EK-M9312-TM-001

# M9312
# bootstrap/terminator
# module technical
# manual

digital equipment corporation • maynard, massachusetts

1st Edition, July 1978



Printed in U.S.A.

**This document was set on DIGITAL's DECset-8000 computerized typesetting system.**

# CONTENTS

# CONTENTS (Cont)

# FIGURES



# TABLES

# TABLES (Cont)

# CHAPTER 1
# INTRODUCTION

## 1.1 GENERAL DESCRIPTION

The M9312 Bootstrap/Terminator Module (Figure 1-1) contains a complete set of Unibus termination resistors along with 512 words of read only memory (ROM) that can be used for diagnostic routines, the console emulator routine, and bootstrap programs. Twelve jumpers (W-1 through W-12) are provided to allow compatibility with any Unibus PDP-11 system. Paragraph 1.6 outlines the use of these jumpers. Five sockets on the M9312 allow the user to interchange ROMs, enabling the module to be used with any Unibus PDP-11 system and boot any peripheral device by simply adding or changing ROMs. One socket is solely used for a diagnostic ROM (PDP-11/60 and 11/70 systems) or a ROM which contains the console emulator routine and diagnostics for all other PDP-11 systems. The other four sockets accept ROMs which contain bootstrap programs. One or two bootstrap programs may be contained in a particular ROM; however, some devices may require two or more ROMs to contain their particular bootstrap programs. ROM placement is outlined in Appendix B.

Diagnostics, bootstrap programs, and the console emulator routine are all selectable through the Address Offset Switch Bank on the M9312 (Paragraph 2.9). Appendix C shows the necessary switch configurations and addresses for various M9312 routines. These switch settings and addresses depend on the particular socket the ROMs are placed in. M9312 routines may be initiated in the following ways:

1. *External boot switch.* This switch is connected to the module via Faston tabs TP1 and TP2. The Address Offset Switch Bank is used to select various M9312 routines.

2. *System power-up.* This feature is enabled or disabled via switch S1-2. Again the Address Offset Switch Bank is used to select various M9312 routines.

3. *Programmer console load address and start sequence.* The programmer loads the starting address of a particular routine found in Appendix C.

**NOTE**

**When the programmer console load address and start sequence is used to start the console emulator routine, the Address Offset Switch Bank setting determines whether or not diagnostics are run.**

### 1.1.1 Scope

This manual is designed to provide DIGITAL Field Service and customer maintenance personnel with sufficient installation and operation information to install and maintain the M9312 Bootstrap/Terminator Module.

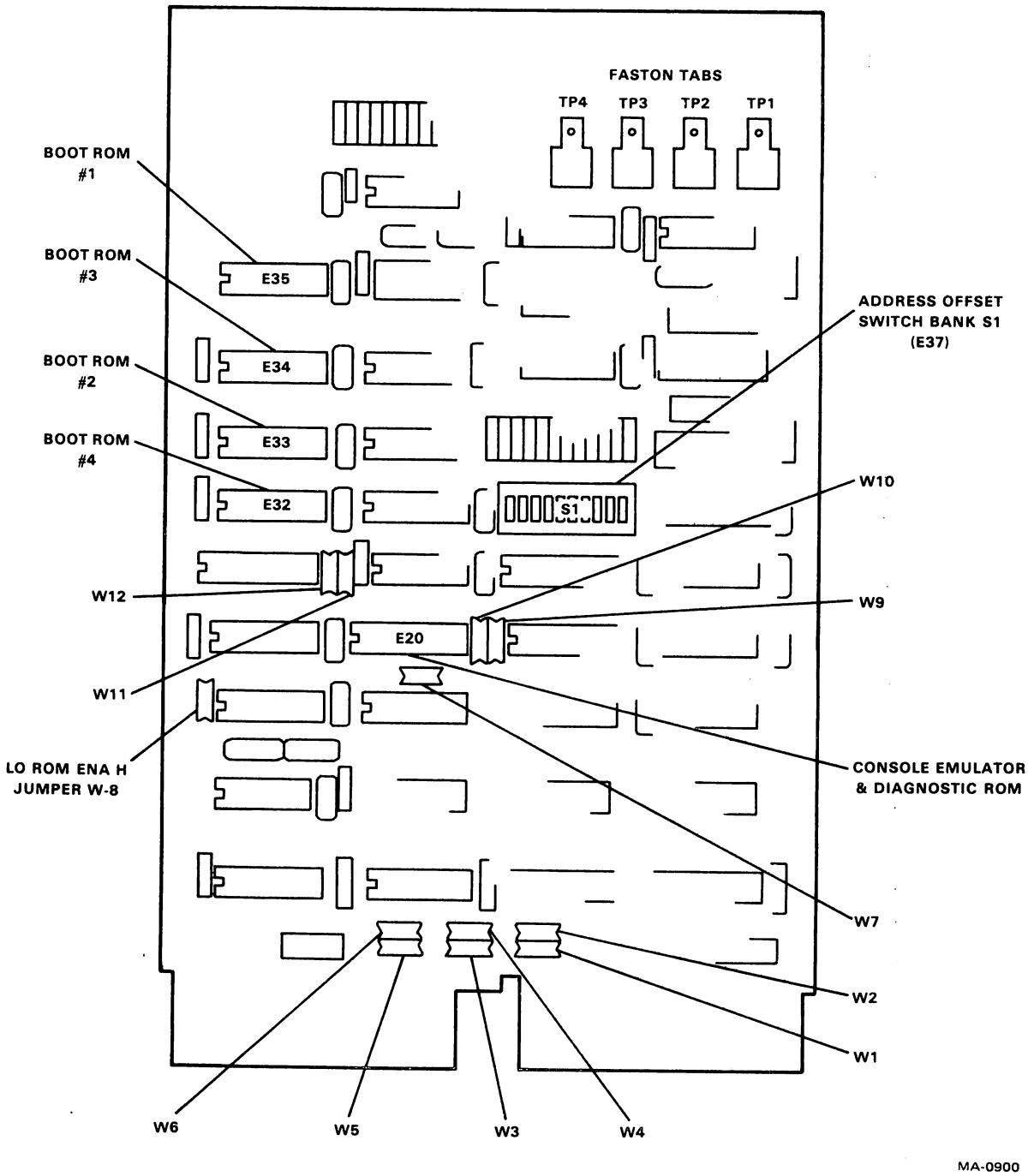



Figure 1-1 M9312 Bootstrap/Terminator Module

### 1.1.2 Related Documentation

Table 1-1 lists related documentation which supplements the information in this manual.

**Table 1-1 Related Documentation**

| Title | Document Number | Media* |
|---|---|---|
| PDP-11 Processor Handbooks | | Hard copy |
| PDP-11/34 User's Guide | EK-11034-UG-001 | Hard copy |
| PDP-11/04 User's Guide | EK-1104-OP-002 | Hard copy |
| Drawing Director | B-DD-M9312-00 | Microfiche |

*Hard copy documents can be ordered from: Digital Equipment Corporation, 444 Whitney Street, Northboro, MA 01532, Attention: Communication Services (NR2/M15), Customer Services Section.

For information concerning microfiche libraries, contact: Digital Equipment Corporation, Micropublishing Group, PK3-2/T12, 129 Parker Street, Maynard, MA 01754.

## 1.2 DEFINITION OF TERMS

Bootstrap Program

A bootstrap program is any program which loads another (usually larger) program into computer memory from a peripheral device.

Bootstrap

Bootstrap and bootstrap programs are used interchangeably.

Boot

Initiate execution of a bootstrap program.

## 1.3 PHYSICAL DESCRIPTION

The M9312 is a double-height extended module [21.6 × 14 cm (8-1/2 × 5-1/2 in)] which plugs into the A and B terminator slots on the PDP-11 backplane. External connections are made via four Faston tabs (TP1, TP2, TP3, TP4) provided at the handle end of the module.

## 1.4 ELECTRICAL SPECIFICATIONS

| | |
|---|---|
| Power Consumption | +5 Vdc ± 5 percent at 1.5 A typical |
| Electrical Interfaces | The Unibus interface is standard using 8837 and 8640 receivers, 8881 drivers, and 8641 transceivers. |

### 1.4.1 External Electrical Interfaces

The external interface consists of four Faston tabs (TP1, TP2, TP3, and TP4) provided at the handle end of the module. These inputs were designed to operate from either a mechanical switch or a TTL output (standard or open collector). Operation from TTL outputs is restricted to circuits inside the standard DIGITAL enclosure having the same logic reference as the M9312. With mechanical switches or TTL outputs, returns (TP2 and TP3) must be used. All inputs have overvoltage protection for up to ±16 V continuous; remote switch operation may require additional protection. When remote switch operation is used, the switch should be electrically isolated from the remote device. Both signal inputs (TP1 and TP4) have filtering and do not recognize an active input (below threshold voltage) until the end of a 10 to 18 μs delay, with any interruptions resetting the delay. The threshold range is between +0.45 and +0.75 Vdc to logic reference. The following is a description of each input.

TP1 Boot signal input. For positive or negative inputs, represents a 1000-ohm load in series with a forward biased diode to logic reference, and a 20,000-ohm pull-up. Switch bounce noise must not exceed 70 ms. Triggering is disabled for a maximum of 8 ms, starting when the M9312 power supply voltage rises from less than 3.0 V to greater than 4.5 V.

TP2 Boot signal return. Represents a 1000-ohm load to logic reference.

TP3 Power-up boot enable return. Represents a 1000-ohm load to logic reference.

TP4 Power-up boot enable input. For positive or negative inputs, represents a 1000-ohm load in series with a forward-biased diode to logic reference, and a 20,000-ohm pull-up. This input must remain either a logical 1 or 0 for the entire power-up sequence.

### 1.4.2 Electrical Prerequisites

Power and Ground Pinouts +5 Vdc: Pins AA2, BA2
GND: Pins AC2, AT1, BC2, BT1

(Refer to Table 1-2 for pin assignments.)

**Table 1-2 M9312 Pin Assignments**

| Pin | Signal | Pin | Signal |
|---|---|---|---|
| AA1 | BUS INIT L | BA1 | SPARE |
| AA2 | POWER (+5 V) | BA2 | POWER (+5 V) |
| AB1 | BUS INTR L | BB1 | SPARE |
| AB2 | TEST POINT | BB2 | TEST POINT |
| AC1 | BUS D00 L | BC1 | BUS BR5 L |
| AC2 | GROUND | BC2 | GROUND |
| AD1 | BUS D02 L | BD1 | *W-6 IN: BUF VECTOR L |
| AD2 | BUS D01 L | | W-6 OUT: BAT BACKUP +5 V |
| AE1 | BUS D04 L | BD2 | BUS BR4 L |
| AE2 | BUS D03 L | BE1 | INT* SSYN |
| AF1 | BUS D06 L | BE2 | PAR: DET |
| AF2 | BUS D05 L | BF1 | BUS AC LO L |
| AH1 | BUS D08 L | BF2 | BUS DC LO L |
| AH2 | BUS D07 L | BH1 | BUS A01 |
| AJ1 | BUS D10 L | BH2 | BUS A00 L |
| AJ2 | BUS D09 L | BJ1 | BUS A03 L |
| AK1 | BUS D12 L | BJ2 | BUS A02 L |
| AK2 | BUS D11 L | BK1 | BUS A05 L |
| AL1 | BUS D14 L | BK2 | BUS A04 L |
| AL2 | BUS D13 L | BL1 | BUS A07 L |
| AM1 | BUS PA L | BL2 | BUS A06 L |
| AM2 | BUS D15 L | BM1 | BUS A09 L |
| AN1 | P1 | BM2 | BUS A08 L |
| AN2 | BUS PB L | BN1 | BUS A11 L |
| AP1 | P0 | BN2 | BUS A10 L |

*W-6 must be out for operation in a modified Unibus slot.

Table 1-2 M9312 Pin Assignments (Cont)

| Pin | Signal | Pin | Signal |
|---|---|---|---|
| AP2 | BUS BBSY L | BP1 | BUS A13 L |
| AR1 | BAT BACKUP +15 V | BP2 | BUS A12 L |
| AR2 | BUS SACK L | BR1 | BUS A15 L |
| AS1 | BAT BACKUP -15 V | BR2 | BUS A14 L |
| AS2 | BUS NPR L | BS1 | BUS A17 L |
| AT1 | GROUND | BS2 | BUS A16 L |
| AT2 | BUS BR7 L | BT1 | GROUND |
| AU1 | +20 V | BT2 | BUS C1 L |
| AU2 | BUS BR6 L | BU1 | BUS SSYN L |
| AV1 | +20 V | BU2 | BUS C0 L |
| AV2 | +20 V | BV1 | BUS MSYN L |
| | | BV2 | -5 V |

### 1.4.3 Timing

Figure 1-2 shows important timing constraints for the M9312. Values shown are typical.

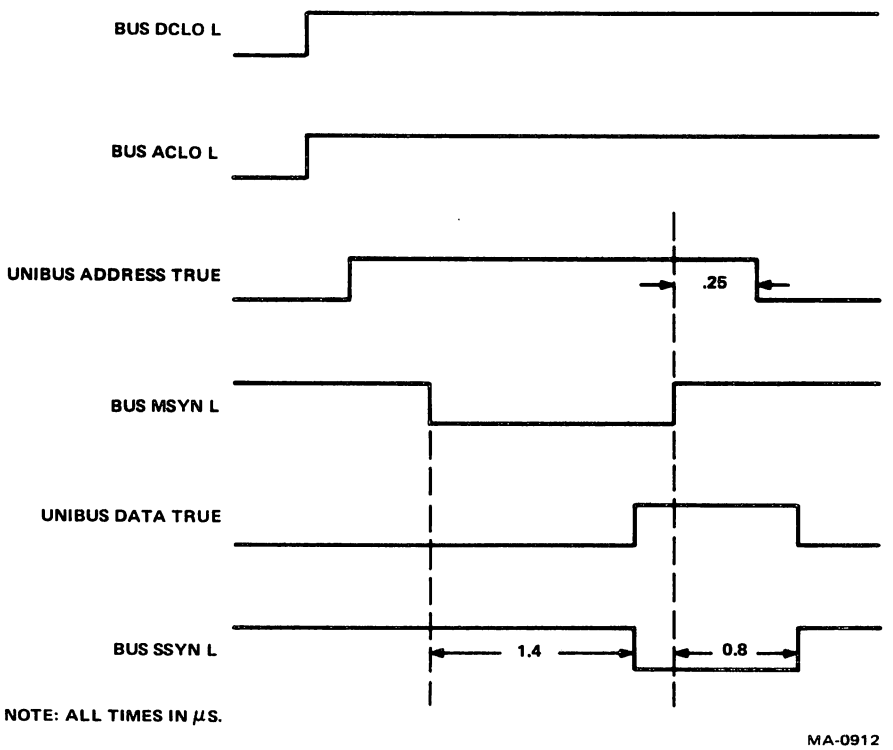



Figure 1-2 M9312 Timing Constraints

### 1.4.4 Operational Environmental Specifications

| | |
|---|---|
| Operating Temperature Range | 5° C (41° F) to 50° C (122° F) |
| Storage Temperature Range | –40° C (–40° F) to 66° C (151° F) |
| Relative Humidity | 10 percent or less to 95 percent with maximum wet bulb of 32° C (90° F) and minimum dew point of 2° C (36° F). |

## 1.5 INSTALLATION

As a universal bootstrap/terminator module, the M9312 in its various configurations can be adapted by the user to meet a variety of boot requirements and system configurations. The following guidelines should be used when installing the module.

1. System power should be turned off.
2. When a M9312 is used no other bootstrap module, such as the M9301, may be used, and only one M9312 can be used in any given PDP-11 system.
3. In PDP-11/04, 11/34, and 11/34A systems without battery backup, TP4 should not be connected to the processor's power supply battery status signals unless boots on all power restarts are desired.
4. On PDP-11 systems containing a Unibus repeater, the M9312 must be installed on the processor side of the repeater.
5. Refer to Paragraphs 1.5.1 and 1.5.3 for power-up boot enable and external boot switches if they are to be used in the system.
6. In PDP-11 systems that have grant pull-up resistors in the processor module (PDP-11/04, 11/34 and 11/34A) and use the M9312 as the terminator for the processor end, jumpers W-1 through W-5 must be out. All other processors require W-1 through W-5 to be in.
7. Jumper W-6 should be in only when the M9312 is used with PDP-11/55, 11/60, and 11/70 systems that support push-button boot.
8. Jumper W-7 must be in for use in all PDP-11 systems.
9. For PDP-11 systems with at least one peripheral device whose Unibus address lies between $765000_8$ and $765776_8$, jumper W-8 should be in. This prevents the M9312 from responding to these addresses.
10. When the M9312 is used with a PDP-11/60 processor, jumpers W-9 and W-10 must be out, and jumpers W-11 and W-12 must be in. For all other current PDP-11 systems, jumpers W-9 and W-10 must be in and W-11 and W-12 must be out.
11. Appendix C should be consulted for the switch settings (S1-1 through S1-10) required for various boot configurations.
12. Bootstrap ROM installation must be sequential beginning with ROM location 1 (Figure 1-1), for all PDP-11 systems except the PDP-11/60, whether or not the console emulator routine is used.

13. In PDP-11/60 systems, when only one boot ROM is used it must be installed in location 2. If bootstraps are to be started from the console emulator routine, locations 1 and 2 must both contain ROMs. Additional ROMs must be installed first in location 3 and then in location 4 (Figure 1-1).

### 1.5.1 Power-Up Boot Enable

Automatic booting on power-up can be enabled or disabled using the power-up boot enable switch (S1-2). If this switch is set to the OFF position, the processor will execute its power-up routine normally, obtaining a new program counter (PC) from memory location $24_8$ and a new processor status word (PSW) from location $26_8$. When the switch is in the ON position during a power-up, the processor will obtain its new PC and PSW from locations $773024_8$ and $773026_8$ respectively. The address of the Offset Switch Bank (S1-1 and S1-3 through S1-10) is $773024_8$ ($773224_8$ if the processor traps to $224_8$ on power-up).

The function performed by the power-up boot enable switch (S1-2) can be duplicated by an external switch using Faston tabs TP3 and TP4. A closed switch connected to TP3 and TP4 is equivalent to S1-2 being ON. When MOS memory is present with battery backup, a battery status signal is generated by the power supply. This signal should be attached to the power-up boot enable input (TP4) on the M9312. If this status signal goes low, it indicates that the contents of the MOS memory are no longer valid. The M9312, sensing the status of the memory, forces a boot on power-up allowing new data to be written into memory. When TP4 is used, switch S1-2 should be off.

If the battery status input is high (logic 1) the M9312 will not automatically boot on power-up, and the processor will obtain its new PC from location $24_8$, and its new PSW from location $26_8$.

### 1.5.2 Boot Selection

For power-up boot or external boot, the boot routine is selected by nine switches (S1-1 and S1-3 through S1-10) provided on the M9312. Appendix C shows switch configurations necessary for various boot routines.

### 1.5.3 External Boot Switch

A device can be externally booted by using an external boot switch connected to Faston tabs TP1 and TP2. When TP1 and TP2 are connected, BUS ACLO L will be asserted, causing the processor to perform a power-down. Upon releasing the switch, BUS ACLO L will be unasserted, initiating a power-up sequence in the CPU and M9312 address assertion ($773000_8$).

# CHAPTER 2
# HARDWARE DESCRIPTION

## 2.1 GENERAL

The M9312 Bootstrap/Terminator Module, through the use of interchangeable socketed ROMs, can be used on all PDP-11 processors. The description that follows will hold true for most applications. Various portions of the circuitry will be analyzed separately for clarity. M9312 circuit schematics (D-CS-M9312-0-1) will be referenced throughout the description.

## 2.2 ROM MEMORY

The five ROMs used for the console emulator routine, diagnostics and bootstrap programs in the M9312 are socketed to allow reconfiguration with a minimum of effort. Only half of each ROM is used by Unibus systems, the other half is reserved for use by other systems. The module has 512 words of read only memory. The lower 256 words (addresses $765000_8$ through $765776_8$) are used for the storage of ASCII console and diagnostic routines. The diagnostics (discussed further in Chapter 6) are rudimentary CPU and memory diagnostics. The upper 256 words (addresses $773000_8$ through $773776_8$) are used for storage of Bootstrap programs. These locations are divided further into four 64-word segments. If necessary more than one segment may be used for a boot program. Figure 2-1 illustrates the segregation of the ROMs.

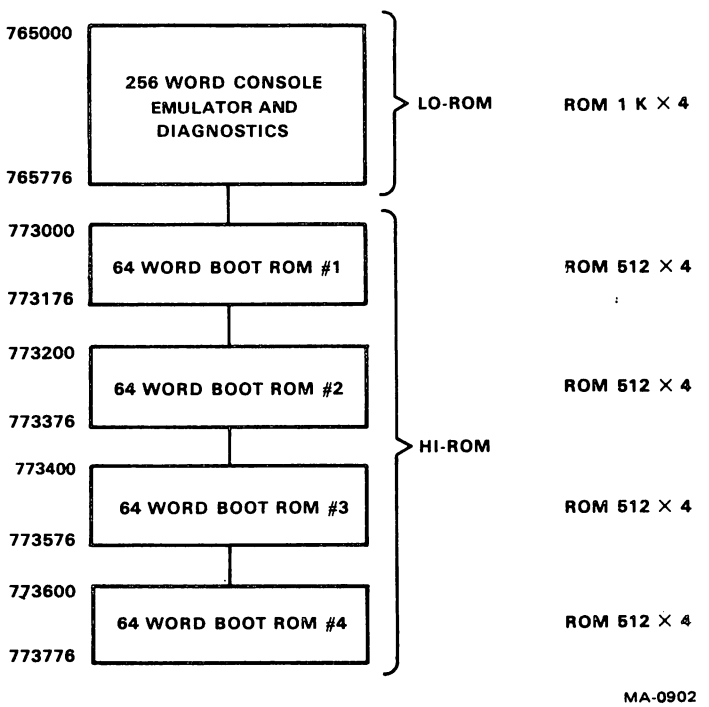



Figure 2-1 ROM Segregation

### 2.2.1 ROM Specifications

All PROM/ROM memories used on the M9312 must meet the requirements of Digital Equipment Corporation purchase specification 23-000A9-01 for the 512 × 4 (boot) ROMS and 23-000F1-01 for the 1K × 4 (CPU) ROM.

### 2.2.2 ROM Format

The following 64-word ROM format is required when writing a device boot which requires only one boot ROM in conjunction with the M9312 Bootstrap/Terminator Module.

1. The boot program must begin with a ROM header block.
2. Word address $24_8$ of the ROM must remain reserved and be set to $173000_8$.
3. Word address $26_8$ of the ROM must remain reserved and be set to $340_8$.
4. The last word of the ROM must be a CRC-16 word for the previous 63 words.

The following ROM format is required for device boots that need more than one ROM.

1. The first ROM follows the previously stated format.
2. The first word on each continued ROM must contain $177776_8$.
3. The last word of each continued ROM must contain a CRC-16 word for the previous 63 words.
4. Any continued ROM that would occupy word address $773224_8$ must reserve this location and put $173000_8$ in it.
5. Any continued ROM that would occupy word address $773226_8$ must reserve this location and put $340_8$ in it.

**2.2.2.1 ROM Header** – As previously stated the beginning of a boot program must contain a header section. This header section is described below:

| | |
|---|---|
| First word | Contains the ASCII identifier which consists of two characters with a zero-parity bit that will be used by the console emulator to identify a device for booting. |
| Second word | Contains the offset from this point to the next ROM header. If there is only one ROM header, this must point to the invisible first word of the next ROM. |
| Third word | Power-up entry point for unit zero, no diagnostics. |
| Fourth word | Power-up entry point for unit zero, diagnostic enabled. |
| Fifth word | Contains 000000, indicating unit 0 for instruction in previous word. |
| Sixth word | Entry point to ROM boot from the console emulator, R(0) must contain the unit number right justified. Enter here with C bit set if diagnostics are not desired. If diagnostics are desired, the C bit should not be set. |

| | |
|---|---|
| Seventh word | Address of the control/status register of the device to be booted. |
| Eighth word | Entry point when unit number of device to be booted is other than 0. This word moves R(7) to R(4). |
| Ninth word | Contains a branch instruction (BCC) to a link to the secondary diagnostic code. |

### 2.2.3 ROM Data

**2.2.3.1 ROM Data Transfer** – Data stored in the ROMs is addressed four bits at a time. These four bits are shifted through the output latches (E11 and E12) until a 16-bit word is ready to be transferred to the Unibus. A block diagram of this procedure is shown in Figure 2-2. Table 2-1 shows the relationship between the data word bit number and the output of the ROMs. It should be noted that bits 10, 11, and 12 must be stored inverted.

**2.2.3.2 ROM Data Organization** – As previously stated all ROM memory on the M9312 is four bits wide. Table 2-1 shows how 16-bit words are organized in the ROMs.

**Table 2-1 ROM Data Organization**

| ROM Output | Data | Word | Bit | Number |
|---|---|---|---|---|
| 4 | 15 | $\overline{11}$ | 7 | 3 |
| 3 | 14 | $\overline{10}$ | 6 | 2 |
| 2 | $\overline{13}$ | 9 | 5 | 1 |
| 1 | $\overline{12}$ | 0 | 4 | 8 |
| ROM Address Data Word 1 | $3_8$ | $2_8$ | $1_8$ | $0_8$ |
| ROM Address Data Word 64 | $377_8$ | $376_8$ | $375_8$ | $374_8$ |

**NOTE**
**Data word bits 10, 11, and 12 are stored inverted.**

### 2.2.4 Requirements for New ROMs and Uses of the M9312

M9312 specification A-SP-M9312-0-8 must be referred to for new uses of the M9312 and for creation of new ROMs.

## 2.3 POWER-UP SEQUENCE

Typically all PDP-11 computers perform a power-up sequence each time power is applied to their CPU module(s). This sequence is as follows.

1. +5 Vdc comes true.
2. BUS DCLO L is unasserted by power supply.
3. BUS ACLO L is unasserted by power supply.
4. BUS INIT asserts for approximately 100 ms.

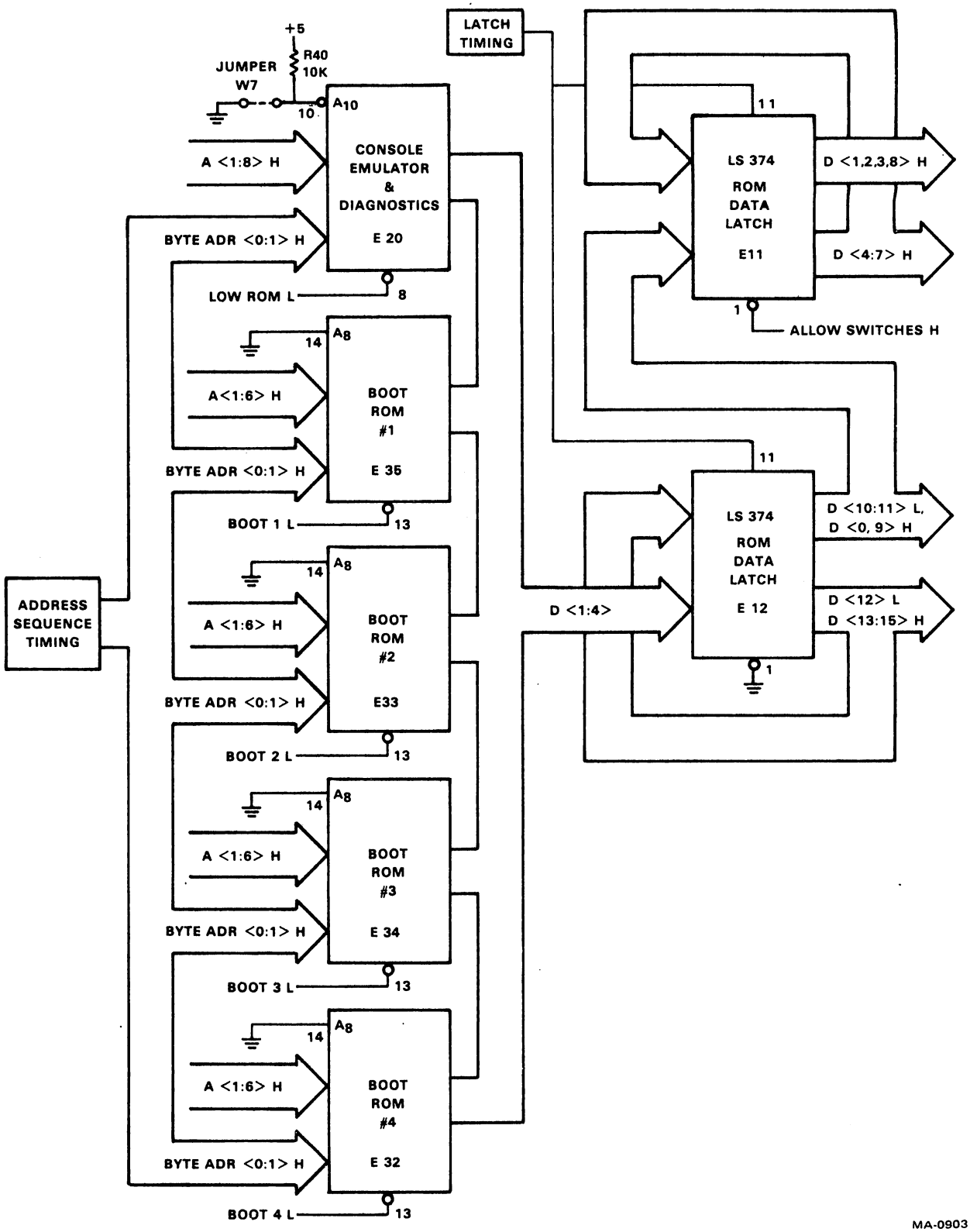



Figure 2-2 ROM Data Transfer

5. Processor accesses memory location $24_8$ for new PC.

6. Processor accesses memory location $26_8$ for new PSW.

7. Processor begins running program at new PC contents.

With an M9312 Bootstrap/Terminator Module in the PDP-11 computer system, on power-up the user can optionally force the processor to read its new PC from a ROM memory location and the Offset Switch Bank on the M9312 (Unibus location $773024_8$). A switch (S1-2) on the M9312 or an external switch on Faston tabs TP3 and TP4 can enable or disable this feature. The new PSW will be read from a location (Unibus location $773026_8$) in the M9312 memory. This new PC and PSW will then direct the processor to a program (typically a bootstrap) in the M9312 ROM memory (Unibus memory locations $773000_8$ through $773776_8$, and $765000_8$ through $765776_8$).

If the boot enable switch (S1-2) is off, an external switch or logic level can be used to make the processor execute a boot program on power-up. Programs in the M9312 can also be initiated by program jumps to their starting addresses or through the START switch feature of a programmer's switch console if one is available in the system.

## 2.4 POWER-UP BOOTING LOGIC

The status of every Unibus PDP-11 power supply is described by the two Unibus control lines BUS ACLO L and BUS DCLO L. The condition of these two lines in relation to the +5 V output of the power supply is defined by Unibus specifications as summarized in Figure 2-3.

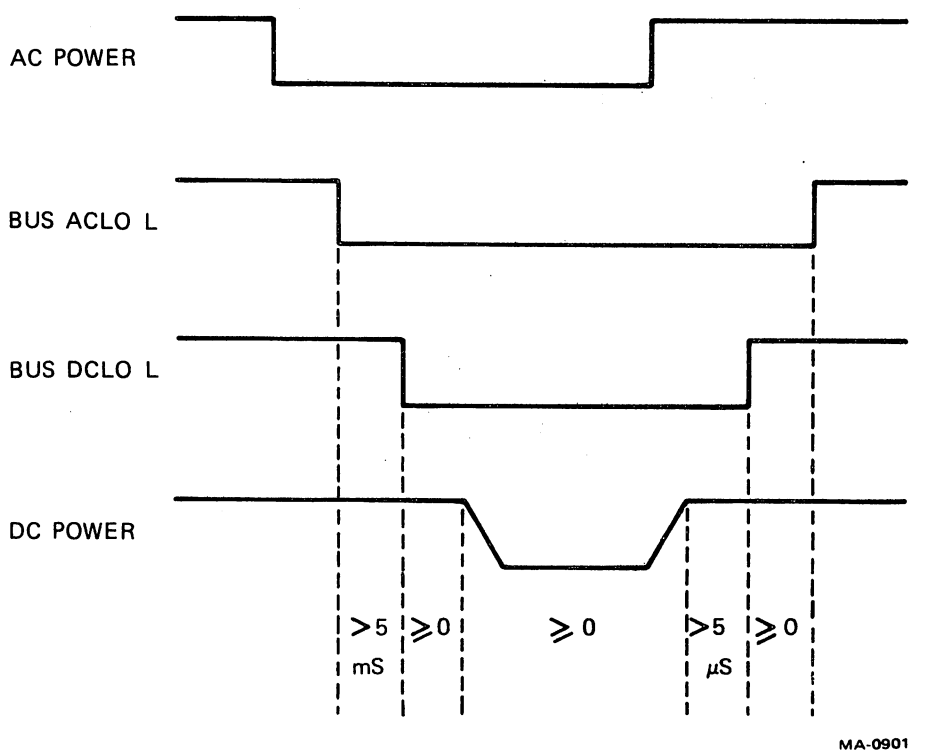



Figure 2-3 Power-Down/Power-Up Sequence

### 2.4.1 Power-Up and Power-Down

On the M9312, power-up sequences are detected by the circuitry shown in Figure 2-4. When +5 V first becomes true, both BUS ACLO L and BUS DCLO L are asserted low. Assuming the power-up boot-enable switch is closed a high to low transition out of E8 (pin 3) (POWER-UP BOOT L) triggers the one shot E7 (pin 5) which asserts Unibus address lines BUS A 09 L, BUS A 10 L, and BUS A 12 L through BUS A 17 L for up to 300 ms. The logic shown in Figure 2-5 generates these Unibus addresses.

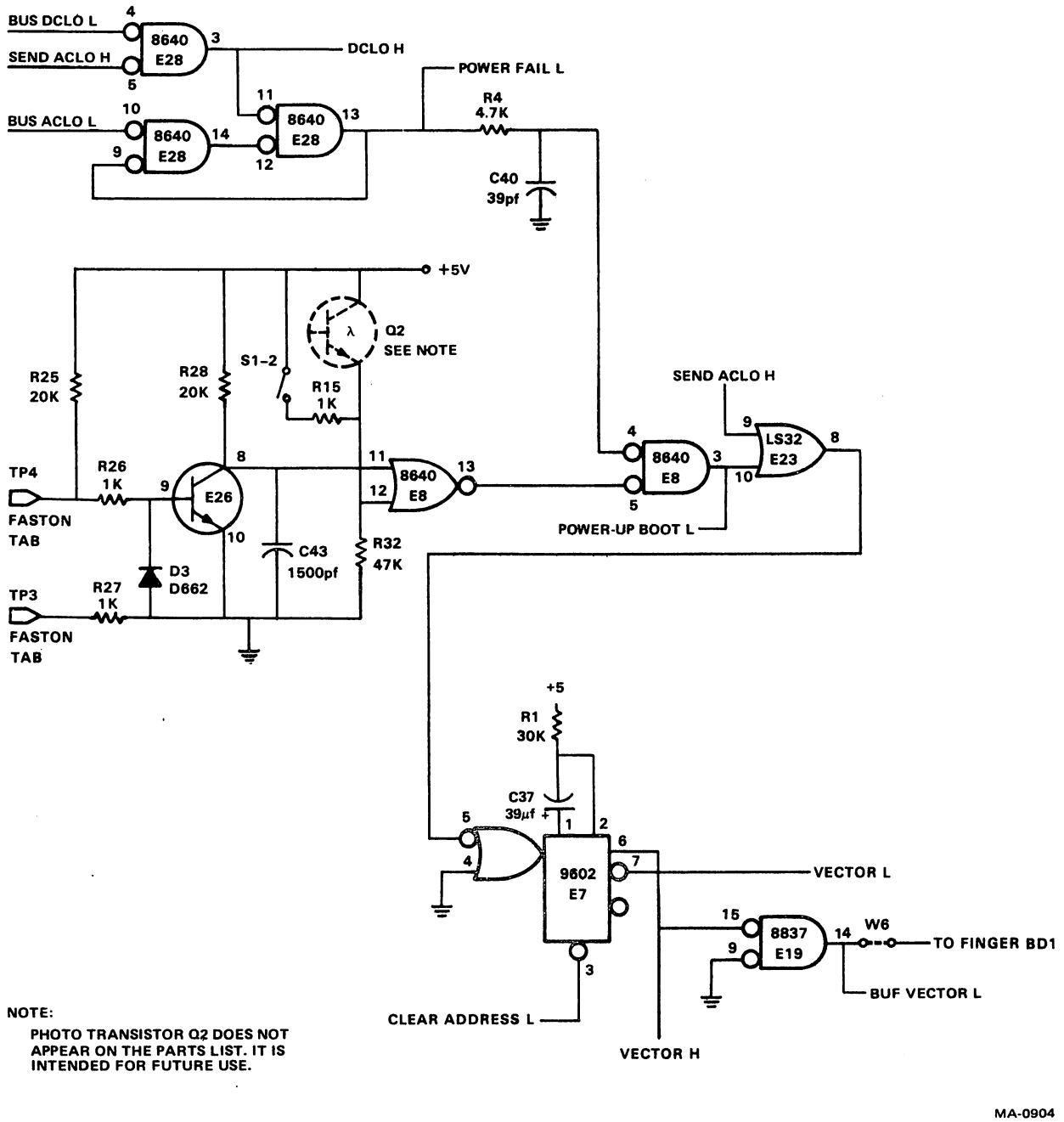



Figure 2-4 Power-Up Boot Logic

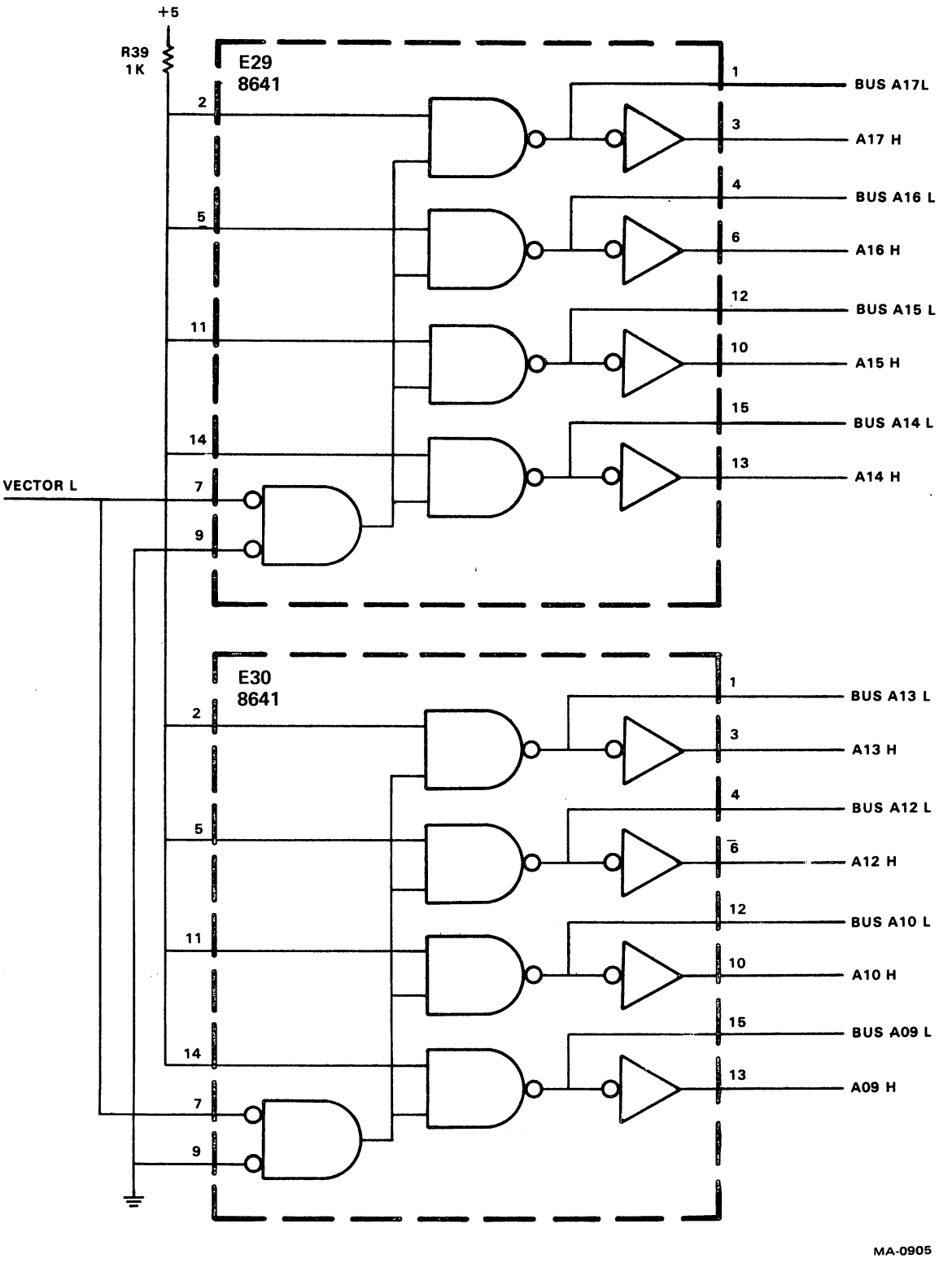



Figure 2-5 Address Generation Logic

### 2.4.2 Processor Reads New Program Counter

During the 300 ms maximum assertion time, the central processor will be performing its power up sequence. When the processor attempts to read a new program counter (PC) address from memory location $24_8$, the address bits enabled by the one shot E7 are logically ORed to generate the address $773024_8$. This location is an address in the M9312 ROM space and the address of the Offset Switch Bank, which contains the starting address of a specific routine.

### 2.4.3 Processor Reads New Status Word

Having obtained a new PC from location $773024_8$, the processor then attempts to read a new processor status word (PSW) from memory location $26_8$. The address bits enabled by the one shot E7 (pin 7) are logically ORed to generate the address $773026_8$ which is also in the M9312 ROM address space. Once this transfer is completed, the processor unasserts MSYN L and 150 ns (minimum) later the M9312 clears its asserted addresses. The M9312 unasserts SSYN L 800 ns after the unassertion of MSYN L freeing the Unibus. The 300 ms one-shot (E7 pin 7) guarantees enough time for any PDP-11 processor to complete the two memory transfers described, before releasing the address lines.

### 2.4.4 Power-Up Boot Enable Switch

The power-up boot enable switch (S1-2) can be used to disable the logic shown in Figure 2-4. With this switch off (TP3 and TP4 open), the output of E8 (pin 3) will always be low, preventing one-shot E7 (pin 7) from ever being set on power restarts. Faston tabs TP3 and TP4 are provided to allow S1-2 to be remotely duplicated or accept a battery status input. Note that when Faston tab TP4 is used, S1-2 must be left in the off position.

## 2.5 EXTERNAL BOOT CIRCUIT

The processor can be activated externally by connecting Faston tabs TP1 and TP2 or by applying a logic '0' to TP1 (Figure 2-6). This sets flip-flop E6 (pin 15), which then generates an asserted BUS ACLO L signal on the Unibus. Upon seeing this Unibus signal, the processor will begin a power-down routine, anticipating a real power failure. After completing this routine, the processor will then wait for the unassertion of BUS ACLO L, at which time it will perform a power-up sequence through location $24_8$ and $26_8$.

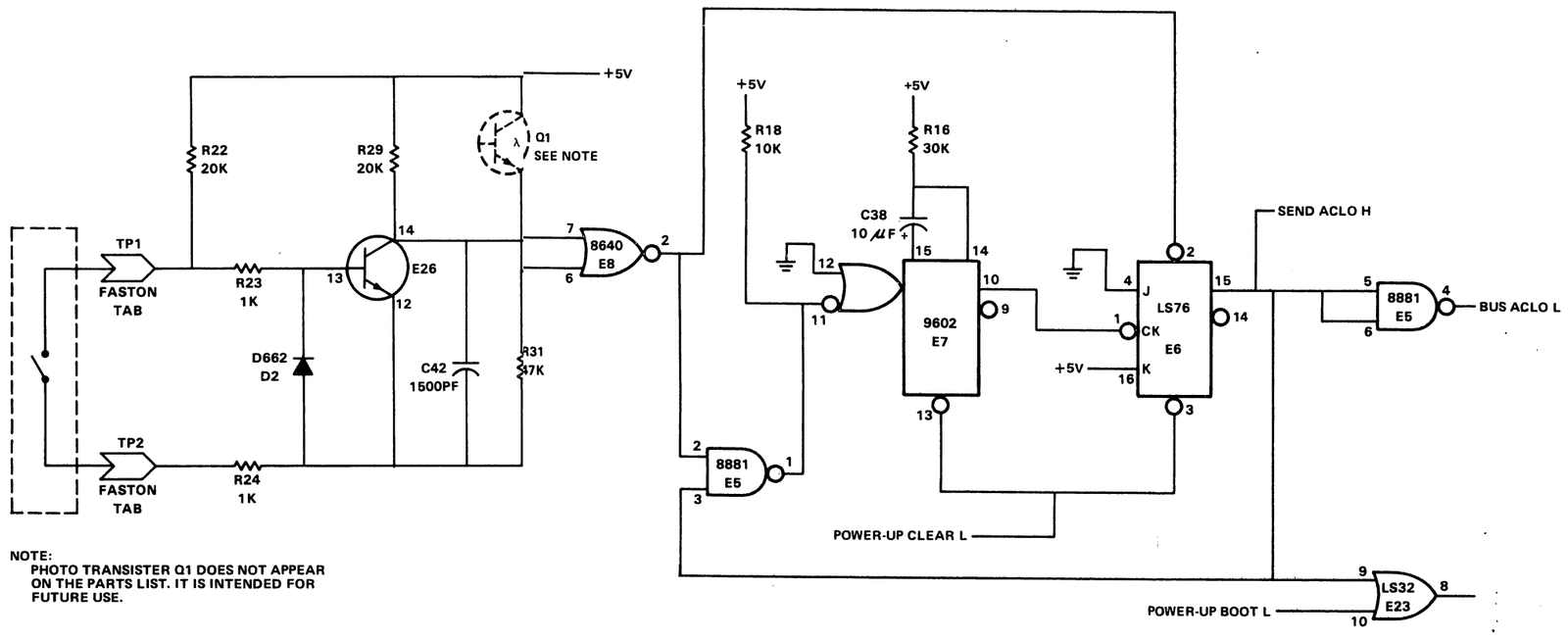
Upon release of the external boot switch or return to logic '1' at TP1, the set input to flip-flop E6 (pin 2) is unasserted and one-shot E7 on the M9312 is triggered, causing a 100 ms timeout. At the end of the timeout, BUS ACLO L is unasserted and the 300 ms one shot E7 (pin 10) is triggered. The processor is then forced to read its new PC and PSW from locations $773024_8$ and $773026_8$ respectively. The external boot timing is shown in Figure 2-7.

## 2.6 POWER-UP TRANSFER DETECTION LOGIC

Any time one-shot E7 (Pin 6) is set, bus address lines [BUS A (9, 10, 12:17) L] are asserted. The circuit shown in Figure 2-8 clears the address lines until BUS DCLO L becomes unasserted. When a boot occurs and the new PC and PSW have been transferred, the circuitry has received two MSYN signals and MSYN COUNT L goes low. At this time the bus address lines are unasserted. If the above transfers do not occur within 300 ms, the address lines are unasserted.

## 2.7 POWER-UP CLEAR

The circuit shown in Figure 2-9 holds the external boot circuit clear for approximately 8 ms after the dc supply voltage exceeds the threshold voltage of 3.0 to 4.5 V. This is to ensure that the external boot circuit does not cause a boot on power-up. The circuit uses the M9312's dc supply voltage to determine if a real power failure has occurred, because an assertion BUS ACLO L and BUS DCLO L may occur without a real power failure.




NOTE:
PHOTO TRANSISTER Q1 DOES NOT APPEAR ON THE PARTS LIST. IT IS INTENDED FOR FUTURE USE.

Figure 2-6 External Boot Logic

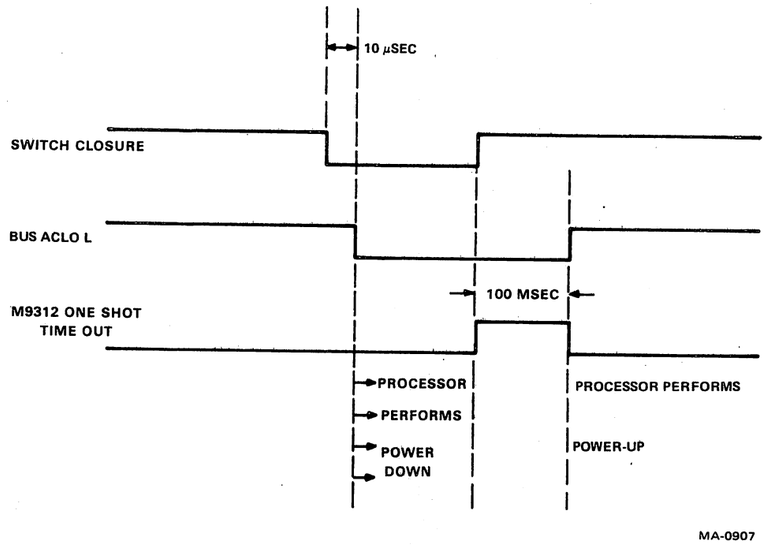



Figure 2-7 External Boot Timing

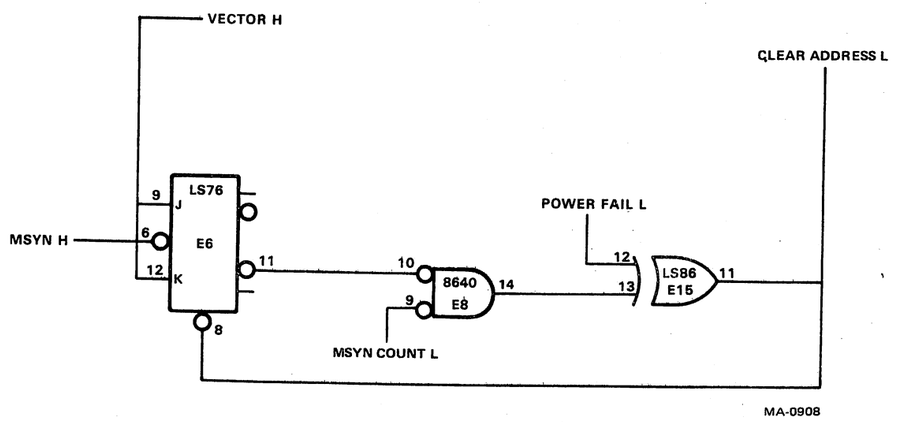



Figure 2-8 Power-Up Transfer Detection Logic

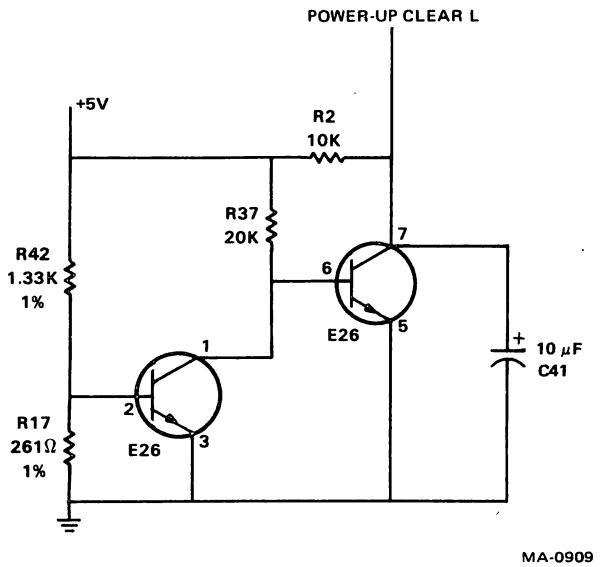



Figure 2-9 Power-Up Clear Circuitry

## 2.8 ADDRESS DETECTION LOGIC

### 2.8.1 M9312 Address Space

Address detection logic on the M9312 detects Unibus addresses within the address space $773000_8$ through $773776_8$ and $765000_8$ through $765776_8$. It also recognizes the specific address $773024_8$. Some processors, such as the PDP-11/60, trap to locations $224_8$ and $226_8$ on power-up for their new PC and PSW. The M9312 also recognizes the specific address $773224_8$. Figure 2-10 illustrates the M9312's address detection logic.

### 2.8.2 Memory Access Constraints

Upon receiving a recognized Unibus address and BUS MSYN, the M9312 ROM data output is transferred to the Unibus data lines (BUS D00 L through BUS D15 L) and BUS SSYN L is enabled. Conditions which must be met before transferring the ROM data and returning BUS SSYN are as follows:

1. Detection of the Unibus address $773XXX_8$ or $765XXX_8$ where $XXX_8$ is any even address. Installing W-8 disables detection of address $765XXX_8$.

2. Transfer being performed is a DATI operation where BUS C1 L is not asserted.

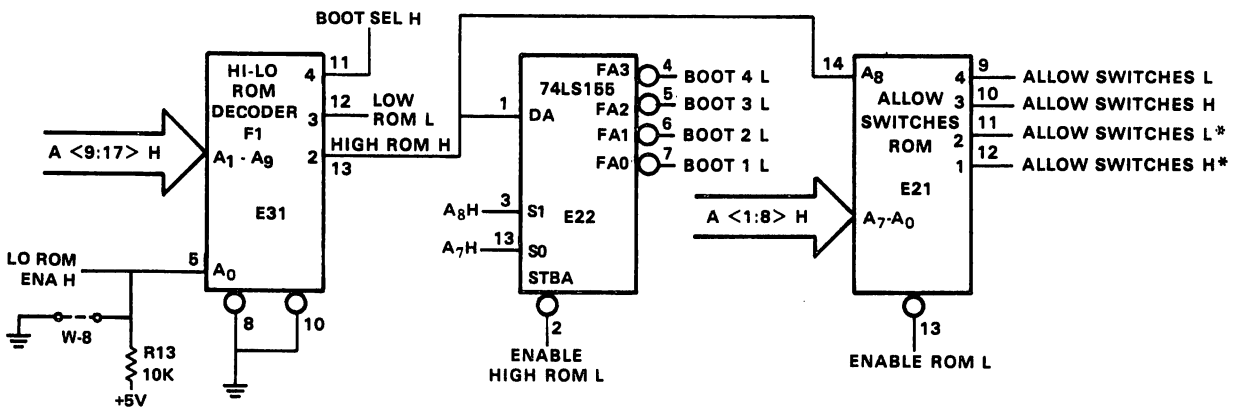
3. An asserted BUS MSYN L control signal has been obtained.

### 2.8.3 LO ROM ENA H Jumper

LO ROM ENA H jumper W-8 allows the user to disable the M9312 detection of Unibus addresses $765000_8$ through $765776_8$. Disabling the detection of these addresses (W-8 in) becomes essential when that memory space is being used by other peripheral device(s) in the system. Users should note that when detection of these addresses is disabled the console emulator and diagnostic routines in the M9312 are eliminated (not addressable).

## ALLOW SWITCHES ROM

| ADDRESSES A <8:1> H | HIGH ROM H | ENABLE ROM L | JUMPERS W-9 AND W-10 | JUMPERS W-11 AND W-12 | ALLOW SWITCHES L | ALLOW SWITCHES H |
|---|---|---|---|---|---|---|
| XXX | X | H | X | X | H | H |
| XXX | L | L | IN | OUT | H | L |
| XXX | L | L | OUT | IN | H | L |
| 024 | H | L | IN | OUT | L | H |
| DEFAULT | X | L | IN | OUT | H | L |
| 224 | H | L | OUT | IN | L | H |
| DEFAULT | X | L | OUT | IN | H | L |




## HI-LO ROM DECODER

| ADDRESSES A <17:9> | LO ROM ENA H | BOOT SEL H | LOW ROM L | HIGH ROM H |
|---|---|---|---|---|
| 765XXX | H | H | L | L |
| 765XXX | L | L | H | L |
| 773XXX | X | H | H | H |
| DEFAULT | X | L | H | L |

## 74LS155

| $A_8$ | $A_7$ | ENABLE HIGH ROM L | HIGH ROM H | BOOT 1 L | BOOT 2 L | BOOT 3 L | BOOT 4 L |
|---|---|---|---|---|---|---|---|
| L | L | L | H | L | H | H | H |
| L | H | L | H | H | L | H | H |
| H | L | L | H | H | H | L | H |
| H | H | L | H | H | H | H | L |
| X | X | X | L | H | H | H | H |
| X | X | H | X | H | H | H | H |

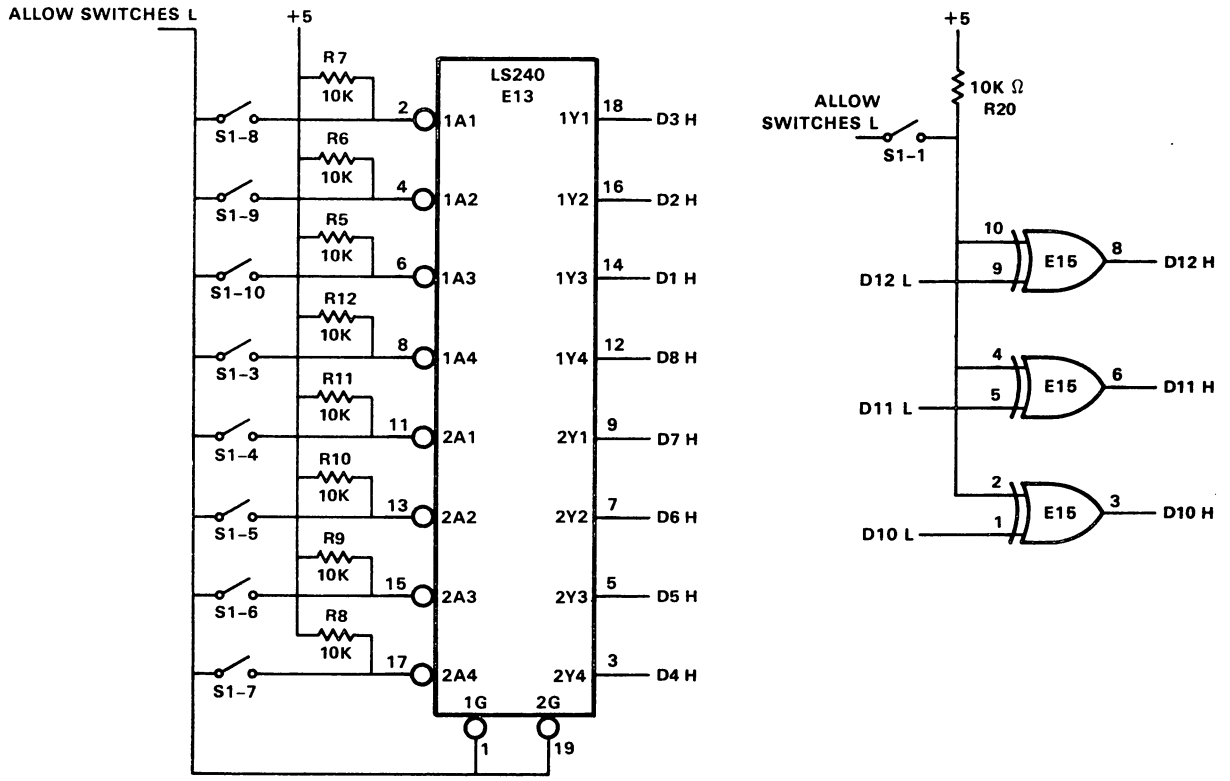
* THESE OUTPUTS ARE USED WHEN MODULE IS BEING USED WITH PROCESSORS THAT OBTAIN THEIR NEW PC AT LOCATION $224_{(8)}$ AND PSW AT $226_{(8)}$.

MA-0910

Figure 2-10 Address Detection Logic

## 2.9 ADDRESS OFFSET SWITCH BANK

As previously mentioned, on a power-up boot or an external boot, the M9312 forces the processor to obtain its new PC from location $773024_8$ instead of location $24_8$. When the M9312 address detection logic decodes address $773024_8$, it enables (via ALLOW SWITCHES L) the Address Offset Switch Bank (Figure 2-11). The contents of the switches S1-1 and S1-3 through S1-10 combined with the contents of the specified address in M9312 ROM memory produce a new PC for the CPU. The new PC will point to HI or LO ROM memory depending upon the position of the switch S1-1. With S1-1 OFF the new PC will point the processor to a starting address of a bootstrap program (addresses $773000_8$ through $773776_8$) in M9312 ROM memory. When S1-1 is ON, the new PC will point the processor to a starting address of a program in the M9312 console emulator and diagnostic ROM (addresses $765000_8$ through $765776_8$). Several programs can be included in M9312 memory with any one being user selectable through the Address Offset Switch Bank. Appendix C shows the relationship between the switches and the devices to be booted.




OFFSET SWITCHES AND CORRESPONDING BUS ADDRESS BITS

| S1 SWITCHES | 1 | | | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | * |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CORRESPONDING BUS ADDRESS BITS | 12 | 11 | 10 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| BINARY VALUE OF SWITCHES | | | | X | X | X | X | X | X | X | X | 0 |
| OCTAL VALUE OF SWITCHES | | | | Y | | | Y | | | Y | | |

*NO SWITCH IS PROVIDED FOR SETTING BIT 0, THEREFORE ONLY EVEN ADDRESS (FOR THE PC) MAY BE SET.

MA - 0911

Figure 2-11 Address Offset Switch Bank

## 2.10 M9312 TERMINATOR

The terminator section of the M9312 consists of four resistor pack circuits, each containing the required pull-up (178 ohms) and pull-down (383 ohms) resistors for proper Unibus termination. Since PDP-11/04, 11/34 and 11/34A processors contain Bus Grant pull-up resistors, and other processors do not, the M9312 allows a choice of whether or not to pull up the BUS grant lines. When jumpers W-1 through W-5 are in, these lines are pulled up; when these jumpers are out, the grant lines are not pulled up.

# CHAPTER 3
# CONSOLE EMULATOR

## 3.1 GENERAL

The console emulator routine is available when a ROM which contains the routine is used on the M9312. This ROM is placed in the console emulator and diagnostic ROM socket shown in Figure 1-1.

## 3.2 USING THE CONSOLE EMULATOR

The system will execute a console emulator power-up routine when power is supplied to the system, the boot switch is pressed, or the correct address from Table 3-1 is loaded and started, provided jumper W-8 is out, W-7 is in, and the Address Offset Switch Bank is set according to Table 3-1. If diagnostics are selected, secondary diagnostics (tests 6 and 7) will run after the console emulator routine, just before a boot. Primary diagnostic tests 1 through 4 are always executed before the console emulator routine. Completion of the primary diagnostic tests will be followed by the register display routine. The contents of R0, R4, R6, and R5 will be printed out on the terminal. An @ sign will be printed at the beginning of the next line of the terminal, indicating that the console emulator routine is waiting for input from the operator.

**Table 3-1 Console Emulator Switch Requirements**

| | Octal Address† | Address Offset Switch Bank S1 | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 1 | 2* | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| Console Emulator with Diagnostics | 165020 | ON | – | OFF | OFF | OFF | OFF | ON | OFF | OFF | OFF |
| Console Emulator without Diagnostics | 165144 | ON | – | OFF | OFF | ON | ON | OFF | OFF | ON | OFF |

*When switch S1-2 is ON, power-up boot is enabled; when S1-2 is OFF power-up boot is disabled.

†The octal address can be loaded and program started from a programmer's console or switch register, if a power-up start or external boot start from the M9312 is not used.

The following symbols will be used in this discussion.

<SB>: Space bar
<CR>: Carriage return key
X: Any octal number 0–7

The four console functions can be exercised by pressing keys, as follows:

| Function | Keyboard Strokes |
|---|---|
| Load Address | L<SB> XXXXXX <CR> |
| Examine | E<SB> |
| Deposit | D<SB> XXXXXX <CR> |

The first digit typed will be the most significant digit. The last digit typed will be the least significant digit. If an address or data word contains leading zeros, these zeros can be omitted when loading the address or depositing the data. An example using the load, examine, deposit, and start functions follows. Assume a user wishes to:

1. Load address 700
2. Examine location 700
3. Deposit 777 into location 700
4. Examine location 700
5. Start at location 700.

To accomplish this, the procedure below must be followed.

| | Operator Input | Terminal Display |
|---|---|---|
| 1. | Turns on power | XXXXXX XXXXXX XXXXXX XXXXXX |
| 2. | L<SB> 700<CR> | @L 700 |
| 3. | E<SB> | @E 000700 XXXXXX |
| 4. | D<SB> 777<CR> | @D 777 |
| 5. | E<SB> | @E 000700 000777 |
| 6. | S<CR> | @S |

**NOTE**
**The console emulator routine will not work with odd addresses. Even addresses must always be used.**

### 3.2.1 Successive Operations

**3.2.1.1 Examine** – Successive examine operations are permitted. The address is loaded for the first examine only. Successive examine operations cause the address to increment and will display consecutive addresses along with their contents. For example, to examine addresses 500–506, the following procedure may be used.

| Operator Input | Terminal Display |
|---|---|
| L<SB>500<CR> | @L 500 |
| E<SB> | @E 000500 XXXXXX |
| E<SB> | @E 000502 XXXXXX |
| E<SB> | @E 000504 XXXXXX |
| E<SB> | @E 000506 XXXXXX |

**3.2.1.2 Deposit** – Successive deposit operations are permitted. The procedure is identical to that used with examine. For example, to deposit 60 into location 500, 2 into location 502, and 4 into location 504:

| Operator Input | Terminal Display |
|---|---|
| L<SB>500<CR> | @L 500 |
| D<SB>60<CR> | @D 60 |
| D<SB>2<CR> | @D 2 |
| D<SB>4<CR> | @D 4 |

**3.2.1.3 Alternate Deposit-Examine Operations** – This mode of operation will not auto-increment the address. The location addressed will contain the last data which was deposited. For example, to load address 500 and deposit 1000, 2000, and 5420 with examine operations after every deposit:

| Operator Input | Terminal Display |
|---|---|
| L<SB>500<CR> | @L 500 |
| D<SB>1000<CR> | @D 1000 |
| E<SB> | @E 000500 001000 |
| D<SB>2000<CR> | @D 2000 |
| E<SB> | @E 000500 002000 |
| D<SB> 5420<CR> | @D 5420 |
| E<SB> | @E 000500 005420 |

**3.2.1.4 Alternate Examine-Deposit Operations** – If an examine is the first instruction after a load sequence and is alternately followed by deposits and examines, the address will not be incremented, and the address will contain the last data which was deposited. The prior example applies to this operation, and with the exception of the order of examine and deposit, the end result is the same.

### 3.2.2 Limits of Operation

The M9312 console emulator can directly manipulate the lower 28K of memory and the 4K I/O page. See Chapter 5 for an explanation of techniques required to access addresses above the lower 28K.

## 3.3 BOOTSTRAPS STARTED FROM THE CONSOLE EMULATOR

Once the @ symbol has been displayed in response to system power-up, or pressing the boot switch, the system is ready to boot a device the operator selects. The procedure is as follows.

1. Load paper tape, magtape, disk, etc., into the peripheral to be booted, if required.

2. Verify that the peripheral indicators signify that the peripheral is ready (if applicable).

3. Find the boot command code in Table 3-2 that corresponds to the peripheral to be booted. Type the code obtained from the table. (The @ sign will be returned at this point if the correct boot ROM has not been installed, or if a non-existent code is typed in. If the register display is printed first, the emulator is indicating that at least one boot ROM socket is available for boot ROM installation.)

4. If there is more than one unit of a given peripheral, type the unit number to be booted (0–7). If no number is typed, the default number will be 0.

5. Type <CR>, which initiates the boot.

Table 3-2 Boot Command Codes

| Interface | Device | Description | Command Code |
|---|---|---|---|
| RL11 | RL01 | Disk Memory | DL |
| RX11 | RX01 | Floppy disk system | DX |
| RK11C,D | RK03,05/05J | DECpack disk | DK |
| TC11 | TU55/56 | Dual DECtape | DT |
| RX211 | RX02 | Double density floppy disk system | DY |
| RK611 | RK06/07 | Disk drive | DM |
| TM11/A11/B11 | TS03,TU10 | Magnetic tape (9 track, 800 bits/in, NRZ) | MT |
| RH11/RH70 | TU16,TM02 | Magnetic tape | MM |
| TA11 | TU60 | Dual magnetic tape | CT |
| PC11 | | High speed reader | PR |
| DL11-A | | Low speed reader | TT |
| RP11 | RP02/03 | Moving head disk | DP |
| RH11/RH70 | RP04/05/06 RM02/03 | Moving head disk | DB |
| RH11/RH70 | RS03/04 | Fixed head disk | DS |

Before booting a device always remember:

1. The medium (paper tape, disk, magtape, cassette, etc.) must be placed in the peripheral to be booted prior to booting.

2. The machine will not be under the control of the console emulator routine after booting.

3. The program which is booted in must:

   a. Be self-starting or

   b. Allow the user to load another program by using the CONT function or

   c. Be startable from the console emulator or switch register after having been booted in.

### 3.3.1 Booting the High-Speed Reader Using the Console Emulator

To load the CPU diagnostic for a PDP-11/34 computer system with a high-speed reader, perform the following procedure.

1. Place the HALT/CONT switch in the CONT position.

2. Obtain a @ symbol by either turning on system power or actuating the boot switch. (R0, R4, SP, and old PC will be printed prior to the @ symbol.)

3. Place the absolute loader paper tape (coded leader section) in the high speed reader.

4. Type PR<CR>.
   The absolute loader tape will be loaded and the machine will halt.

5. Remove the absolute loader and place the leader of the program, in this case a CPU diagnostic, in the reader.

6. Move the HALT/CONT switch to HALT and then return it to CONT. The diagnostic will be loaded and the machine will halt (normal for this program; non-diagnostic programs may or may not be self-starting.)

7. If program is not self-starting, activate the BOOT/INIT switch. This will restart the console emulator routine.

8. Using the console emulator, deposit desired functions into the software switch register (a memory address) location. (See the diagnostic for the software switch register's actual location and significance.)

9. Using the console emulator, load the starting address, and start the program as described earlier in this section.

### 3.3.2 Booting a Disk Using the Console Emulator

To boot the system's RK05 disk, which contains the CPU diagnostics that you want to run, perform the following procedure:

1. Verify that the HALT/CONT switch is in the CONT position and the write lock switch on the RK11 peripheral is in the ON position.

2. Turn on system power or press the console boot switch. The system terminal displays R0, R4, SP and old PC in octal numbers followed by a @ symbol on the next line.

3. Place the disk pack in drive 0.

4. When the RK05 load light appears, the system is ready to be booted.

5. Type DK<CR>.
   This causes the loading of the bootstrap routine into memory and the execution of that routine.

6. The program should identify itself and initiate a dialogue (which will not be discussed here).

## 3.4 RECOVERING FROM ERRORS IN THE CONSOLE EMULATOR ROUTINE

Table 3-3 describes the effects of entering information incorrectly to the console emulator routine. The following symbols are used in the table.

(9) Represents a non-octal number (8 or 9)

(Y) Represents:
   1. All keys (other than numerics) which are unknown.

   2. Keys which are known but do not constitute a valid code in the context which they are entered.

Refer to previous sections for a discussion of the correct method operating the console emulator routine.

**Table 3-3 Deposit Errors: Useful Examples**

| Error | Result | Remedy | Operator | Terminal |
|---|---|---|---|---|
| L was followed by a key other than (SB). | Terminal display will immediately return an @ to signify an unknown code. No address is loaded. | Try again | L(Y) | @L<br>@ |
| An illegal (non octal) number (8 or 9) is typed after the correct load entrance, within an otherwise valid number. | Upon receipt of the illegal number, the Console Emulator will ignore the entire address and return an @. | Try again | L<SB><br>XXX9 | @L XXX9<br>@ |
| An alphabetic key is typed after the correct load entrance within an otherwise valid number. | Same as illegal number. | Try again | L<SB><br>XXXY | @L XXXY<br>@ |
| The most significant octal number in a six bit address is greater than one. | An address will be loaded. However, the state of the most significant address bit will be determined by bit 15 only:<br><br>2 = 0<br>3 = 1<br>4 = 0<br>5 = 1<br>6 = 0<br>7 = 1 | Try again if required | L<SB><br>6XXXXX<br><CR> | @L 6XXXXX<br>@<br>(address loaded 0XXXXX) |
| An unwanted but legal octal number is loaded. | Unwanted address will be loaded. | Try again | | |
| An extra (seventh) octal number is typed. | The loaded number will be incorrect. The system will accept any size number but will only remember the last six characters typed in. | Try again | L<SB><br>1XXXXX<br><CR> | @L 1XXXXX<br>@<br>(Actually Loads XXXXXX) |
| A memory location higher than the highest memory location available in the machine is loaded. | No errors will result unless a deposit, examine, or start is attempted, causing Bus to hang up. | Try again | L<SB><br>1XXXXX<br><CR> | L 1XXXXXX<br>@ |
| L, <SB> and number were entered correctly, <CR> was not entered. | Machine will wait indefinitely for <CR>. @ will not be returned | Type <CR> | L<SB><br>XXXXX<br><CR> | @L XXXXXX<br>@ |

Table 3-3 Deposit Errors: Useful Examples (Cont)

| Error | Result | Remedy | Operator | Terminal |
|---|---|---|---|---|
| Examine or start is attempted to a memory location which is higher than the highest available memory location in the machine (I/O page can be examined) or to an odd memory location. | The system will hang up. | Actuate the boot switch | E<SB> or S<CR> | @E or @S (stops responding) |
| Examine is performed without loading an address prior to first examine. | An examine operation of an unknown address will be performed. It is possible that the machine may attempt to examine an address which does not exist. If this happens the system will hang-up. | Try again or boot if system hangs up. | E<SB> | |
| Start is performed without loading an address prior to starting. | Start at an unknown location will occur. | Depress the boot switch, load correct address and then start. | | |
| D was followed by a key other than a space or a valid second character of a boot command code. | Terminal display will immediately return a @ to signify an unknown code. If the register display is printed before the @ sign, the emulator is indicating that at least one socket is still available for a boot ROM. | Try again | D(Y) | @D(Y)<br>@ |
| Deposit is attempted to a memory location which is higher than the highest available memory location in the machine (with the exception of the I/O page). | The system will hang up when <CR> is executed. | Depress the boot switch and start over again. | | |

Table 3-3 Deposit Errors: Useful Examples (Cont)

| Error | Result | Remedy | Operator | Terminal |
|---|---|---|---|---|
| Deposit is performed without loading an address or knowing what address has been previously loaded. | a. Data will be written over and lost | a. Immediately following the error, perform an examine to determine the location which was accessed. Restore original contents if known. | | |
| | b. Machine might hang up. | b. Actuate the boot switch. | | |
| Deposit into an odd address is attempted. | The system will hang up when <CR> is executed. | Actuate the boot switch. | | |

**Escape Route**
If an entry has not been completed and the user realizes that an incorrect or unwanted character has been entered, press the rubout or delete key. This action will void the entire entry and allow the user to try again.

**Machine "Hang-Up"**
Machine has "hung-up" (halted or gone into program loop) if the terminal does not respond to any keyboard entry.

# CHAPTER 4
# BOOTSTRAPPING

The routines to bootstrap a device typically read in the first sector, block, or 512 words from the device into location 0 through 512 of memory. The exception to this rule is the paper tape boot. The paper tape boot is unique in that it can do no error checking and the secondary bootstrap (the absolute loader, for example) is read into the upper part of memory. The actual locations loaded by the paper tape boot are partially determined by the secondary bootstrap itself and by the size routine which determines the highest available memory address within the first 28K. The flexible disk (or floppy) reads sector 1 on track 1 into consecutive locations starting at 0. The magnetic tape boots read the second block into consecutive locations starting with 0. If no errors are detected in the device, the bootstraps normally transfer control to location 0 in order to execute the secondary bootstrap just loaded. The only exception to this starting address concerns the paper tape boots. They transfer control to location XXX374, where XXX is determined initially by the size routine to be at the top of memory. This is where the absolute loader has just been loaded.

If a device error is detected, a reset will be executed and the bootstrap will try again. The bootstrap will be retried indefinitely until it succeeds without error unless the user (operator) intervenes. The advantage of retrying the boot is that if a particular device being booted is not on-line or not loaded, perhaps because of a power failure, the boot will give the device a chance to power-up (essential for disks).

A magnetic tape transport, however, will not automatically reload itself after a power failure and restart. This situation requires user intervention. The user must reload the magtape and bring it back on-line, at which time the magtape bootstrap, which will have been continually attempting to boot the tape, will succeed.

# CHAPTER 5
# EXTENDED ADDRESSING

## 5.1 GENERAL

This chapter applies to use of the M9312 in PDP-11 systems which have no console. When the memory of a PDP-11 system is extended beyond 28K, the processor is able to access upper memory through the memory management system. However, the console emulator normally allows the user to access only the lower 28K of memory. This chapter provides an explanation of the method by which the user can gain access to upper memory in order to read or modify the contents of any location. The reader should be familiar with the concepts of memory management in the KD11-E processor.

## 5.2 VIRTUAL AND PHYSICAL ADDRESSES

Addresses generated in the processor are called virtual addresses, and will be 16 bits in length. Physical addresses refer to actual locations in memory. They are asserted on the Unibus and may be up to 18 bits in length (for 128K memories).

## 5.3 ADDRESS MAPPING WITHOUT MEMORY MANAGEMENT

With memory management disabled (as is the case following depression of the boot switch), a simple hardware mapping scheme converts virtual addresses to physical addresses. Virtual addresses in the 0 to 28K minus 2 range are mapped directly into physical addresses in the range from 0 to 28K minus 2. Virtual addresses on the I/O page, in the range from 28K to 32K minus 2 ($160000_8$ to $177776_8$), are mapped into physical addresses in the range from 124K to 128K minus 2, or the last 4K of memory.

## 5.4 ADDRESS MAPPING WITH MEMORY MANAGEMENT

With memory management enabled, a different mapping scheme is used. In this scheme, a relocation constant is added to the virtual address to create a physical or "relocated" address.

Virtual address space consists of eight 4K banks where each bank can be relocated by the relocation constant associated with that bank. The procedure specified in this section allows the user to:

1. Create a virtual address to type into the load address command.

2. Determine the relocation constant required to relocate the calculated virtual address into the desired physical address.

3. Enable or disable the memory management hardware.

## 5.5 CREATION OF A VIRTUAL ADDRESS

The easiest way to create a virtual address is to divide the 18-bit physical address into two separate fields – a virtual address and a physical bank number. The virtual address is represented by the lower 13 bits and the physical bank by the upper 5 bits. The lower 3 bits of the physical bank number (bits 13, 14, 15) represent the virtual bank number (Table 5-1). Thus if bits 13, 14, and 15 are all 0s, the virtual bank selected is 0. The user should calculate the relocation constant according to Table 5-2. He can then deposit this constant in the relocation register associated with virtual bank 0 (Table 5-1).

**Table 5-1 Unibus Address Assignments**

| Virtual Address | Virtual Bank | Relocation Register | Descriptor Register |
|---|---|---|---|
| 160000–177776 | 7 | 172356 | 172316 |
| 140000–157776 | 6 | 172354 | 172314 |
| 120000–137776 | 5 | 172352 | 172312 |
| 100000–117776 | 4 | 172350 | 172310 |
| 060000–077776 | 3 | 172346 | 172306 |
| 040000–057776 | 2 | 172344 | 172304 |
| 020000–037776 | 1 | 172342 | 172302 |
| 000000–017776 | 0 | 172340 | 172300 |

**Table 5-2 Relocation Constants**

| Physical Bank Number | Relocation Constant | Physical Bank Number | Relocation Constant |
|---|---|---|---|
| 37 | 007600 | 17 | 003600 |
| 36 | 007400 | 16 | 003400 |
| 35 | 007200 | 15 | 003200 |
| 34 | 007000 | 14 | 003000 |
| 33 | 006600 | 13 | 002600 |
| 32 | 006400 | 12 | 002400 |
| 31 | 006200 | 11 | 002200 |
| 30 | 006000 | 10 | 002000 |
| 27 | 005600 | 7 | 001600 |
| 26 | 005400 | 6 | 001400 |
| 25 | 005200 | 5 | 001200 |
| 24 | 005000 | 4 | 001000 |
| 23 | 004600 | 3 | 000600 |
| 22 | 004400 | 2 | 000400 |
| 21 | 004200 | 1 | 000200 |
| 20 | 004000 | 0 | 000000 |

One relocation register exists for each of the eight virtual banks. In addition to the relocation registers, each bank has its own descriptor register which provides information regarding the types of access allowed (read-only, read or write, or no access).

The memory management logic also provides various forms of protection against unauthorized access. The corresponding descriptor register must be set up along with the relocation register to allow access anywhere within the 4K bank.

For example, assume a user wishes to access location $533720_8$. The normal access capability of the console is 0 to 28K. This address (533720) is between the 28K limit and the I/O page (760000–777776), and consequently must be accessed as a relocated virtual address, with memory management enabled. The virtual address is 13720 in physical bank 25 and is derived as follows.

All locations in bank 25 may be accessed through virtual addresses 000000–017776. The relocation and descriptor registers in the processor are still accessible since their addresses are within the I/O page. (Note that access to the I/O page is not automatically relocated with memory management, while access to the I/O page is automatically relocated when memory management is not used.)

The relocation constant for physical bank 25 is 005200. This constant is added in the relocation unit to the virtual address, as shown, yielding 533720.

| | |
|---|---|
| 013720 | Virtual address |
| *520000* | Relocated constant (Table 5-2) |
| 533720 | Physical address |

The Unibus addresses of the relocation registers and the descriptor registers are given in Table 5-1. The relocation constant to be loaded into the relocation register for each 4K bank is provided in Table 5-2. The data to be loaded in the descriptor register to provide read/write access to the full 4K is always 077406.

The Unibus address of the control register to enable memory management is 177572. This register is loaded with the value 000001 to enable memory management, and loaded with 0 to disable it.

To complete the example previously described (accessing location 533720), the console routine would be as follows:

```
@L      172340      /Access relocation register for virtual bank 0.
@D        5200      /Deposit code for physical bank 25.
@L      172356      /Access relocation register for virtual bank 7.
@D        7600      /Deposit code for the I/O page.
@L      172300      /Access descriptor register, virtual bank 0.
@D       77406      /Deposit code for read/write access to 4K.
@L      172316      /Access descriptor register, virtual bank 7.
@D       77406      /Deposit code for read/write access to 4K.
@L      177572      /Access control register.
@D            1     /Enable memory management.
@L       13720      /Load virtual address of location desired.
@E                  /Examine the data in location 533720.
                    /Data will be displayed.
```

## 5.6 CONSTRAINTS

Loading a new relocation constant into the relocation register for virtual bank 0 will cause virtual addresses 000000–017776 to access the new physical bank. A second bank can be made accessible by loading the relocation constant and descriptor data into the relocation and descriptor registers for virtual bank 1 and accessing the location through virtual address 020000–037776. Seven banks are accessible in this manner, by loading the proper constants, setting up the descriptor data, and selecting the proper virtual address. Bank 7 (I/O page) must remain relocated to physical bank 37 as it is accessed by the CPU to execute the console emulator routine.

Memory management is disabled by clearing (loading with 0s) control register 177572. It should always be disabled prior to typing a boot command.

The start command automatically disables memory management and the CPU begins executing at the physical address corresponding to the address specified by the previous load address command. Pressing the boot switch automatically disables memory management. The contents of the relocation registers are not modified.

The HALT/CONT switch has no effect on memory management.

# CHAPTER 6
# DIAGNOSTICS

## 6.1 GENERAL

The diagnostics in this chapter are standard for the M9312 when used in all PDP-11 computers. Paragraph 6.2 explains the diagnostics used in lower order systems such as the PPD-11/04 and 11/34, where the console emulator routine and diagnostics are to be used. An explanation of switch settings (S1-1, and S1-3 through S1-10) necessary to select specific routines is contained in Paragraphs 2.9 and 3.1. Paragraph 6.3 explains the diagnostics used for PDP-11/60 and 11/70 computers. No console emulator routine is present when used with these systems.

NOTE

**LO ROM ENA H jumper W-8 must be out in order to run diagnostics and/or the console emulator routine.**

## 6.2 DIAGNOSTICS

An explanation of the seven CPU and memory diagnostic tests follows. Three types of tests are included in the M9312 diagnostics:

1. Primary CPU tests
2. Secondary CPU tests
3. Memory test

**Primary CPU Tests**

The primary CPU tests exercise all unary and double operand instructions with all source modes. These tests do not modify memory. If a failure is detected, a branch-self (BR) will be executed. The run light will stay on, because the processor will hang in a loop. If no failure is detected in tests 1–4, the processor will emerge from the last test and enter the register display routine (console emulator).

**Test 1 – Single Operand Test** – This test executes all single operand instructions using destination mode 0. The basic objective is to verify that all single operand instructions operate; it also provides a cursory check on the operation of each instruction, while ensuring that the CPU decodes each instruction in the correct manner.

Test 1 tests the destination register in its three possible states: zero, negative, and positive. Each instruction operates on the register contents in one of four ways.

1. Data will be changed via a direct operation, i.e., increment, clear, decrement, etc.

2. Data will be changed via an indirect operation, i.e., arithmetic shifts, add carry, and subtract carry.

3. Data will be unchanged, but operated upon via a direct operation, i.e., clear a register already containing zeros.

4. Data will be unchanged but examined via a non-modifying instruction (TEST).

NOTE

**When operating upon data in an indirect manner, the data is modified by the state of the appropriate condition code. Arithmetic shift will move the C bit into or out of the destination. This operation, when performed correctly, implies that the C bit was set correctly by the previous instruction. There are no checks on the data integrity prior to the end of the test. However, a check is made on the end result of the data manipulation. A correct result implies that all instructions manipulated the data in the correct way. If the data is incorrect, the program will hang in a program loop until the machine is halted.**

**Test 2 – Double Operand, All Source Modes** – This test verifies all double operand, general, and logical instructions, each in one of the seven addressing modes (excludes mode 0). Thus, two operations are checked: the correct decoding of each double operand instruction, and the correct operation of each addressing mode for the source operand.

Each instruction in the test must operate correctly in order for the next instruction to operate. This interdependence is carried through to the last instruction (bit test) where, only through the correct execution of all previous instructions, a data field is examined for a specific bit configuration. Thus, each instruction prior to the last serves to set up the pointer to the test data.

Two checks on instruction operation are made in test 2. One check, a branch on condition, is made following the compare instruction, while the second is made as the last instruction in the test sequence.

Since the GO-NO GO tests reside in ROM memory, all data manipulation (modification) must be performed in destination mode 0 (register contains data). The data and addressing constants used by test 2 are contained within the ROM.

It is important to note that two different types of operations must execute correctly in order for this test to operate:

1. Those instructions that participate in computing the final address of the data mask for the final bit test instruction.

2. Those instructions that manipulate the test data within the register to generate the expected bit pattern.

Detection of an error within this test results in a program loop.

**Test 3 – Jump Test Modes 1, 2, and 3** – The purpose of this test is to ensure correct operation of the jump instruction. The test is constructed so that only a jump to the expected instruction will provide the correct pointer for the next instruction.

There are two possible failure modes that can occur in this test:

1. The jump addressing circuitry will malfunction causing a transfer of execution to an incorrect instruction sequence or non-existent memory.

2. The jump addressing circuitry will malfunction in such a way as to cause the CPU to loop.

The latter case is a logical error indicator. The former, however, may manifest itself as an after-the-fact error. For example, if the jump causes control to be given to other routines within the M9312, the interdependent instruction sequences would probably cause a failure to eventually occur. In any case, the failing of the jump instruction will eventually cause an out of sequence or illogical event to occur. This in itself is a meaningful indicator of a malfunctioning CPU.

This test contains a jump mode 2 instruction that is not compatible across the PDP-11 line. However, it will operate on any PDP-11 within this test, due to the unique programming of the instruction within test 3. Before illustrating the operation, it is important to understand the differences of the jump mode 2 between machines.

On the PDP-11/05, 11/10, 11/15, and 11/20 processors, for the jump mode 2 [JMP(R)+C, the register (R) is incremented by 2 prior to execution of the jump. On the PDP-11/04, 11/34, 11/35, 11/40, 11/45, 11/50, 11/55, and 11/70 processors, (R) is used as a jump address and incremented by 2 after execution of the jump.

In order to overcome this incompatability, the JMP(R)+ is programmed with (R) pointing back on the jump itself. On PDP-11/05, 11/10, 11/15, and 11/20 processors, execution of the instruction would cause (R) to be incremented to point to the following instruction, effectively continuing a normal execution sequence.

On the PDP-11/04, 11/34, 11/35, 11/40, 11/45, 11/50, 11/55, and 11/70 processors, the use of the initial value of (R) will cause the jump to loop back on itself. However, correct operation of the auto-increment will move (R) to point to the next instruction following the initial jump. The jump will then be executed again. However, the destination address will be the next instruction in sequence.

**Test 4 - Single Operand, Non-Modifying Byte Test** - This test focuses on the one single operand instruction, the TST. TST is a special case in the CPU execution flow since it is a non-modifying operation. Test 4 also tests the byte operation of this instruction. The TSTB instruction will be executed in mode 1 (register deferred) and mode 2 (register deferred, auto-increment).

The TSTB is programmed to operate on data which has a negative value most significant byte and a zero (not negative) least significant byte.

In order for this test to operate properly, the TSTB on the low byte must first be able to access the even addressed byte and then set the proper condition codes. The TSTB is then re-executed with the auto-increment facility. After the auto-increment, the addressing register should be pointing to the high byte of the test data. Another TSTB is executed on what should be the high byte. The N bit of the condition codes should be set by this operation.

Correct execution of the last TSTB implies that the auto-increment recognized that a byte operation was requested, thereby only incrementing the address in the register by one, rather than two. If the correct condition code has not been set by the associated TSTB instruction, the program will loop.

Upon successful completion of test 4, the register display routine is enabled. This routine will be followed by a prompt character (@) on the next line.

An example of a typical printout follows.

| | XXXXXX | XXXXXX | XXXXXX | XXXXXX |
|---|---|---|---|---|
| @ | | | | |
| | R0 | R4 | R6 | R5 |
| Prompt Character | | | (Stack Pointer) | (Old PC) |

NOTES

1. **X signifies an octal number (0–7).**

2. **Whenever there is a power-up routine or the boot switch is released on PDP-11/04 and PDP-11/34 machines, the PC at this time will be stored in R5. The contents of R5 are then printed as the old PC shown in the example.**

3. **The prompting character string indicates that diagnostics have been run and the processor is operating.**

**Secondary CPU and Memory Tests**
The secondary CPU tests modify memory and involve the use of the stack pointer. The JMP and JSR instructions and all destination modes are tested. If a failure is detected, these tests, unlike the primary tests, will execute a halt.

Secondary CPU and memory diagnostics are run immediately after test 4 when they have been evoked by means other than the console emulator, provided that the correct microswitches have been set. If the console emulator has been entered at the completion of test 4, the secondary CPU and memory diagnostics will be run when the appropriate boot command is given.

The M9312 reacts to a false boot command (an invalid address code) by returning to the console emulator routine. This should not be interpreted as a diagnostic test failure.

**Test 6 – Double Operand, Modifying, Byte Test** – The objective of this test is to verify that the double-operand, modifying instructions will operate in the byte mode. Test 6 contains three subtests:

1. Test source mode 2, destination mode 1, odd and even bytes
2. Test source mode 3, destination mode 2
3. Test source mode 0, destination mode 3, even byte.

The move byte (MOVB), bit clear byte (BICB), and bit set byte (BISB) are used within test 6 to verify the operation of the modifying double-operand functions.

Since modifying instructions are under test, memory must be used as a destination for the test data. Test 6 uses location 500 as a destination address. Later, in test 7 and the memory test, location 500 is used as the first available storage for the stack.

Note that since test 6 is a byte test, location 500 implies that both 500 and 501 are used for the byte tests (even and odd, respectively). Thus, in the word of data at 500, odd and even bytes are caused to be all 0s and then all 1s alternately throughout the test. Each byte is modified independently of the other.

**Test 7 - JSR Test** - The JSR is the first test in the GO-NO GO sequence that utilizes the stack. The jump subroutine command (JSR) is executed in modes 1 and 6. After the JSR is executed, the subroutine which was given control will examine the stack to ensure that the correct data was placed in the correct stack location (500). The routine will also ensure that the line back register points to the correct address. Any errors detected in this test will result in a halt.

**Test 8 - Dual Addressing and Data Check** - Finally the memory test performs both dual addressing and data check of all the available memory on the system below 28K. This test will leave all of memory clear. Like the secondary tests the memory test will halt when an error is detected. At the time the memory error halt is executed, R4 will contain the address at which the failure was detected. R0 will contain the failing data pattern and R6 will contain the expected data pattern. Thus after a memory failure has occurred, the user can enter the console emulator and have this information printed out immediately by display routine. (See section on console emulator.)

## 6.3 DIAGNOSTICS (PDP-11/60 AND 11/70)

The M9312 provides basic diagnostic tests for the CPU, memory, and cache when used with PDP-11/60 and PDP-11/70 computers. All diagnostic tests reside in ROM memory locations 765000 through 765776 (console emulator routine is eliminated.) These diagnostics test the basic CPU including the branches, the registers, all addressing modes, and many of the instructions in the PDP-11 repertoire. Memory from virtual address 1000 to the highest available address up to 28K will also be checked. After main memory has been verified, with the cache off, the cache memory will be tested to verify that hits occur properly. Main memory will be scanned again to ensure that the cache is working properly throughout the 28K of memory to be used in the boot operation. If one of the cache memory tests fails, the operator can attempt to boot the system anyway by pressing continue. This will cause the program to force misses in both groups of the cache before going to the bootstrap section of the program. The following is a list of M9312 diagnostic tests.

| | |
|---|---|
| TEST 1 | This test verifies the unconditional branch |
| TEST 2 | Test CLR, MODE 0, and BMI, BVS, BHI, BLT, BLOS |
| TEST 3 | Test DEC, MODE 0, and BPL, BEQ, BGE, BLE |
| TEST 4 | Test ROR, MODE 0, and BVC, BHIS, BNE |
| TEST 5 | Test register data path |
| TEST 6 | Test ROL, BCC, BLT |
| TEST 7 | Test ADD, INC, COM, and BCS, BLE |
| TEST 10 | Test ROR, DEC, BIS, ADD, and BLO |
| TEST 11 | Test COM, BIC, and BGT, BLE |
| TEST 12 | Test SWAB, CMP, BIT, and BNE, BGT |
| TEST 13 | Test MOVB, SOB, CLR, TST and BPL, BNG |
| TEST 14 | Test JSR, RTS, RTI, and JMP |
| TEST 15 | Test main memory from virtual 001000 to last address |
| | Cache memory diagnostic tests |
| TEST 16 | Test cache data memory |
| TEST 17 | Test memory with the data cache on |

**Test 1 - Verify the Unconditional Branch**
The registers and condition codes are all undefined when this test is entered and they should remain that way upon completion of this test.

**Test 2 - Test CLR, MODE 0, and BMI, BVS, BHI, BLT, BLOS**
The registers and condition codes are all undefined when this test is entered. Upon completion of this test, the SP (R6) should be zero and only the Z flip-flop will be set.

**Test 3 - Test DEC, MODE 0, and BPL, BEQ, BGE, BLE**
Upon entering this test, the condition codes are: N = 0, Z = 1, V = 0, and C = 0.
The registers are: R0 = ?, R1 = ?, R2 = ?, R3 = ?, R4 = ?, R5 = ?, and SP = 000000.
Upon completion of this test, the condition codes will be: N = 1, Z = 0, V = 0, and C = 0.
The registers affected by the test are: SP = 177777.

**Test 4 - Test ROR, MODE 0, and BVC, BHIS, BNE**
Upon entering this test, the condition codes are: N = 1, Z = 0, V = 0, and C = 0.
The registers are: R0 = ?, R1 = ?, R2 = ?, R3 = ?, R4 = ?, R5 = ?, and SP = 177777.
Upon completion of this test, the condition codes will be: N = 0, Z = 0, V = 1, and C = 1.
The registers affected by the test are: SP = 077777.

**Test 5 - Test Register Data Path**
Upon entering this test, the condition codes are: N = 0, Z = 0, V = 1, and C = 1.
The registers are: R0 = ?, R1 = ?, R2 = ?, R3 = ?, R4 = ?, R5 = ?, and SP = 077777.
Upon completion of this test, the condition codes will be: N = 0, Z = 1, V = 0, and C = 0.
The registers are left as follows: R0 = 125252, R1 = 000000, R2 = 125252, R3 = 125252, R4 = 125252, R5 = 125252, and SP = 125252.

**Test 6 - Test ROL, BCC, BLT**
When this test is entered, the condition codes are: N = 0, Z = 1, V = 0, and C = 0.
The registers are: R0 = 125252, R1 = 000000, R2 = 125252, R3 = 125252, R4 = 125252, R5 = 125252, SP = 125252.
Upon completion of this test, the condition codes are: N = 0, Z = 0, V = 1, and C = 1.
The registers are left unchanged except for R2 which should now equal 052524.

**Test 7 - Test ADD, INC, COM, and BCS, BLE**
When this test is entered, the condition codes are: N = 0, Z = 0, V = 1, and C = 1.
The registers are: R0 = 125252, R1 = 000000, R2 = 052524, R3 = 125252, R4 = 125252, R5 = 125252, SP = 125252.
Upon completion of this test, the condition codes are: N = 0, Z = 1, V = 0, and C = 0.
The registers are left unchanged except for R3 which now equals 000000, and R1 which is also 000000.

**Test 10 - Test ROR, DEC, BIS, ADD, and BLO**
When this test is entered, the condition codes are: N = 0, Z = 1, V = 0, and C = 0.
The registers are: R0 = 125252, R1 = 000000, R2 = 052524, R3 = 000000, R4 = 125252, R5 = 125252, and SP = 125252.
Upon completion of this test, the condition codes are: N = 1, Z = 0, V = 0, and C = 0.
The registers are left unchanged except for R1 which should equal 177777, and R4 which should now equal 052525.

**Test 11 - Test COM, BIC, and BGT, BLE**
When this test is entered, the condition codes are: N = 1, Z = 0, V = 0, and C = 0.
The registers are: R0 = 125252, R1 = 177777, R2 = 052524, R3 = 000000, R4 = 052525, R5 = 125252, and SP = 125252.

Upon completion of this test, the condition codes are: N = 0, Z = 0, V = 1, and C = 1.
The registers are left unchanged except for R0 which should now equal 052525, and R1 which should now equal 052524.

**Test 12 - Test SWAB, CMP, BIT, and BNE, BGT**
When this test is entered, the condition codes are: N = 0, Z = 0, V = 1, and C = 1.
The registers are: R0 = 052525, R1 = 052524, R2 = 052524, R3 = 000000, R4 = 052525, R5 = 125252, SP = 125252.
Upon completion of this test, the condition codes are: N = 0, Z = 0, V = 0, and C = 1.
The registers are now: R0 = 052525, R1 = 052125, R2 = 052524, R3 = 000000, R4 = 052525, R5 = 052525, SP = 12525.
**Test 13 - Test MOVB, SOB, CLR, TST and BPL, BNE**
When this test is entered, the condition codes are: N = 0, Z = 0, V = 0, and C = 1.
The registers are: R0 = 052525, R1 = 052125, R2 = 052524, R3 = 000000, R4 = 052525, R5 = 052525, and SP = 125252.
Upon completion of this test, the condition codes are: N = 0, Z = 1, V = 0, and C = 0.
R0 is decremented by an SOB instruction to 000000; R1 is cleared and then incremented around to 000000.

**Test 14 - Test JSR, RTS, RTI, JMP**
This test first sets the stack pointer to 000776, and then verifies that JSR, RTS, RTI, and JMP all work properly.

On entry to this test, the stack pointer (SP) is initialized to 000776 and is left that way on exit.

**Test 15 - Test Main Memory from 1000 to Highest Available Address up to 28K**
This test will test main memory with the cache disabled, from virtual address 001000 to the last address (up to 28K). The memory is sized before testing begins. If the data does not compare properly, the test will halt at either 165516 or 165536. If a parity error occurs, the test will halt at address 165750, with PC + 2 on the stack.

In this test the registers are initialized as follows: R0 = 001000, R1 = DATA READ, R2 = 001000, R3 = 177746 (cache control register), R4 = count value, R5 = last memory address, SP = 000776.

The following two tests are cache memory tests. If either of them fails to run successfully it will come to a halt in the M9312 ROM. If you desire to try to boot your system anyway, you can press continue and the program will force misses in both groups of the cache and go to the bootstrap that has been selected.

**Test 16 - Test Cache Data Memory**
This test will check the data memory in the cache, on the PDP-11/60. There is only one group (1K), on PDP-11/70 there are two groups, (1/2K) each. The test loads 0552525 into an address, complements it twice and then reads the data, then it checks to ensure that the data was a hit. Then the sequence is repeated on the same address with 125252 as the data. All cache memory data locations are tested in this way. If either group fails and the operator presses continue the program will try to boot with the cache disabled.

The registers are initialized as follows for this test: R0 = 4000 (address), R1 = 2 (count), R2 = 1000 (count), R3 = 177746 (control register), R4 = 125252 (pattern), R5 = (last memory address), SP = 000776 (flag of zero pushed on stack).

**Test 17 – Test Memory with the Data Cache On**
This test checks virtual memory from 001000 thru last address to ensure that you can get hits all the way up through main memory. On the PDP-11/70, it starts with group 1 enabled, then tests group 0, and finally checks memory with both groups enabled. On the PDP-11/60, the test is done with the whole cache enabled.

Upon entry the registers will be set up as follows: R0 = 001000 (address), R1 = 3 (pass count), R2 = (first address), R3 = 177746 (control register), R5 = (last memory address), SP = 775

Upon completion of this test main memory from vitual address 001000 thru last address will contain its own virtual address.

# APPENDIX A
# M9312 JUMPERS

## A.1 INTRODUCTION

The M9312 Bootstrap/Terminator Module is compatible with any PDP-11 system through the use of 12 jumpers (W-1 through W-12.) Table A-1 explains the function of these jumpers. Table A-2 shows necessary jumper configurations for various PDP-11 systems.

Table A-1 Jumper Explanation

| Jumper | Function |
|---|---|
| W1 | Connects Pull Up for BUS BG6 H (When in) |
| W2 | Connects Pull Up for BUS BG7 H (When in) |
| W3 | Connects Pull Up for BUS NPG H (When in) |
| W4 | Connects Pull Up for BUS BG5 H (When in) |
| W5 | Connects Pull Up for BUS BG4 H (When in) |
| W6 | Connects BUF VECTOR L to finger BD1 (IN for PDP-11/70) |
| W7 | RESERVED (Always in) |
| W8 | Connects LO ROM ENABLE H (When out) |
| W9, W10 | Install for power-up boot to $773024_{(8)}$ (IN for all CPUs except 11/60) |
| W11, W12 | Install for power-up boot to $773224_{(8)}$ (IN for PDP-11/60 only) |

Table A-2 Jumper Configurations

| PDP-11 Systems | Jumpers | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | W1 | W2 | W3 | W4 | W5 | W6 | W7 | W8 | W9 | W10 | W11 | W12 |
| PDP-11/04/34/34A (Modified Unibus Device Machine) with the M9312 in slot AB2, 3, 4 of processor backplane | OUT | OUT | OUT | OUT | OUT | OUT | IN | OUT | IN | IN | OUT | OUT |
| PDP-11/70 with M9312 in slot EF1 of processor backplane (See note) | IN | IN | IN | IN | IN | IN | IN | OUT | IN | IN | OUT | OUT |
| PDP-11/60 with M9312 in the last AB slot of the last memory backplane | IN | IN | IN | IN | IN | OUT | IN | OUT | OUT | OUT | IN | IN |
| Other Unibus CPUs with an M9312, or an M9312 in place of an M9301 or equivalent | IN | IN | IN | IN | IN | OUT | IN | OUT | IN | IN | OUT | OUT |

NOTES

**If a Unibus repeater is used in the system, the M9312 must be installed on the processor side of the repeater.**

**For boot on power-up the 11/70 requires the following.**

1. **7010329 backplane ECO 8 (wire list Rev J or later)**
2. **M8130 ECO's 1, 2 and 3 (CS Rev C or later) Jumpers set as follows: W1-IN, W2-OUT, W3-IN, W4-OUT, W5-OUT, W6-OUT, W7-OUT, W8-IN, W9-OUT**
3. **M8138 ECO 5 (CS Revs C1 or E or later)**
4. **M9312 ECO 1 (CS Rev B or later)**

# APPENDIX B
# M9312 ROMs

The M9312, through the use of socketed ROMs, can boot most PDP-11 peripheral devices, run diagnostic routines or execute a console emulator routine. The Address Offset Switch Bank (Paragraph 2.9 and Appendix C) is used to select one or more of these functions. Five sockets are used to accept ROMs with these various routines. The console emulator and diagnostic ROM socket (E20) accepts ROMs which contain either a console emulator and diagnostic routine or just a diagnostic routine for the PDP-11/60 and 11/70. The last two characters in the pattern number for these ROMs will always be "F1." The other four sockets (locations E32, E33, E34, and E35) accept ROMs which contain bootstrap programs. The last two characters of the pattern number for boot ROMs will always be "A9." Although it is not required that a particular device boot ROM go in a particular socket, the order in which these sockets are filled is important. Table B-1 shows the necessary installation order for Boot ROMs.

**Table B-1 Boot ROM Installation Order**

| Order of Installation | ROM Socket # | Location # |
|---|---|---|
| First Boot ROM* | E35 | 1 |
| Second Boot ROM | E33 | 2 |
| Third Boot ROM | E34 | 3 |
| Fourth Boot ROM | E32 | 4 |

*On PDP-11/60 if only one ROM, then it must be installed in ROM location #2 to do a power-up boot.

# APPENDIX C
# M9312 ADDRESS OFFSET SWITCH BANK

## C.1 INTRODUCTION

Tables C-1 through C-11 contain the necessary information to boot a particular device either on power-up boot, pushbutton boot, or console load address start sequence. A more detailed explanation of these switches and their functions may be found in Paragraphs 2.9 and 3.2, Table 3-1, Figure 2-11, and Appendix F.

Table C-1 ROM P/N 23-751A9

| Device to be Booted | ROM Location from App B | Diag-nostic | Unit | Starting Address* | Switch Settings S1–for Power-Up Boot or Push-Button Boot of Device | | | | | | | | | | SWR Code |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | |
| RL01 | ROM 1 | NO | 0 | 173004 | OFF | † | OFF | OFF | OFF | OFF | OFF | OFF | ON | OFF | NA |
| RL01 | ROM 1 | YES | 0 | 173006 | OFF | † | OFF | OFF | OFF | OFF | OFF | OFF | ON | ON | 012 |
| RL01 | ROM 2 | NO | 0 | 173204 | OFF | † | OFF | ON | OFF | OFF | OFF | OFF | ON | OFF | NA |
| RL01 | ROM 2 | YES | 0 | 173206 | OFF | † | OFF | ON | OFF | OFF | OFF | OFF | ON | ON | 212 |
| RL01 | ROM 3 | NO | 0 | 173404 | OFF | † | ON | OFF | OFF | OFF | OFF | OFF | ON | OFF | NA |
| RL01 | ROM 3 | YES | 0 | 173406 | OFF | † | ON | OFF | OFF | OFF | OFF | OFF | ON | ON | 412 |
| RL01 | ROM 4 | NO | 0 | 173604 | OFF | † | ON | ON | OFF | OFF | OFF | OFF | ON | OFF | NA |
| RL01 | ROM 4 | YES | 0 | 173606 | OFF | † | ON | ON | OFF | OFF | OFF | OFF | ON | ON | 612 |

*The starting address for the console load address and start sequence.

† S1-2: when ON, power-up boot is enabled; when OFF, power-up boot disabled. (Must be OFF for PDP-11/60. See PDP-11/60 documentation for power-up boot.)

**NOTES**

**If diagnostics are to be run, a CPU diagnostic ROM must be in location E20, and jumper W8 must be out.**

**When booting from the console emulator, a device mnemonic and unit number are required. If unit number is not entered, it is assumed to be zero. Listed below are the device mnemonic and unit numbers supported by this ROM, and the device CSR address assumed by the boot code.**

| Device | Mnemonic | Unit | CSR |
|---|---|---|---|
| RL01 | DL | 0–3 | 774400 |

Table C-2 ROM P/N 23-752A9

| Device to be Booted | ROM Location from App B | Diag-nostic | Unit | Starting Address* | Switch Settings S1 – for Power-Up Boot or Push-Button Boot of Device | | | | | | | | | | SWR Code |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | |
| RK06/07 | ROM 1 | NO | 0 | 173004 | OFF | † | OFF | OFF | OFF | OFF | OFF | OFF | ON | OFF | NA |
| RK06/07 | ROM 1 | YES | 0 | 173006 | OFF | † | OFF | OFF | OFF | OFF | OFF | OFF | ON | ON | 012 |
| RK06/07 | ROM 2 | NO | 0 | 173204 | OFF | † | OFF | ON | OFF | OFF | OFF | OFF | ON | OFF | NA |
| RK06/07 | ROM 2 | YES | 0 | 173206 | OFF | † | OFF | ON | OFF | OFF | OFF | OFF | ON | ON | 212 |
| RK06/07 | ROM 3 | NO | 0 | 173404 | OFF | † | ON | OFF | OFF | OFF | OFF | OFF | ON | OFF | NA |
| RK06/07 | ROM 3 | YES | 0 | 173406 | OFF | † | ON | OFF | OFF | OFF | OFF | OFF | ON | ON | 412 |
| RK06/07 | ROM 4 | NO | 0 | 173604 | OFF | † | ON | ON | OFF | OFF | OFF | OFF | ON | OFF | NA |
| RK06/07 | ROM 4 | YES | 0 | 173606 | OFF | † | ON | ON | OFF | OFF | OFF | OFF | ON | ON | 612 |

*The starting address for the console load addres and start sequence.
† S1-2: when ON, power-up boot is enabled; whn OFF, power-up boot disabled. (Must be OFF for PDP-11/60. See PDP-11/60 documentation for power-up boot.)

**NOTES**

**If diagnostics are to be run, a CPU diagnostic ROM must be in location E20, and jumper W8 must be out.**

**When booting from the console emulator, a device mnemonic and unit number are required. If unit number is not entered, it is assumed to be zero. Listed below are the device mnemonic and unit numbers supported by this ROM, and the device CSR address assumed by the boot code.**

| Device | Mnemonic | Unit | CSR |
|---|---|---|---|
| RK06/07 | DM | 0–7 | 777440 |

Table C-3 ROM P/N 23-753A9

| Device to be Booted | ROM Location from App B | Diag-nostic | Unit | Starting Address* | Switch Settings S1 –for Power-Up Boot or Push-Button Boot of Device | | | | | | | | | | SWR Code |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | |
| RX01 | ROM 1 | NO | 0 | 173004 | OFF | † | OFF | OFF | OFF | OFF | OFF | OFF | ON | OFF | NA |
| RX01 | ROM 1 | YES | 0 | 173006 | OFF | † | OFF | OFF | OFF | OFF | OFF | OFF | ON | ON | 012 |
| RX01 | ROM 2 | NO | 0 | 173204 | OFF | † | OFF | ON | OFF | OFF | OFF | OFF | ON | OFF | NA |
| RX01 | ROM 2 | YES | 0 | 173206 | OFF | † | OFF | ON | OFF | OFF | OFF | OFF | ON | ON | 212 |
| RX01 | ROM 3 | NO | 0 | 173404 | OFF | † | ON | OFF | OFF | OFF | OFF | OFF | ON | OFF | NA |
| RX01 | ROM 3 | YES | 0 | 173406 | OFF | † | ON | OFF | OFF | OFF | OFF | OFF | ON | ON | 412 |
| RX01 | ROM 4 | NO | 0 | 173604 | OFF | † | ON | ON | OFF | OFF | OFF | OFF | ON | OFF | NA |
| RX01 | ROM 4 | YES | 0 | 173606 | OFF | † | ON | ON | OFF | OFF | OFF | OFF | ON | ON | 612 |

*The starting address for the console load address and start sequence.

† S1-2: when ON, power-up boot is enabled; when OFF, power-up boot disabled. (Must be OFF for PDP-11/60. See PDP-11/60 documentation for power-up boot.)

**NOTES**

**If diagnostics are to be run, a CPU diagnostic ROM must be in location E20, and jumper W8 must be out.**

**When booting from the console emulator, a device mnemonic and unit number are required. If unit number is not entered, it is assumed to be zero. Listed below are the device mnemonic and unit numbers supported by this ROM, and the device CSR address assumed by the boot code.**

| Device | Mnemonic | Unit | CSR |
|---|---|---|---|
| RX01 | DX | 0–1 | 777170 |

Table C-4 ROM P/N 23-755A9

| Device to be Booted | ROM Location from App B | Diag-nostic | Unit | Starting Address* | Switch Settings S1–for Power-Up Boot or Push-Button Boot of Device | | | | | | | | | | SWR Code |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | |
| RP02/03 | ROM 1 | NO | 0 | 173004 | OFF | † | OFF | OFF | OFF | OFF | OFF | OFF | ON | OFF | NA |
| RP02/03 | ROM 1 | YES | 0 | 173006 | OFF | † | OFF | OFF | OFF | OFF | OFF | OFF | ON | ON | 012 |
| RP02/03 | ROM 2 | NO | 0 | 173204 | OFF | † | OFF | ON | OFF | OFF | OFF | OFF | ON | OFF | NA |
| RP02/03 | ROM 2 | YES | 0 | 173206 | OFF | † | OFF | ON | OFF | OFF | OFF | OFF | ON | ON | 212 |
| RP02/03 | ROM 3 | NO | 0 | 173404 | OFF | † | ON | OFF | OFF | OFF | OFF | OFF | ON | OFF | NA |
| RP02/03 | ROM 3 | YES | 0 | 173406 | OFF | † | ON | OFF | OFF | OFF | OFF | OFF | ON | ON | 412 |
| RP02/03 | ROM 4 | NO | 0 | 173604 | OFF | † | ON | ON | OFF | OFF | OFF | OFF | ON | OFF | NA |
| RP02/03 | ROM 4 | YES | 0 | 173606 | OFF | † | ON | ON | OFF | OFF | OFF | OFF | ON | ON | 612 |
| RP04/05/06 RM02/03 | ROM 1 | NO | 0 | 173050 | OFF | † | OFF | OFF | OFF | ON | OFF | ON | OFF | OFF | NA |
| RP04/05/06 RM02/03 | ROM 1 | YES | 0 | 173052 | OFF | † | OFF | OFF | OFF | ON | OFF | ON | OFF | ON | 056 |
| RP04/05/06 RM02/03 | ROM 2 | NO | 0 | 173250 | OFF | † | OFF | ON | OFF | ON | OFF | ON | OFF | OFF | NA |
| RP04/05/06 RM02/03 | ROM 2 | YES | 0 | 173252 | OFF | † | OFF | ON | OFF | ON | OFF | ON | OFF | ON | 256 |
| RP04/05/06 RM02/03 | ROM 3 | NO | 0 | 173450 | OFF | † | ON | OFF | OFF | ON | OFF | ON | OFF | OFF | NA |
| RP04/05/06 RM02/03 | ROM 3 | YES | 0 | 173452 | OFF | † | ON | OFF | OFF | ON | OFF | ON | OFF | ON | 456 |
| RP04/05/06 RM02/03 | ROM 4 | NO | 0 | 173650 | OFF | † | ON | ON | OFF | ON | OFF | ON | OFF | OFF | NA |
| RP04/05/06 RM02/03 | ROM 4 | YES | 0 | 173652 | OFF | † | ON | ON | OFF | ON | OFF | ON | OFF | ON | 656 |

*The starting address for the console load address and start sequence.

† S1-2: when ON, power-up boot is enabled; when OFF, power-up boot disabled. (Must be OFF for PDP-11/60. See PDP-11/60 documentation for power-up boot.)

**NOTES**

**If diagnostics are to be run, a CPU diagnostic ROM must be in location E20, and jumper W8 must be out.**

**When booting from the console emulator, a device mnemonic and unit number are required. If unit number is not entered, it is assumed to be zero. Listed below are the device mnemonic and unit numbers supported by this ROM, and the device CSR address assumed by the boot code.**

| Device | Mnemonic | Unit | CSR |
|---|---|---|---|
| RP02/03 | DP | 0–7 | 776714 |
| RP04/05/06 RM02/03 | DB | 0–7 | 776700 |

Table C-5 ROM P/N 23-756A9

| Device to be Booted | ROM Location from App B | Diag-nosic | Unit | Starting Address* | Switch Settings S1 – for Power-Up Boot or Push-Button Boot of Device | | | | | | | | | | SWR Code |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | |
| RK03/05/05J | ROM 1 | NO | 0 | 173004 | OFF | † | OFF | OFF | OFF | OFF | OFF | OFF | ON | OFF | NA |
| RK03/05/05J | ROM 1 | NO | 2 | 173164 | OFF | † | OFF | OFF | ON | ON | ON | OFF | ON | OFF | NA |
| RK03/05/05J | ROM 1 | YES | 0 | 173006 | OFF | † | OFF | OFF | OFF | OFF | OFF | OFF | ON | ON | 012 |
| RK03/05/05J | ROM 2 | NO | 0 | 173204 | OFF | † | OFF | ON | OFF | OFF | OFF | OFF | ON | OFF | NA |
| RK03/05/05J | ROM 2 | NO | 2 | 173364 | OFF | † | OFF | ON | ON | ON | ON | OFF | ON | OFF | NA |
| RK03/05/05J | ROM 2 | YES | 0 | 173206 | OFF | † | OFF | ON | OFF | OFF | OFF | OFF | ON | ON | 212 |
| RK03/05/05J | ROM 3 | NO | 0 | 173404 | OFF | † | ON | OFF | OFF | OFF | OFF | OFF | ON | OFF | NA |
| RK03/05/05J | ROM 3 | NO | 2 | 173564 | OFF | † | ON | OFF | ON | ON | ON | OFF | ON | OFF | NA |
| RK03/05/05J | ROM 3 | YES | 0 | 173406 | OFF | † | ON | OFF | OFF | OFF | OFF | OFF | ON | ON | 412 |
| RK03/05/05J | ROM 4 | NO | 0 | 173604 | OFF | † | ON | ON | OFF | OFF | OFF | OFF | ON | OFF | NA |
| RK03/05/05J | ROM 4 | NO | 2 | 173764 | OFF | † | ON | ON | ON | ON | ON | OFF | ON | OFF | NA |
| RK03/05/05J | ROM 4 | YES | 0 | 173606 | OFF | † | ON | ON | OFF | OFF | OFF | OFF | ON | ON | 612 |
| TU55/56 | ROM 1 | NO | 0 | 173034 | OFF | † | OFF | OFF | OFF | OFF | ON | ON | ON | OFF | NA |
| TU55/56 | ROM 1 | YES | 0 | 173036 | OFF | † | OFF | OFF | OFF | OFF | ON | ON | ON | ON | 042 |
| TU55/56 | ROM 2 | NO | 0 | 173234 | OFF | † | OFF | ON | OFF | OFF | ON | ON | ON | OFF | NA |
| TU55/56 | ROM 2 | YES | 0 | 173236 | OFF | † | OFF | ON | OFF | OFF | ON | ON | ON | ON | 242 |
| TU55/56 | ROM 3 | NO | 0 | 173434 | OFF | † | ON | OFF | OFF | OFF | ON | ON | ON | OFF | NA |
| TU55/56 | ROM 3 | YES | 0 | 173436 | OFF | † | ON | OFF | OFF | OFF | ON | ON | ON | ON | 442 |
| TU55/56 | ROM 4 | NO | 0 | 173634 | OFF | † | ON | ON | OFF | OFF | ON | ON | ON | OFF | NA |
| TU55/56 | ROM 4 | YES | 0 | 173636 | OFF | † | ON | ON | OFF | OFF | ON | ON | ON | ON | 642 |

*The starting address for the console load address and start sequence.

† S1-2: when ON, power-up boot is enabled; when OFF, power-up boot disabled. (Must be OFF for PDP-11/60. See PDP-11/60 documentation for power-up boot.)

NOTES

If diagnostics are to be run, a CPU diagnostic ROM must be in location E20, and jumper W8 must be out.

When booting from the console emulator, a device mnemonic and unit number are required. If unit number is not entered, it is assumed to be zero. Listed below are the device mnemonic and unit numbers supported by this ROM, and the device CSR address assumed by the boot code.

| Device | Mnemonic | Unit | CSR |
|---|---|---|---|
| RK03/05/05J | DK | 0–7 | 777404 |
| TU55/56 | DT | 0–7 | 777342 |

Table C-6 ROM P/N 23-757A9

| Device to be Booted | ROM Location from APP B | Diag-nostic | Unit | Starting Address* | Switch Settings S1 – for Power-Up Boot or Push-Button Boot of Device | | | | | | | | | | SWR Code |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | |
| TU16 | ROM 1 | NO | 0 | 173004 | OFF | † | OFF | OFF | OFF | OFF | OFF | OFF | ON | OFF | NA |
| TU16 | ROM 1 | YES | 0 | 173006 | OFF | † | OFF | OFF | OFF | OFF | OFF | OFF | ON | ON | 012 |
| TU16 | ROM 2 | NO | 0 | 173204 | OFF | † | OFF | ON | OFF | OFF | OFF | OFF | ON | OFF | NA |
| TU16 | ROM 2 | YES | 0 | 173206 | OFF | † | OFF | ON | OFF | OFF | OFF | OFF | ON | ON | 212 |
| TU16 | ROM 3 | NO | 0 | 173404 | OFF | † | ON | OFF | OFF | OFF | OFF | OFF | ON | OFF | NA |
| TU16 | ROM 3 | YES | 0 | 173406 | OFF | † | ON | OFF | OFF | OFF | OFF | OFF | ON | ON | 412 |
| TU16 | ROM 4 | NO | 0 | 173604 | OFF | † | ON | ON | OFF | OFF | OFF | OFF | ON | OFF | NA |
| TU16 | ROM 4 | YES | 0 | 173606 | OFF | † | ON | ON | OFF | OFF | OFF | OFF | ON | ON | 612 |

*The starting address for the console bad address and start sequence.
† S1-2: when ON, power-up boot is enabled; when OFF, power-up boot disabled. (Must be OFF for PDP-11/60. See PDP-11/60 documentation for power-up boot.)

**NOTES**

**If diagnostics are to be run, a CPU diagnostic ROM must be in location E20, and jumper W8 must be out.**

**When booting from the console emulator, a device mnemonic and unit number are required. If unit number is not entered, it is assumed to be zero. Listed below are the device mnemonic and unit numbers supported by this ROM, and the device CSR address assumed by the boot code.**

| Device | Mnemonic | Unit | CSR |
|---|---|---|---|
| TU16 | MM | 0–7 | 772440 |

Table C-7 ROM P/N 23-758A9

| Device to be Booted | ROM Location from App B | Diag-nostic | Unit | Starting Address* | Switch Settings S1 – for Power-Up Boot or Push-Button Boot of Device | | | | | | | | | | SWR Code |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | |
| TU10 TS03 | ROM 1 | NO | 0 | 173004 | OFF | † | OFF | OFF | OFF | OFF | OFF | OFF | ON | OFF | NA |
| TU10 TS03 | ROM 1 | YES | 0 | 173006 | OFF | † | OFF | OFF | OFF | OFF | OFF | OFF | ON | ON | 012 |
| TU10 TS03 | ROM 2 | NO | 0 | 173204 | OFF | † | OFF | ON | OFF | OFF | OFF | OFF | ON | OFF | NA |
| TU10 TS03 | ROM 2 | YES | 0 | 173206 | OFF | † | OFF | ON | OFF | OFF | OFF | OFF | ON | ON | 212 |
| TU10 TS03 | ROM 3 | NO | 0 | 173404 | OFF | † | ON | OFF | OFF | OFF | OFF | OFF | ON | OFF | NA |
| TU10 TS03 | ROM 3 | YES | 0 | 173406 | OFF | † | ON | OFF | OFF | OFF | OFF | OFF | ON | ON | 412 |
| TU10 TS03 | ROM 4 | NO | 0 | 173604 | OFF | † | ON | ON | OFF | OFF | OFF | OFF | ON | OFF | NA |
| TU10 TS03 | ROM 4 | YES | 0 | 173606 | OFF | † | ON | ON | OFF | OFF | OFF | OFF | ON | ON | 612 |

*The starting address for the console load address and start sequence.
† S1-2: when ON, power-up boot is enabled; when OFF, power-up boot disabled. (Must be OFF for PDP-11/60. See PDP-11/60 documentation for power-up boot.)

**NOTES**

**If diagnostics are to be run, a CPU diagnostic ROM must be in location E20, and jumper W8 must be out.**

**When booting from the console emulator, a device mnemonic and unit number are required. If unit number is not entered, it is assumed to be zero. Listed below are the device mnemonic and unit numbers supported by this ROM, and the device CSR address assumed by the boot code.**

| Device | Mnemonic | Unit | CSR |
|---|---|---|---|
| TU10 TS03 | MT | 0–7 | 772522 |

Table C-8 ROM P/N 23-759A9

| Device to be Booted | ROM Location from App B | Diag-nostic | Unit | Starting Address* | Switch Settings S1 – for Power-Up Boot or Push-Button Boot of Device | | | | | | | | | | SWR Code |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | |
| RS03/04 | ROM 1 | NO | 0 | 173004 | OFF | † | OFF | OFF | OFF | OFF | OFF | OFF | ON | OFF | NA |
| RS03/04 | ROM 1 | YES | 0 | 173006 | OFF | † | OFF | OFF | OFF | OFF | OFF | OFF | ON | ON | 012 |
| RS03/04 | ROM 2 | NO | 0 | 173204 | OFF | † | OFF | ON | OFF | OFF | OFF | OFF | ON | OFF | NA |
| RS03/04 | ROM 2 | YES | 0 | 173206 | OFF | † | OFF | ON | OFF | OFF | OFF | OFF | ON | ON | 212 |
| RS03/04 | ROM 3 | NO | 0 | 173404 | OFF | † | ON | OFF | OFF | OFF | OFF | OFF | ON | OFF | NA |
| RS03/04 | ROM 3 | YES | 0 | 173406 | OFF | † | ON | OFF | OFF | OFF | OFF | OFF | ON | ON | 412 |
| RS03/04 | ROM 4 | NO | 0 | 173604 | OFF | † | ON | ON | OFF | OFF | OFF | OFF | ON | OFF | NA |
| RS03/04 | ROM 4 | YES | 0 | 173606 | OFF | † | ON | ON | OFF | OFF | OFF | OFF | ON | ON | 612 |

*The starting address for the console load address and start sequence.
†S1-2: when ON, power-up boot is enabled; when OFF, power-up boot disabled. (Must be OFF for PDP-11/60. See PDP-11/60 documentation for power-up boot.)

**NOTES**

**If diagnostics are to be run, a CPU diagnostic ROM must be in location E20, and jumper W8 must be out.**

**When booting from the console emulator, a device mnemonic and unit number are required. If unit number is not entered, it is assumed to be zero. Listed below are the device mnemonic and unit numbers supported by this ROM, and the device CSR address assumed by the boot code.**

| Device | Mnemonic | Unit | CSR |
|---|---|---|---|
| RS03/04 | DS | 0–7 | 772040 |

Table C-9 ROM P/N 23-760A9

| Device to be Booted | ROM Location from App B | Diag-nostic | Unit | Starting Address* | Switch Settings S1 – for Power-Up Boot or Push-Button Boot of Device | | | | | | | | | | SWR Code |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | |
| PC05 | ROM 1 | NO | 0 | 173004 | OFF | † | OFF | OFF | OFF | OFF | OFF | OFF | ON | OFF | NA |
| PC05 | ROM 1 | YES | 0 | 173006 | OFF | † | OFF | OFF | OFF | OFF | OFF | OFF | ON | ON | 012 |
| PC05 | ROM 2 | NO | 0 | 173204 | OFF | † | OFF | ON | OFF | OFF | OFF | OFF | ON | OFF | NA |
| PC05 | ROM 2 | YES | 0 | 173206 | OFF | † | OFF | ON | OFF | OFF | OFF | OFF | ON | ON | 212 |
| PC05 | ROM 3 | NO | 0 | 173404 | OFF | † | ON | OFF | OFF | OFF | OFF | OFF | ON | OFF | NA |
| PC05 | ROM 3 | YES | 0 | 173406 | OFF | † | ON | OFF | OFF | OFF | OFF | OFF | ON | ON | 412 |
| PC05 | ROM 4 | NO | 0 | 173604 | OFF | † | ON | ON | OFF | OFF | OFF | OFF | ON | OFF | NA |
| PC05 | ROM 4 | YES | 0 | 173606 | OFF | † | ON | ON | OFF | OFF | OFF | OFF | ON | ON | 612 |
| LO SPD RDR | ROM 1 | NO | 0 | 173034 | OFF | † | OFF | OFF | OFF | OFF | ON | ON | ON | OFF | NA |
| LO SPD RDR | ROM 1 | YES | 0 | 173036 | OFF | † | OFF | OFF | OFF | OFF | ON | ON | ON | ON | 042 |
| LO SPD RDR | ROM 2 | NO | 0 | 173234 | OFF | † | OFF | ON | OFF | OFF | ON | ON | ON | OFF | NA |
| LO SPD RDR | ROM 2 | YES | 0 | 173236 | OFF | † | OFF | ON | OFF | OFF | ON | ON | ON | ON | 242 |
| LO SPD RDR | ROM 3 | NO | 0 | 173434 | OFF | † | ON | OFF | OFF | OFF | ON | ON | ON | OFF | NA |
| LO SPD RDR | ROM 3 | YES | 0 | 173436 | OFF | † | ON | OFF | OFF | OFF | ON | ON | ON | ON | 442 |
| LO SPD RDR | ROM 4 | NO | 0 | 173634 | OFF | † | ON | ON | OFF | OFF | ON | ON | ON | OFF | NA |
| LO SPD RDR | ROM 4 | YES | 0 | 173636 | OFF | † | ON | ON | OFF | OFF | ON | ON | ON | ON | 642 |

*The starting address for the console load address and start sequence.
† S1-2: when ON, power-up boot is enabled; when OFF, power-up boot disabled. (Must be OFF for PDP-11/60. See PDP-11/60 documentation for power-up boot.)

**NOTES**

**If diagnostics are to be run, a CPU diagnostic ROM must be in location E20, and jumper W8 must be out.**

**When booting from the console emulator, a device mnemonic and unit number are required. If unit number is not entered, it is assumed to be zero. Listed below are the device mnemonic and unit numbers supported by this ROM, and the device CSR address assumed by the boot code.**

| Device | Mnemonic | Unit | CSR |
|---|---|---|---|
| HI SPD RDR | PR | N/A | 777550 |
| LO SPD RDR | TT | N/A | 777560 |

Table C-10 ROM P/N 23-761A9

| Device to be Booted | ROM Location from App B | Diagnostic | Unit | Starting Address* | Switch Settings S1 – for Power-Up Boot or Push-Button Boot of Device | | | | | | | | | | SWR Code |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | |
| TU60 | ROM 1 | NO | 0 | 173004 | OFF | † | OFF | OFF | OFF | OFF | OFF | OFF | ON | OFF | NA |
| TU60 | ROM 1 | YES | 0 | 173006 | OFF | † | OFF | OFF | OFF | OFF | OFF | OFF | ON | ON | 012 |
| TU60 | ROM 2 | NO | 0 | 173204 | OFF | † | OFF | ON | OFF | OFF | OFF | OFF | ON | OFF | NA |
| TU60 | ROM 2 | YES | 0 | 173206 | OFF | † | OFF | ON | OFF | OFF | OFF | OFF | ON | ON | 212 |
| TU60 | ROM 3 | NO | 0 | 173404 | OFF | † | ON | OFF | OFF | OFF | OFF | OFF | ON | OFF | NA |
| TU60 | ROM 3 | YES | 0 | 173406 | OFF | † | ON | OFF | OFF | OFF | OFF | OFF | ON | ON | 412 |
| TU60 | ROM 4 | NO | 0 | 173604 | OFF | † | ON | ON | OFF | OFF | OFF | OFF | ON | OFF | NA |
| TU60 | ROM 4 | YES | 0 | 173606 | OFF | † | ON | ON | OFF | OFF | OFF | OFF | ON | ON | 612 |

*The starting address for the console load address and start sequence.
† S1-2: when ON, power-up boot is enabled; when OFF, power-up boot disabled. (Must be OFF for PDP-11/60. See PDP-11/60 documentation for power-up boot.)

**NOTES**

**If diagnostics are to be run, a CPU diagnostic ROM must be in location E20, and jumper W8 must be out.**

**When booting from the console emulator, a device mnemonic and unit number are required. If unit number is not entered, it is assumed to be zero. Listed below are the device mnemonic and unit numbers supported by this ROM, and the device CSR address assumed by the boot code.**

| Device | Mnemonic | Unit | CSR |
|---|---|---|---|
| TU60 | CT | 0–1 | 777550 |

Table C-11 ROM P/N 811A9

| Device to be Booted | ROM Location from App B | Diag-nostic | Unit | Starting Address* | Switch Settings S1 – for Power-Up Boot or Push-Button Boot of Device | | | | | | | | | | SWR Code |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | |
| RX02 | ROM 1 | NO | 0 | 173004 | OFF | † | OFF | OFF | OFF | OFF | OFF | OFF | ON | OFF | NA |
| RX02 | ROM 1 | YES | 0 | 173006 | OFF | † | OFF | OFF | OFF | OFF | OFF | OFF | ON | ON | 012 |
| RX02 | ROM 2 | NO | 0 | 173204 | OFF | † | OFF | ON | OFF | OFF | OFF | OFF | ON | OFF | NA |
| RX02 | ROM 2 | YES | 0 | 173206 | OFF | † | OFF | ON | OFF | OFF | OFF | OFF | ON | ON | 212 |
| RX02 | ROM 3 | NO | 0 | 173404 | OFF | † | ON | OFF | OFF | OFF | OFF | OFF | ON | OFF | NA |
| RX02 | ROM 3 | YES | 0 | 173406 | OFF | † | ON | OFF | OFF | OFF | OFF | OFF | ON | ON | 412 |
| RX02 | ROM 4 | NO | 0 | 173604 | OFF | † | ON | ON | OFF | OFF | OFF | OFF | ON | OFF | NA |
| RX02 | ROM 4 | YES | 0 | 173606 | OFF | † | ON | ON | OFF | OFF | OFF | OFF | ON | ON | 612 |

*The starting address for the console load address and start sequence.
†S1-2: when ON, power-up boot is enabled; when OFF, power-up boot disabled. (Must be OFF for PDP-11/60. See PDP-11/60 documentation for power-up boot.)

**NOTE**

**If diagnostics are to be run, a CPU diagnostic ROM must be in location E20, and jumper W8 must be out.**

**When booting from the console emulator, a device mnemonic and unit number are required. If unit number is not entered, it is assumed to be zero. Listed below are the device mnemonic and unit numbers supported by this ROM, and the device CSR address assumed by the boot code.**

| Device | Mnemonic | Unit | CSR |
|---|---|---|---|
| RX02 | DY | 0–1 | 777170 |

# APPENDIX D
# M9312 FASTON TAB CONNECTIONS

## D.1 INTRODUCTION

The M9312 Faston tabs are defined in Table D-1. One of the frequent applications of the M9312 is to replace the M9301. Table D-2 provides the necessary information needed to make the appropriate substitutions with respect to the Faston tabs.

**Table D-1 Faston Tab Description**

| Faston Tab | Function Performed |
|---|---|
| TP1 | Boot input |
| TP2 | Return for boot input |
| TP3 | Return for enable boot on power-up |
| TP4 | Enable boot on power-up |

**Table D-2 Faston Tab Substitution**

| Wire Connection | | | |
|---|---|---|---|
| From | | To | |
| Module | Faston Tab | Module | Faston Tab |
| M9301 | TP1 | M9312 | TP4 |
| M9301 | TP2 | M9312 | TP1 |
| M9301 | TP3 | M9312 | TP3 |

# APPENDIX E
# CROSS REFERENCE AND IDENTIFICATION TABLES

## E.1 CROSS REFERENCE TABLES

Table E-1 provides a guide used to identify a particular ROM by part number. Table E-2 is a cross reference table which identifies each device with its corresponding controller.

**Table E-1 Cross Reference ROM P/N to ROM Table Number**

| Part Number 23- | Function Performed by ROM | | | | See Table |
|---|---|---|---|---|---|
| 233F1 | Diagnostic for 11/60/70 ROM ID=B0 | | | | F-1 |
| 248F1 | ASCII Console and Diagnostic for 11/04/05/34/35/40/45/50/55 ROM ID=A0 | | | | F-2 |
| | The following ROMS are Bootstrap ROMs | | | | |
| | **Bootable Devices** | | | | |
| | **Mnemonic** | **First Device In ROM** | **Mnemonic** | **Second Device in ROM** | |
| 751A9 | DL | RL01 | NA | NA | C-1 |
| 752A9 | DM | RK06/07 | NA | NA | C-2 |
| 753A9 | DX | RX01 | NA | NA | C-3 |
| 811A9 | DY | RX02 | NA | NA | C-11 |
| 755A9 | DP | RP02/03 | DB | RP04/5/6 RM02/3 | C-4 |
| 750A9 | DK | RK03/05 | DT | TU55/56 | C-5 |
| 757A9 | MM | TU16/E16 TM02/3 | NA | NA | C-6 |
| 758A9 | MT | TU10/TS03 | NA | NA | C-7 |
| 759A9 | DS | RS03/04 | NA | NA | C-8 |
| 760A9 | PR | PC05 | TT | LO SPD RDR | C-9 |
| 761A9 | CT | TU60 | NA | NA | C-10 |

Table E-2 Cross Reference Device to Controller

| Device | Controller |
|---|---|
| RL01 | RL11 |
| RK06/07 | RK611 |
| RX01 | RX11 |
| RX02 | RX211 |
| RP02/03 | RP11C/E |
| RP04/05/06 RM02/03 | RH11/70 |
| RK03/05 | RK11C/D |
| TU55/56 | TC11 |
| TU16/E16 TM02/03 | RH11/70 |
| TU10/E10 TS03 | TM11/A11/B11 |
| RS03/04 | RH11/70 |
| PC05 (HI SPD RDR) | PC11/R11 |
| LO SPD RDR (ASR33) | DL11A/W |
| TU60 | TA11 |

## E.2 ROM IDENTIFICATION

When the ROM configuration of an M9312 already installed in a system is not known, it is desirable to identify the ROM configuration without removing the module. This can be accomplished by running diagnostic CZM9B, or by examining the data in five specific locations and using Table E-3 to identify the ROM. The locations are as follows:

1. 765774 Diagnostic ROM
2. 773000 ROM 1
3. 773200 ROM 2
4. 773400 ROM 3
5. 773600 ROM 4

By comparing the data observed at the above locations with Table E-3 you can identify the type and location of each ROM in the module.

Table E-3 ROM Identification

| Octal Data | Mnemonic | P/N 23- | See Table |
|---|---|---|---|
| 040460 | A0 | 248F1 | F-2 |
| 041060 | B0 | 233F1 | F-1 |
| 041524 | CT | 761A9 | C-10 |
| 042113 | CI | 761A9 | C-10 |
| 042113 | DK | 756A9 | C-5 |
| 042114 | DL | 751A9 | C-1 |
| 042115 | DM | 752A9 | C-2 |
| 042120 | DP | 755A9 | C-4 |
| 042123 | DS | 759A9 | C-8 |
| 042130 | DX | 753A9 | C-3 |
| 042131 | DY | 811A9 | C-11 |
| 046515 | MM | 757A9 | C-6 |
| 046524 | MT | 758A9 | C-7 |
| 056122 | PB | 760A9 | C-9 |
| 177776 | This is a Continuation ROM of a Multiple-ROM Boot | | |
| XXX777 | Bad ROM or No ROM Present | | |

# APPENDIX F
# DIAGNOSTIC AND CONSOLE EMULATOR ROMS

## F.1 DIAGNOSTIC ROM (P/N 23-233F1)

There are no special M9312 switch settings that pertain to this ROM. The only way these diagnostics can be executed is by entering a bootstrap at the entry point which calls for diagnostics to be run. This ROM allows the user to boot via the console switch register. This can be done as follows:

1. Load Address 765744.
2. Set switch register according to Table F-1
3. Start.

Table F-1 Console Switch Register Settings for Diagnostic ROM

| 15 | 14 | 13 | 12 | 11 | 10 | 09 | 08 | 07 | 06 | 05 | 04 | 03 | 02 | 01 | 00 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| NA | NA | NA | NA | | | | | | | | | | | | |
| | | | | Octal Unit Number | | | SWR Code from Boot ROM Tables | | | | | | | | |

## F.2 CONSOLE EMULATOR AND DIAGNOSTIC ROM (P/N 23-248F1)

This ROM contains an ASCII console emulator routine and diagnostics for use with PDP-11/04, 11/05, 11/34, 11/35, 11/40, 11/45, 11/50, 11/55. To enter the ASCII console via power-up boot or pushbutton boot, the M9312 switches must be set according to Table F-2.

Table F-2 Switch Settings for ASCII Console and Diagnostic ROM

| Function to be Booted | ROM Location | Diag-nostic | Starting Address* | S1 Switch Settings for Power-Up Boot or Push-Button Boot | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| Console Emulator | E20 | NO | 165144 | ON | † | OFF | OFF | ON | ON | OFF | OFF | ON | OFF |
| Console Emulator | E20 | YES | 165020 | ON | † | OFF | OFF | OFF | OFF | ON | OFF | OFF | OFF |

*The starting address for the console load address and start sequence.
†S1-2: When ON, power-up boot is enabled; when OFF, power-up boot is disabled.

M9312 BOOTSTRAP/TERMINATOR
TECHNICAL MANUAL
EK-M9312-TM-001

# Reader's Comments

**Your comments and suggestions will help us in our continuous effort to improve the quality and usefulness of our publications.**

What is your general reaction to this manual? In your judgment is it complete, accurate, well organized, well written, etc.? Is it easy to use? _____

_____

_____

_____

What features are most useful? _____

_____

_____

_____

What faults or errors have you found in the manual? _____

_____

_____

_____

Does this manual satisfy the need you think it was intended to satisfy? _____

Does it satisfy *your* needs? _____ Why? _____

_____

_____

_____

☐ Please send me the current copy of the *Technical Documentation Catalog*, which contains information on the remainder of DIGITAL's technical documentation.

Name _____ Street _____
Title _____ City _____
Company _____ State/Country _____
Department _____ Zip _____

Additional copies of this document are available from:

Digital Equipment Corporation
444 Whitney Street
Northboro, Ma 01532
Attention: Communications Services (NR2/M15)
Customer Services Section

Order No. EK-M9312-TM-001

Fold Here

Do Not Tear - Fold Here and Staple

FIRST CLASS
PERMIT NO. 33
MAYNARD, MASS.

BUSINESS REPLY MAIL
NO POSTAGE STAMP NECESSARY IF MAILED IN THE UNITED STATES

Postage will be paid by:

Digital Equipment Corporation
Technical Documentation Department
Maynard, Massachusetts 01754

digital

digital equipment corporation

Printed in U.S.A.