragraphs discuss the operation of the KT24 UNIBUS map option.

6.2 UNIBUS MAP

The UNIBUS map is the interface between the UNIBUS and main memory. It responds as a slave device to UNIBUS signals and is used to convert 18-bit UNIBUS addresses to 22-bit memory addresses.

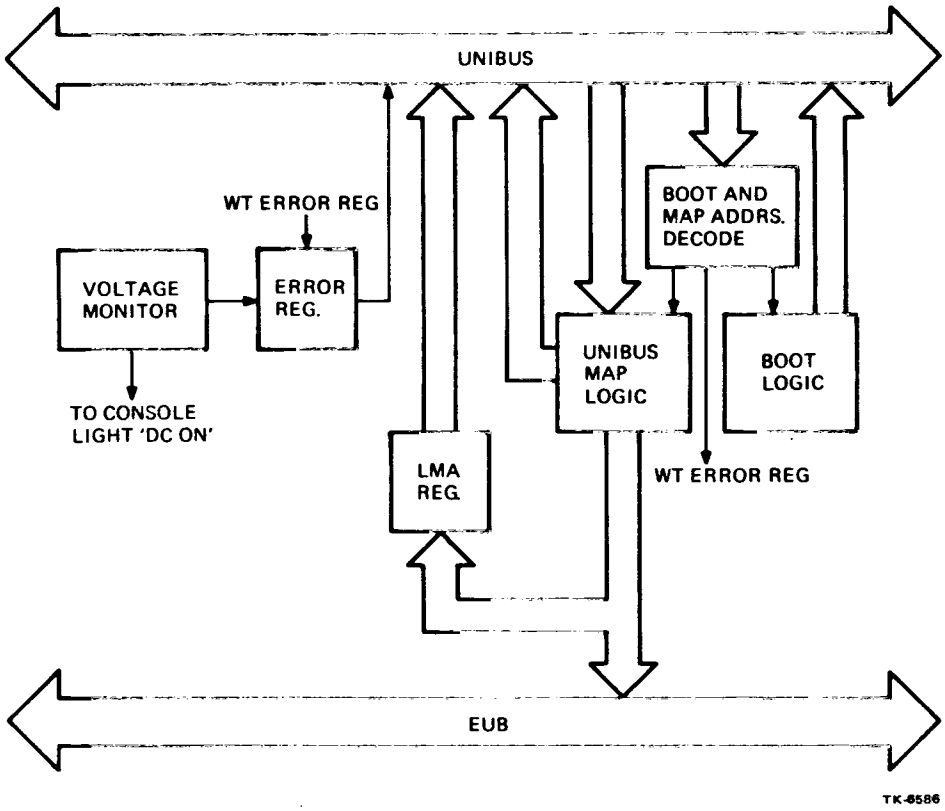


Figure 6-1 Map Module Block Diagram

UNIBUS address space is 128KW of which the top 4KW addresses are reserved for the CPU and I/O registers. This upper 4KW of UNIBUS address space is called the I/O page (Figure 6-2). The lower 124KW of UNIBUS address space can be used by the UNIBUS map to reference physical memory.

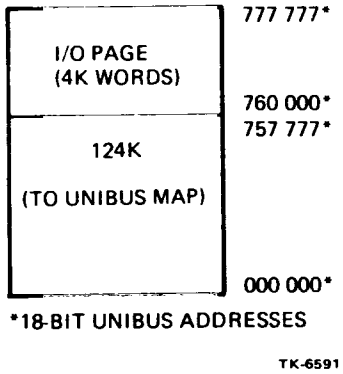


Figure 6-2 UNIBUS Address Space

The UNIBUS map consists of 32 mapping registers (K4-6), a 21-bit adder (K4-8), and associated logic. The mapping registers are 21-bits wide and can be accessed in two different ways:

1. Direct Access – The mapping registers are accessed via their UNIBUS address. Because the mapping registers are 21-bits wide, two UNIBUS transactions are required for each read or write of the registers.

2. Indirect Access – When the UNIBUS map is enabled, the upper five bits of the UNIBUS address are used to select the appropriate register to be used in relocating the 18-bit UNIBUS address.

The 21-bit wide UNIBUS map registers are allotted 64 addresses in the I/O page (Table 6-1). It should be noted that the last mapping register (addresses $17\ 770\ 374_8$ and $17\ 770\ 376_8$) can be read or written, but cannot be used to map UNIBUS addresses because it would be used by addresses in the range of the I/O page ($17\ 760\ 000_8$ – $17\ 777\ 777_8$).

The UNIBUS map does relocation by adding one of 31 UNIBUS map registers, which contain a relocation constant, to bits ⟨12:01⟩ of the UNIBUS address to create a 22-bit physical address (PA). The 22-bit PA is used to reference physical memory. When the UNIBUS map is disabled, its operation is transparent to the user and the incoming UNIBUS address is used to reference the first 124KW of physical memory and the I/O page.

Figure 6-3 shows a block diagram of the UNIBUS map and its associated control logic.

Table 6-1 Access to UNIBUS Map Registers

Register	Physical Address for Direct Read or Write Lo	Hi	UNIBUS Address Mapped via Register
0	17 770 200	17 770 202	000 000 – 017 777
1	17 770 204	17 770 206	020 000 – 037 777
2	17 770 210	17 770 212	040 000 – 057 777
3	17 770 214	17 770 216	060 000 – 077 777
4	17 770 220	17 770 222	100 000 – 117 777
5	17 770 224	17 770 226	120 000 – 137 777
6	17 770 230	17 770 232	140 000 – 157 777
7	17 770 234	17 770 236	160 000 – 177 777
10	17 770 240	17 770 242	200 000 – 217 777
11	17 770 244	17 770 246	220 000 – 237 777
12	17 770 250	17 770 252	240 000 – 257 777
13	17 770 254	17 770 256	260 000 – 277 777
14	17 770 260	17 770 262	300 000 – 317 777
15	17 770 264	17 770 266	320 000 – 337 777
16	17 770 270	17 770 272	340 000 – 357 777
17	17 770 274	17 770 276	360 000 – 377 777
20	17 770 300	17 770 302	400 000 – 417 777
21	17 770 304	17 770 306	420 000 – 437 777
22	17 770 310	17 770 312	440 000 – 457 777
23	17 770 314	17 770 316	460 000 – 477 777
24	17 770 320	17 770 322	500 000 – 517 777
25	17 770 324	17 770 326	520 000 – 537 777
26	17 770 330	17 770 332	540 000 – 557 777
27	17 770 334	17 770 336	560 000 – 577 777
30	17 770 340	17 770 342	600 000 – 617 777
31	17 770 344	17 770 346	620 000 – 637 777
32	17 770 350	17 770 352	640 000 – 657 777
33	17 770 354	17 770 356	660 000 – 677 777
34	17 770 360	17 770 362	700 000 – 717 777
35	17 770 364	17 770 366	720 000 – 737 777
36	17 770 370	17 770 372	740 000 – 757 777
37*	17 770 374	17 770 376	

*Can be read or written into, but is not used for mapping.

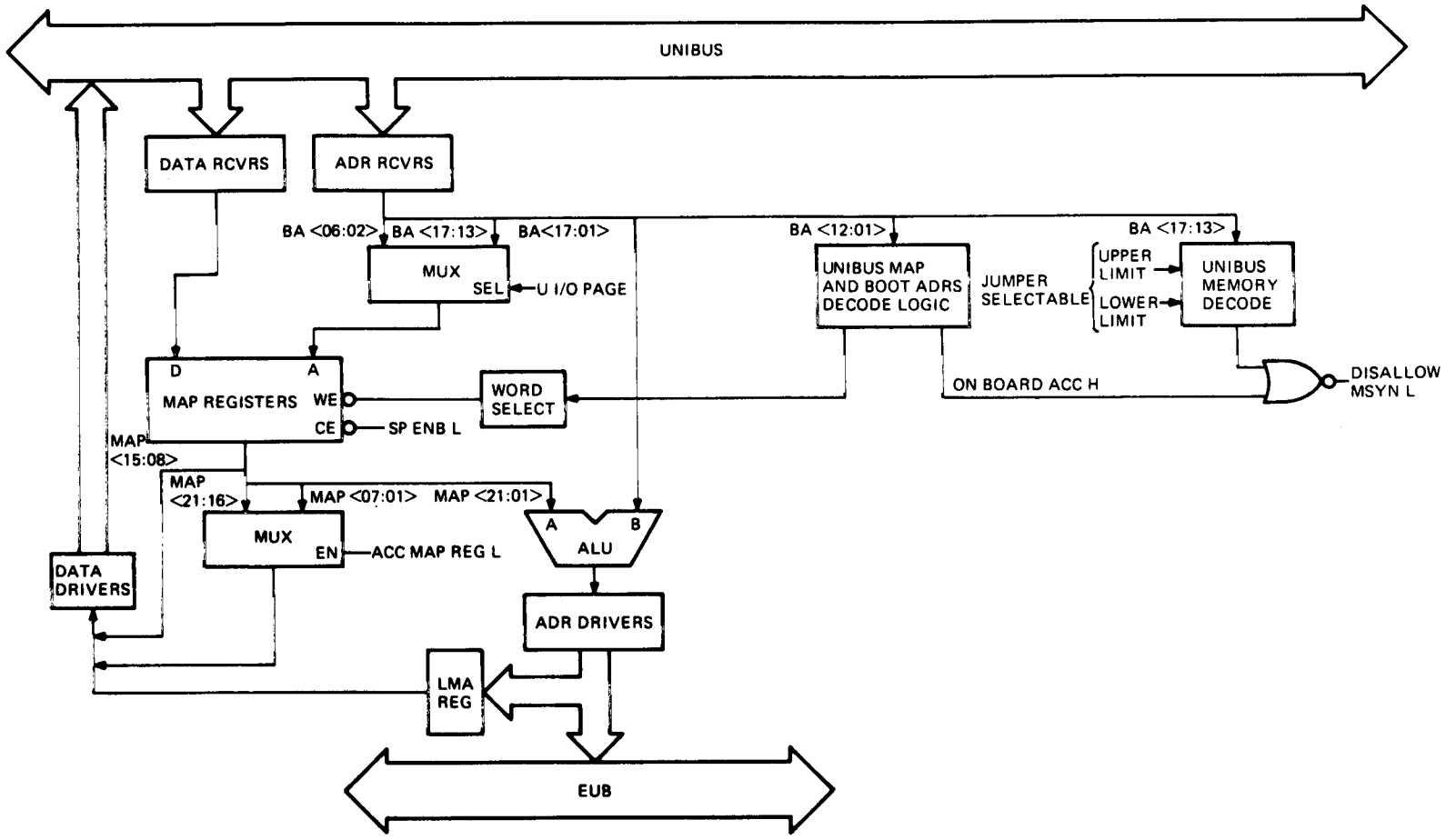


Figure 6-3 UNIBUS Map Block Diagram

6.2.1 MSYN Steering

The MSYN steering logic located on K1-3 is used to control the generation of MEM MSYN in the PDP-11/24 when the KT24 UNIBUS map is present in the system. The MSYN steering logic controls MEM MSYN in the following ways.

1. When the UNIBUS map is disabled, the MSYN steering logic performs the following functions:

   Determines if the CPU or a DMA device is performing the transfer.

   Allows the assertion of MEM MSYN.

2. When the UNIBUS map is enabled, the MSYN steering logic performs the following functions:

   - Determines if the CPU or a DMA device is performing the transfer.
   - Delays the assertion of MEM MSYN to allow the UNIBUS map to generate the physical address.
   - Enables the relocated address onto the EUB.
   - Strobes the relocated address into the last mapped address (LMA) register.

3. The MSYN steering logic also disallows the assertion of MEM MSYN when one of the KT24 UNIBUS map option I/O page registers is accessed.

6.2.2 Map Control

The map control logic (K1-4) is used to control the reading and writing of the 32 map registers and the last mapped address (LMA) register. The control logic consists of the UNIBUS map and boot address decode PLA (E92), a delay line (E31), and associated read/write and buffer control logic. The address decode PLA (E92) decodes the UNIBUS address to determine if a read or write of a map register or the LMA register is to be performed. If a read or write of one of these registers is intended, the PLA will assert ON BOARD ACC H and, depending on the register to be accessed, ACC HI LMA L, ACC LO LMA L, or ACC MAP REG L. ON BOARD ACC H is used to accomplish the following during a read/write of the map and LMA registers.

1. Assert DISALLOW MEM MSYN. This keeps MSYN, intended for the map or LMA registers, from being sent to memory.

2. ON BOARD ACC H is ANDed with BUS MSYN H to generate VALID ADD H which loads counter E81 with a three (3) and causes the input of delay line E31 to be asserted. The delay line is used to negate ENB WRITES H during a write operation, assert ENB READ DAT L during a read operation, and assert UBI SSYN H.

Figure 6-4 shows the relationship of these signals to VALID ADD H.

Figure 6-5 shows the logic used to select whether the high word or low word of the selected register is to be read or written.

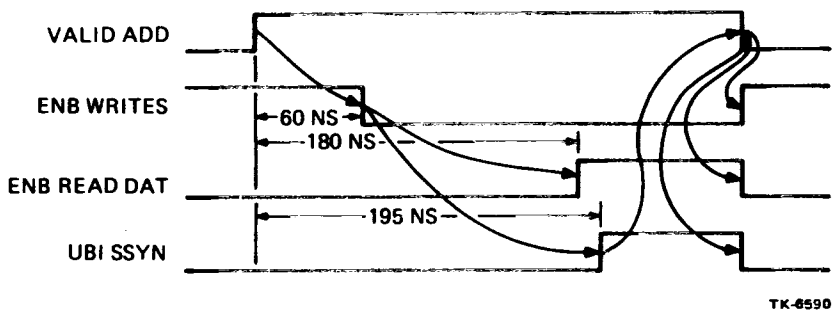


Figure 6-4 Map Control Timing

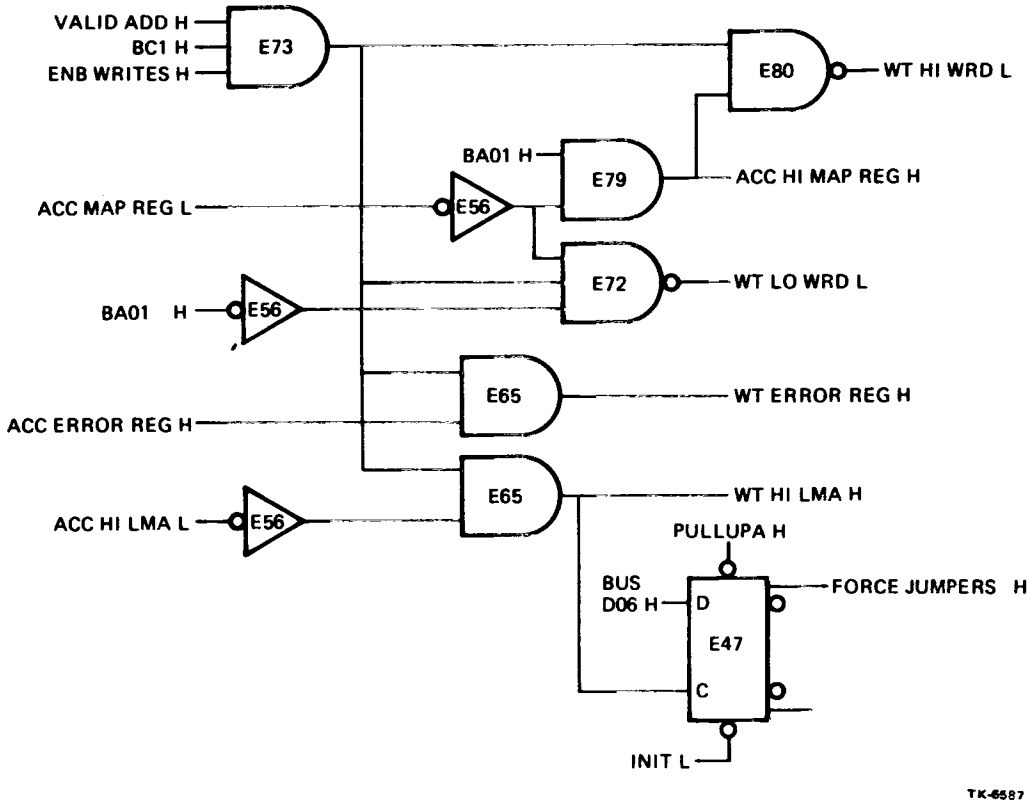


Figure 6-5 Word Control

6.2.3 Map Addressing and Relocation

Relocation expands the 18-bit UNIBUS address to the 22-bit main memory address. This allows the UNIBUS to access any location in main memory. This relocation or mapping of addresses is done by adding the contents of one of the mapping registers to bits ⟨12:01⟩ of the incoming UNIBUS address.

All mapping registers in the UNIBUS map are 21 bits wide. A 22nd bit, which is not writable and is always read as zero, acts as the lowest-order bit for each register. Each register specifies the 22-bit base PA of a 4K page residing on any word boundary in memory. The reason for using word boundaries is that the mapping logic does not recognize if a byte operation is being executed.

Figure 6-6 illustrates the construction of a PA by the UNIBUS map. Bits ⟨17:13⟩ of the 18-bit UNIBUS address select a map register. The remaining bits ⟨12:00⟩ of the UNIBUS address are used as an offset into the page to which the mapping register is pointing.

When an address is taken from the UNIBUS, the mapping register is automatically selected and the contents read out. Then the 22-bit base address contained in the selected map register is added to the 12-bit offset in the UNIBUS address to form the PA. This mapping function is very similar to that performed by memory management.

The program controls this process both by selecting the contents of the mapping registers and by its ability to enable and disable the UNIBUS map relocation function. Since the 32 UNIBUS map registers may be accessed directly from the UNIBUS, a sophisticated DMA device could also manipulate them. Thus, virtually unlimited-length transfers may be performed.

The UNIBUS map is enabled by the assertion of UBMAP L (bit ⟨05⟩ of SR3). The UNIBUS address lines are received and buffered by logic on K1-5. When a mapping operation is being done, a multiplexer (E69,E62) selects bus address bits BA ⟨17:13⟩, to generate the map register address RAM A ⟨4:0⟩. The output of the selected map register, MAP ⟨21:01⟩, is then sent to the adder, which is made up of ALUs E36, E35, E34, E42, E44 and carry generators E43 and E45, where it is added to BA ⟨12:01⟩ to create MA ⟨21:01⟩. MA ⟨21:01⟩, along with BA⟨00⟩ and control bits BC1 and BC0, are used to create the PA and control information. The relocated physical address is asserted on the memory bus (EUB) via buffers E16, E25 and E33 and strobed into the LMA register.

6.2.4 Addressing Limits

There are 31 mapping registers which can be accessed indirectly by the UNIBUS for relocation. The actual number in use is determined by two sets of five jumpers (K1-4) which are used to set the upper limit, W3–W7, and the lower limit, TP1–TP10 (wire wrap posts).

Two regions of UNIBUS addresses may be mapped to main memory. These regions are referred to as the lower and upper window. The lower window expands upward from 0K; the upper window expands downward from 124K. (Refer to Figure 6-7.)

The lower limit jumpers are used to select the first address that will not be mapped to main memory. The lower limit allows the UNIBUS space that will be mapped (via the lower window) to main memory to be expanded upward from UNIBUS address $000\ 000_8$ up to $760\ 000_8$ in 4K word segments. If the lower limit jumpers are set to $000\ 000_8$, the lower window does not exist.

The upper limit jumpers are used to select the first address that will be mapped to main memory via the upper window. The upper limit allows the UNIBUS space that will be mapped to main memory to be expanded downward from UNIBUS address $760\ 000_8$ in 4K word segments. If the upper window jumpers are set to $760\ 000_8$, the upper window does not exist.

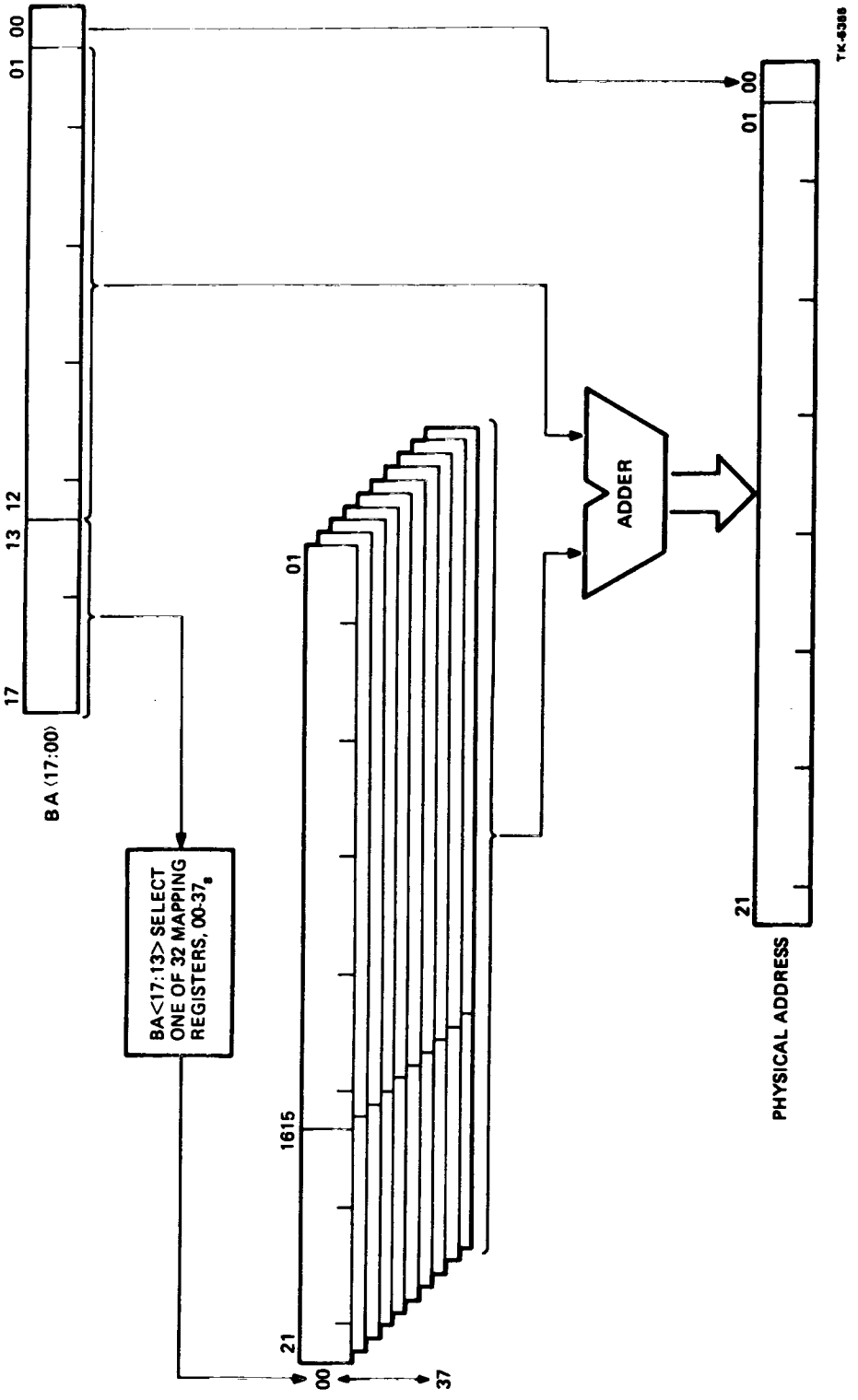


Figure 6-6 Construction of the PA

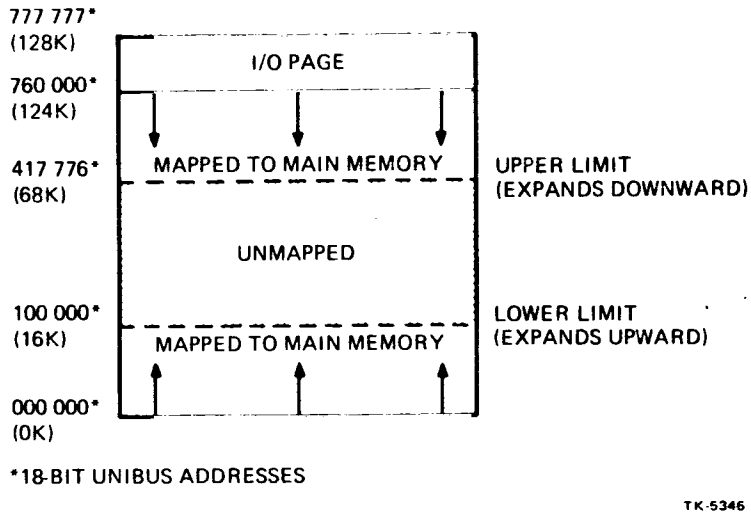


Figure 6-7 UNIBUS Map Addressing Limits

Figure 6-7 shows how the addressing limits affect UNIBUS space. When the upper and lower limits are set to address 000 000$_8$ (0K) or 760 000$_8$ (124K) or if the limits overlap (e.g., lower limit = 600 000$_8$ (96K), upper limit = 400 000$_8$ (64K)), all of the UNIBUS space, with the exception of the I/O page, will be passed to main memory.

The UNIBUS I/O page is always mapped to the main memory I/O page, regardless of the contents of the 32nd map register or the state of the UNIBUS map.

The signal K1-4 FORCE JUMPERS H, when asserted, causes a default of the addressing limits to the all jumpers out condition. This signal is used for diagnostic and maintenance purposes to force the UNIBUS map addressing limits to a known condition. The force jumpers bit is set under program control. For proper operation of this bit, any conflicting UNIBUS devices or memory may need to be removed.

6.3 BOOT LOGIC

The KT24 UNIBUS map option contains boot logic that is compatible with the M9312 bootstrap. The bulk of this logic is located on K1-4, K1-9 and K1-10 of the print set. This logic consists of four interchangable device boot ROMs, the PDP-11/24 diagnostic ROM, and the necessary logic, switches, and jumpers to implement the bootstrap programs. Paragraph 2.8.3 of this manual explains the function of the switches and jumpers. The following paragraphs explain the operation of the boot logic.

A boot is enabled by one of the following actions:

1. By system powerup.
2. By moving the front panel switch to the BOOT position and then back to the CONT position.
3. By loading the starting address of the boot program and starting the processor.

When a boot on system powerup or via the front panel switch is performed, the boot starting address is determined by S1 and S3–S10 of switch pack E58.

The boot logic consists of an address decode PLA, a decoder, a delay line, a counter, a shifter, a buffer, and associated logic. This circuitry is used to decode a boot address, assemble a 16-bit word from a four-bit ROM, and assert it on the UNIBUS.

The address decode PLA (E92) decodes the boot addresses and enables one of the four device boot ROMs or the CPU diagnostic ROM via decoder (E88). With S2 of E58 closed (boot logic enabled), K1-4 BOOT READ L is asserted by the address decode PLA. K1-4 BOOT READ L sets the counter (E81) to 0 and is ANDed with K1-3 BUS MSYN H to assert the input of an oscillator made up of a delay line (E31) along with AND gates E57 and E67. T90 H toggles the input to the delay line generating the clock pulses needed to increment the counter (E81) and clock the shifter. As the counter is incremented, two address bits, LSB H and MSB H, are generated. These address bits are used to address four contiguous four-bit words in the enabled ROM. These four words are assembled into one 16-bit word in the shifter (E41 and E50). After the fourth clock pulse to the counter (INC H), the oscillator is stopped and the 16-bit word generated from the ROM is buffered and then asserted on the UNIBUS bus by the assertion ENB READ DAT L and UBI SSYN H is asserted.

Figure 6-8 shows the timing associated with the generation of a 16-bit word from one of the five ROMs.

When the switches (E58 S1, S3–S10) are to supply the boot starting address, ACC SWITCHES L and BOOT 1 L are asserted by the address decode logic. UNIBUS address 773 024_8 is the address of switch pack E58. This action causes the switches (E58 S3–S10) to replace BUF ROM ⟨16:09⟩ H, and BOOT ROM 1 supplies BUF ROM ⟨08:01⟩ H to the buffer. This action allows the CPU to read the switches and thereby select the boot function to be performed. In addition, switch S1 will select 165 XXX_8 or 173 XXX_8, through the control of XOR gates E49. Device boot ROM 1 must be installed for the switch pack to function properly.

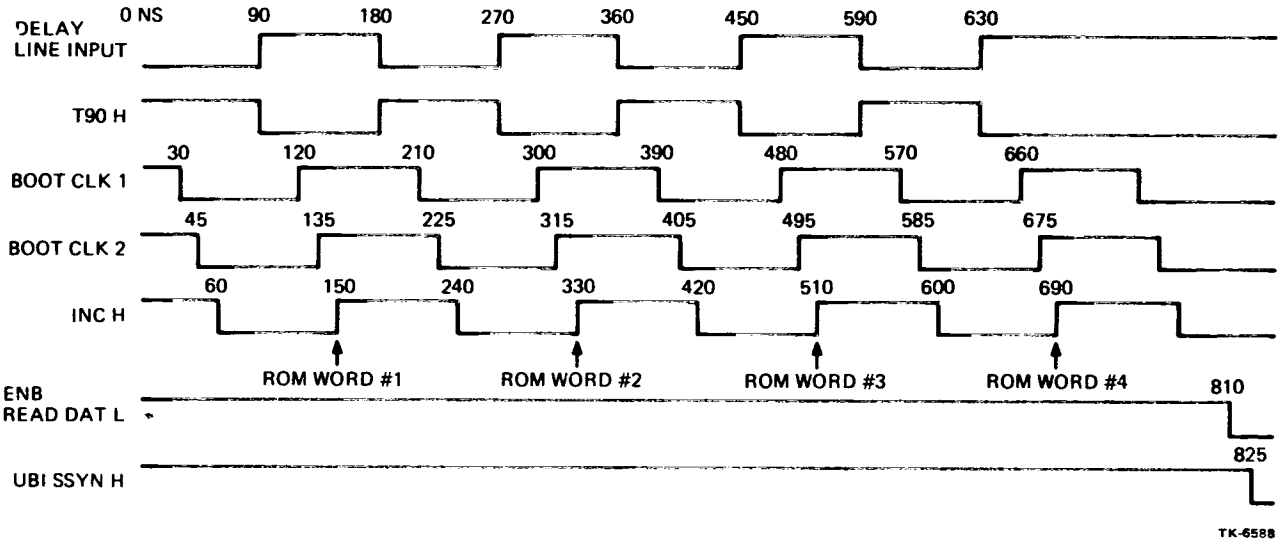


Figure 6-8 Boot Timing

6.4 VOLTAGE MONITOR

The voltage monitor on the KT24 UNIBUS map option monitors the voltages supplied to the PDP-11/24 and memory. If one of the supply voltages is out of tolerance, an error condition exists and the power-fail bit in the CPU error register is set. Additionally, the DC ON light located on the PDP-11/24 console will flash at a rate of 5 Hz as long as the out-of-tolerance condition exists. Table 6-2 lists the supply voltages and their limits.

Table 6-2 Voltage Limits

Supply Voltage	Limits Lo	Hi
+5.1 V	+4.64 V	+5.56 V
+5 B	+4.54 V	+5.66 V
+15 V	+13.35 V	+16.65 V
−15 V	−13.35 V	−16.65 V
+15 B*	+13.35 V	+16.65 V
+12 B*	+10.68 V	+12.84 V
−15 B*	−13.35 V	−16.65 V
−12 B*	−10.68 V	−12.84 V

*Selected by switch S3 of E6.
ON = ± 15 B
OFF = ± 12 B

The voltage monitor (K1-2) uses a comparator network (E15, E23 and E7) to compare the various voltages used in the PDP-11/24 to a fixed reference voltage. If any one of these voltages goes out of tolerance, the output of its respective comparator will go low. When the output of any comparator goes low, the power-fail bit (E48) of CPU error register is set, and a 5 Hz oscillator comprised of a timer (E24) and its associated circuitry is enabled causing the DC ON light to flash. When the power-fail bit of the CPU register has been set, it is cleared by an INIT or by writing to the CPU error register. Switches S1, S2 and S4 of E6 are used to isolate the different voltage comparators for maintenance purposes.

6.5 REGISTERS

The KT24 UNIBUS option contains the UNIBUS map registers, the last mapped address (LMA) register, and the CPU error register. The following paragraphs describe the registers.

6.5.1 UNIBUS Map Registers

There are 32 mapping registers, each containing a 21-bit mapping offset. Each mapping register is contained in two 16-bit registers; one register contains bits ⟨15:01⟩ (low word), and the other register contains bits ⟨21:16⟩ (high word). Table 6-1 lists the register address. Figure 6-9 shows the bit configuration of the registers.

6.5.2 Last Mapped Address (LMA) Register

The LMA is a two-word register located at addresses $17\ 777\ 734_8$ (low word) and $17\ 777\ 736_8$ (high word) that contains the 22-bit memory (EUB) address of the last mapped address. This register is used for maintenance purposes and also contains the memory control lines, C1 and C0, and the force jumpers bit. Table 6-3 describes the LMA bits. Figure 6-10 shows the LMA configuration.

6.5.3 CPU Error Register

The CPU error register is located at address $17\ 777\ 766_8$. This register is read-only; any attempt to write to it will clear the power-fail bit. Table 6-4 describes the CPU error register bits and Figure 6-11 shows the register configuration.

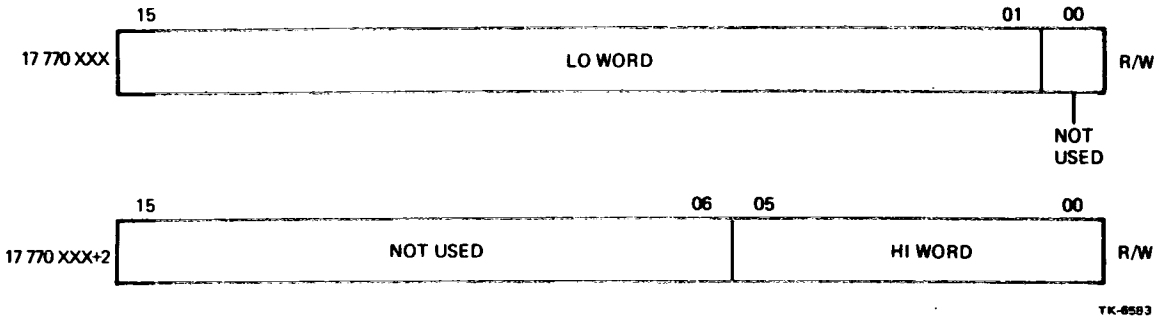


Figure 6-9 Map Registers

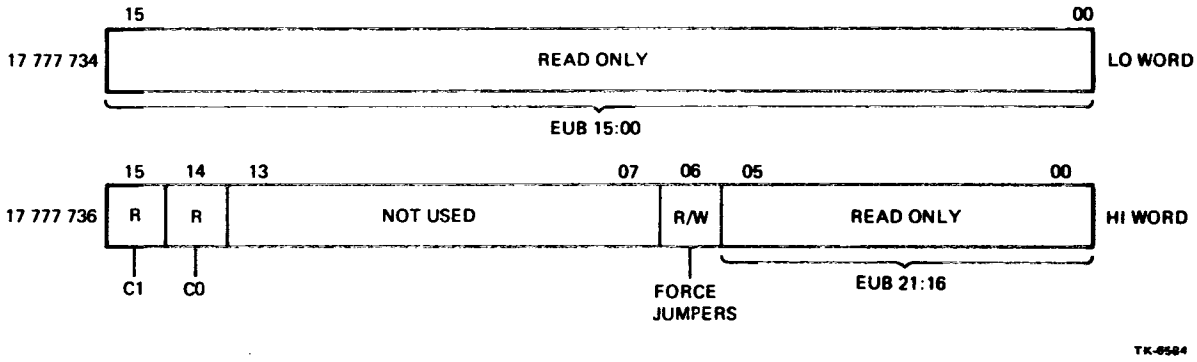


Figure 6-10 LMA Registers

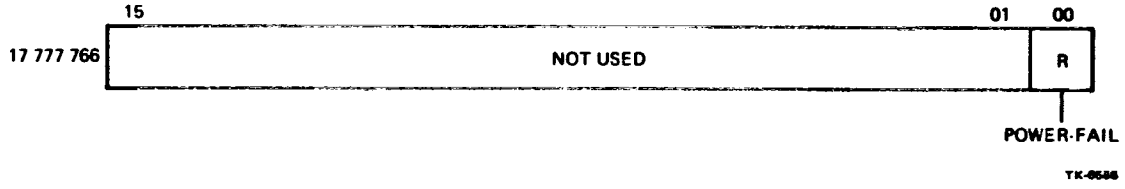


Figure 6-11 CPU Error Register

Table 6-3 LMA Bit Description

Bit	Description
Low Word	
15:00	Contains EUB ⟨15:00⟩ of the last mapped address.
High Word	
15	Memory control line C1 (Table 4-2).
14	Memory control line C0 (Table 4-2).
13:07	Not used.
06	Force jumpers bit. When set, it forces the UNIBUS addressing limits to the default condition (all jumpers out).
05:00	Contains EUB ⟨21:16⟩ of the last mapped address.

Table 6-4 CPU Error Register Description

Bit	Description
15:01	Not used.
00	Power-fail bit, set when one of the processor voltages is (was) out of tolerance.

CHAPTER 7
FLOATING-POINT PROCESSOR FUNCTIONAL DESCRIPTION

7.1 INTRODUCTION

The KEF11-A floating-point processor (FPP) is a microcode option for use with the PDP-11/24 processor. The KEF11-A is completely software compatible with all FP11 floating-point processors. Both single (32-bit) and double (64-bit) precision floating-point capabilities are available together with other features such as floating-to-integer and integer-to-floating data conversion.

The FPP microcode resides in two control chips packaged on one 40-pin hybrid DIP. The FPP requires the MMU chip to be present in the system (in addition to the base machine chip) because the MMU chip contains the floating-point accumulators and status registers (Figure 7-1).

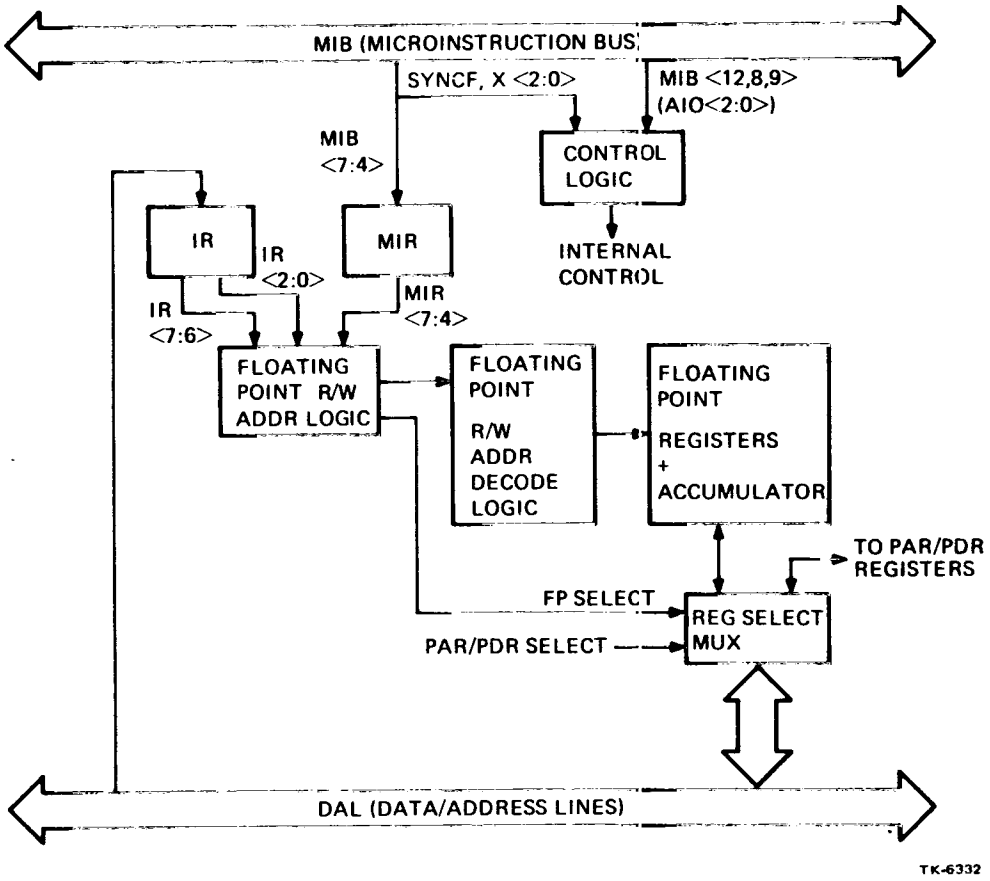


Figure 7-1 Block Diagram of Floating-Point Logic on MMU

7.2 FLOATING-POINT FORMATS

The FPP requires its input data (operands) to be formatted. This formatting allows the FPP to process operands in a meaningful way and produce correct results. There are four formats for operands input to the FPP: short-integer format (I), long-integer format (L), single-precision floating format (F), and double-precision floating format (D).

Output data from the FPP is also formatted. This output data is in the form of:

1. FPP status information and FPP exception information required by the CPU.
2. Data sent to memory (via the CPU), which must be in I, L, F, or D format.

7.2.1 FPP Integer Formats

There are two integer formats, short (I) and long (L). The short-integer format is 16 bits long and the long-integer format is 32 bits long. Data words (operands) in integer format are represented in 2's complement notation. In both I and L formats, the most significant bit of the data word is the sign bit. Figure 7-2 shows the integers +5 and −5 in both I and L formats.

Figure 7-3 also illustrates the formats in which integers are arranged in memory. Integers will be sent to memory in one of these formats.

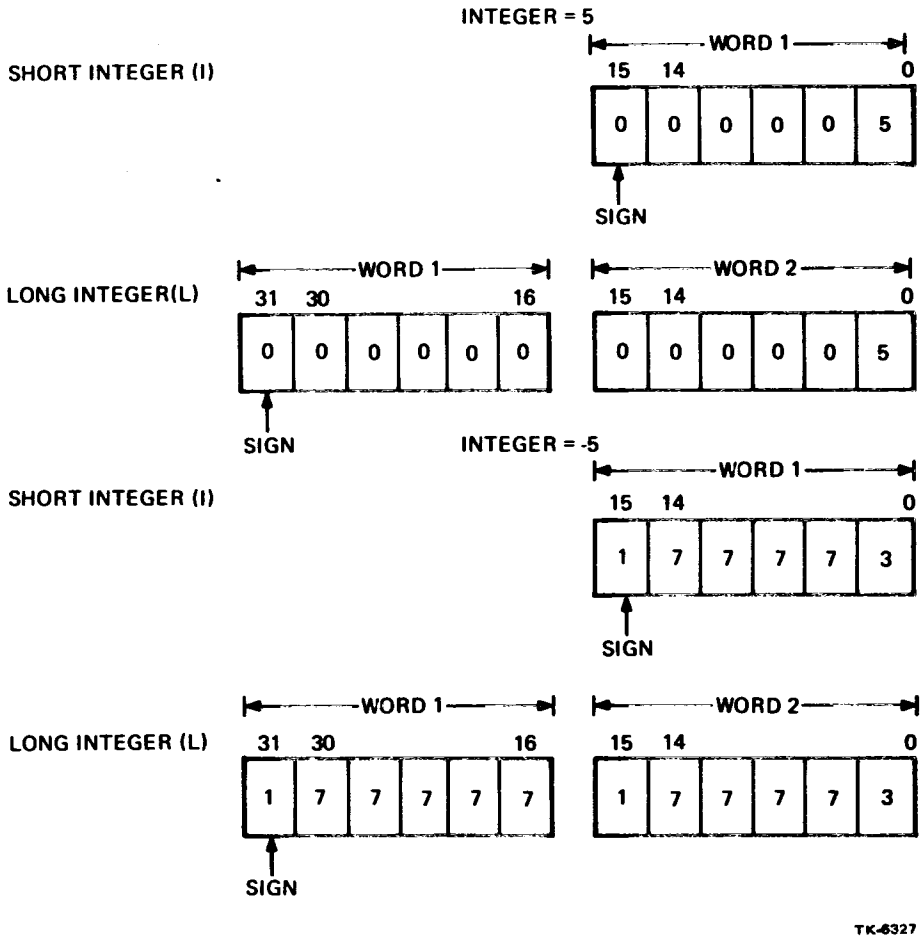


Figure 7-2 Integer Formats

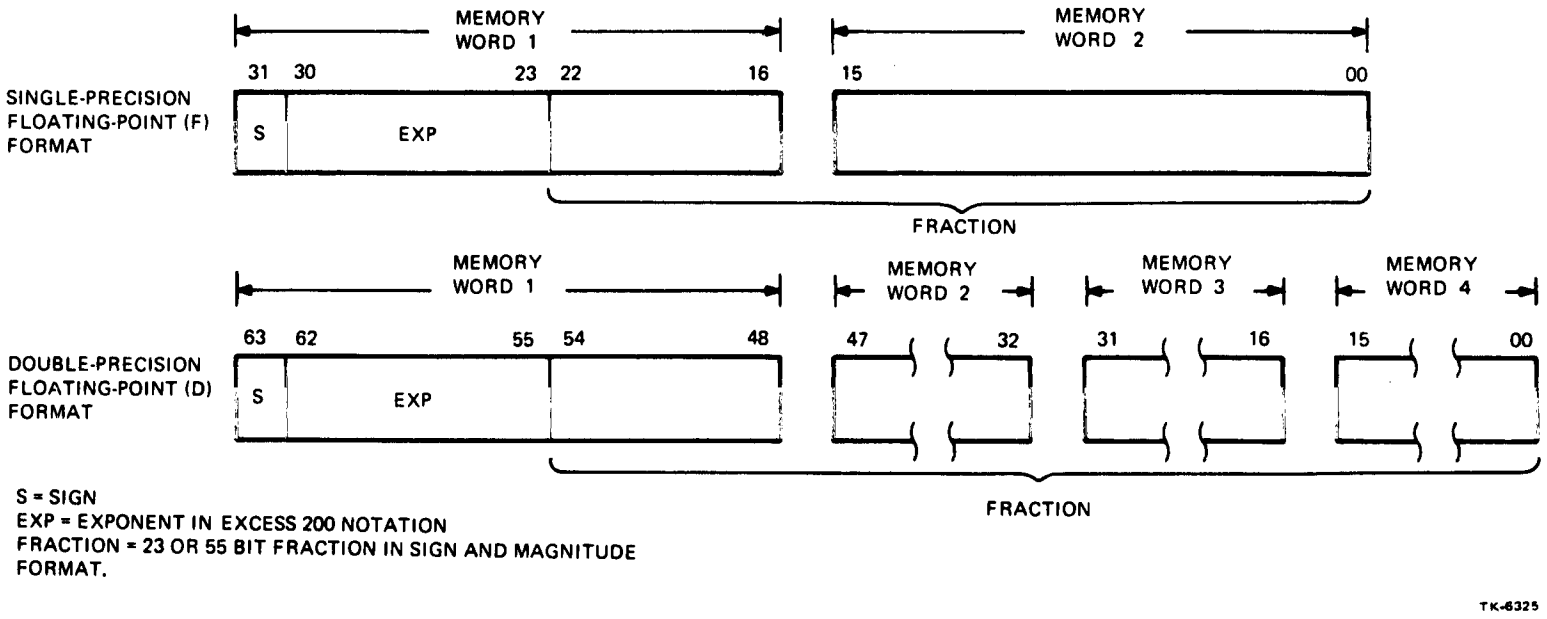


Figure 7-3 Floating-Point Data Formats

7.2.2 FPP Precision Formats

There are two floating-point precision formats, single-precision (F) and double-precision (D). The single-precision format is 32 bits long and the double-precision format is 64 bits long. Figure 7-3 shows that the most significant bit of the double (single-precision) or quad (double-precision) word is the sign of the fraction (and the floating-point number being represented). The next eights bits contain the value of the exponent, expressed in excess 200 notation. The remaining bits (23 for single-precision, 55 for double-precision) contain the fraction. The fraction and its associated sign bit are expressed in sign and magnitude notation.

7.2.3 Floating-Point Data Word

Figure 7-4 illustrates the formats in which floating-point numbers are arranged in memory. Floating-point numbers sent to memory must be in one of these formats. Floating-point numbers received by the FPP are arranged as illustrated in Figure 7-5.

The sign bit, exponent bits, and fraction bits in the FPP data word have the same values as the data word in memory. Note, however, that the FPP data word has more bits than its counterpart in memory. This is because the FPP has provisions for generating an overflow bit and a "hidden" bit.

For purposes of discussion, the FPP data word can be thought of as being divided into two major parts:

1. A fraction, with its associated sign bit, hidden bit, and overflow bit.
2. An exponent.

7.2.3.1 Floating-Point Fraction – The fraction is expressed in sign and magnitude notation. The following simple example illustrates the idea behind sign and magnitude notation.

	2's Complement Notation	Sign and Magnitude Notation
+2	000010	Sign 000010 Magnitude
−2	111110	Sign 100010 Magnitude

Only a change of sign bit is required to change the sign of a number in sign and magnitude notation. Note that a positive number is the same in both sign-magnitude and 2's complement notation.

Unnormalized floating-point fractions have a range from approximately 0 through 2 as shown in Figure 7-6. The FPP, however, normalizes all unnormalized fractions. That is, the fractions are adjusted such that there is a 0 to the left of the binary point (bit 63) and a 1 to the right of the binary point (bit 62). Thus, normalized fractions range in magnitude from 0.1000...to 0.1111 or from 1/2 to approximately 1.

Bit 62 is called the hidden bit. Recall that since fractions are normalized by the FPP, the bit immediately to the right of the binary point (bit 62) is always a 1. This bit is dropped when a fraction is sent to memory and appended when a fraction is received from memory. This procedure allows one extra bit of significance in floating-point fraction representation.

The fraction overflow bit (bit 63) is set during certain arithmetic operations. For example, during addition, certain sums will produce an overflow such as 0.1000... + 0.100... which yields 1.000.... This result must be normalized, so that FPP right-shifts the fraction one place and increases the exponent by one.

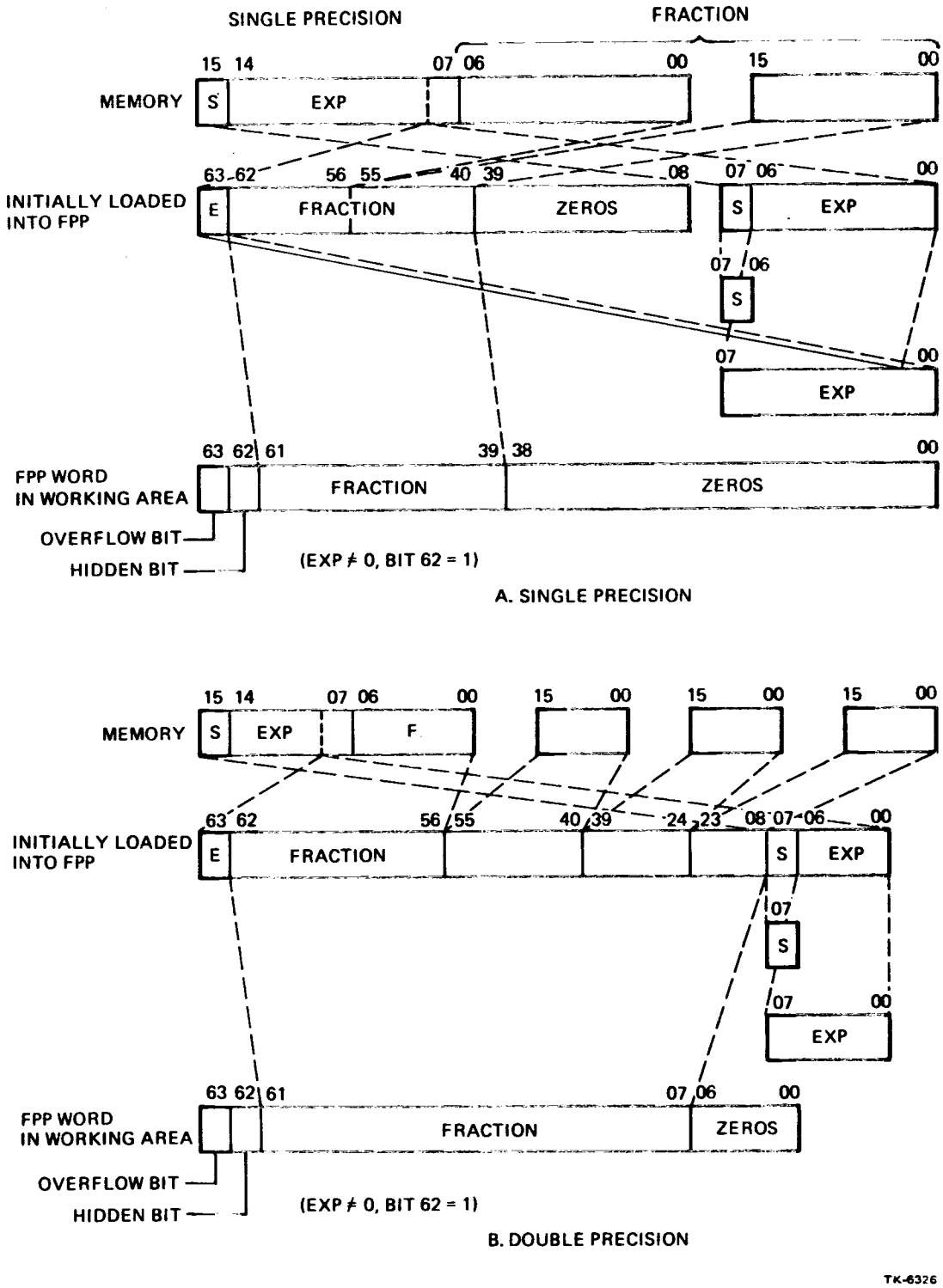


Figure 7-4 Floating-Point Data Words

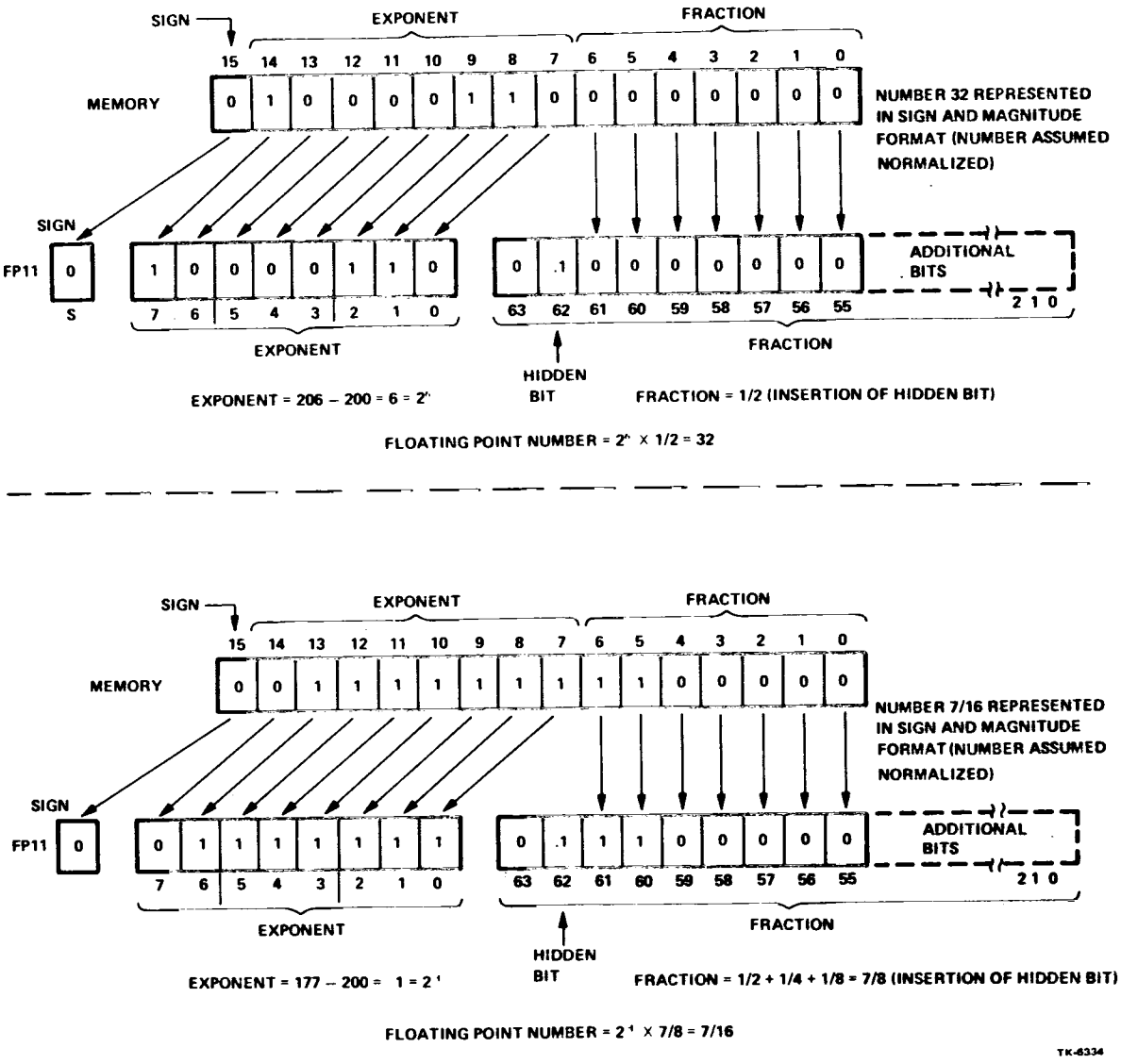


Figure 7-5 Interpretation of Floating-Point Numbers

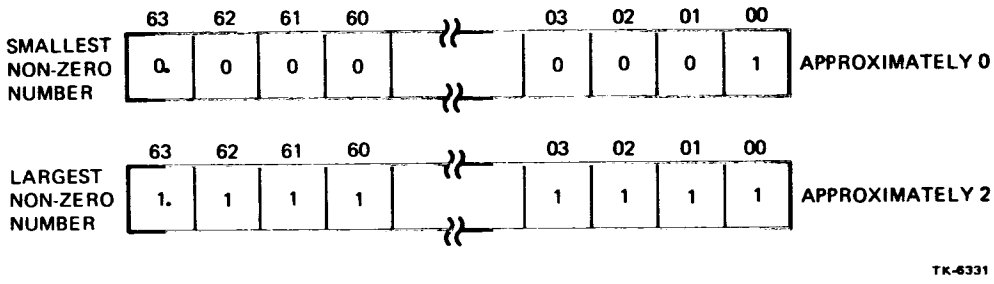
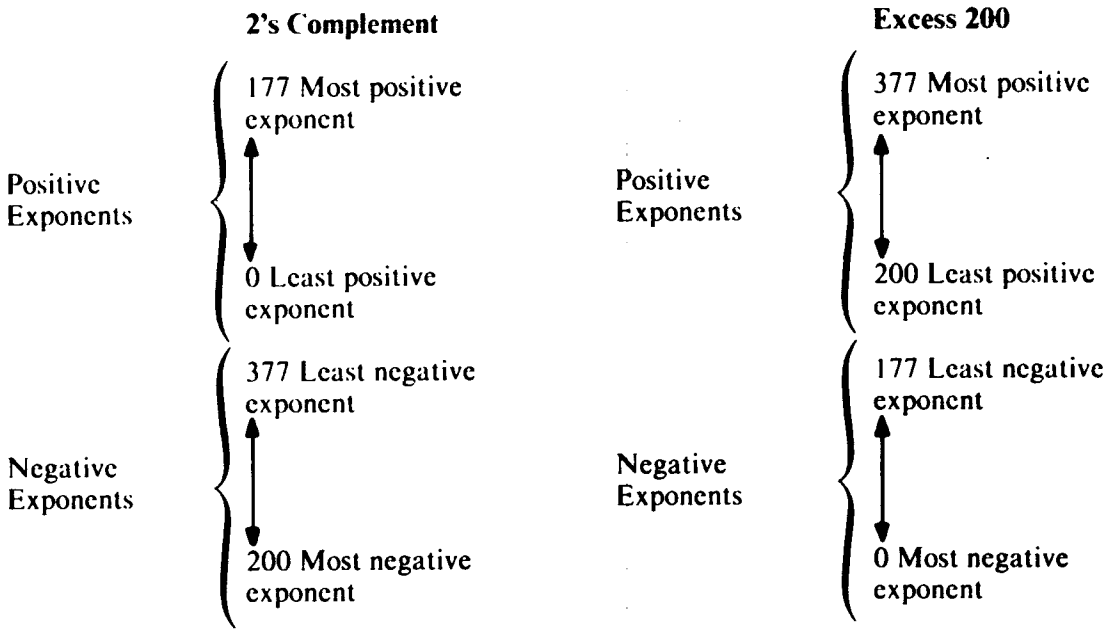


Figure 7-6 Unnormalized Floating-Point Fraction

7.2.3.2 Floating-Point Exponent – The eight-bit floating-point exponent is expressed in excess 200 notation. The chart below illustrates the relationship between exponents in 2's complement notation and exponents in excess 200 notation.



Note that an exponent in excess 200 notation is obtained by simply adding 200 to the exponent in 2's complement notation. Thus, eight-bit exponents in excess 200 notation range from 0 to 377 (or from -200 to $+177$). A number with an exponent of -200 is treated by the FPP as 0.

For example, the number 0.1_2 is actually 0.1×2^0, and the exponent is represented as 10 000 000 because 200_8 represents an exponent of zero.

7.2.4 Processing of Floating-Point Exceptions

Location 244 is the interrupt vector used to handle all floating-point interrupts. A total of six possible interrupts can occur. These possible interrupt exceptions are encoded in the FPP exception code (FEC) register. The interrupt exception codes represent an offset into a dispatch table, which routes the program to the right error-handling routine. The dispatch table is a function of the software. The FEC for each exception is briefly described in Table 7-1.

Refer to the *PDP-11 Processor Handbook* for further details concerning FPP exceptions.

In addition to the FEC register, the CPU contains a 16-bit floating-exception address (FEA) register, which stores the address of the last floating-point instruction that caused a floating-point exception.

7.3 FLOATING-POINT PROCESSOR STATUS REGISTER (FPS)

The floating-point status register provides mode and interrupt control for the FPP as well as conditions codes resulting from the execution of the previous floating-point instruction. Figure 7-7 illustrates the bits of the FPS, and Table 7-2 provides a description of each bit in the register.

Table 7-1 FPP Exception Codes

FP11-A Exception Code	Definition
02	Floating Op Code Error – The FPP causes an interrupt for an erroneous op code.
04	Floating Divide by Zero – Division by zero causes an interrupt if FID bit in the FPS register is not set.
06	Floating (or Double) Integer Conversion Error
10	Floating Overflow
12	Floating Underflow
14	Floating Undefined Variable

Note: The traps for exception codes 06, 10, 12, and 14, can be enabled in the FPP program status register. All traps are disabled if FID bit is set.

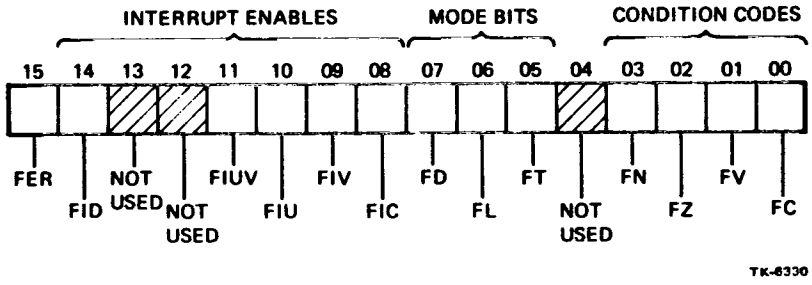


Figure 7-7 Floating-Point Processor Status Register

Table 7-2 Floating-Point Processor Status Register

Bit	Mnemonic	Description
15	FER	This bit indicates an error condition of the FPP.
14	FID	Floating Interrupt Disable – All interrupts by the FPP are disabled when this bit is on. Primarily for maintenance use. Normally clear.
13		Not used.
12		Not used.
11	FIUV	Floating Interrupt on Undefined Variable – If −0 is obtained from memory, an interrupt occurs. If the FIUV bit is not set −0 can be loaded and stored; however, any arithmetic operation treats it as if it was a +0.
10	FIU	Floating Interrupt on Underflow – When this bit is set, an underflow condition causes a floating underflow interrupt. The result of the operation causing the interrupt is correct except for the exponent, which is off by 400. If the FIU is not set and underflow occurs, the result is set to zero.
09	FIV	Floating Interrupt on Overflow – When this bit is set, floating overflow causes an interrupt. The result of the operation causing the interrupt is correct except for the exponent, which is off by 400.
08	FIC	Floating Interrupt on Integer Conversion Error – When this bit is set and the store convert floating-to-integer instruction causes FC to be set (indicating a conversion error), an interrupt occurs. When an interrupt occurs, the destination register is cleared and the source register is untouched. When FIC is reset, the result of the operation is the same; however, no interrupt occurs.
07	FD	Double-Precision Mode Bit – This bit, when set, specifies double-precision (64-bit) format and, when reset, specifies single-precision (32-bit) format.
06	FL	Long-Precision Integer Mode Bit – This bit is employed during conversion between integer and floating-point format. If set, double-precision 2's complement integer format of 32 bits is specified; if reset, single-precision 2's complement integer format of 16 bits is specified.
05	FT	Truncate Bit – This bit, when set, causes the result of any floating-point operation to be truncated rather than rounded.
04		Not used.
03:00	FN, FZ, FV, and FC	These bits are the four floating-point condition codes, which can be loaded in the CPU's N, Z, V, and C condition codes, respectively. This is accomplished by the copy floating condition codes (CFCC) instruction.

7.4 FLOATING-POINT INSTRUCTION FORMATS

Paragraph 7.4 provides a description of the FPP floating-point instructions and instruction formats. A description of the floating-point accumulators is also provided.

7.4.1 Floating-Point Accumulators

The MMU chip contains the six general-purpose accumulators (AC0–AC5) used by the FPP. These accumulators are 64-bit read/write scratchpad memories with nondestructive readout.

Each accumulator is interpreted as being either 32 or 64 bits long, depending on the instruction and the FPP status. If an accumulator is interpreted as being 64 bits long, 64 bits of data occupy the entire accumulator. If an accumulator is interpreted as being 32 bits long, 32 bits of data occupy only the left-most 32 bits of an accumulator as shown in Figure 7-8.

The floating-point accumulators are used in numeric calculations and interaccumulator data transfers. AC0–AC3 are used for all data transfers between the FPP and the CPU or memory.

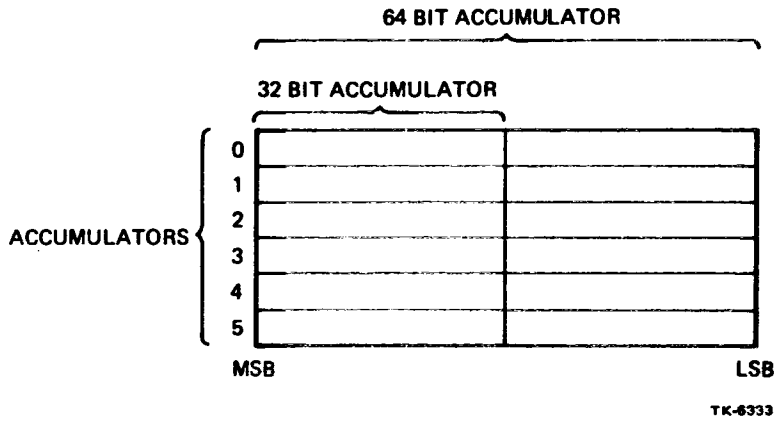


Figure 7-8 Floating-Point Accumulators

7.4.2 Instruction Formats

An FPP instruction must be in one of five formats. These formats are summarized in Figure 7-9.

The two-bit AC field (bits 6 and 7) allows selection of scratchpad accumulators 0 through 3 only.

If address mode 0 is specified with formats F1 or F2, bits ⟨2:0⟩ are used to select a floating-point accumulator. Only accumulators 0–5 can be specified in mode 0. If 6 or 7 is specified in bits ⟨2:0⟩ in mode 0, the FP11 traps if floating-point interrupts are enabled (FID = 0). The FEC will indicate an illegal op code error (exception code 02).

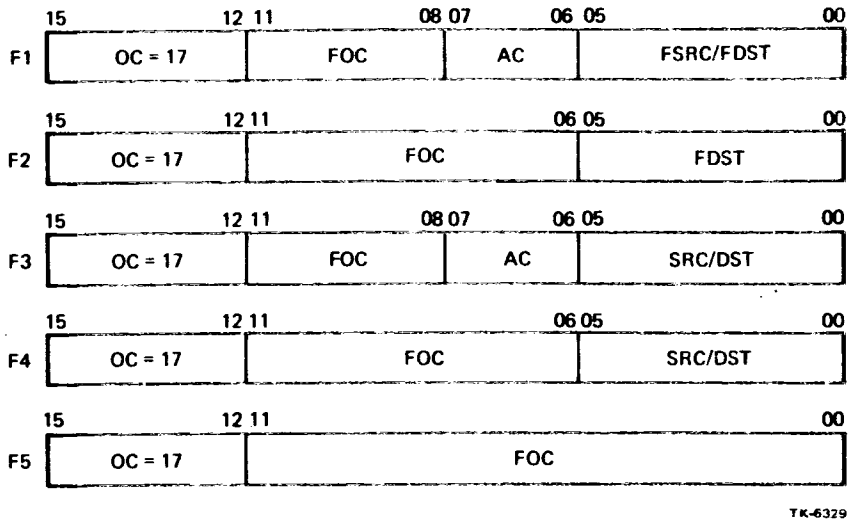


Figure 7-9 Instruction Formats

The fields of the various instruction formats (as summarized in Table 7-3) are interpreted as follows.

Mnemonic	Description
OC	Operation Code – All floating-point instructions are designated by a four-bit op code of 17_8.
FOC	Floating Operating Code – The number of bits in this field varies with the format; the code is used to specify the actual floating-point operation.
SRC	Source – A six-bit source field identical to that in the PDP-11 instruction.
DST	Destination – A six-bit destination field identical to that in a PDP-11 instruction.
FSRC	Floating Source – A six-bit field used only in format F1. It is identical to SRC, except in mode 0 when it references a floating-point accumulator instead of a CPU general register.
FDST	Floating Destination – A six-bit field used in formats F1 and F2. It is identical to DST, except in mode 0 when it references a floating-point accumulator instead of a CPU general register.
FAC	Floating Accumulator – A two-bit field used in formats F1 and F3 to specify FP11 scratchpad accumulators 0–3.

Table 7-3 Format of FPP Instructions

Instruction	Instruction Format	Mnemonic
Absolute	F2	ABSF FDST ABSD FDST
Add	F1	ADDF FSRC, AC ADD FSRC, AC
Clear	F2	CLRF FDST CLRD FDST
Compare	F4	CMPF FSRC, AC CMPD FSRC, AC
Copy Floating Condition Codes	F5	CFCC
Divide	F1	DIVF FSRC, AC DIVD FSRC, AC
Load	F1	LDF FSRC, AC LDD FSRC, AC
Load Convert	F1	LDCFD FSRC, AC FDCDF FSRC, AC
Load Convert Integer	F1	LDCIF SRC, AC LDCID SRC, AC LDCLF SRC, AC LDCLD SRC, AC
Load Exponent	F3	LDEXP SRC, AC
Load FP11'S Program Status	F4	LDFPS SCR
Modulo	F1	MODF FSRC, AC MODD ESRC, AC
Multiply	F1	MULF FSRC, AC MULD FSRC, AC
Negate	F2	NEGF FDST NEGD FDST
Set Double Mode	F5	SETD
Set Floating Mode	F5	SETF

Table 7-3 Format of FPP Instructions (Cont)

Instruction	Instruction Format	Mnemonic
Set Integer Mode	F5	SETI
Set Long Integer Mode	F5	SETL
Store	F1	STF AC, FDST STD AC, FDST
Store Convert	F1	STCFD AC, FDST STCDF AC, FDST
Store Convert Floating to Integer	F3	STCFI AC, DST STCFL AC, DST STCDI AC, DST STCDL AC, DST
Store Exponent	F3	STEXP AC, DST
Store FP11'S Program Status	F4	STFPS DST
Store FP11'S Status	F4	STST DST
Subtract	F1	SUBF FSRC, AC SUBD FSRC, AC
Test	F2	TSTF FDST TSTD FDST

7.5 FLOATING-POINT INSTRUCTIONS

7.5.1 Arithmetic Instructions

The arithmetic instructions (add, subtract, multiply, divide) require one operand in a source (a floating-point accumulator in mode 0, a memory location otherwise) and one operand in a destination accumulator. The instruction is executed by the FPP and the result is stored in the destination accumulator.

The Compare instruction also requires one operand in a source and one operand in a destination accumulator. However, the two operands remain in their respective locations after the instruction is executed by the FPP, and there is no transfer of the result.

7.5.2 Floating-Modulo Instruction

The Floating-Modulo (MOD) instruction causes the FPP to multiply two floating-point operands, separate the product into integer and fractional parts, and store one or both parts as floating-point numbers. The whole-number portion goes into an odd-numbered accumulator and the fraction goes into an even-numbered accumulator.

The whole-number portion of the number, when expressed as a floating-point number, contains an exponent greater than 201 in excess 200 notation, which means the the whole number has a decimal value of some number greater than one and less than UPLIM, where UPLIM is the greatest possible number that can be represented by the FPP.

The fractional portion of the number, when expressed as a floating-point number, contains an exponent less than or equal to 201 in excess 200 notation. This means that the fraction has a value less than one and greater than LOLIM, where LOLIM is the smallest possible number that can be represented by the FPP.

7.5.3 Load Instruction

The Load instruction causes the FPP to take an operand from a source and copy it into an accumulator. In mode 0 the source is a floating-point accumulator; otherwise, the source is a memory location.

7.5.4 Store Instruction

The Store instruction causes the FPP to take an operand from a source accumulator and transfer it to a destination. This destination is a floating-point accumulator in mode 0 and a memory location otherwise.

7.5.5 Load Convert (Double-to-Floating, Floating-to-Double) Instructions

The Load Convert Double-to-Floating (LDCDF) instruction causes the FPP to assume that the source specifies a double-precision floating-point number. The FPP then converts that number to single-precision and places this result in the destination floating accumulator. If the floating-truncate (FT) status bit is set, the number is truncated. If the FT bit is not set, the number is rounded.

The Load Convert Floating-to-Double (LDCFD) instruction causes the FPP to assume that the source specifies a single-precision number. The FPP then converts that number to double-precision by appending 32 zeros to the single-precision word and places this result in the destination accumulator.

Note that for both Load Convert instructions, the number to be converted is originally in the source (a floating-point accumulator in mode 0, a memory location otherwise) and is transferred to the destination accumulator after conversion.

7.5.6 Store Convert (Double-to-Floating, Floating-to-Double) Instructions

The Store Convert Double-to-Floating (STCDF) instruction causes the FPP to convert a double-precision number located in the source accumulator to a single-precision number. The FPP then transfers this result to the specified destination. If the floating-truncate (FT) bit is set, the floating-point number is truncated. If the FT bit is not set, the number is rounded. If the MSB (bit 31) of the double-precison segment of the word is a 1, 1 is added to the single-precision segment of the word, depending on the prior conditions set up by the FD bit in the FPS register; otherwise, the single-precison segment remains unchanged.

The Store Convert Floating-to-Double (STCFD) instruction causes the FPP to convert a single-precision number located in the source accumulator to a double-precision number. The FPP then transfers this result to the specified destination. The single-to-double precision conversion is obtained by appending zeros equivalent to the double-precision segment of the word (Figure 7-10).

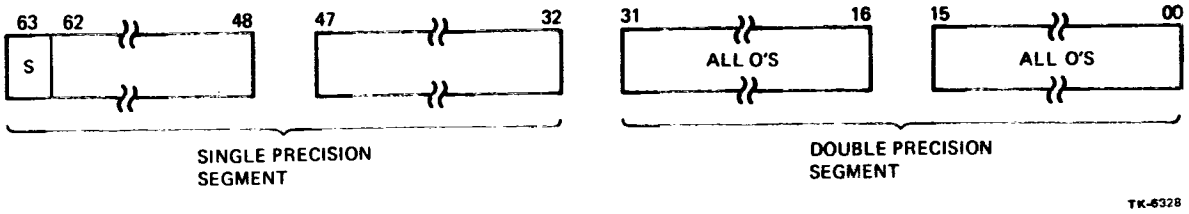


Figure 7-10 Single-to-Double Precision Appending

Note that for both Store Convert instructions, the number to be converted is originally in the source accumulator and is tranferred to the destination (a floating-point accumulator in mode 0, a memory location otherwise) after conversion.

7.5.7 Clear Instruction

The Clear instruction causes the FPP to clear a floating-point destination by setting all its bits to 0.

7.5.8 Test Instruction

The Test instruction causes the FPP to test the sign and exponent of a floating-point number and update the FPP status accordingly. The number tested is obtained from the destination (a floating-point accumulator in mode 0, a memory location otherwise). The FC and FV bits are cleared. The FN bit is set only if the destination is negative. The FZ bit is set only if the exponent of the destination is zero. If the FIUV status bit is set, a trap occurs (after the test instruction is executed) if a minus zero is encountered, indicating an undefined variable.

7.5.9 Absolute Instruction

The Absolute instruction causes the FPP to take the absolute value of a floating-point number by forcing its sign bit to 0. If mode 0 is specified, the sign of the number in the floating-point destination accumulator is forced to 0. The exponent of the number is tested, and if it is 0, 0s are written into the accumulator. If the exponent is nonzero, the accumulator is unaffected.

7.5.10 Negate Instruction

The Negate instruction causes the FPP to complement the sign of the operand and store the result in the same location.

7.5.11 Load Exponent Instruction

The Load Exponent instruction causes the FPP to load an exponent from the source (a floating-point accumulator in mode 0, a memory location otherwise) into the exponent field of the destination floating accumulator. In order to do this, the 16-bit, 2's complement exponent from the source must be converted to an 8-bit number in excess 200 notation. This process is described further in the following paragraph.

Assume that the 16-bit, 2's complement exponent is coming from memory. The possible legal range of 16-bit numbers in memory is from $000\ 000_8$ to $177\ 777_8$. On the other hand, the possible legal range of exponents in the FPP falls into two classes.

1. Positive exponents (0_8 through 177_8) – When 200_8 is added to any of these numbers, the sum stays within the legal 8-bit exponent field (i.e., from 200_8 through 377_8).

2. Negative exponents ($177\ 601_8$ through $177\ 777_8$) – When 200_8 is added to any of these numbers, the sum stays within the legal 8-bit exponent field (i.e., from 1_8 through 177_8).

7.5.12 Load Convert Integer-to-Floating Instruction

The Load Convert Integer-to-Floating instruction takes a 2's complement integer from memory and converts it to a floating-point number in sign and magnitude format. If short-integer mode is specified, the number from memory is 16 bits and is converted to a 24-bit fraction (single-precision) or a 56-bit fraction (double-precision), depending on whether floating or double mode is specified. In each case the appropriate exponent is assigned.

When loading a long integer in the single-precision floating mode, and if the long integer contains more than 24 significant digits, then less significant digits will be truncated with some loss of accuracy.

7.5.13 Store Exponent Instruction

The Store Exponent (STEXP) instruction causes the CPU to access a floating-point number in the FPP, extract the 8-bit exponent field from this number, and subtract a constant of 200_8 (since the exponent is expressed in excess 200_8 notation). The exponent is then stored in the destination as a 16-bit, 2's complement, right-justified number.

The legal range of exponents if from 0_8 to 377_8, expressed in excess 200 notation. This means that the number stored ranges from -200_8 to 177_8 after the constant of 200 has been subtracted.

7.5.14 Store Convert Floating-to-Integer Instruction

The Store Convert Floating-to-Integer instruction causes the FPP to take a floating-point number and convert it to an integer for transfer to a destination.

The four classes of this instruction are as follows:

1. STCFI – Convert single-precision, 24-bit fraction to a 16-bit integer (short-integer mode).
2. STCFL – Convert single-precision, 24-bit fraction to a 32-bit integer (long-integer mode).
3. STCDI – Convert double-precision, 56-bit fraction to a 16-bit integer (short-integer mode).
4. STCDL – Convert double-precision, 56-bit fraction to a 32-bit integer (long-integer mode).

The (normalized) floating-point number to be converted is transferred to the FPP. The FPP works with the sign bit and one of the following.

1. The 15 MSBs of the fraction for floating-to-integer and double-to-floating conversion.
2. The 31 MSBs of the fraction for double-to-long conversion.
3. The entire fraction for floating-to-long conversion.

The FPP subtracts 201_8 from the exponent to determine if the floating-point number is a fraction. If the result of the subtraction is negative, the exponent is less than 201_8, and the absolute value of the floating-point number is less than 1. When converted to an integer, the value of this number is 0; a conversion error occurs, the FZ bit is set, and 0s are sent to the destination. If the result of the subtraction is positive (or zero), it indicates that the exponent is greater than (or equal to) 201_8, and the floating-point number can be converted to a nonzero integer.

7.5.15 Load FPP's Program Status Instruction

This instruction causes the FPP to transfer 16 bits from the location specified by the source to the floating-point status (FPS) register. These 16 bits contain status information for use by the FPP in order to enable and disable interrupts, set and clear mode bits, and set condition codes.

7.5.16 Store FPP's Program Status Instruction

This instruction causes the FPP to transfer the 16 bits of the FPS register to the specified destination.

7.5.17 Store FPP's Status Instruction

The Store FP11's Status (STST) instruction causes the FPP to read the contents of the floating exception code (FEC) and floating exception address (FEA) registers when a floating-point exception (error) occurs.

If mode 0 addressing is enabled, only the FEC is sent to the destination accumulator. If mode 0 addressing is not enabled, the FEC is stored in memory followed by the FEA. In memory, the FEA data occupies all 16 bits of its memory location, while the FEC data occupies only the lower 4 bits of its location.

When an error occurs and the interrupt trap in the CPU is enabled, the CPU traps to interrupt vector 244_8. The program then issues the STST instruction to determine the type of error.

NOTE

The STST instruction should be used only after an error has occurred, since in all other cases the instruction contains irrelevant data or contains the conditions that occurred after the last error.

7.5.18 Copy Floating Condition Codes Instruction

The Copy Floating Condition Codes (CFCC) instruction causes the FPP to copy the four floating condition codes (FN, FZ, FV, FC) into the CPU condition codes (N, Z, V, C).

7.5.19 Set Floating Mode Instruction

The Set Floating Mode (SETF) instruction causes the FPP to clear the FD bit (bit ⟨07⟩ of the FPS register) and indicate single-precision operation.

7.5.20 Set Double Mode Instruction

The Set Double Mode (SETD) instruction causes the FPP to set the FD bit (bit ⟨07⟩ of the FPS register) and indicate double-precision operation.

7.5.21 Set Integer Mode Instruction

The Set Integer Mode (SETI) instruction causes the FPP to clear the FL bit (bit ⟨06⟩ of the FPS) and indicate that short-integer mode (16 bits) is specified.

7.5.22 Set Long-Integer Mode Instruction

The Set Long-Integer Mode (SETL) instruction causes the FPP to set the FL bit (bit ⟨06⟩ of the FPS) and indicate that long-integer mode (32 bits) is specified.

7.6 FLOATING-POINT INSTRUCTION EXECUTION

Floating-point macroinstructions are directly loaded into the PLA input register (PIR) of the base machine control chip via the DAL. The base machine control chip decodes the floating-point instruction and issues an unconditional branch microinstruction to the floating-point chip via the MIB. It should be noted that every control chip latches the microinstruction into its microinstruction register to determine if it is the selected chip. The floating-point chip decodes the microinstruction and branches to the location in its control store which decodes the specified floating-point routine. The floating-point chip also asserts its chip select line (K13 CSEL 2 L) indicating that it has recognized its address in the jump microinstruction. After the floating-point routine has been completed, the floating-point chip issues an unconditional jump back to the base machine control chip. If the floating-point chip does not assert K13 CSEL 2 L or if there is no floating-point chip present in the system, K7 CTL ERR H will be asserted at PBT2. K7 CHIP RESET H will also be asserted at the end of PBT2 to initialize the LSI chips in the system. K7 CHIP RESET H, when asserted, forces control of the processor back to the base machine control chip and the service microstate to be entered. K7 CTL ERR H being asserted causes a trap through the vector at 10_8. This trap indicates an attempt to execute a reserved instruction; in this case, the missing floating-point instruction. K13 CHIP RESET H will be cleared on the following MCLK after service information has been loaded into the system.

CHAPTER 8
COMMERCIAL INSTRUCTION SET PROCESSOR

8.1 INTRODUCTION

The commercial instruction set processor (CISP) is a microcode option (KEF11-B) for use with the PDP-11/24 processor. The KEF11-B is completely software compatible with the KE44-A CISP, which is used with the PDP-11/44 processor. The KEF11-B option significantly extends the capability of the PDP-11/24 in the area of commercial data processing.

The commercial instruction set (CIS) is a series of instructions for manipulating byte strings which provide improved COBOL performance, text editing, and word processing capabilities. CIS includes instructions for character handling and decimal arithmetic operations. Each of these instructions has two forms, register and in-line. CIS register form instructions load descriptors into the general registers and then perform the instruction. CIS in-line form instructions call the descriptors through descriptor address pointers. The CIS also includes load two descriptors and load three descriptors instructions. These instructions augment the register form of the instructions by making it easy to load descriptors into registers. Descriptors are used to identify character strings and consist of the starting address of the string and the length of the string. A complete description of the function of the PDP-11/24 commercial instruction set and its instructions is provided in the *PDP-11 Processor Handbook*.

The CISP microcode resides in six control chips packaged on one 40-pin, double-width hybrid package. The CISP requires only the base machine chip to be installed in the system to use the commercial instruction set. The CISP option may be added or removed from the system simply by installing or removing the CISP hybrid.

The microcode for the various CIS instructions is divided among the six CISP chips. Table 8-1 lists the location of the microcode for the CIS instructions. All CIS instructions are dispatched through the first CISP chip. All in-line descriptor calculations are also done in this chip before dispatch.

8.2 CHARACTER DATA TYPES

There are three different character data types. The "character" is a single byte, and is an abbreviated string of length 1. The "character string" is a contiguous group of bytes in memory. The third is a "character set."

The character is an eight-bit byte.

The character is used as an operand by CIS instructions. When it appears in a general register, the character is in the low order half; the high order half of the register must be zero. When it appears in the instruction stream, the character is in the low order half of a word; the high order half of the word must be zero. If the high order half of a word which contains a character is non-zero, the effect of the instruction which uses it will be unpredictable.

Table 8-1 Commercial Instruction Set

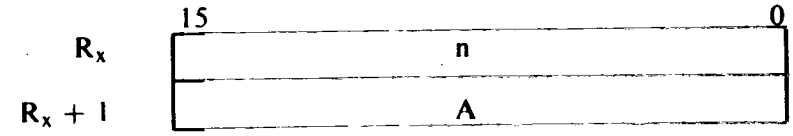
Instruction	Op Code	Data	Form	Description
Control				
L2Dr	07602r	–	In-Line	Load two descriptors
L3Dr	07606r	–	In-Line	Load three descriptors
Byte Manipulation				
CMPC	076044	Byte	Register	Compare character
CMPCI	076144	Byte	In-Line	Compare character
LOCC	076040	Byte	Register	Locate character
LOCCI	076140	Byte	In-Line	Locate character
MATC	076045	Byte	Register	Match character
MATCI	076145	Byte	In-Line	Match character
MOVC	076030	Byte	Register	Move character
MOVCI	076130	Byte	In-Line	Move character
MOVRC	076031	Byte	Register	Move reverse justified character
MOVRCI	076131	Byte	In-Line	Move reverse justified character
MOVTC	076032	Byte	Register	Move translated character
MOVTCI	076132	Byte	In-Line	Move translated character
SCANC	076042	Byte	Register	Scan character
SCANCI	076142	Byte	In-Line	Scan character
SKPC	076041	Byte	Register	Skip character
SKPCI	076141	Byte	In-Line	Skip character
SPANC	076043	Byte	Register	Span character
SPANCI	076143	Byte	In-Line	Span character
Arithmetic				
ADDN	076050	Numeric	Register	Add
ADDNI	076150	Numeric	In-Line	Add
ADDP	076070	Packed	Register	Add
ADDPI	076170	Packed	In-Line	Add
ASHN	076056	Numeric	Register	Arithmetic shift
ASHNI	076156	Numeric	In-Line	Arithmetic shift
ASHP	076076	Packed	Register	Arithmetic shift
ASHPI	076176	Packed	In-Line	Arithmetic shift
CMPN	076052	Numeric	Register	Compare numeric
CMPNI	076152	Numeric	In-Line	Compare numeric
CMPP	076072	Packed	Register	Compare packed
CMPPI	076172	Packed	In-Line	Compare packed
CVTLN	076057	Numeric	Register	Convert long to numeric
CVTLNI	076157	Numeric	In-Line	Convert long to numeric
CVTLP	076077	Packed	Register	Convert long to packed
CVTLPI	076177	Packed	In-Line	Convert long to packed
CVTNL	076053	Numeric	Register	Convert numeric to long
CVTNLI	076153	Numeric	In-Line	Convert numeric to long
CVTPL	076073	Packed	Register	Convert packed to long

Table 8-1 Commercial Instruction Set (Cont)

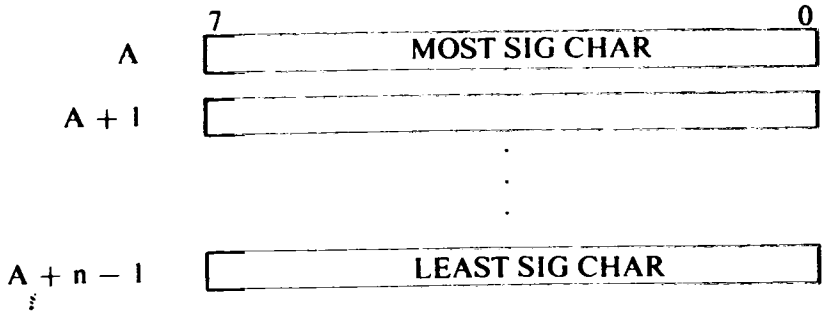
Instruction	Op Code	Data	Form	Description
CVTPLI	076173	Packed	In-Line	Convert packed to long
CVTNP	076055	Numeric	Register	Convert numeric to packed
CVTNPI	076155	Numeric	In-Line	Convert numeric to packed
CVTPN	076054	Packed	Register	Convert packed to numeric
CVTPNI	076154	Packed	In-Line	Convert packed to numeric
DIVP	076075	Packed	Register	Divide
DIVPI	076175	Packed	In-Line	Divide
MULP	076074	Packed	Register	Multiply
MULPI	076174	Packed	In-Line	Multiply
SUBN	076051	Numeric	Register	Subtract
SUBNI	076151	Numeric	In-Line	Subtract
SUBP	076071	Packed	Register	Subtract
SUBPI	076171	Packed	In-Line	Subtract

A character string is a contiguous sequence of bytes in memory that begins and ends on a byte boundary. It is addressed by its most significant character (lowest address). The highest address is the least significant character. It is specified by a two-word descriptor with the attributes of length and lowest address. The length is an unsigned binary integer which represents the number of characters in the string and may range from 0 to 65,535. A character string with zero length is said to be vacant; its address is ignored. A character string with non-zero length is said to be occupied.

The character string descriptor is used as an operand by CIS instructions. It appears in two consecutive general registers or in two consecutive words in memory pointed to by a word in the instruction stream. The following shows the descriptor for a character string of length "n" starting at address "A" in memory.



The following shows the character string in memory.



A character set is a subset of the 256 possible characters that can be encoded in a byte. It is specified by a descriptor which consists of the address of a 256-byte table and an 8-bit mask. The address is of the zeroeth byte in the table. Each byte in the table specifies up to eight orthogonal character subsets of which the corresponding character is a member. The mask selects which combinations of these orthogonal subsets comprise the entire character set. In effect, each bit in the mask corresponds to one of eight orthogonal subsets that may be encoded by the table. The mask specifies the union of the selected subsets into the character set. Typical sets would be: upper case, lower case, non-zero digits, end of line, etc.

8.3 DECIMAL STRING DATA TYPES

Two classes of decimal string data types–numeric strings and packed strings–are defined. Both have similar arithmetic and operational properties; they primarily differ in the representation of signs and the placement of digits in memory.

The numeric string data types are signed zoned, unsigned zoned, trailing overpunch, leading overpunched, trailing separate and leading separate. The packed string data types are signed packed and unsigned packed. Instructions which operate on numeric strings permit each numeric string operand to be separately specified. Similarly, packed string instructions permit each packed string operand to be separately specified. Thus, within each of the two classes of decimal strings, the operands of an instruction may be of any data type within the appropriate class.

Decimal strings exist in memory as contiguous bytes which begin and end on a byte boundary. They represent numbers consisting of 0 to 31_{10} digits, in either sign-magnitude or absolute-value form. Sign-magnitude strings (signed) may be positive or negative; absolute-value strings (unsigned) respresent the absolute value of the magnitude. Decimal numbers are whole integer values with an implied decimal radix point immediately beyond the least significant digit. They may be conceptually extended with zero digits beyond the most significant digit.

8.4 CIS INSTRUCTION EXECUTION

CIS macroinstructions are directly loaded into the PLA input register (PIR) of the base machine control chip via the DAL. The base machine control chip decodes the instruction and issues an unconditional branch microinstruction to the initial CIS chip, via the MIB. The CIS chip decodes the microinstruction and branches to the location in its control store which contains the CIS routine. It should be noted that every control chip latches the microinstruction into its microinstruction register to determine if it is the selected chip. The CIS chip decodes the microinstruction and branches to the location in its control store which contains the CIS routine.

The CIS chip also asserts its chip select line (K13 CSEL 1 L), indicating that it has recognized that it is the destination of the unconditional jump microinstruction. Further decoding of the macroinstruction is done in the initial CIS chip to determine where the microroutine specified by the macroinstruction resides. If the microroutine resides in the initial CIS chip, the routine is executed, and, upon completion, an unconditional branch to the base machine chip is performed. However, if the microroutine resides in one of the other five CIS chips, the initial CIS chip issues an unconditional branch to the CIS chip which contains the routine. The initial CIS chip will now clear its chip select line and the CIS chip which contains the necessary routine will assert its chip select line and execute the routine. When one of the other five CIS chips recognize the jump microinstruction, it will assert K13 CSEL 1 L. Upon completion of the CIS routine, an unconditional branch microinstruction to the base machine chip is issued.

If the base machine attempts an unconditional microjump to the first CIS chip, and the chip is not present in the system or does not respond, the absence of K13 CSEL1 L or K13 CSEL2 L will cause CTL ERR H to be asserted at PBT2. CHIP RESET H will also be asserted at PBT2 to initialize the other LSI chips in the system. This will cause a re-selection of the base machine control chip, just as though an unconditional microjump was directed at it. K7 CHIP RESET H will be cleared on the following MCLK after service information has been loaded into the base machine chip. The base machine then enters the service microstate and K7 CTL ERR H will cause a trap through the vector at 10_8 to tell the user that the CIS processor is not present or is not functioning properly.

The same process will occur if the first CIS chip fails in its attempt to microjump to another CIS chip. This informs the user that the specific CIS instruction cannot be executed.

CHAPTER 9
MAINTENANCE

This chapter describes the basic troubleshooting procedures and diagnostics used with the PDP-11/24.

9.1 TROUBLESHOOTING PROCEDURES

The tests contained in this section are used to diagnose basic malfunctions of the PDP-11/24. These tests are not to replace MAINDEC diagnostics but are to be used in cases when the PDP-11/24 is not functioning well enough to run diagnostics. If the machine is functioning well enough to run diagnostics, refer to Paragraph 9.2 of this manual.

When performing the following tests, each test is to be performed in sequence. When a fault is encountered a detailed analysis and repair of the fault should be done before proceeding to the next test. Note that some of the tests require the next test to be performed when a failure occurs, thereby further isolating the cause of the failure.

Before starting the test procedure, the PDP-11/24 must be set to the following conditions:

1. Turn the CPU keyswitch to the DC OFF position.
2. Turn the console terminal ON.

TEST 1

1. Place the console terminal in the LOCAL mode.
2. Check to see that the console terminal is operating properly.

This test is necessary and is not to be skipped. The PDP-11/24 does not have a "lights and switches console;" the ASCII console, therefore, must be operational to have control over and visibility into the PDP-11/24 processor. If the console is operating properly, proceed to Test 2. If not, repair the console and then proceed to Test 2.

TEST 2

1. Place the processor HALT/CONT/BOOT switch in the HALT position.
2. Turn the processor keyswitch to the LOCAL position.
3. Check the processor fans to see if they are spinning.

If the fans are spinning, proceed to Test 3. If the fans are not spinning, perform the following steps. These steps assume the presence of a DEC power controller (874 or equivalent).

1. For all PDP-11/24 variations check to be certain there is ac power at the CPU box. The following steps will verify this.

    a. Check to ensure that the cabinet ac power cord is securely plugged into its ac receptacle.

b. Check the neon lights on the power controller. They will be lit if ac is present at its input.

c. Check the circuit breaker on the power controller. It should be in the ON (1) position.

d. Check the CPU power cord. It should be plugged into an unswitched outlet on the power controller. If necessary, plug a known good load (e.g., a lamp) into an outlet on the power control.

e. If there is ac power at the CPU, proceed to step 2 for the BA11-L (5-1/4 in) mounting box or step 3 for the BA11-A (10-1/2 in) mounting box.

2. Perform the BA11-L (5-1/4 in) mounting box power check.

Check the CPU circuit breaker. It should be in the ON (1) position. The circuit breaker is reached by sliding the CPU a few inches out of its wraparound (Figure 2-5). If the circuit breaker is ON the problem may be one or more of the following:

WARNING

The power supply in the BA11-L box is not de-energized when the front-panel keyswitch is in the DC OFF postion.

a. A defective H777 power supply.
b. A blown fuse in the fuse holder next to the CPU circuit breaker.
c. Defective cabling between the CPU front panel and the power supply.
d. A defective front-panel module.

To diagnose a problem in the front panel and its associated cabling, short pins 1 and 3 of J5 (Figure 9-1) at the back of the mounting box. If the CPU powers up, the problem is with the front panel or its cabling.

3. Perform the BA11-A (10-1/2 in) mounting box power check.

Check the CPU circuit breaker located on the back panel of the mounting box (Figure 9-2). It should be in the ON (1) position. If the circuit breaker is already on the problem may be one or more of the following:

WARNING

The power supply in the BA11-A box is not de-energized when the front-panel keyswitch is in the DC OFF position.

a. A defective H7140 power supply.

WARNING

Very high voltages are stored on the capacitors contained in the power supply used in the BA11-A box. DO NOT open the power supply without a thorough understanding of its operation.

b. Defective cabling between the CPU front panel and the power supply.
c. A defective front-panel module.

To diagnose a problem in the front panel and its cabling, short pins 1 and 3 of J3 or J4 at the back of the CPU box. If the CPU powers up, the problem is the front-panel module or its cabling.

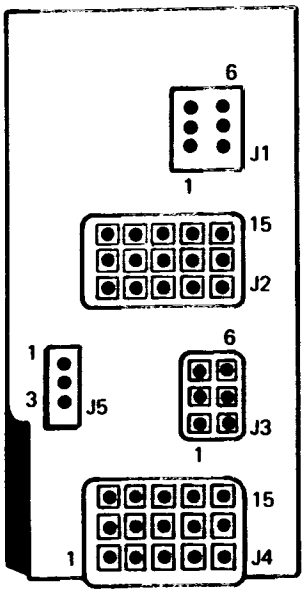


Figure 9-1 BA11-L Distribution Panel

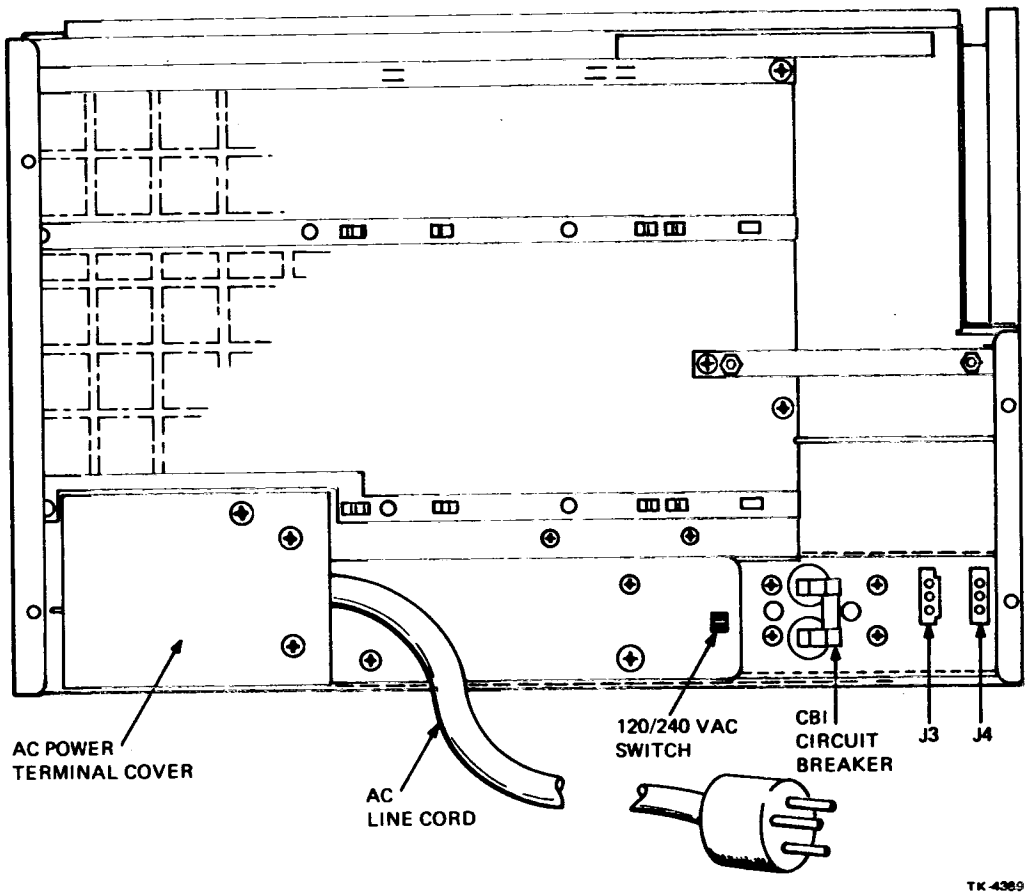


Figure 9-2 BA11-A Backpanel Components

TEST 3

The CPU is now operating minimally, at least to the point that its cooling fans are working. Now inspect the front panel LED labeled DC ON and proceed as indicated in the following steps.

1. DC ON is lit steadily – Proceed to Test 4.

2. DC ON is not lit – This indicates that +5 Vdc is not present at the CPU. The most likely cause is a defective power supply.

3. DC ON is blinking – This condition can exist when the UNIBUS map option module (M7134) is present in the system and indicates that +5 Vdc is present but one of the CPU voltages, including +5 V, is out of tolerance or E6, S3 on the M7134 module is improperly configured. The out-of-tolerance voltage can be isolated by reconfiguring S1–S4 of switch pack E6. Paragraph 2.8.3.4 of this manual explains the function of these switches.

TEST 4

The CPU power is now assumed to be okay.

1. If the M7134 UNIBUS map option module is present in the system, the voltages are checked by its voltage monitoring circuitry.

2. If the M7134 module is not present in the system, the voltages can be checked with a voltmeter or oscilloscope. The voltages can be checked at the backplane power connectors. Figure 9-3 shows the connector, and Table 9-1 gives the signal for each pin.

After verifying the CPU voltages proceed to Test 5.

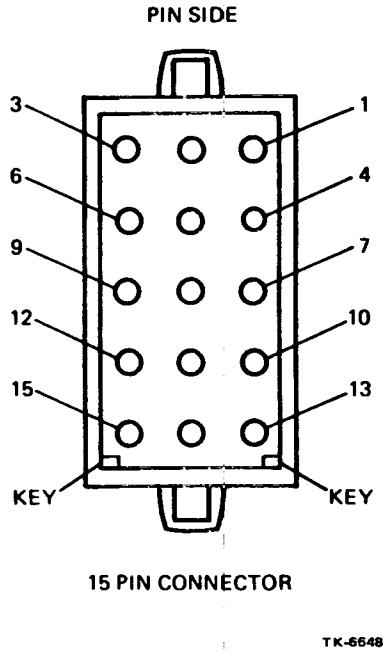


Figure 9-3 Backplane Power Connector Pin Designations

Table 9-1 PDP-11/24 Backplane Power Connectors

Pin	Signal P2	P4
1	+5 V	+5 V
2	Not Used	Not Used
3	+15 V	+15 V
4	+5 V	+5 V
5	Not Used	Not Used
6	Not Used	+12/15 B
7	Not Used	Not Used
8	GND	GND
9	GND	GND
10	Not Used	Not Used
11	Not Used	Not Used
12	+5 B	Not Used
13	Not Used	Not Used
14	−15 V	−15 V
15	Not Used	−12/15 B

TEST 5
Check the condition of the front panel and module LEDs. They should be as follows.

LED	Condition
Front Panel	
RUN	OUT
DC ON	ON
BATT	*
PROC	OUT
CPU Module	
CLK	ON
1	*
0	*
Memory Modules	
GREEN	ON
RED	OFF

If all the LEDs are correctly lit, proceed to Test 6. If the LEDs are not correctly lit, refer to Table 9-2.

* Indicates an indeterminate condition.

Table 9-2 Problems Indicated by System LEDs

LED	Condition	Possible Problem
Front Panel		
DC ON	Off	+5 Vdc is not present.
	Blinking	CPU voltage or voltages are out of tolerance.
RUN PROC	On Off	A defective terminal, terminal cable, or serial line unit.
PROC RUN	On Off	Processor has powered-up to a micro-loop, possibly due to a defective battery backup unit (if present), defective memory, or a defective power supply.
RUN PROC	On On	CPU disregarded setting of HALT switch, indicating a defective front panel or front-panel cable. Check to be sure the HALT/CONT/BOOT switch is in the HALT position and the keyswitch is in the LOCAL position.
CPU Module		
CLK	Off	A defective CPU module, CPU chip, or power supply.
Memory Module		
GREEN	Off	Indicates a lack of +5 B on the memory module. This may be caused by a blown fuse on the memory module or a defective power supply.
RED	On	A defective memory module or a memory voltage problem.

TEST 6
The terminal should now be displaying:

```
000000
@
```

If the terminal output is correct, proceed to Test 7. If the terminal output is not correct, perform the following procedures to help isolate the problem.

1. If the terminal printed text but the text is garbled, the problem may be the baud rate, stop bits, or parity settings. Check to be sure the terminal and CPU are set up properly.

2. If the terminal printed nothing, the problem could be:

   a. The terminal and CPU setup mentioned in step 1.

   b. The connection between the terminal and the CPU. Check to see that the terminal is connected to the SLU connector assembly and that the SLU connector assembly is connected to the CPU.

   c. The EIA set-up jumpers on the CPU do not match the terminal configuration.

NOTE
The PDP-11/24 is only capable of EIA operation.

   d. The terminal cabling.

      1) If hardwired to the CPU, the terminal should connect to the CPU through a null-modem cable, BC22-A or equivalent.

      2) A bad terminal cable, which can be checked by shorting pins 2 and 3 of the terminal's cable. The terminal should now echo characters typed on it. If not, the problem lies in the terminal cable or its associated cabling.

   e. Faulty serial line logic on the CPU module (M7133). This can be checked by performing the following steps:

      1) Place S5 of E124 in the ON position.

      2) Type a 'U' at the terminal.
      An '*' should be echoed.

      3) Type an '*' at the terminal.
      A 'U' should be echoed.

      If the characters are not echoed, the fault lies in the SLU logic. If the characters are echoed, the PDP-11 processor on the CPU module is defective.

      4) Place S5 of E124 in the OFF position.

      5) Proceed to Test 7 to further isolate the problem.

(See Appendix D, Section D.2 for M7133-YA CPU switch/jumper locations.)

TEST 7
While watching the front-panel RUN light type a ⟨space⟩ at the console terminal. If the character was received by the PDP-11/24, the RUN light will flash once.

The terminal display should now be:

```
@?
@
```

If the RUN light did not flash, the terminal is not getting through to the PDP-11/24 CPU chip.

TEST 8
Type '0/' at the console terminal. The terminal should now be displaying:

@0/nnnnnn_

where nnnnnn represents a 16-bit octal integer and _ represents a space.

For MS11-L memory, nnnnnn will most likely be 000000, although different numbers do not necessarily indicate a problem. Proceed to Test 9 if the terminal output is reasonable.

If the terminal display is:

@0/?
@

the PDP-11/24 has failed to access memory location 00 000 000. This may be due to an incorrect setup of the switches on the memory module(s) or a number of the previously mentioned problems.

TEST 9
The CPU now has memory location 0 opened. Type '177777' followed by ⟨return⟩. 177777 will be deposited as the new contents of location 0. The terminal display should now be:

@0/nnnnnn_177777
@

Type '/'. The terminal should now display:

@/177777_

This indicates that the data pattern has been correctly read from memory. If the response is correct, proceed to Test 10. If the response is not correct, a failure in the data paths, memory, UNIBUS, or CPU, of the machine is indicated. To further isolate the problem, type ⟨return⟩ followed by '765132/'. The terminal should display:

@765132/177777_

The data is now being read from the CPU diagnostic ROM. If the same bits are still missing, the fault lies either in the UNIBUS data lines or the CPU.

TEST 10
Type '0' followed by ⟨return⟩. This changes the contents of location 0 to 000000. Type '/'. The terminal should now display:

@/177777_0
@/000000_

If the expected response is received, proceed to Test 11. Any difference in the response indicates an error in the data paths of the machine. The data paths can be checked further by typing a ⟨return⟩ followed by '765144/'. The terminal should now display:

@765144/000000_

The data is now being read from the CPU diagnostic ROM. If the same bits are still incorrect, the fault is with either the UNIBUS data lines or the CPU.

TEST 11

The previous tests have partially verified that the PDP-11/24 memory subsystem is working and that the CPU path to memory is functional. It is now time to load into memory and run a very short program. Type ⟨return⟩. If the PDP-11/24 replies with a '?', disregard it. Next, type the following underlined text.

```
@4/nnnnnn_ 6⟨LF⟩
000006/nnnnnn_ 0⟨LF⟩
000010/nnnnnn_ 12⟨LF⟩
000012/nnnnnn_ 0⟨CR⟩
@ 1000/nnnnnn_ 240⟨LF⟩
001002/nnnnnn_ 776⟨LF⟩
001004/nnnnnn_ 0⟨CR⟩
@ R6/nnnnnn_ 1000⟨CR⟩
@ 1000G
001000
@
```

Immediately after the 'G' was typed, the RUN LED should have flashed once and the console terminal should have displayed the text shown. If this test ran successfully, proceed to Test 12. If the console displayed nothing after the 'G' was typed, and the RUN and PROC LEDs are ON, the CPU halt logic is defective. Before attempting to locate the problem, make sure the HALT/CONT/BOOT switch is in the HALT position and the keyswitch is in the LOCAL position.

TEST 12

Place the HALT/CONT/BOOT switch in the CONT position. No action should occur. Type 'P' at the console terminal. The RUN and PROC LEDs should both be lit and the console should now display:

```
001000
@P
```

If the terminal display is correct, proceed to Test 13.

Three possible faults are:

1. Terminal display:

   ```
   001000
   @P
   001002
   @
   ```

   The CPU halt logic is defective and will not permit the CPU to run. The trouble could be with the front panel, cabling, or the CPU.

2. Terminal display:

   ```
   001000
   @P
   000010
   @
   ```

   The CPU serviced a bus error during execution of the program. Examine memory locations 1000 and 1002 to make sure they contain 000240 and 000776, respectively. If so, this is probably a CPU problem.

3. Terminal display:

```
001000
@P
000014
@
```

The CPU serviced a reserved instruction trap during execution of the program. Examine memory locations 1000 and 1002 to ensure they contain 000240 and 000776, respectively. If so, this probably reflects a CPU error.

TEST 13

The CPU is now running the simple program loaded in memory. Momentarily press the BREAK key on the console terminal. The terminal should now display:

```
nnnnnn
@
```

The address represented by nnnnnn should be either 001000 or 001002. Any other address represents a fault. Examine the program to verify that memory still contains the correct instructions. If so, the CPU may be at fault.

Failure of the PDP-11/24 to halt may be due to the terminal not sending the BREAK character or the failure of SLU1 to detect the break. The failure may also be any of those failures discussed under Test 11. Proceed to Test 14 to further isolate any faults.

TEST 14

Type 'P' at the console terminal. The RUN and PROC LEDs should be lit and the console should display:

```
@P
```

Place the HALT/CONT/BOOT switch in the HALT position. The RUN and PROC LEDs should go off and the console terminal should display:

```
nnnnnn
@
```

The address represented by nnnnnn should once again be 001000 or 001002. Any other address represents the same failure discussed in Test 13.

If the CPU can be halted via the front panel switch but not via the BREAK key on the console terminal, the problem is in either the terminal's break generation logic or the PDP-11/24's break detection logic.

TEST 15

Place the keyswitch in the LOC DSBL position. Type 'P' at the console terminal. The RUN and PROC LEDs should light and the console terminal should display:

```
@P
```

If the RUN light flashes and the console displays:

```
@P
001002 (or 001000)
@
```

the error is most likely a defective front panel or improper set up of the jumpers on the front panel, inhibiting the panel-lock function.

TEST 16

The CPU should now be functional to the point that it is able to run the diagnostic contained in the M9312 diagnostic ROM. Return the keyswitch to the LOCAL position. The PDP-11/24 should halt and the console should print the ODT prompt.

Return the HALT/CONT/BOOT switch to the CONT position.

The M9312 diagnostic may now be started by typing '165006G' at the console. The RUN and PROC LEDs should light. The diagnostic takes approximately 6 seconds per 128K words of memory.

After the diagnostic has run, the console terminal should display:

```
@165006G ssssssss
165006
@
```

The 22-bit octal address represented by 'ssssssss' is the last address plus 2 found (to be working) during the memory diagnostic. It should conform to the expected memory size of the system. Refer to Paragraph 1.3.8 for details.

If the PDP-11/24 halts at any other address, it indicates a fault. The halt instructions contained in the M9312 are detailed in Table 9-3.

Certain errors cause the console to be unable to print. The state of the diagnostic may then be determined by examining the display register LEDs on the CPU module (M7133). They indicate the state of the diagnostic as detailed in Table 3-17.

Table 9-3 M9312 Halt Errors

Error	Displayed Address	Description
Bad Address	165146	This error is caused by trapping to location 4 at any time prior to executing the memory test on the first 4K of memory.
CPU Error	165150	This error indicates a failure with either the base instruction set or the EIS instruction set.
Memory Error	165552	This error indicates a memory system failure.
SLU Error	165706	This error indicates a data error in the console SLU.

TEST 17

At this point, the PDP-11/24 should be able to boot the XXDP+ diagnostic medium. Place the XXDP+ media in the drive and perform the correct bootstrap procedure. If the media boots, you may then proceed to run XXDP+ diagnostics. If not, the problem may be in the specific peripheral device or the DMA logic of the processor.

If the optional M7134 UNIBUS map module is installed, it may be removed and a bootstrap program toggled in. This may allow the PDP-11/24 to be bootstrapped. The toggle-in boots can be found in the *XXDP+ DEC/X11 Programming Card* (EK-0XXDP-MC).

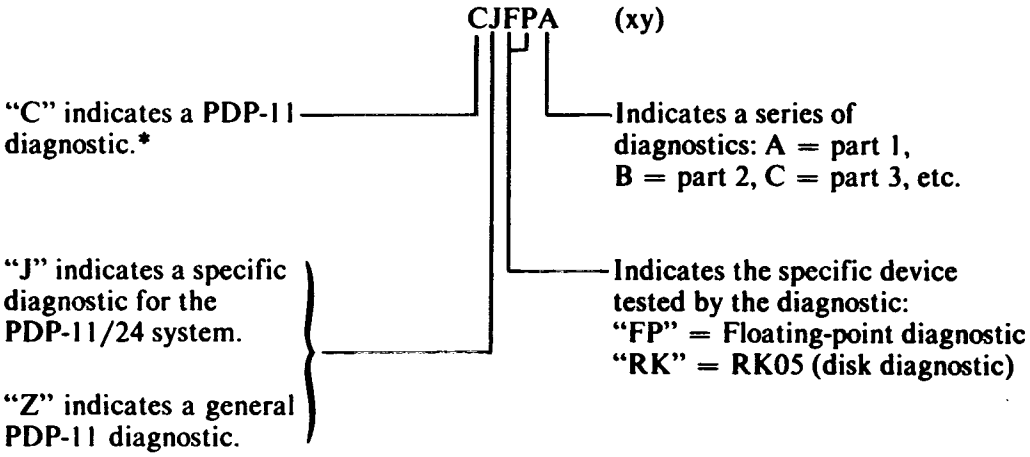
9.2 MAINDEC DIAGNOSTIC PROGRAMS

The MAINDEC diagnostics are external programs which are listed in Table 9-4. These programs are loaded and executed with the MAINDEC diagnostic package XXDP+. This package includes monitor programs, device driver programs and utility programs for the CPU and peripheral devices.

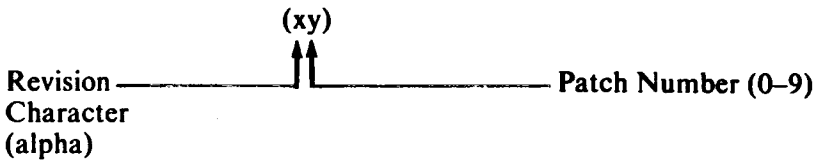
Diagnostics 1 through 7 should be executed in the sequence as listed to obtain the proper results. Each diagnostic test assumes the successful completion of the preceding test.

9.2.1 Diagnostic Designations

The designations assigned to the PDP-11 family of diagnostic programs are described as follows.



The "xy" designation that appears after the MAINDEC diagnostic listing on the table contains the following information.



* Not used as part of the diagnostic filename on diagnostic disk pack or on magnetic tape.

Table 9-4 PDP-11/24 MAINDEC Diagnostic Programs

MAINDEC Diagnostic	Operating Sequence	Title
CJKDB <xy>	1	PDP-11/24 CPU
CJKDA <xy>	2	KTF11-A Memory Management
CJKDF <xy>	3	PDP-11/24 Options
CZM9B <xy>	4	M9312, PDP-11/24, PDP-11/44 UBI Boot
CKKUA <xy>*	5	PDP-11/24, PDP-11/44 UBI Map
CZMSD <xy>	6	MS11-L/M Memory
CZMSP <xy>	6	MS11-L/M or -P Memory
DZKAQ <xy>	7	PDP-11 Power-Fail
CJKDC <xy>†	8	KEF11-A FPP Part 1
CJKDD <xy>†	9	KEF11-A FPP Part 2
CJKDH <xy>‡	10	KEF11-B CISP

*If KT24 map option is present.
†If KEF11-A floating-point processor option is present.
‡If KEF11-B commercial instruction set option is present.

9.2.2 Running Diagnostics

For a detailed explanation of how to run each diagnostic, refer to the appropriate diagnostic documentation.

The following paragraphs list each diagnostic, which diagnostics should be successfully completed before running the specified diagnostic, and the switch register bits and their use. The switch register referred to is a software switch register maintained at memory location 000 176_8.

9.2.2.1 CJKDB – PDP-11/24 CPU Diagnostic – No prerequisites.

Switch Register Bit	Octal Designation	Use
15	100000	Halt on error
13	020000	Inhibit error printout
01	000002	CIS chip set present
00	000001	Skip traps test

Bits ⟨14, 12:02⟩ are not used.

NOTE: Bit 15 and 13 only effect part 3 of the CPU diagnostic. Parts 1 and 2 always halt on an error.

9.2.2.2 CJKDA – KTF11-A Memory Management Diagnostic – Prerequisites are CJKDB and the PDP-11/24 Diagnostic ROM.

Switch Register Bit	Octal Designation	Use
15	100000	Halt on error
14	040000	Loop on test
13	020000	Inhibit error type outs
12	010000	Inhibit trace trap
11	004000	Inhibit subtest iteration
10	002000	Bell on error
09	001000	Loop on error
08	000400	Loop on test in bits ⟨07:00⟩
07:00	000XXX	Test number

9.2.2.3 CJKDF – 11/24 Option Diagnostic – Prerequisites are CJKDB and a turnaround jumper installed on SLU2.

Switch Register Bit	Octal Designation	Use
15	100000	Halt on error
14	040000	Scope loop
13	020000	Inhibit error type out
10	002000	Inhibit error flags test
09	001000	Loop on error
07	000200	Inhibit SLU2 wraparound test
06	000100	Inhibit LTC tests
05	000040	Inhibit all SLU tests
04	000020	Inhibit SLU1 testing
03	000010	Inhibit SLU2 testing

Bits ⟨12:11, 08, 02:00⟩ are not used.

9.2.2.4 CZM9B – M9312, 11/24, 11/44 UBI Boot Diagnostic

Switch Register Bit	Octal Designation	Use
15	100000	Halt on error
13	020000	Inhibit error type outs
10	002000	Bell on error

Bits ⟨12:11, 09:00⟩ are not used.

9.2.2.5 CKKUA - 11/24, 11/44 UBI Map Diagnostic - Prerequisites are KT24 Map Option (M7134), CJKDB and CJKDA.

Switch Register Bit	Octal Designation	Use
15	100000	Halt on error
14	040000	Loop on test
13	020000	Inhibit error type outs
12	010000	Inhibit trace trap
11	004000	UNIBUS memory trap
10	002000	Bell on error
09	001000	Loop on error
08	000400	Loop on test in bits (05:00)
07	000200	Inhibit multiple error type outs
06	000100	Unused
05:00	0000XX	Select UNIBUS memory tests

9.2.2.6 CZMAP - MS11L/P Memory Diagnostic - Prerequisites are CJKDB, CJKDA, and CKKUA.

Switch Register Bit	Octal Designation	Use
15	100000	Halt on error
14	040000	Loop on test
13	020000	Inhibit error type outs
12	010000	Inhibit relocation
11	004000	Quick verify
10	002000	Bell on error
09	001000	Loop on error
08	000400	Halt program
07	000200	Detailed error reports
06	000100	Inhibit configuration map
05	000040	Limit maximum errors per band
04	000020	Wide terminal (132 columns or more)
03	000010	Test mode
02	000004	Test mode
01	000002	Test mode
00	000001	Detect single bit errors

9.2.2.7 DZKAQ - PDP-11 Power-Fail Diagnostic

Switch Register Bit	Octal Designation	Use
15	100000	Set = Halt at end of test pass
14	040000	Set = Disable error type outs Clear = Enable error type outs

Bits (13:00) are not used.

9.2.2.8 CKJDC, CJKDD – KEF11-A FPP Diagnostic Part 1 and 2 – Prerequisites are CJKDB, CJKDA, and the KEF11-A chip set.

Switch Register Bit	Octal Designation	Use
15	100000	Halt on error
14	040000	Loop on current test
13	020000	Inhibit error type outs
12	010000	Inhibit T-bit trapping
11	004000	Inhibit iterations
10	002000	Ring bell on error
09	001000	Loop on error
08	000400	Loop on test in bits (06:00)
07	000200	Unused
06:00	000XXX	Test number

9.2.2.9 CJKDH – KEF11-B CISP Diagnostic – Prerequisites are CJKDB, CZMSD, CJKDF and the KEF11-B chip set.

Switch Register Bit	Octal Designation	Use
15	100000	Halt on error
14	040000	Scope loop
13	020000	Inhibit error typeout
12	010000	Unused
11	010000	Inhibit subtest iterations
05	000040	Test CIS control chip 9
04	000020	Test CIS control chip 8
03	000010	Test CIS control chip 7
02	000004	Test CIS control chip 6
01	000002	Test CIS control chip 5
00	000001	Test CIS control chip 4

Bits <10:06> are not used.

If bits (05:00) are not set, the diagnostic checks all six chips.

APPENDIX A
COMMUNICATIONS REGULATOR

The commmunications (comm) regulator (part no. 5413764) produces two voltages, +15 V and −15 V, that are used by the CPU and optional communication devices in the PDP-11/24. It is mounted in the H777UA-UB next to the MOS memory regulator. The following paragraphs contain functional and detailed descriptions of the comm regulator circuitry.

A.1 FUNCTIONAL DESCRIPTION

Figure A-1 is a functional block diagram of the comm regulator. For discussion purposes, the comm regulator is divided into two parts: the +15 V circuit and the −15 V circuit.

In the +15 V circuit, a buck regulator, the pass switch, filter and controller form a closed loop circuit that produces +15 V from an unregulated raw dc voltage (23–44 V). The crowbar circuit protects the +15 V load from excessively high voltage at the filter output due to a +15 V circuit malfunction. To protect the regulator the +15 V is internally current limited if the output is overloaded.

The −15 V circuit is a flyback regulator, operating from the same bulk voltage supply.

A.2 +15 V CIRCUIT DETAILED DESCRIPTION

The pass switch (Q2) is repeatedly turned on and off, generating a pulse train across D7 at the LC filter input (L1, C7, C8 and C9). When current flows through D5 and D6 (generated by the pulse width modulator E2), the pass switch is turned on and the raw dc voltage appears across D7. Diode D7 is forward biased when the pass switch is turned off, clamping the filter input to approximately −1 V. The output voltage of the filter is equal to the time averaged (dc) value of the voltage across D7.

The magnitude of the raw dc voltage and the conduction time of the pass switch determine the filter output voltage. The filter output voltage is fed back to the controller, E1, that regulates the filter output to +15 V. Regulation is accomplished by varying the conduction time of the pass switch. The major components of the controller are operational amplifier E1 and the 555 timer E2.

Amplifier E1 compares approximately one-third of the filter output to a 5.1 V zener reference and amplifies the difference. The amplifier output is an error voltage which increases (decreases) as the output decreases (increases).

The output of the controller is generated by the 555 timer, E2, and is synchronized to the H777 master clock signal which is a 0 to +12 V squarewave with a period of 50 μs.

When the clock signal, P1 pins 5 and 6, goes high, Q7 is momentarily turned on and the timer is first reset and then triggered. Once triggered, the timer output at E2 pin 3 is latched to +12 V producing a current in R20, D5 and D6, and the pass switch turns on. Triggering E2 also turns off its internal discharge transistor at E2 pin 7 and C8 charges towards +15 V along a fixed exponential curve. When the voltage at E2 pin 6 equals the error voltage at E2 pin 5, the timer is reset. (E2 pin 3 goes low, the pass switch turns off, and the voltage across D7 is clamped to approximately 0.6 V). The conduction time of the pass switch increases (decreases) as the error voltage at E2 pin 5 increases (decreases).

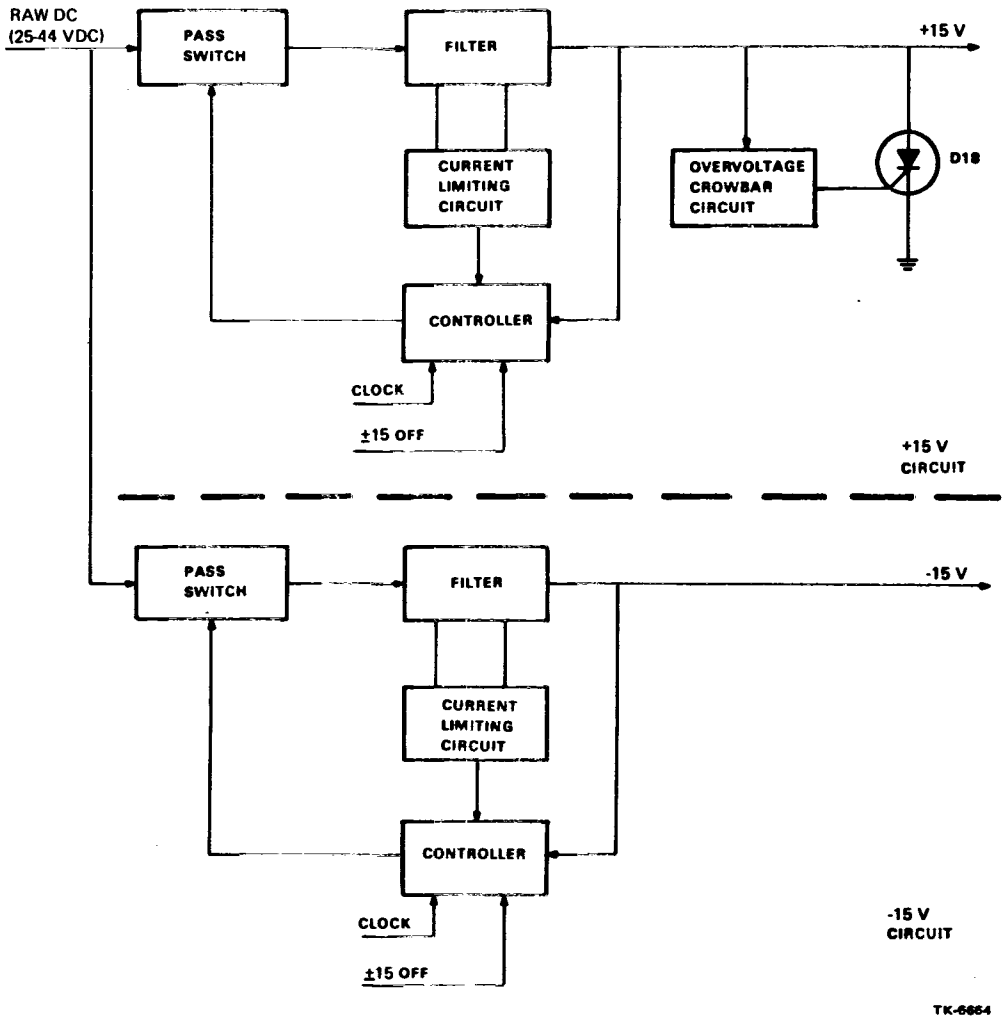


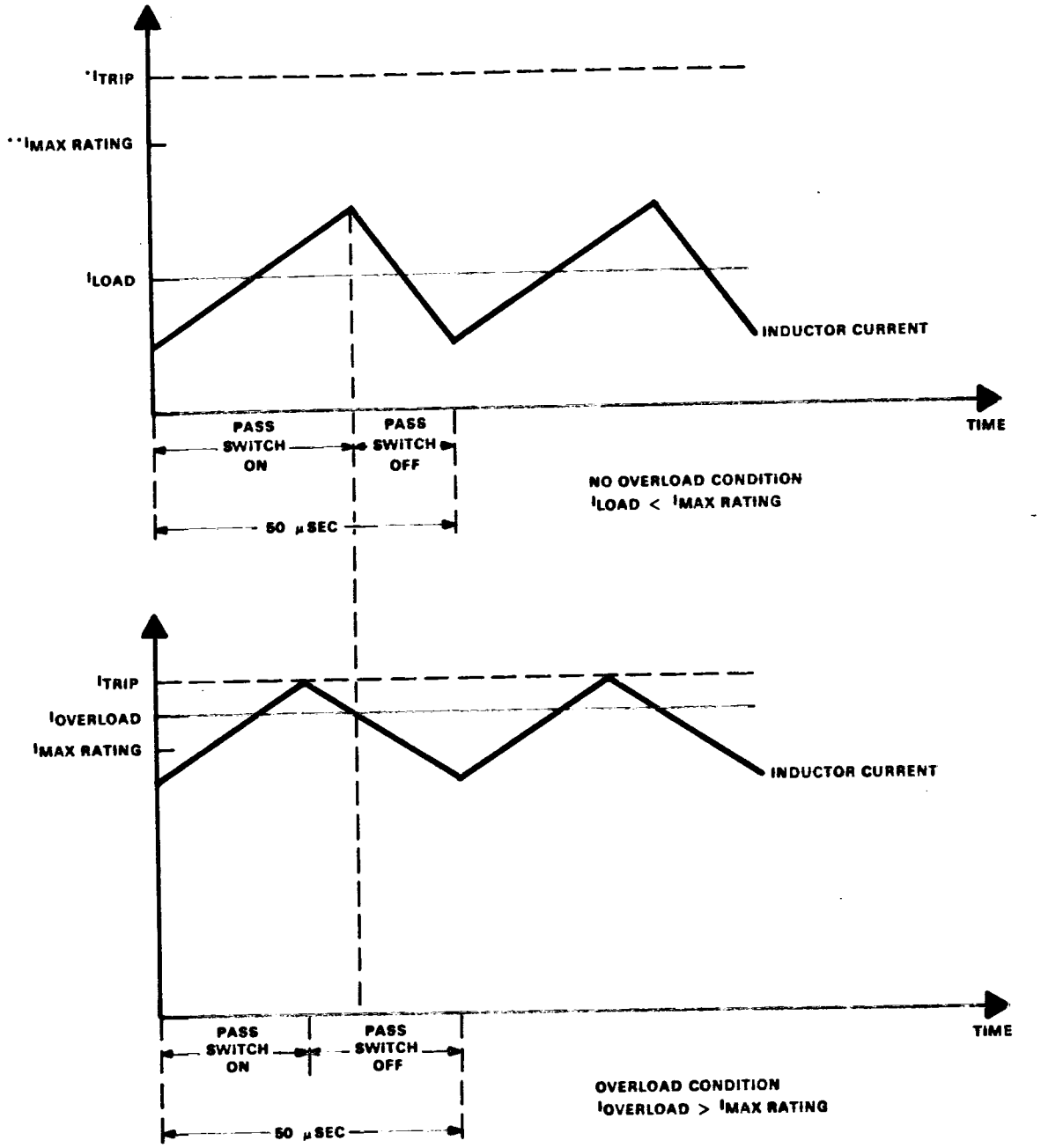
Figure A-1 Comm Regulator Block Diagram

Note that if the error voltage is very high, the timer is not reset until current limit is initiated and Q6 turns on. The timer does not react to a trigger input at E2 pin 2 until the trailing (rising) edge of the reset input at E2 pin 4. Therefore, when the clock goes high the timer is first reset (ensuring an end to the previous timer cycle) and then triggered (starting a new timer cycle).

NOTE

It is important that E2 be replaced only with an identical part.

Transistors Q4, Q5, and Q6 are the major components of the current limiting circuit that protects the +15 V circuitry from a current overload. The inductor current and the voltage drop across R8 increase when the pass switch is on and decrease when the pass switch is off. If the inductor current reaches approximately 4.7 A during pass switch on time, the voltage drop across R8 is sufficient to turn on Q4; subsequently, Q5 and Q6 also turn on. The error voltage at E2 pin 5 is clamped to approximately 0.6 V and is lower than the voltage at E2 pin 6, so the timer is reset. Q6 also shorts the drive current from E2 pin 3 (via R20) removing the drive from the regulator. The pass switch is then turned off limiting the peak inductor current to approximately 4.7 A. The current output to the load is limited to approximately 4 A and the output voltage decreases. Waveforms that apply to the +15 V current limiting circuit are shown in Figure A-2.



*ITRIP IS THE VALUE OF THE INDUCTOR CURRENT AT WHICH THE CURRENT LIMITING CIRCUIT TURNS OFF THE PASS SWITCH

**IMAX RATING IS THE MAXIMUM CURRENT THAT CAN BE SUPPLIED TO THE LOAD WITHOUT CAUSING THE PASS SWITCH TO TURN OFF PREMATURELY.

MA-0017

Figure A-2 Overcurrent Limiting Waveforms

Zener diode D14 and SCRs D12 and D13 are the major components of the overvoltage crowbar circuit which protects the +15 V load from an excessively high voltage at the filter output (i.e., raw dc appearing at the filter output due to a pass switch failure).

If the filter output becomes higher than approximately 17 V, D14 conducts and the voltage difference between the output voltage and the zener voltage (17 V) appears across R16. Current flows to the gate of D13 so that D13 and, subsequently, D12 are fired. Once D12 is fired, the output load current is bypassed to ground and the ouput voltage is clamped to approximately 1.6 V. D11 removes drive from the output by shunting the current from E2 pin 3 and R20.

SCR (D12) conducts until its anode current falls below approximately 75 mA for approximately 50 μs. This crowbar is a recycling type (nonlatching) that will allow the regulator to restart if the overvoltage condition was caused externally and is corrected.

A.3 −15 V CIRCUIT DETAILED DESCRIPTION

The switch, filter and controller form a closed loop circuit that produces a regulated −15 V from an unregulated raw dc voltage (23–44 V). The −15 V regulator is a flyback type utiliziing L2 to transfer the energy.

The switch (Q9) is repeatedly turned on and off, generating a pulse train across L2. When current flows into the base of Q11, the pass switch is turned on and the raw dc voltage appears across L2. During the ON or charge time, diode D24 is reverse biased, the inductor current increases and current to the load is provided by C27, C23 and C28. When the switch Q9 is turned off, D24 is forward biased and the inductor current flows into the load.

The voltage magnitude at the filter output is determined by the magnitude of the raw dc voltage and the conduction time of the switch.

The filter output voltage is fed back to the controller which regulates the filter output at −15 V by varying the conduction time of the switch. The major components of the controller are operational amplifier E4 and the 555 timer E3.

Amplifier E4 compares the filter ouput voltage to a −5.1 V reference generated across D19 and amplifies the difference. Zener diode D20 (V_Z = 5.6 V) acts as a level shifter. The control voltage at E3 pin 5 has a positive polarity, while the op amp output varies around the 0 V level. The control voltage increases (decreases) as the filter output voltage becomes less (more) negative.

The controller output is generated by E3 and is synchronized to the clock signal which is a 0 to +12 V squarewave with a period of 50 μs.

When the clock signal goes high, Q7 is momentarily turned on and the timer is first reset and then triggered. Once triggered, the timer output at E3 pin 3 is latched to +12 V producing a current in R38 which is bypassed to ground via Q12 during the time that the clock signal is high (Q12 turned on). Triggering E3 also turns off the internal discharge transistor at E3 pin 7, and C23 charges towards +15 V along a fixed exponential curve. When the voltage at E3 pin 6 equals the control voltage at E3 pin 5, the timer is reset. (E3 pin 3 goes low and C23 is clamped to approximately 0.6 V because the discharge transistor is on.) Current flows into the base of Q11, and the switch is turned on from the time the clock signal goes low until E3 is reset. The conduction time of the switch increases (decreases) as the control voltage increases (decreases). Note that if the control voltage is very high, the timer is not reset until Q7 is turned on. The timer does not react to a trigger input at E3 pin 2 until the trailing (rising) edge of the reset input at E3 pin 4. Therefore, when the clock goes high, the timer is first reset (ensuring an end to the previous timer cycle) and then triggered (starting a new timer cycle).

During each 50 μs clock cycle, the conduction time of the switch is between 0 and 25 μs; the exact time is dependent on the control voltage.

The current limiting circuitry is internal to the -15 V circuit and is designed primarily to limit the current in Q9 and L2. Transistor Q13 is the major component of the current limiting circuit which protects the -15 V circuitry. The inductor current and the voltage across R46 increase when switch Q9 is on and decrease when the switch is off. If the inductor current reaches approximately 8.5 A during switch on time, the voltage developed across R46 is sufficient to turn on Q13. The control voltage at E3 pin 5 is clamped to approximately 0.6 V and is lower than the voltage at E3 pin 6, so the timer is reset. The switch is turned off and the peak inductor current is limited to approximately 8.5 A. As the overload increases towards 7.5 A, the conduction time of the switch is decreased and the filter output voltage becomes less negative. Waveforms that apply to the -15 V current limiting circuit are shown in Figure A-2.

In the event an external overvoltage is applied to the output, D23 shunts the drive to the switch, stopping operation of the regulator until the output is less than approximately -17 V. D8 provides reverse voltage protection and load protection if D24 should fail.

APPENDIX B
FPP INSTRUCTION SET

Table B-1 contains the instruction set of the FPP. Some of the symbology may not be familiar, therefore, a brief description follows.

1. A bit in the FPS register, designated FD, determines whether single- or double-precision floating-point format is specified. If the flip-flop is cleared, single-precision is specified and is designated by F. If the flip-flop is set, doudle-precision is specified and is designated by D. Examples are NEGF, NEGD, and SUBD.

   NOTE

   Only the assembler or compiler differentiates between NEGF and NEGD or LDCID or LDCLD instructions. The floating-point does not differentiate between the instructions but depends upon the value of FD and FL as usually controlled by SETD, SETF, SETC, and SETI instructions (i.e., SETD, SETI → LDCID, SETD, SETL → LDCLD).

2. FPS bit designated FL, determines whether short-integer or long-integer format is specified. If the flip-flop is cleared, short-integer format is specified and is designated by I. If the flip-flop is set, long-integer format is specified and is designated by L. Examples are SETI and SETL.

3. Several convert-type instructions use the following symbology.

   $C_{IL,}FD$ – Convert integer long to floating double.
   $C_{FD,}$ – Convert floating double to integer long.
   $C_{F,D}$ or $D_{D,F}$ – Convert single-floating to double-floating or convert double-floating to single-floating.

4. UPLIM is defined as the largest possible number that can be represented in floating-point format. This number has an exponent of 377 (excess 200 notation) and a fraction of all 1s. Note the UPLIM is dependent on the format specified. LOLIM is defined as the smallest possible number that is not identical to 0. This number has an exponent of 001 and a fraction of all 0s except for the hidden bit.

5. The following conventions are used when referring to address locations.

   (xxxx) = the contents of the location specified by xxxx.
   ABBS (address) = absolute value of (address).
   EXP (address) = exponent of (address) in excess 200 notation.

6. Some of the octal codes listed in Table 6-4 are in the form of mathematical expressions.

Example 1: LDFPS Instruction

Mode 3, register 7 specified (F instruction format)

170100 + SRC
SRC field is equal to 37.
Basic op code is 170100.
SRC and basic op code are added to yield 170137.

Example 2: LDF Instruction

FAC2, mode 2, and register 6 specified (F1 instruction format).

172400 + FAC * 100 + FSRC

FAC = 2

2 * 100 = 200

172400 + 200 = 172600
FSRC is equal to 26.

172600 + 26 + 172626

7. AC + 1 means that the accumulator field (bits 6 and 7 in formats F1 and F3) is logically ORed with 01.

   Example:

   Accumulator field = bits 6 and 7 = AC2 = 10. AC + 1 = 11.

The information in Table 7-4 is expressed in symbolic notation to provide the reader with a quick reference to the function of each instruction.

Table B-1 FPP Instruction Set

Mnemonic	Instruction Description	Octal Code
ABSF FDST ABSD FDST	Absolute FDST ← minus (FDST) ⟨ 0; otherwise FDST (FDST) FC ← 0 FV ← 0 FX ← 1 if exp (FDST) = 0; otherwise FZ ← 0 FN ← 0	170600+FDST F2 Format
ADDF FSRC, FAC ADDD FRSR, FAC	Floating Add FAC ← (FAC) + (FSRC) if FAC + (FSRC) ⟨LOLIM; otherwise FAC ← 0 FC ← 0 FV ← 1 if FAC ⟩UPLIM; otherwise FV ← 0 FX ← if (FAC) = 0; otherwise FZ ← 0 FN ← 1 if (FAC) ⟨ 0; otherwise FN ← 0	172000+FAC*100+FSRC F1 Format
CLRF FDST CLRD FDST	Clear FDST ← 0 FC ← 0 FV ← 0 FZ ← 1 FN ← 0	170400+FDST F2 Format
CMPF FSRC, FAC CMPD FSRC, FAC	Floating Compare FC ← 0 FV ← 0 FZ ← 1 if (FSRC) = 0; otherwise FZ ← 0 FN ← 1 if (FSRC) ⟨ 0; otherwise FN ← 0	173400+FAC*100+FSRC F1 Format
CFCC	Copy Floating Condition Codes C ← FC V ← FV Z ← FZ N ← FN	170000 F5 Format

Table B-1 FPP Instruction Set (Cont)

Mnemonic	Instruction Description	Octal Code
DIVF FSRC, FAC DIVD FSRC,	Floating Divide AC ← (FAC)/(FSRC) if |(FAC)/(FSRC)| ⟩ LOLIM; otherwise FAC ← 0 FC ← 0 FV ← 1 if FAC ⟩ UPLIM; otherwise FV ← 0 FZ ← 1 if EXP (FAC) = 0; otherwise FZ ← 0 FN ← 1 if (FAC) ⟨ 0; otherwise FN ← 0	174400+FAC*100+FSRC F1 Format
LDF FSRC, FAC or LDD FSRC, FAC	Floating Load FAC ← (FSRC) FC ← 0 FV ← 0 FZ ← 1 if (FAC) = 0; otherwise FZ ← 0 FN ← 1 if (FAC) ⟨ 0; otherwise FN ← 0	172400+FAC*100+FSRC F1 Format
LDCDF FSRC, FAC LDCFD FSRC, FAC	Load Convert Double-to-Floating Floating-to-Double FAC ← CDF or CFD (FSRC) FC ← 0 FV ← 1 if FAC ⟩ UPLIM; otherwise FV ← 0 FV ← 0 FZ ← 1 if (FAC) = 0; otherwise FZ ← 0 FN ← 1 if (FAC) ⟨ 0; otherwise FN ← 0 If the current format is single-precision floating-point (FD = 0), the source is assumed to be a double-precision number and is converted to single-precision. If the floating-truncate bit is set, the number is truncated; otherwise, it is rounded. If the current format is double-precision (FD = 1), the source is assumed to be a single-precision number and loaded left-justified in the AC. The lower half of the AC is cleared.	177400+FAC*100+FSRC F1 Format F, D-single-precision to double-precision floating D, F-double-precision to single-precision floating

Table B-1 FPP Instruction Set (Cont)

Mnemonic	Instruction Description	Octal Code
LDCIF SRC, FAC LDCID SRC, FAC LDCLF SRC, FAC LDCLD SRC, FAC LDCIF = Single Integer to Single Float LCID = Single Integer to Double Float LDCLF = Long Integer to Single Float LDCLD = Long Integer to Double Float	Load and Convert from Integer Floating FAC ← CIL,FD (SRC) FC ← 0 FV ← 0 FZ ← 1 if (FAC) = 0; otherwise FZ ← 0 FN ← 1 if (FAC) ⟨ 0; otherwise CIL,FD specifies conversion from a 2's complement integer with precision I to L to a floating-point number of precision F or D. If integer flip-flop IL = 0, a 16-bit integer (I) is double specified, and if IL = 1, a 32-bit integer (L) is specified. If floating-point flip-flop FD = 0, a 32-bit floating-point number (F) is specified, and if FD = 1, a 64-bit floating-point number (D) is specified. If a 32-bit integer is specified and addressing mode 0 or immediate mode is used, the 16 bits of the source register are left justified, and the remaining 16 bits are zeroed before the conversion.	177000+FAC*100+SRC F3 Format
LDEXP SRC, FAC	Load Exponent FAC SIGN ← (AC SIGN) FAC EXP ← (SRC) + 200 only if ABS (SRC) ⟨ 177 FAC FRACTION ← (FAC FRACTION) FC ← 0 FV ← 1 if (SRC) ⟩ 177; otherwise FV ← 0 FZ ← 1 if EXP (FAC) = 0; otherwise FZ ← 0 FN ← 1 if (FAC) ⟨ 0; otherwise FN ← 0	176400+AC*100+SRC F3 Format
LDFPS SRC	Load FP11-A's Program Status Word FPS ← (SRC)	170100+SRC F4 Format

Table B-1 FPP Instruction Set (Cont)

Mnemonic	Instruction Description	Octal Code
MODF FSRC, FAC MODD FSRC, FAC	Floating Modulo FAC + 1 ← integer part of (FAC) * (FSRC) FAC ← fractional part of (FAC) * (FSRC) − (FAC + 1) if (FAC) * (FSRC) ⟩ LOLIM or FIU = 1; otherwise AC ← 0 FC ← 0 FV ← 1 if FAC ⟩ UPLIM; otherwise FV ← 0 FZ ← 1 if (FAC) = 0; otherwise FZ ← 0 FN ← 1 if (FAC) ⟨ 0; otherwise FN ← 0	171400+AC*100+FSRC F1 Format
MODF FSRC, FAC MODD FSRC, FAC (cont)	The product of FAC and FSRC is 48 bits in single-precision floating-point format or 59 bits in double-precision floating-point format. The integer part of the product |(FAC) * (FSRC)| is found and stored in FAC + 1. The fractional part is then obtained and stored in FAC. Note that multiplication by 10 can be done with zero error, allowing decimal digits to be stripped off with no loss in precision.	
MULF FSRC, FAC MULD FSRC, FAC	Floating Multiply FAC ← (FAC) * (FSRC) if (FAC) * (FSRC) ⟩ LOMLIM; otherwise FAC ← 0 FC ← 0 FV ← 1 if FAC ⟩ UPLIM; otherwise FV ← 0 FZ ← 1 if (FAC) = 0; otherwise FZ ← 0 FN ← 1 if (FAC) ⟨ 0; otherwise FN ← 0	17100+FAC*100FSRC F1 Format

Table B-1 FPP Instruction Set (Cont)

Mnemonic	Instruction Description	Octal Code
NEGF FDST NEGD FDST	Negate FDST minus (FDST) if EXP (FDST)≠0; otherwise FDST ← 0 FC ← 0 FV ← 0 FZ ← 1 if EXP (FDST) = 0; otherwise FZ ← 0 FN ← 1 if (FDST) < 0; otherwise FN ← 0	170700+FDST F2 Format
SETD	Set Floating Double Mode FD ← 1	170011 F5 Format
SETF	Set Floating Mode FD ← 0	170001 F5 Format
SETI	Set Integer Mode FL ← 0	170002 F5 Format
SETL	Set Long-Integer Mode FL ← 1	170012 F5 Format
STF FAC, FDST STD FAC, FDST	Floating Store FDST ← (FAC) FC ← FC FV ← FV FZ ← FZ FN ← FN	17400+FAC*100+FDST F1 Format

Table B-1 FPP Instruction Set (Cont)

Mnemonic	Instruction Description	Octal Code
STCFD FAC, FDST STCDF FAC, FDST	Store Convert from Floating-to-Double or Double-to-Floating FDST ← CFD or CDF (FAC) FC ← 0 FV ← 1 if FAC UPLIM; otherwise FV ← 0 FZ ← 1 if (FAC) = 0; otherwise FZ ← 0 FN ← 1 if (FAC) ⟨ 0; otherwise FN ← 0 The STCFD instruction is the opposite of the LDCDF instruction; thus, if the current format is single-precision floating-point (FD = 0), the source is assumed to be a single-precision number and is converted to a double-precision. If the floating truncate bit is set, the number is truncated; otherwise, it is rounded. If the current format is double-precision (FD = 1), the source is assumed to be double-precision number and loaded left-justified in the FAC. The lower half of the FAC is cleared.	176000+FAC*100+FDST F1 Format F, D-single-precision to double-precision floating D, F-double precision to single-precision floating
STCFI FAC, DST STCFL FAC, DST STCDI FAC, DST STCDL FAC, DST	Store Convert from Floating-to-Integer Destination receives converted FAC if the resulting integer number can be represented in 16 bits (short integer) or 32 bits (long integer). Otherwise, destination is zeroed and C-bit is set.	175400+FAC*100+DST F3 Format

Table B-1 FPP Instruction Set (Cont)

Mnemonic	Instruction Description	Octal Code
STCFI = Single to Single Integer STCFL = Single Float to Long Integer STCDI = Double Float to Single Integer STCDL = Double Float to Long Integer	FV ← 0 FZ ← 1 if (DST) = 0; otherwise FZ ← 0 FN ← 1 if (DST) ⟨ 0; otherwise FN ← 0 C ← FC V ← FV Z ← FZ N ← FN When the conversion is to long integer (32 bits) and address mode is specified, only the most significant 16 bits are stored in the destination register.	
STEXP FAC, DST	Store Exponent DST ← FAC EXPONENT -200_8 FC ← 0 FV ← 0 FZ ← 1 if (DST) = 0; otherwise FZ ← 0 FN ← 1 if (DST) ⟨ 0; otherwise FN ← 0 C ← FC V ← FV Z ← FZ N ← FN	175000+FAC*100+DST
STFPS DST	Store FP11-A's Program Status Words DST ← (FPS)	170200+DST F4 Format
STST DST	Store FP11-A's Status DST ← (FEC) DST + 2 ← (FEA) if not mode 0 or not immediate mode	170300+DST F4 Format

Table B-1 FPP Instruction Set (Cont)

Mnemonic	Instruction Description	Octal Code
SUBF FSRC, FAC SUBD FSRC, FAC	Floating Subtract FAC ← (FAC) − (FSRC) if |(FAC) − (FSRC)| ⟩ LOLIM; otherwise FAC ← 0 FC ← 0 FV ← 1 if AC UPLIM; otherwise FV ← 0 FZ ← 1 if (FAC ⟨ 0; otherwise FZ ← 0 FN ← 1 if (FAC) ⟨ 0; otherwise FN ← 0	173000+FAC*100+FSRC F1 Format
TSTF FDST TSTD FDST	Test Floating FC ← 0 FV ← 0 FZ ← 1 if EXP (FDST) = 0; otherwise FZ ← 0 FN ← 1 if (FDST) ⟨ 0; otherwise FN ← 0	170500+FDST F2 Format

APPENDIX C
PDP-11/24 BACKPLANE ASSIGNMENTS

C.1 MODULE UTILIZATION

Figure C-1 shows the module utilization of the PDP-11/24 processor backplane. The following constraints apply to the use of the PDP-11/24 processor backplane.

1. M9312 bootstrap/terminator module or UNIBUS cable must only be installed in slot 9.
2. NPR (DMA) options must only be installed in slots 7 or 8.
3. Non-NPR options must only be installed in slots 3, 4, 5, 6, or 9.
4. All empty slots, except slot 2, must have grant cards. Slots 7 and 8 must have a double-height grant card (G7273) in rows C and D.

   Slots 3, 4, 5, 6, and 9 may use a single-height grant card (G727A or G7270) in row D or a double-height grant card in rows C and D.

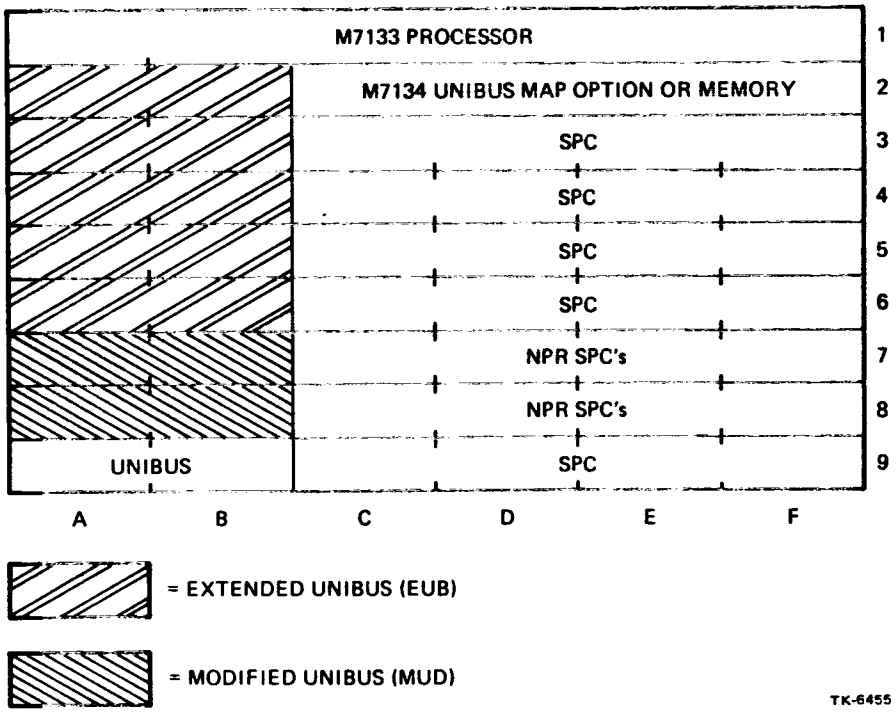


Figure C-1 PDP-11/24 Module Utilization

C.2 STANDARD, MODIFIED AND EXTENDED BACKPLANE LOCATIONS

Figure C-2 shows the pin designations of the standard, modified and extended UNIBUS connectors. The modified and extended UNIBUS differ from the standard UNIBUS in that certain pins have been redesignated. Some ground connections, bus grant signals, and the NPR signal have been removed from the modified and extended UNIBUS and have been redesignated with battery backup voltage pins for MOS memory, parity signal pins, several reserved pins, and test point pins.

Dual-height modules that are standard UNIBUS compatible must not be placed in the modified or extended UNIBUS sections.

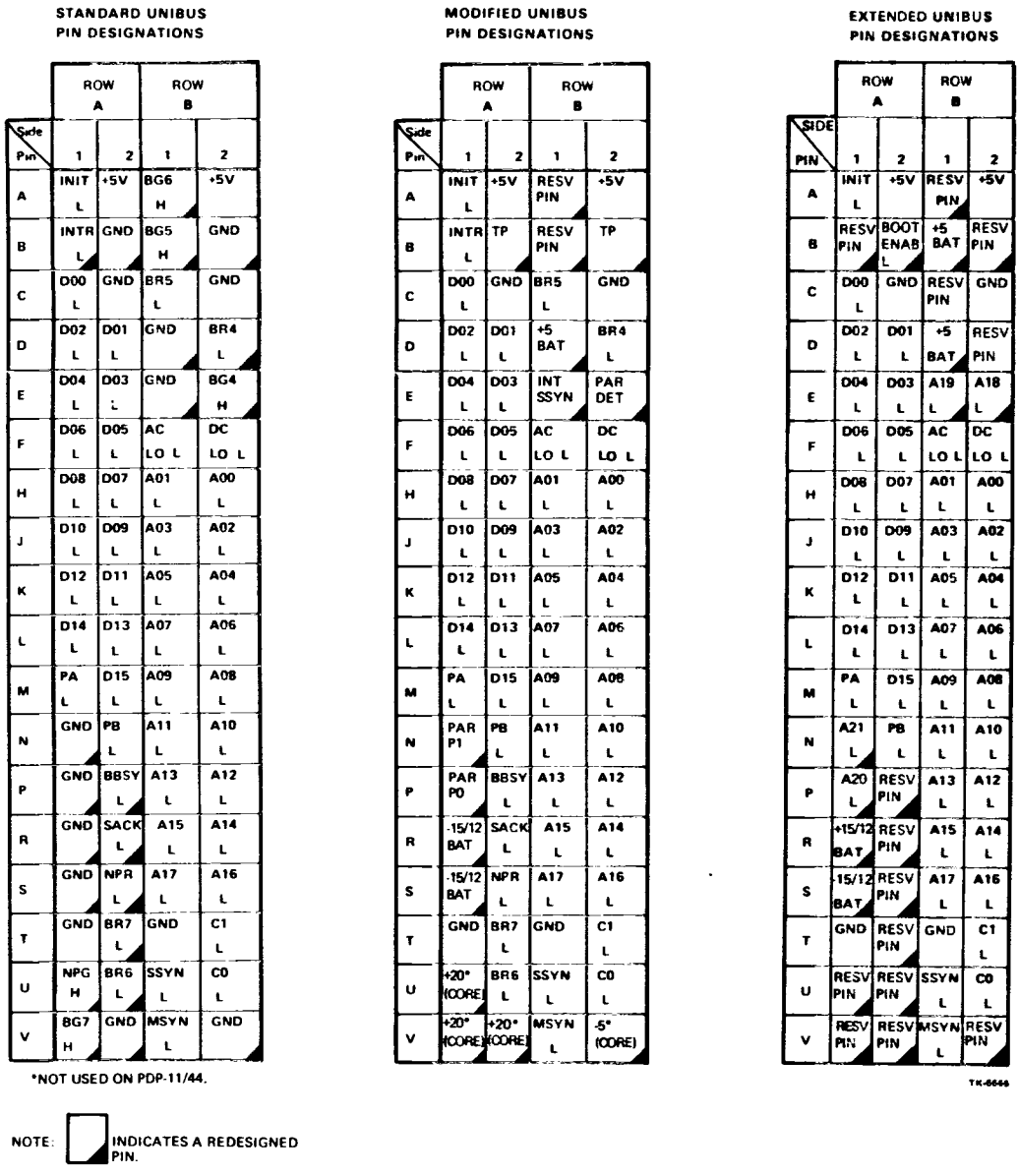


Figure C-2 Standard, Modified and Extended Backplane Pin Assignments

C.3 SPC BACKPLANE LOCATIONS

The small peripheral control sections (Figure C-1) collectively contain all the UNIBUS lines as well as power voltages (+5 V, +15 V, −15 V). These sections can be used by hex-height or quad-height modules containing the control logic for peripheral devices. Figure C-3 shows the pin designations for the SPC connectors.

	ROW C		ROW D		ROW E		ROW F	
SIDE / PIN	1	2	1	2	1	2	1	2
A	NPG (IN)	+5V	TP	+5V	GND A	+5V	ABG OUT	+5V
B	NPG (OUT)	-15V	TP	-15V	ASSYN IN H	-15V	ABG IN	-15V
C	PA L	GND	A SEL 6	GND	A12 L	GND	SSYN L	GND
D	LTC	D15 L	A OUT LOW	BR7 L	A17 L	A15 L	BBSY L	FO1 N1
E	TP	D14 L	A SEL 4	BR6 L	MSYN L	A16 L	FO1 V2	D02 L
F	TP	D13 L	A SEL 0	BR5 L	A02 L	C1 L	D05 L	D06 L
H	D11 L	D12 L	A IN	BR4 L	A01 L	A00 L	D07 L	A INT ENB B
J	A INT B	D10 L	A SEL 2	A BR OUT	SSYN L	C0 L	NPR L	GND A
K	TP	D09 L	A OUT	BG7 SO	A14 L	A13 L	D08 L	A INT B
L	A INT ENBB	D08 L	INIT L	BG7 OUT	A11 L	TP	D03 L	FO1 L2
M	TP	D07 L	AINT ENBA	BG6 SO	AIN	AOUT HIGH	INTR L	FO1 M2
N	DC LO	D04 L	A INT A	BG6 OUT	A OUT LOW	A08 L	FO1 N1	D04 L
P	HALT REQ	D05 L	TP	BG5 S0	A10 L	A07 L	ABR OUT	FO1 P2
R	HALT GRT	D01 L	TP	BG5 OUT	A09 L	ASEL 4	FO1 L2	FO1 N1
S	PB L	D00 L	TP	BG4 S0	ASEL 6	ASEL 0	FO1 M2	FO1 P2
T	GND	D03 L	GND	BG4 OUT	GND	ASEL 2	GND	SACK L
U	+15	D02 L	TP	ABG IN	A06 L	A04 L	A INT A	ABR OUT
V	AC LO	D06 L	ASSYN IN H	ABG OUT	A05 L	A03 L	A INT ENB A	FO1 FO1

TK-4410

Figure C-3 SPC Backplane Pin Assignments

C.4 NPG JUMPER LEAD ROUTING

The NPG line is the UNIBUS grant line for devices that perform data transfers without processor intervention. Continuity of the NPG line is provided by wire jumpers on the backplane with the exception of slots 1, 2, 7 and 8 (slot 1 holds the processor module and slot 2 holds the KT24 map option or memory). Slots 7 and 8 must have a G7273 double-height grant card in rows C and D when an NPR device is not installed. When an NPR device is installed in a slot other than slot 7 or 8, the corresponding jumper wire from pin CA1 to pin CB1 of that slot must be removed. The routing of the NPG signal through the backplane is shown in Figure C-4.

NOTE

If an NPR device is removed from a slot a double-height grant card (G7273) must be installed or the jumper wire from CA1 to CB1 must be reconnected.

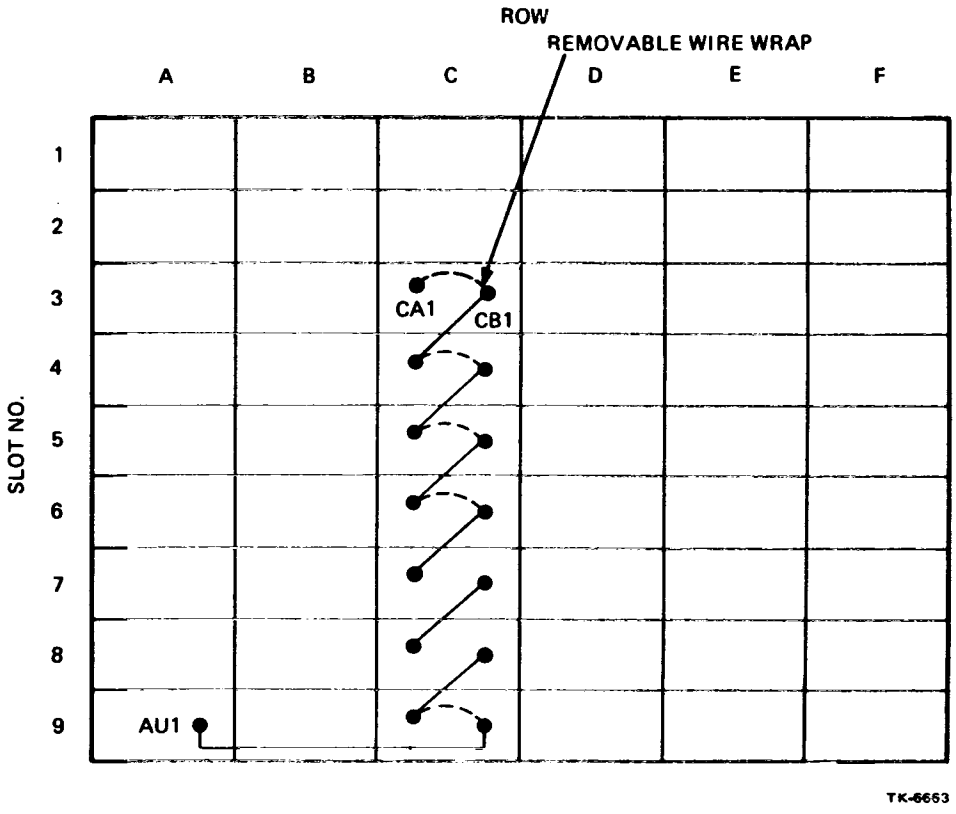


Figure C-4 NPG Jumper Leads Routing

APPENDIX D
PDP-11/24 CPU MODIFICATIONS

D.1 INTRODUCTION

The relayout of the PDP-11/24 processor module, (see Figure D-1), was a Value Engineering project intended to decrease the manufacturing cost of the product through the following methods: eliminating the ECO wires; adding gate array technology; and modifying some components for machine insertion.

Specifically, the relayout involved the replacement of 17 IC's with two gate array chips and repositioning the majority of components on the board. This was accomplished without altering the form, fit or function of the module.

The logic that was taken into the two gate arrays consists of the processor timing logic (timing PLA, counter, buffers and latches), fast data in register, interrupt vector drivers, UART data buffer, line time clock service logic and miscellaneous latches and buffers. See Figures D-2 through D-15 for a more detailed view of the schematics.

Due to these changes in the circuit schematics, the pages that some of the signal names are generated from changed; therefore, in looking through the technical manual, the signal name will be the same, but the K numbers (or page numbers) in front of the signals may have altered. All internal logic in the gate arrays are shown in Sections D.3 and D.5.

D.2 SWITCHES/JUMPERS LOCATION CHANGES FOR M7133-YA

See Table D-1.

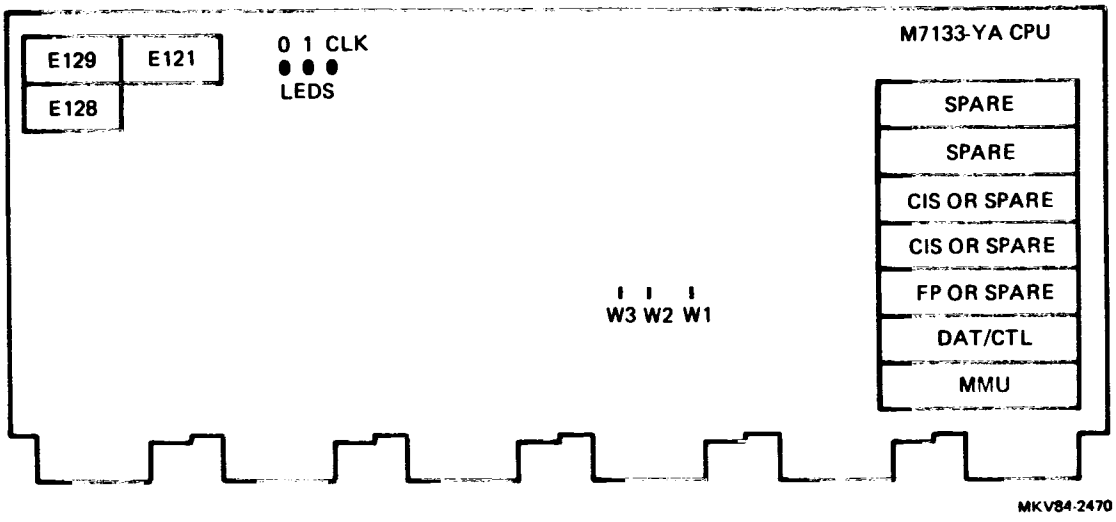


Figure D-1 M7133-YA Unit Assembly

Table D-1 M7133-YA CPU Module Jumper/Switch Selections

W1 Selects boot address when boot on power-up is enabled
IN = 165000_8 (octal)
OUT = 173000_8 (octal)

W2 Controls power fail, power-up action
IN = Boot on power-up from power fail
OUT = Power up via location 24_8

Provided the optional battery back up unit is installed and functioning

W3 Controls kernal halt enable
IN = Allows halt instruction to be executed in kernal mode. In all other modes, a halt instruction traps to 10_8

OUT = All halt instructions trap to 10_8 (except during power fail)

E121 Selects Baud Rates 1 and 2

Baud Rate 1 Baud Rate 2	SW-5 SW-1	SW-6 SW-2	SW-7 SW-3	SW-8 SW-4
50	on	on	on	on
75	on	on	on	off
110	on	on	off	on
134.5	on	on	off	off
150	on	off	on	on
200	on	off	on	off
300	on	off	off	on
600	on	off	off	off
1200	off	on	on	on
1800	off	on	on	off
2000	off	on	off	on
2400	off	on	off	off
3600	off	off	on	on
4800	off	off	on	off
9600	off	off	off	on
19200	off	off	off	off

Note: ON = Closed
OFF = Open

Table D-1 M7133-YA CPU Module Jumper/Switch Selections (Cont)

E128 Selects parity of SLUs

SLU1	SLU2	
SW-3	SW-1	ON = Parity detection enabled OFF = Parity detection disabled

E128 SW-5 Determines the number of stop bits in SLU1
ON = ONE stop bit
OFF = TWO stop bits
Manufacturing tests usually default to one stop bit.

E128 SW-6 No connection

E128 SW-7 Enables Field Service maintenance mode in SLU1
ON = Turn around enabled
OFF = Turn around disabled
This switch affects SLU1 only. Normal operation is off.

E128 SW-8 Determines 18 bit vs 22 bit UNIBUS addressing
ON = No effect on UBXFER
OFF = Forces UBXFER L

E129 SW-1 to SW-6 Selects SLU2 Receive and Transmit Baud Rates

SLU2 Transmit Select SLU2 Receive Select	SW-4 SW-1	SW-5 SW-2	SW-6 SW-3
Selects Baud Rate 1	on	off	off
Selects Baud Rate 2	off	on	off
Fixed 19.2K Baud for transmit and receive	off	off	on

E129 SW-7 to SW-10 Selects SLU1 Receive and Transmit Baud Rates

SLU1 Transmit Select SLU1 Receive Select	SW-9 SW-8	SW-10 SW-7
Selects Baud Rate 1	on	off
Selects Baud Rate 2	off	on

D.3 INTERNAL GATE ARRAY LOGIC (DC369B, DC370B)

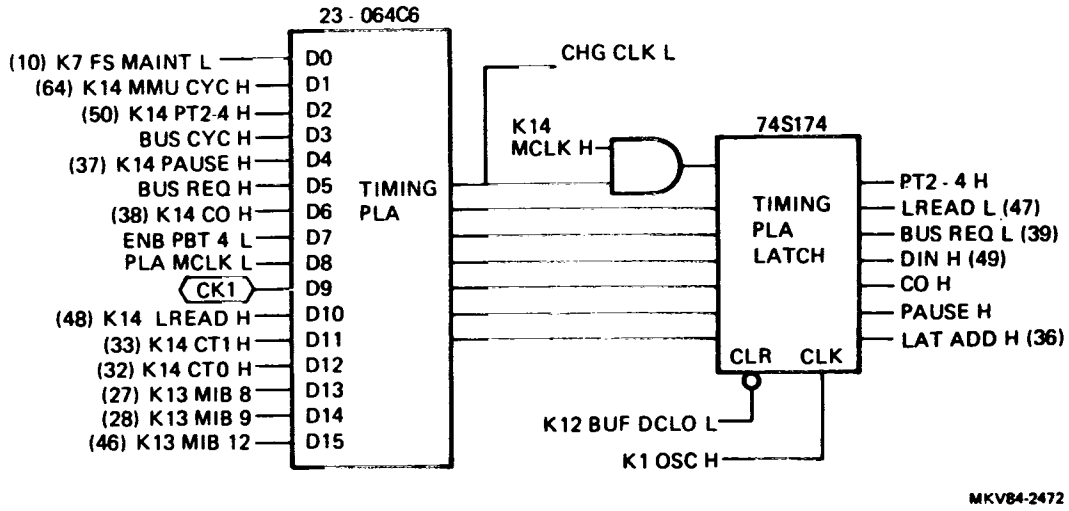


Figure D-2 PLA Control Signals (DC307B)

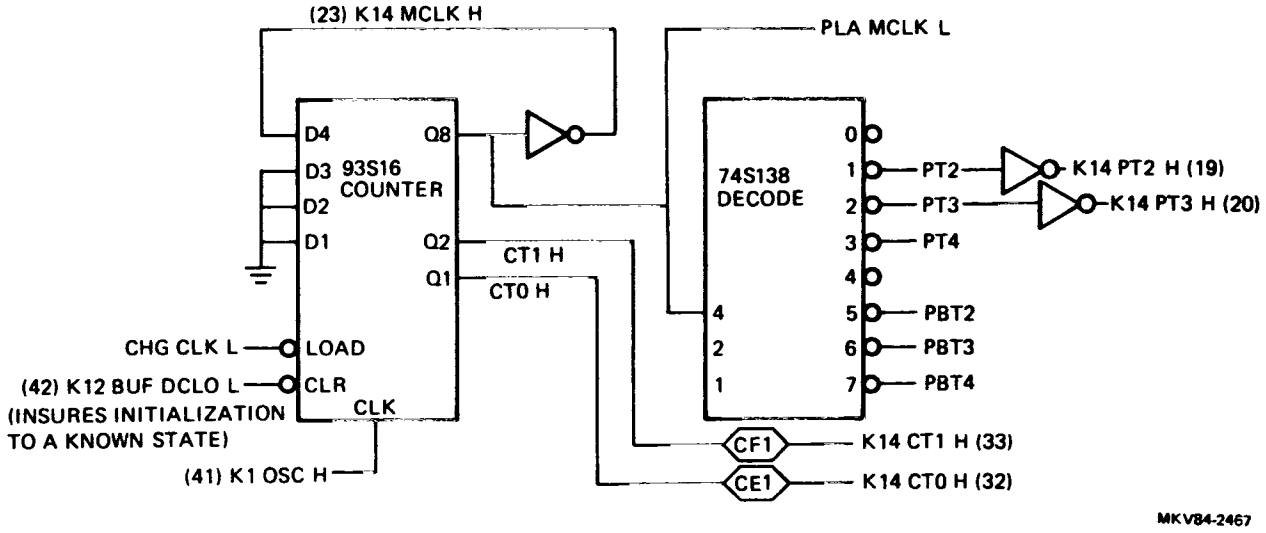


Figure D-3 Phase Time and Phase Bar Time Generation (DC370B)

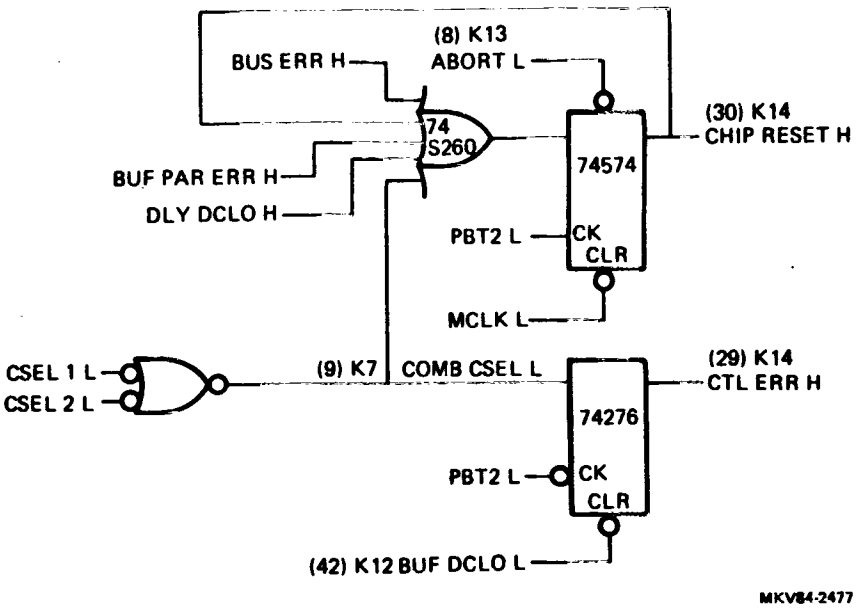


Figure D-4 PDP-11/24 Chip Set Control Logic (DC370B)

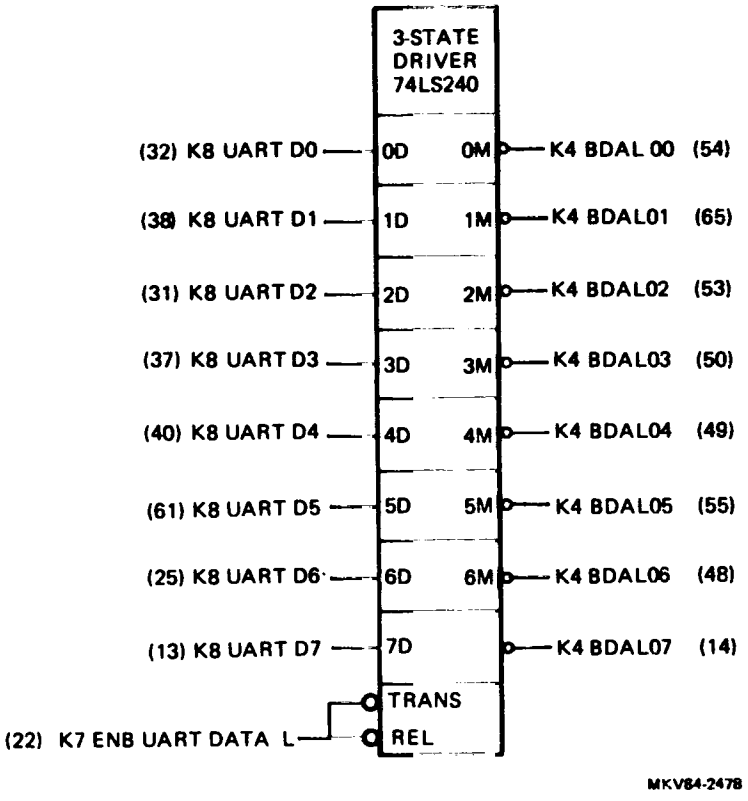


Figure D-5 Serial Line Interface (DC369B)

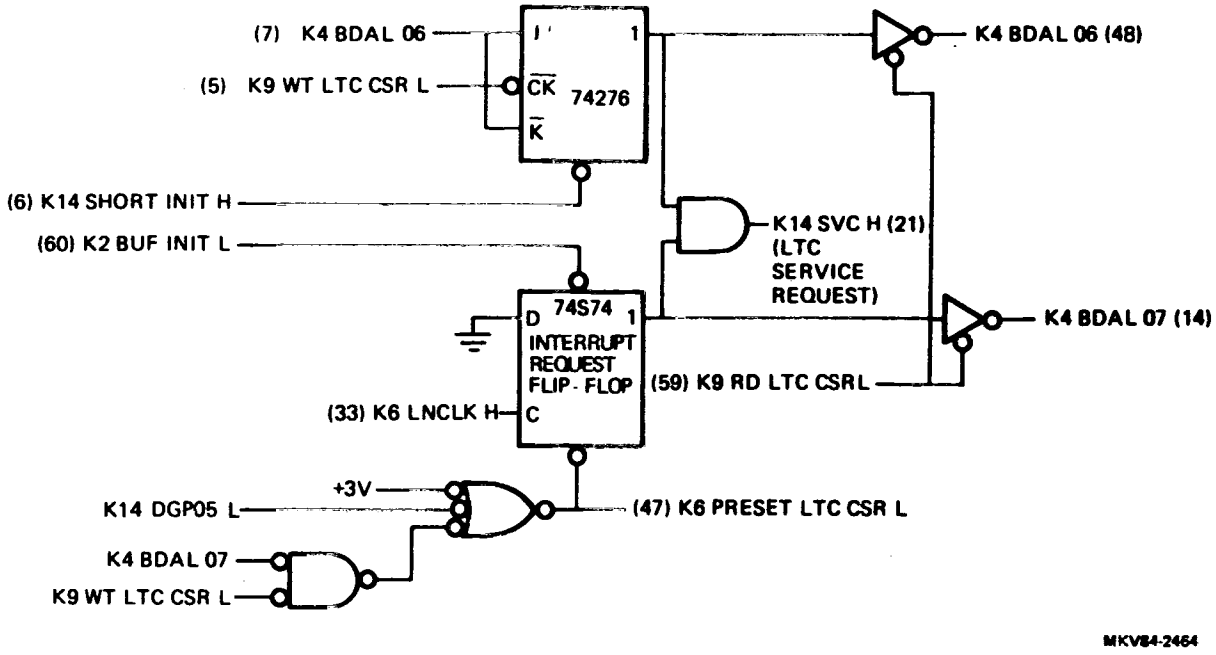


Figure D-6 Line Time Clock (DC369B)

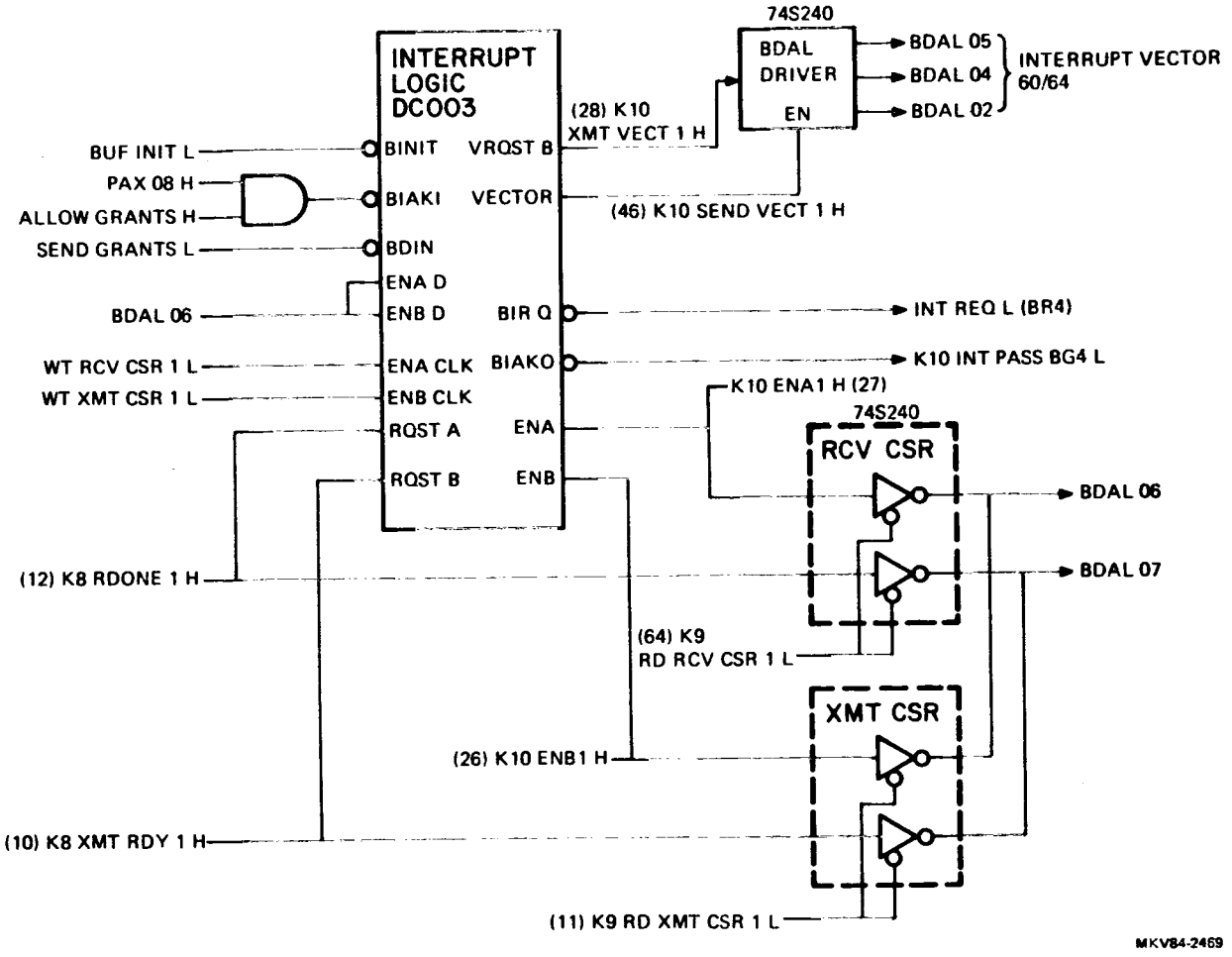


Figure D-7 SLU1 Interrupt Logic (DC369B)

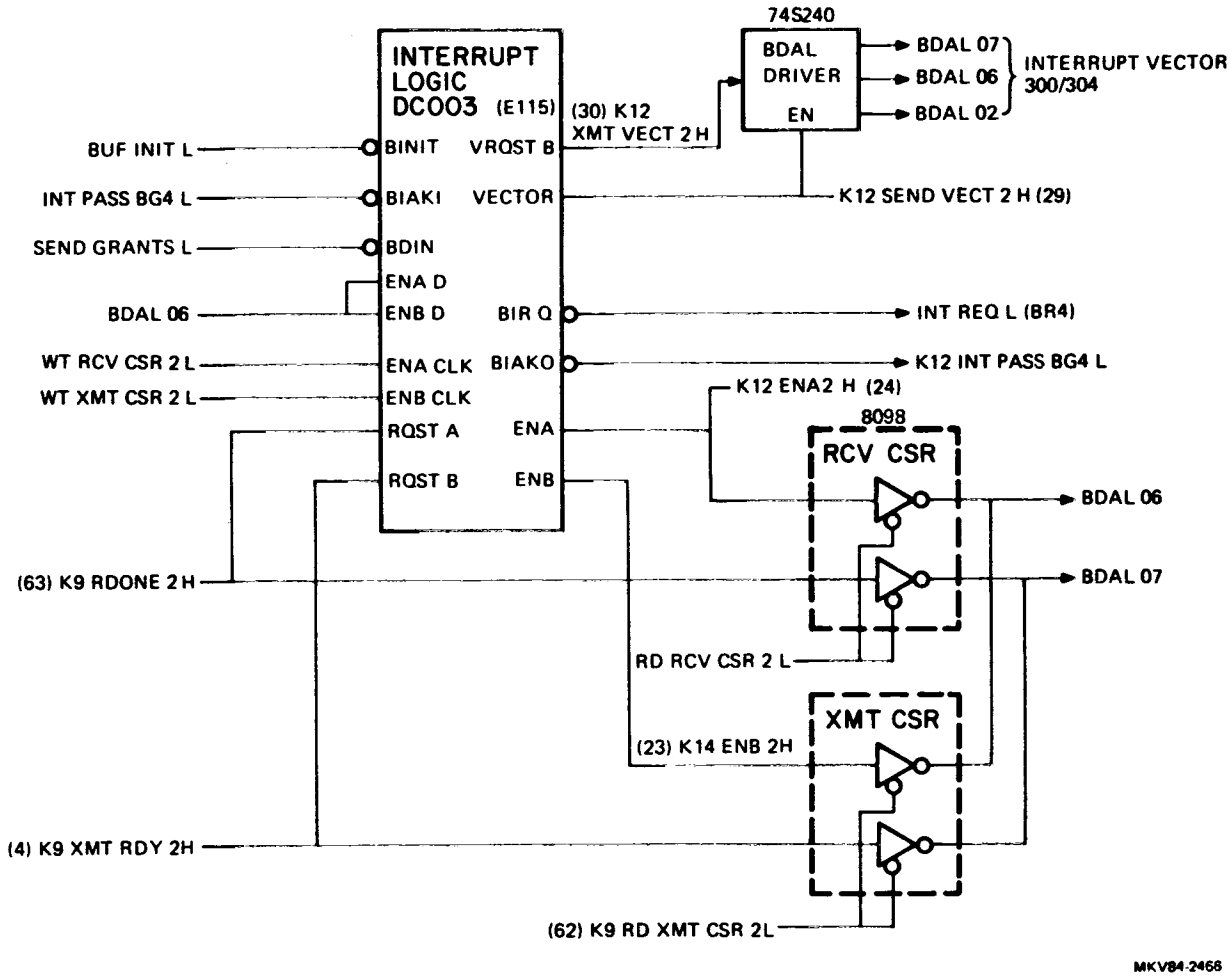


Figure D-8 SLU2 Interrupt Logic (DC369B)

D.4 GATE ARRAY PINOUT (DC369B, DC370B)

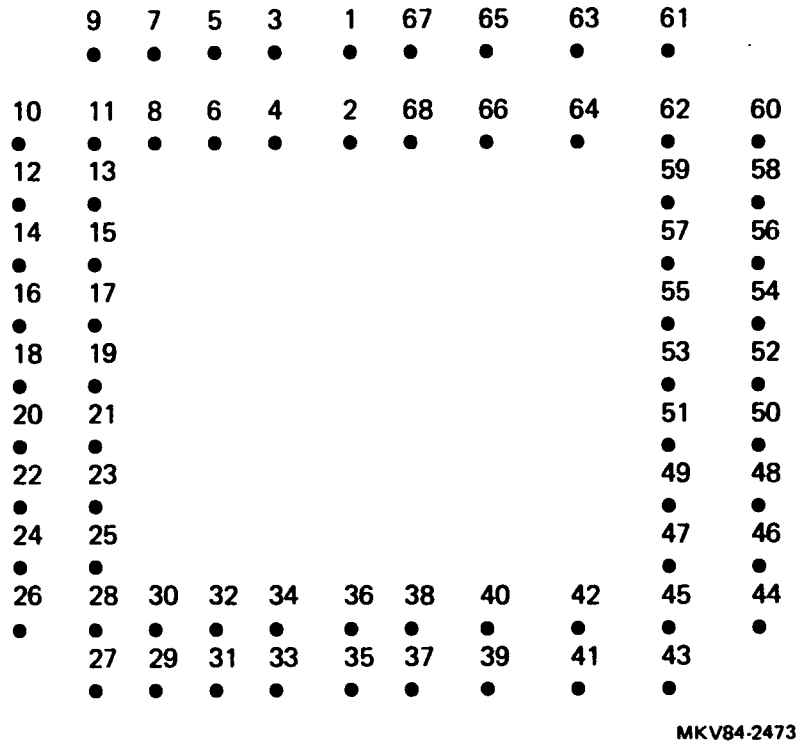


Figure D-9 DC369B, DC370B Pin Out

D.5 ADDITIONAL GATE ARRAY LOGIC (DC369B, DC370B)

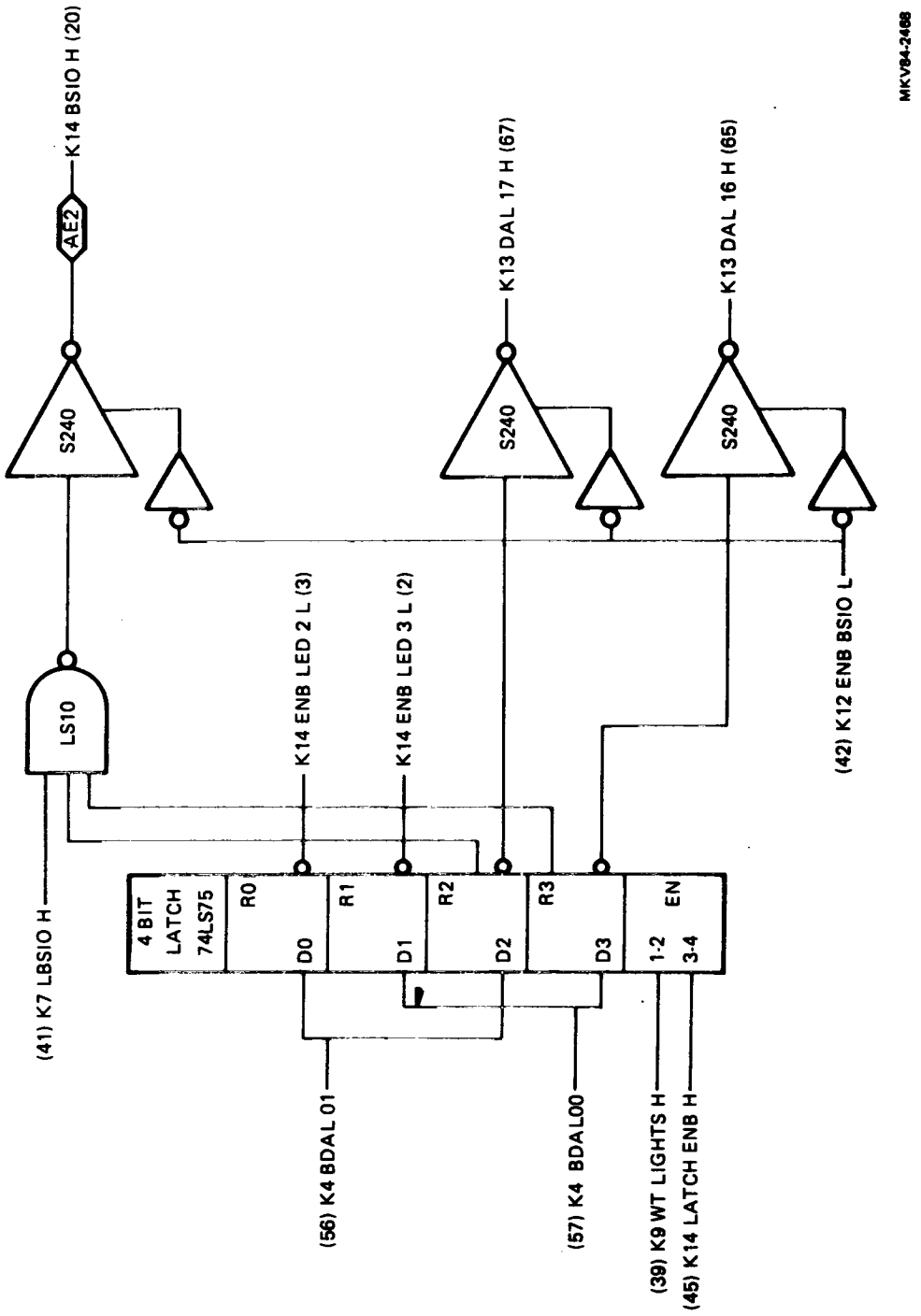


Figure D-10 Latch Address and Write Lights Logic (DC369B)

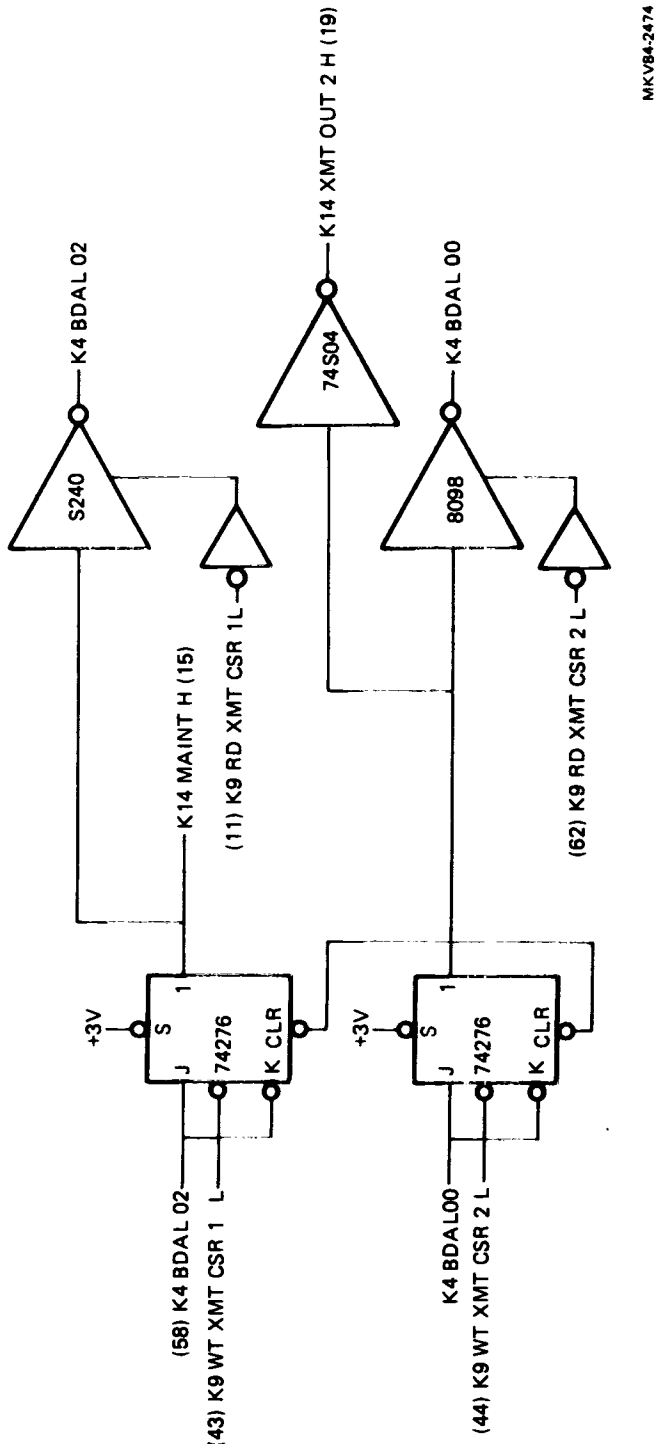


Figure D-11 SLU1 Control and Status (DC369B)

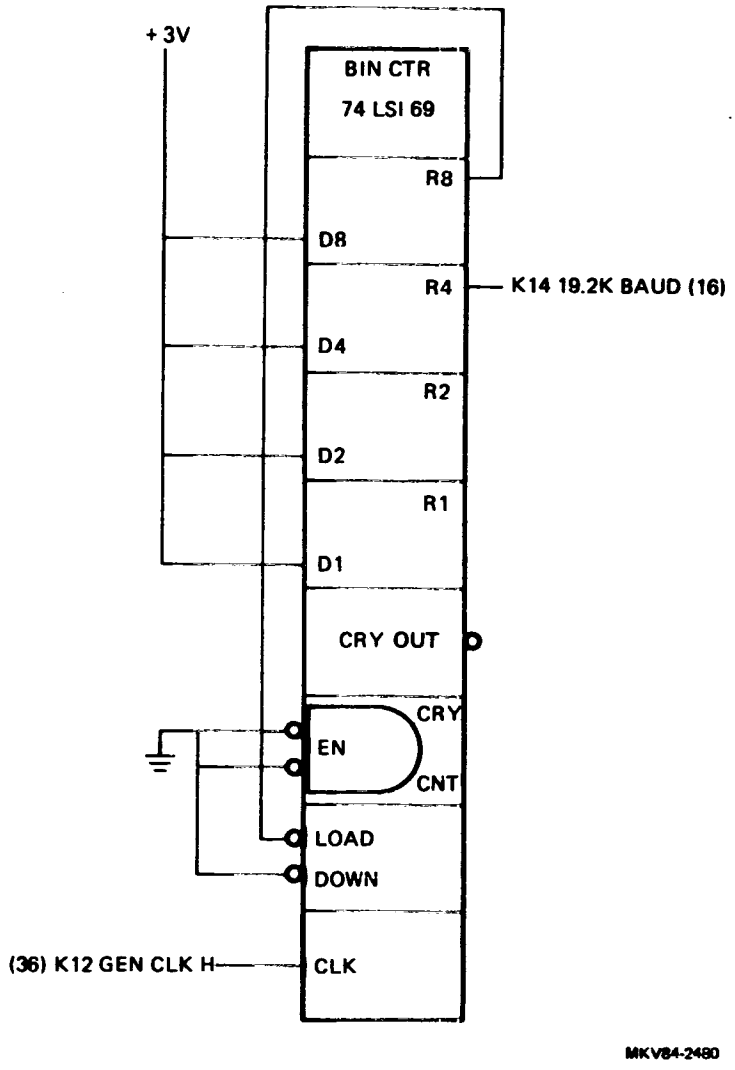


Figure D-12 Binary Counter (DC369B)

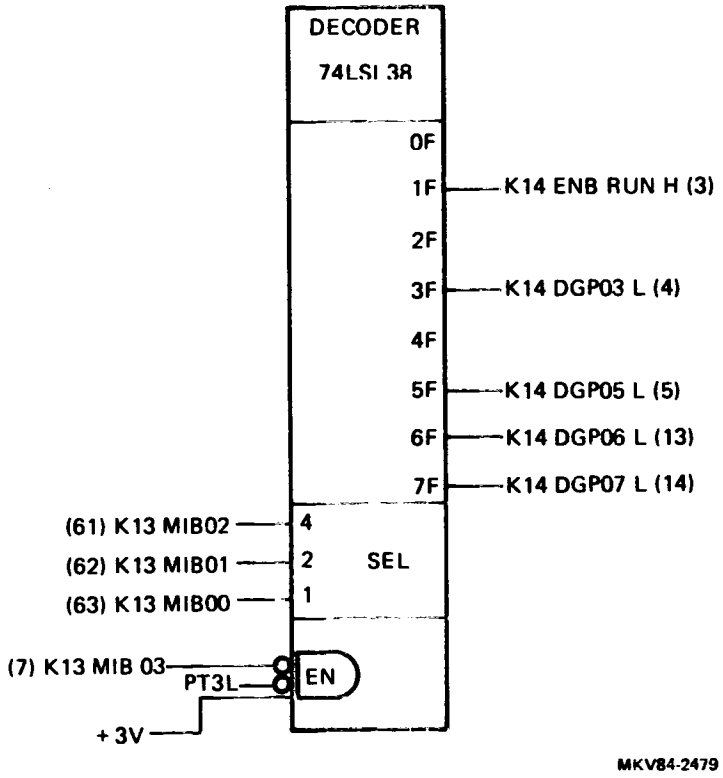


Figure D-13 General Purpose Output Decoder (DC370B)

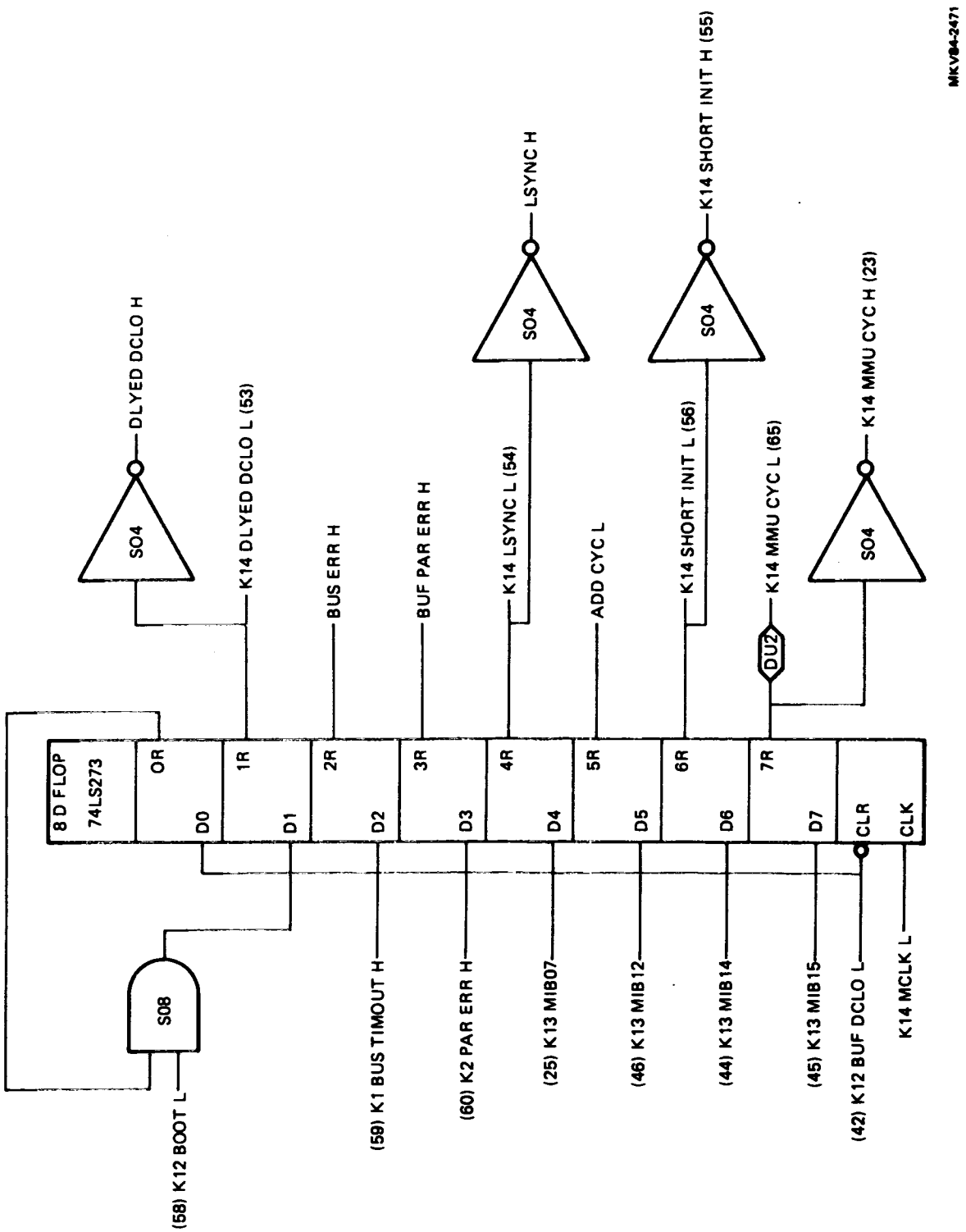


Figure D-14 Timing Control Signals (DC370B)

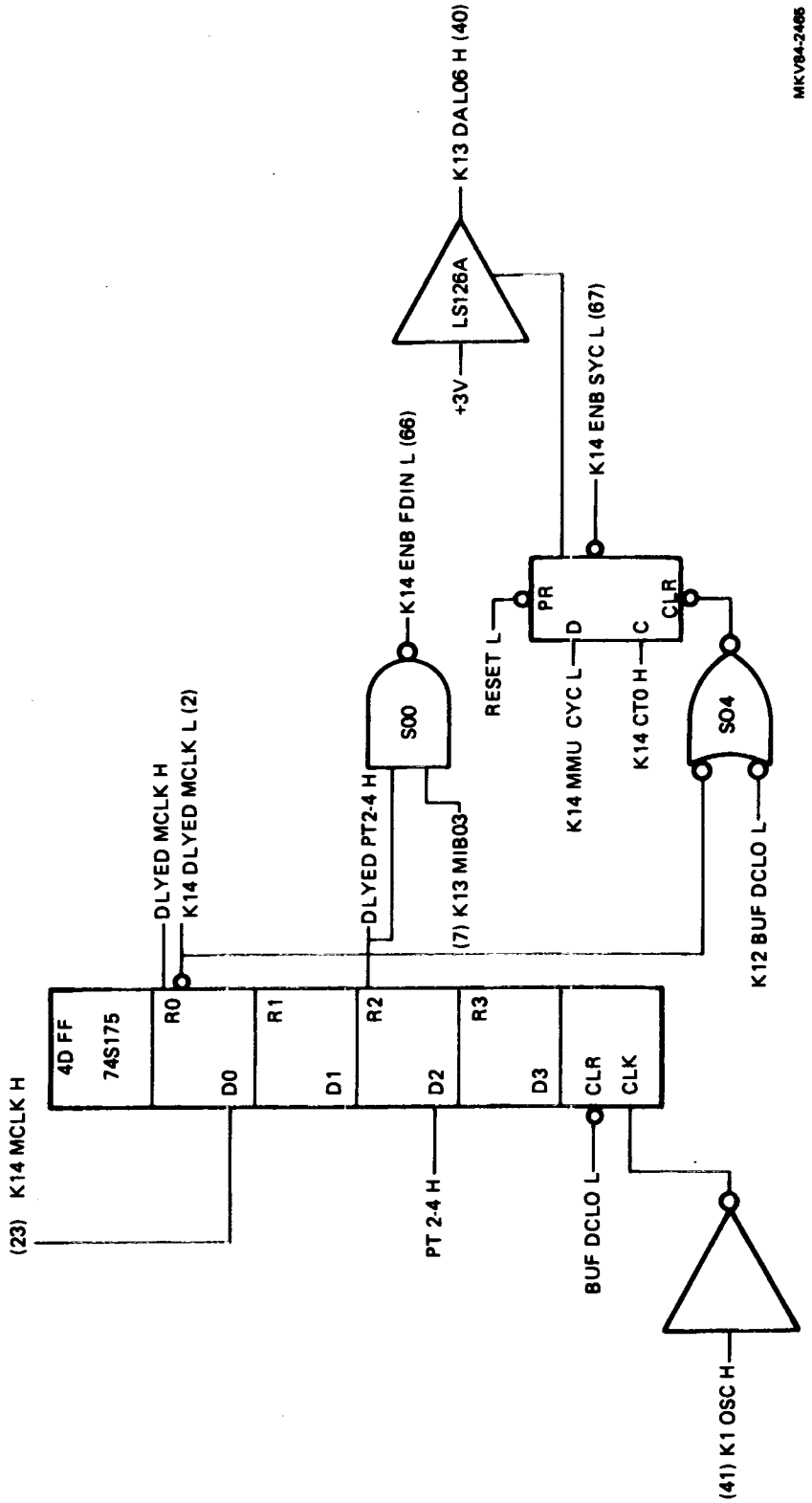


Figure D-15 Fast Data In and Service Logic (DC370B)