

# VAX 4000 Model 100

---

## Customer Technical Information

Order Number: EK-467AA-TI. A01

**September 1992**

This manual provides technical information for the VAX 4000 Model 100 system. It also gives a list of the console commands, and specifications for the system unit, the CPU, the internal SCSI and DSSI devices.

**Revision Information:**            This is a new manual.

**Digital Equipment Corporation  
Maynard, Massachusetts**

---

**September 1992**

The information in this document is subject to change without notice and should not be construed as a commitment by Digital Equipment Corporation. Digital Equipment Corporation assumes no responsibility for any errors that may appear in this document.

The software described in this document is furnished under a license and may be used or copied only in accordance with the terms of such license.

No responsibility is assumed for the use or reliability of software on equipment that is not supplied by Digital Equipment Corporation or its affiliated companies.

Restricted Rights: Use, duplication, or disclosure by the U.S. Government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013.

© Digital Equipment Corporation 1992.

All Rights Reserved.

The postpaid Reader's Comments forms at the end of this document request your critical evaluation to assist in preparing future documentation.

The following are trademarks of Digital Equipment Corporation: RX, ThinWire, VAX, VAXcluster, VAX DOCUMENT, VMS, VT, and the DIGITAL logo.

S2011

This document was prepared using VAX DOCUMENT, Version 2.1.

---

# Contents

<b>Preface</b> .....	vii
<b>1 System Description</b>	
1.1 Model 100 System .....	1-1
1.1.1 Model 100 VAX Architecture Support .....	1-3
1.2 Internal Mass Storage Devices .....	1-5
1.3 Communications Devices .....	1-6
1.3.1 Asynchronous Communications Devices .....	1-6
1.3.2 Synchronous Communications Devices .....	1-6
<b>2 Console Security Feature and System Defaults</b>	
2.1 Returning to Console Mode .....	2-2
2.2 Using the Console Port .....	2-4
2.3 Console Security Feature .....	2-5
2.3.1 Setting the Security Password .....	2-6
2.3.2 Enabling the Console Security Feature .....	2-7
2.3.3 Logging in to Privileged Console Mode .....	2-7
2.3.4 Changing the Security Password .....	2-8
2.3.5 Disabling the Console Security Feature .....	2-8
2.3.6 Exiting from Privileged Console Mode .....	2-9
2.4 Setting the Default Boot Device .....	2-10
2.5 Setting the Default Recovery Action .....	2-11
<b>3 Console Commands</b>	
3.1 BOOT .....	3-1
3.2 CONTINUE .....	3-3
3.3 DEPOSIT .....	3-3
3.4 EXAMINE .....	3-4
3.5 FIND .....	3-6
3.6 HALT .....	3-6

3.7	HELP .....	3-7
3.8	INITIALIZE .....	3-9
3.9	LOGIN .....	3-10
3.10	MOVE .....	3-11
3.11	NEXT .....	3-12
3.12	REPEAT .....	3-14
3.13	SEARCH .....	3-15
3.14	SET .....	3-17
3.15	SHOW .....	3-18
3.16	START .....	3-21
3.17	TEST .....	3-21
3.18	UNJAM .....	3-25
3.19	X—Binary Load and Unload .....	3-25
3.20	! (Comment) .....	3-27

## 4 Hardware Specifications

4.1	System Unit Specifications .....	4-1
4.2	Internal DSSI Device Specifications .....	4-5
4.3	KA52-AA CPU Specifications .....	4-8

## Index

## Figures

1-1	Block Diagram of the VAX 4000 Model 100 .....	1-2
2-1	Returning to Console Mode .....	2-3
2-2	Console Port .....	2-4

## Tables

1-1	Supported Internal Mass Storage Devices .....	1-5
1-2	Supported Asynchronous Devices .....	1-6
1-3	Supported Synchronous Devices .....	1-6
1-4	Synchronous Communications Option Cable Part Numbers .....	1-6
2-1	Alternative Default Boot Devices .....	2-10
2-2	Default Recovery Actions and Associated Values .....	2-11
4-1	System Specifications: VAX 400 Model 100 .....	4-2
4-2	System Unit Metrics .....	4-3

4-3	System Storage Conditions .....	4-3
4-4	System Operating Conditions and Nonoperating Conditions .....	4-3
4-5	RF31T and RF35 Hard Disk Drive Specifications .....	4-5
4-6	TZ30 Tape Drive Specifications .....	4-6
4-7	TZK10 QIC Tape Drive Specifications .....	4-6
4-8	TLZ06 Cassette Tape Drive Specifications .....	4-7
4-9	RRD42 Compact Disc Drive Specifications .....	4-7

---

# Preface

This manual provides technical information for the VAX 4000 Model 100 system. It also gives a list of the console commands, and specifications for the system unit, the CPU, and the internal SCSI devices.

## Audience

This manual is intended for experienced users, for example, system programmers or system managers.

## Structure of This Manual

This manual is divided into four chapters and an index:

- Chapter 1 describes technical information about the Model 100 system.
- Chapter 2 describes the console security feature and how to set system defaults.
- Chapter 3 describes the console commands.
- Chapter 4 gives specifications for the system unit and for internal DSSI devices.

## Additional Information

See the *VAX 4000 Model 100 Operator Information* manual for the list of associated and related documents.

## Conventions

The following conventions are used in this manual:

Convention	Description
MONOSPACE	Text displayed on the screen is shown in monospace type.
<b>boldface type</b>	Boldface type in examples indicates user input. Boldface type in text indicates the first instance of terms defined either in the text, in the glossary, or both.
<i>italic type</i>	Italic type emphasizes important information, indicates variables, and indicates complete titles of manuals.
<i>nn nnn.nnn nn</i>	A space character separates digits in numerals with 5 or more digits. For example, <i>10 000</i> equals <i>ten thousand</i> .
<i>n.nn</i>	A period in numerals signals the decimal point indicator. For example, <i>1.75</i> equals <i>one and three-fourths</i> .
UPPERCASE	Words in uppercase indicate a command.
lowercase	In format descriptions, words in lowercase indicate parameters or arguments to be specified by the user.
	In command syntax descriptions, a vertical bar   separates similar options, one of which you can choose.
<b>Note</b>	A note contains information of special importance to the reader.
Ctrl/ <i>x</i>	Ctrl/ <i>x</i> indicates that you hold down the Ctrl key while you press another key or mouse button (indicated here by <i>x</i> ).
<i>x</i>	A lowercase italic <i>x</i> indicates the generic use of a letter. For example, <i>xxx</i> indicates any combination of three alphabetic characters.
<i>n</i>	A lowercase italic <i>n</i> indicates the generic use of a number. For example, <i>19nn</i> indicates a 4-digit number in which the last 2 digits are unknown.
{ }	In format descriptions, braces indicate required elements. You must choose one of the elements.
[ ]	In format descriptions, brackets indicate optional elements. You can choose none, one, or all of the options.

# 1

---

## System Description

This chapter gives a technical description of the VAX 4000 Model 100. This chapter includes information on the following subjects:

- Model 100 system
- Internal mass storage devices
- Communications devices
- Digital Storage Systems Interconnect (DSSI)
- KA52-AA CPU Specifications

### 1.1 Model 100 System

The Model 100 system uses the KA52 CPU module which is the primary component in the Model 100 system. The module contains the following components:

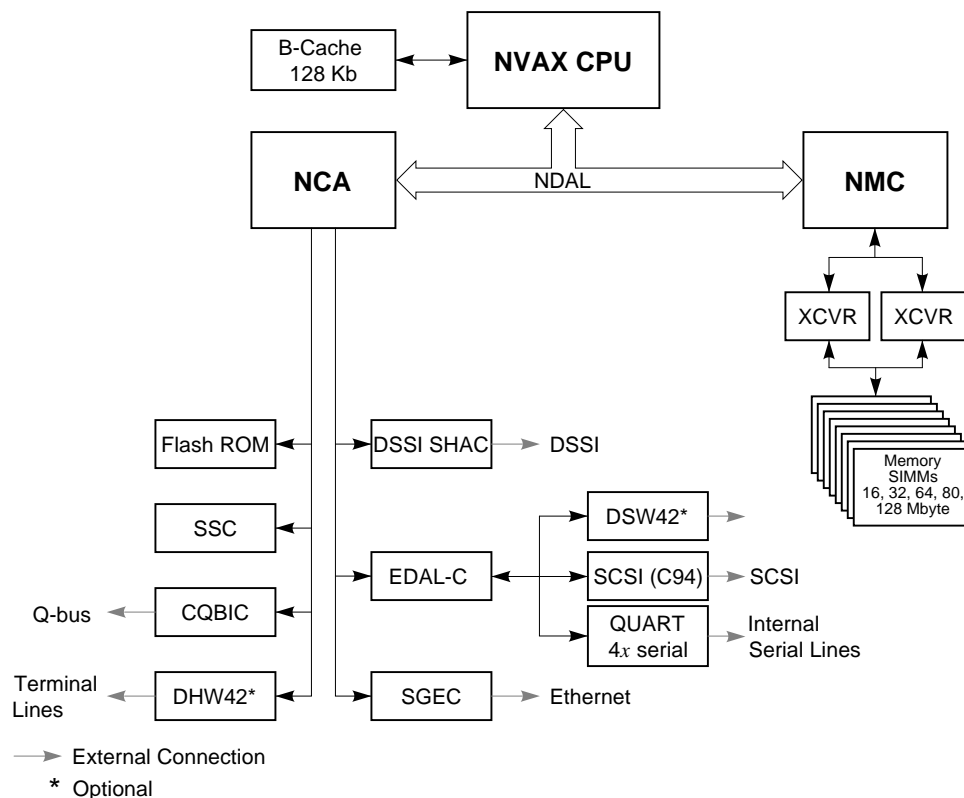
- NVAX processor chip with a 286 MHz clock
- NVAX memory controller (NMC) memory management chip
- NVAX CP bus adapter (NCA)
- DSSI support via the (SHAC) chip
- Q-bus support via the (CQBIC) chip
- SCSI controller and SQWF buffer chip
- Time-of-year (TOY) clock SSC chip
- DC541 (SGEC) chip Ethernet controller for standard or ThinWire Ethernet
- DC7085 (QUART) serial line controller (four serial lines, one with modem control)
- Support for up to 128 M bytes of memory modules



- Support for optional asynchronous communications devices, which provide either 8 or 16 additional DEC423 ports, or 8 additional asynchronous modem control ports
- Support for optional synchronous communications devices, using DSW42 and DHW42 options which provide two synchronous ports

Figure 1-1 is a block diagram of the VAX 4000 Model 100 system.

**Figure 1-1 Block Diagram of the VAX 4000 Model 100**



MLO-009354

### 1.1.1 Model 100 VAX Architecture Support

The KA52 CPU module supports the following VAX data types:

- byte, word, longword, quadword
- character string
- variable-length bit field
- absolute queues
- self-relative queues
- `f_floating point`, `d_floating point`, and `g_floating point`

The operating system uses software emulation to support other VAX data types.

The KA52 CPU module supports the following VAX instructions:

- integer, arithmetic and logical
- address
- variable-length bit field
- control
- procedure call
- miscellaneous
- queue
- character string instructions:
  - MOVC3/MOVC5
  - CMPC3/CMPC5
  - LOCC
  - SCANC
  - SKPC
  - SPANC
- Operating system support
- `f_floating point`, `d_floating point`, and `g_floating point`

The NVAX processor chip provides special microcode assistance to aid the macrocode emulation of the following instruction groups:

- Character string (other than those mentioned previously)
- Decimal string
- CRC
- EDITPC

The operating system uses software emulation to support other VAX instructions.

## 1.2 Internal Mass Storage Devices

Table 1–1 shows the internal mass storage devices that are supported by the Model 100 system.

**Table 1–1 Supported Internal Mass Storage Devices**

Device	Size (inches)	Capacity (bytes)	Description
RF31T	3.5"	381M	Hard disk drive
RF35	3.5"	852M	Hard disk drive
TZ30	5.25	95M	Tape drive
TZK10	5.25	320M or 525M	Tape drive
TLZ06	3.5	Up to 4.0G	Tape drive
RRD42	5.25	600M	CDROM drive

The Model 100 system supports a maximum of three internal DSSI devices, two SCSI devices which are removable media devices. One of the RF-series disks contains factory installed software (FIS). Chapter 4 gives the specifications for each internal DSSI device.

## 1.3 Communications Devices

The Model 100 system supports asynchronous and synchronous communications devices.

### 1.3.1 Asynchronous Communications Devices

Table 1-2 lists the asynchronous devices supported by Model 100 system.

**Table 1-2 Supported Asynchronous Devices**

Device	Description
DHW42-AA	Eight-line DEC423 asynchronous option
DHW42-BA	Sixteen-line DEC423 asynchronous option
DHW42-CA	Eight-line EIA-232 modem asynchronous option
DHW42-UP	Eight-line to 16-line upgrade of the DEC423 asynchronous option

### 1.3.2 Synchronous Communications Devices

Table 1-3 lists the synchronous devices supported by the Model 100 system.

**Table 1-3 Supported Synchronous Devices**

Device	Description
DSW42-AA	Two-line EIA-232/V.24 synchronous module

If you order a different synchronous option cable, you can use different interface standards with the synchronous communications module. Table 1-4 lists each standard and the part number of the corresponding option cable.

**Table 1-4 Synchronous Communications Option Cable Part Numbers**

Standard	Option Cable Part Number
EIA-232/V.24	BC19D-02
EIA-423/V.10	BC19E-02
EIA-422/V.11	BC19B-02

# 2

---

## Console Security Feature and System Defaults

This chapter describes how to set system defaults and how to use the console security feature. It includes information on the following subjects:

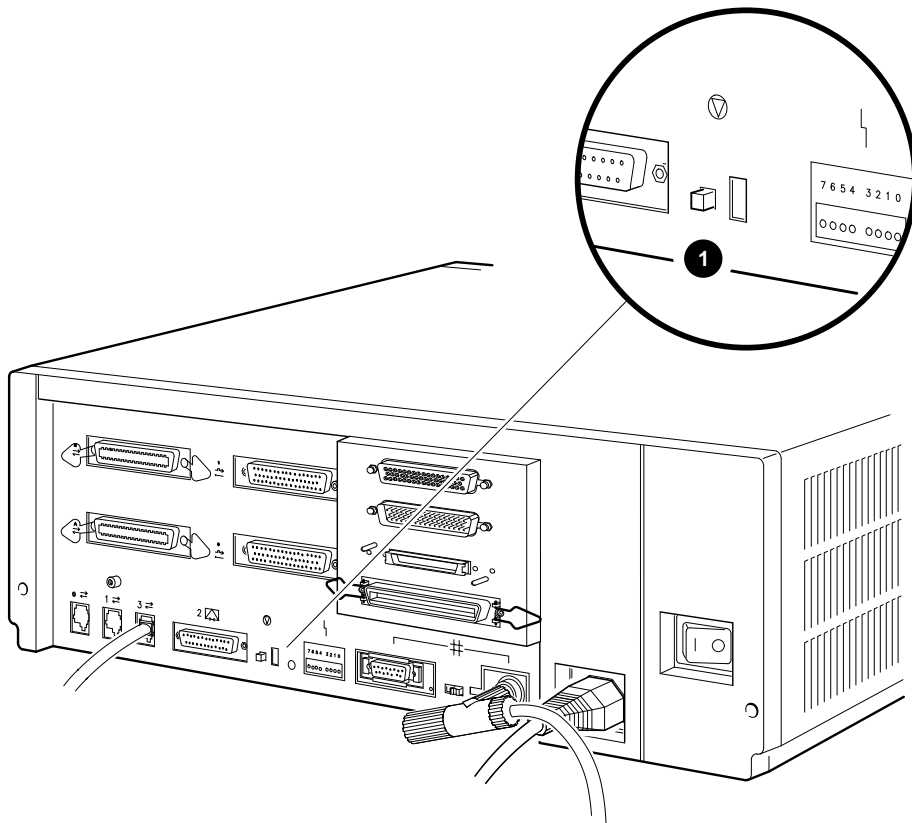
- Returning to console mode
- Using the console port
- Console security feature
- Setting the default boot device
- Setting the default recovery action

## 2.1 Returning to Console Mode

To use the procedures described in this chapter, the system must be in console mode. Before returning to console mode, you must shut down the operating system software if it is running. See the operating system documentation for information on the shutdown procedures. To return to console mode, follow these steps:

1. Shut down the operating system software if it is running.
2. Press the halt button on the back of the system unit see Figure 2-1. The system responds with the console prompt (>>>) on port 3 when it is in console mode.

Figure 2-1 Returning to Console Mode



MLO-009355

❶ Halt Button



## 2.2 Using the Console Port

The VAX 4000 Model 100 system provides a console port through MMJ port 3. Follow these steps:

1. Set the on/off switch on the system unit to the off (O) position.
2. Connect the console terminal to MMJ port 3.
3. Set the break enable switch see Figure 2-2 to the up position (break enable).

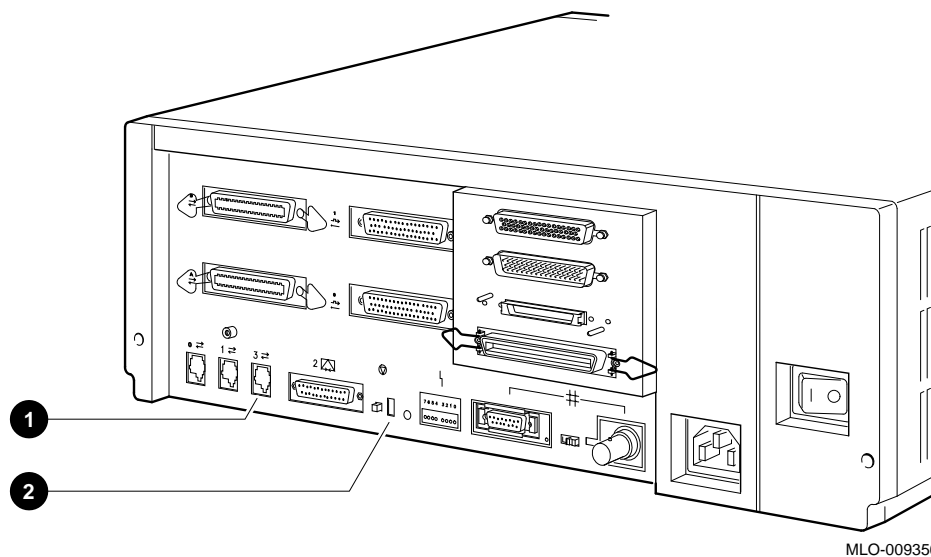
The break enable LED lights when you set the switch to the up position.

4. Set the on/off switch on the system unit to the on (|) position.

The system recognizes the position of the switch only when you set the power switch to the on (|) position.

5. The system will stop at the console prompt (>>>).

Figure 2-2 Console Port



MLO-009356

- 1 MMJ Port 3, Console Port
- 2 Break/Enable Switch

## 2.3 Console Security Feature

The console security feature allows you to disable most of the system console commands. When the security password is set, there are two types of users: privileged users and unprivileged users. Privileged users know the security password and can use the full range of console commands; unprivileged users can use only the following commands:

- LOGIN—Use this command with the security password to become a privileged user.
- BOOT—Use this command without parameters to boot the operating system when the boot device has been set.

Chapter 3 describes the console commands.

The following subsections describe how to do the following tasks:

- Set the security password
- Enable the console security feature
- Log in to privileged console mode
- Change the security password
- Disable the security password
- Exit from privileged console mode

### 2.3.1 Setting the Security Password

The console security feature is disabled when you receive the system. To set the security password on the system, follow these steps:

1. Enter the following command at the console prompt (>>>):

```
>>> SET PSWD
```

The system responds with the following prompt:

```
>>> PSWD1 :
```

---

**Note**

---

- The security password must be a string of **EXACTLY 16** hexadecimal characters (0 to 9 and A to F).
  - Write down the security password and store it in a safe place. If you forget the security password, you must call your Digital Services representative to disable the console security feature.
- 

2. Enter a security password and press Return.

The system does not display the security password as you type it. The system responds with the following prompt:

```
>>> PSWD2 :
```

3. Verify the security password by entering it a second time.

The system does not display the security password as you type it. If you enter the same security password at each prompt, the system saves the security password in nonvolatile memory. The system does not lose the security password when you turn off the system.

If the second security password does not match the first, the system responds with the following error message:

```
?80 ILLEGAL PASSWORD
```

```
>>>
```

4. Repeat steps 1 to 3 if you see an error message.

## 2.3.2 Enabling the Console Security Feature

When you have set the security password, you must enable the console security feature. To enable the console security feature, enter the following command at the console prompt:

```
>>> SET PSE 1
```

## 2.3.3 Logging in to Privileged Console Mode

When the console security feature is enabled, you must enter the security password to log in to privileged console mode. In privileged console mode you can use the full range of console commands. To log in to privileged console mode, follow these steps:

---

### Note

---

You must set the security password before following these steps (see Section 2.3.1).

---

1. Enter the following command:

```
>>> LOGIN
```

The system responds with the following prompt:

```
>>> Password:
```

2. Enter the security password and press Return.

The system does not display the security password as you type it. If you enter the correct security password, the system returns you to the console prompt and you become a privileged user. You can now use the full range of console commands.

If you enter an incorrect security password, the system responds with the following error message:

```
?81 INCORRECT PASSWORD
```

```
>>>
```

3. Repeat steps 1 and 2 if an error message is displayed.

## 2.3.4 Changing the Security Password

You must be a privileged user to change the security password. To change the security password, follow these steps:

1. Follow the procedure in Section 2.3.3 using the current security password to log in to the system.
2. Enter the following command:

```
>>> SET PSWD
```

The system responds with the following prompt:

```
PSWD1 :
```

3. Enter a new security password and press Return.

The system does not display the security password as you type it. The system then responds with the following prompt:

```
PSWD2 :
```

4. Verify the new security password by entering it a second time.

The system does not display the security password as you type it. If you enter the correct, current security password at the >>> prompt, and correctly enter the new security password a second time, the system saves the new security password in nonvolatile memory. The system does not lose the new security password when you turn off the system.

If you incorrectly enter the new security password a second time, the system responds with the following error message:

```
?80 ILLEGAL PASSWORD
```

```
>>>
```

5. Repeat steps 1 to 4 if an error message is displayed.

## 2.3.5 Disabling the Console Security Feature

When you disable the console security feature, all users can use the full range of console commands. To disable the console security feature, follow these steps:

1. Follow the procedure in Section 2.3.3 using the current security password to log in to the system.

2. Enter the following command:

```
>>> SET PSE 0
```

### 2.3.6 Exiting from Privileged Console Mode

When you exit from privileged console mode, privileged users must enter the LOGIN command with the correct password before they can use the full range of console commands. To exit from privileged console mode, enter one of the following commands:

- BOOT (with any supplied parameters)
- CONTINUE
- HALT
- START

Chapter 3 describes each of these commands.

## 2.4 Setting the Default Boot Device

When the system is shipped, it is set to boot from the system disk, DIA0. This RF-series disk holds the factory installed software (FIS).

If the toy battery has discharged, the default boot device may have to be set. Use **SHOW BOOT** to verify.

You can set the system to boot from a different default boot device that holds the operating system software. Table 2–1 shows the alternative default boot devices and their associated VMS™ device names.

**Table 2–1 Alternative Default Boot Devices**

Device	VMS Device Name
Hard disk (DSSI ID 0 to 6)	DIA <u>u</u> <sup>1</sup>
Network (the system boots from a remote system)	EZA0
Tape drive (SCSI ID 0 to 7) <sup>3</sup>	MKA <u>x</u> 00 <sup>2</sup>
Compact disc (SCSI ID 0 to 7) <sup>3</sup>	DKA <u>x</u> 00 <sup>2</sup>

<sup>1</sup>*u* represents the device unit number.

<sup>2</sup>*x* represents the SCSI ID of that device.

<sup>3</sup>This excludes ID number 6 which is the default controller ID.

To set an alternative default boot device, enter the SET BOOT command using the VMS device name of the alternative default boot device. For example, to set the system to boot over the network, enter the following command:

```
>>> SET BOOT EZA0
```

## 2.5 Setting the Default Recovery Action

There are five default recovery actions. You can change the default recovery action by entering the SET HALT command and the value or keyword associated with the action you want to set. Table 2–2 shows the five default recovery actions and their associated values. When the system is shipped, the default recovery action is set to halt.

**Table 2–2 Default Recovery Actions and Associated Values**

Recovery Action Keyword	Associated Value	Result
DEFAULT	0	The system halts and displays the console prompt.
RESTART	1	The system tries to restart the operating system. If it fails to restart the operating system, it halts.
REBOOT	2	The system tries to boot. If it fails to boot, it halts.
HALT	3	The system halts and displays the console prompt.
restart_reboot	4	The system tries to restart the operating system. If it fails to restart the operating system, it tries to boot. If it fails to boot, it halts.

To set an alternative default recovery action, enter the SET HALT command using the value or keyword associated with the recovery action you want to set. For example, to set the system to halt, enter one of the following commands:

```
>>> SET HALT 3
>>> SET HALT HALT
```



# 3

---

## Console Commands

This chapter describes the console commands that you can enter when the system is in console mode. The system displays the console prompt (>>>) when it is in console mode. If the system is running the operating system software, see Chapter 2 for information on returning the system to console mode.

If the console security feature is enabled and a security password is set, you must log in to privileged console mode before using most of these commands. See Chapter 2 for information on the console security feature.

The following sections describe all the console commands, give the command format, and describe the significance of each parameter.

### 3.1 BOOT

The BOOT command initializes the processor and transfers execution to VMB. VMB attempts to boot the operating system from the specified device or list of devices, or from the default boot device if none is specified. The console qualifies the bootstrap operation by passing a boot flags bitmap to VMB in R5.

*Format:*

**BOOT [qualifier-list] [{boot\_device},{boot\_device},...]**

If you do not enter either the qualifier or the device name, the default value is used. Explicitly stating the boot flags or the boot device overrides, but does not permanently change, the corresponding default value.

When specifying a list of boot devices (up to 32 characters, with devices separated by commas and no spaces), the system checks the devices in the order specified and boots from the first one that contains bootable software.

---

**Note**

---

If included in a string of boot devices, the Ethernet device, EZA0, should be placed only as the last device of the string. The system will continuously attempt to boot from EZA0.

---

Set the default boot device and boot flags with the SET BOOT and SET BFLAG commands. If you do not set a default boot device, the processor times out after 30 seconds and attempts to boot from the Ethernet device, EZA0.

*Qualifiers:*

*Command specific:*

/R5:{boot_flags}	A 32-bit hex value passed to VMB in R5. The console does not interpret this value. Use the SET BFLAG command to specify a default boot flags longword. Use the SHOW BFLAG command to display the longword.
{boot_flags}	Same as /R5:{boot_flags}
[device_name]	A character string of up to 32 characters. When specifying a list of boot devices, the device names should be separated by commas and no spaces. Apart from checking the length, the console does not interpret or validate the device name. The console converts the string to uppercase, then passes VMB a string descriptor to this device name in R0. Use the SET BOOT command to specify a default boot device or list of devices. Use the SHOW BOOT command to display the default boot device. The factory default device is the Ethernet device, EZA0. Table 2-1 lists the boot devices supported by the Model 100 system.

*Examples:*

```
>>>SHOW BOOT
DKA300
>>>SHOW BFLAG
00000000
>>>B !Boot using default boot flags and device.
(BOOT/R5:0 DKA300)

2..
-DKA300
```

## 3.2 CONTINUE

The CONTINUE command causes the processor to begin instruction execution at the address currently contained in the program counter (PC). This address is the address stored in the PC when the system entered console mode or an address that the user specifies using the DEPOSIT command. The CONTINUE command does not perform a processor initialization. The console enters program I/O mode.

*Format:*

**CONTINUE**

*Example:*

```
>>>CONTINUE
$      !VMS DCL prompt
```

## 3.3 DEPOSIT

The DEPOSIT command deposits data into the address specified. If you do not specify an address space or data size qualifier, the console uses the last address space and data size used in a DEPOSIT, EXAMINE, MOVE, or SEARCH command. After processor initialization, the default address space is physical memory and the default data size is longword. If you specify conflicting address space or data sizes, the console ignores the command and issues an error message.

*Format:*

**DEPOSIT [qualifier-list] {address} {data} [data...]**

*Qualifiers:*

*Data control:* /B, /W, /L, /Q, /N:{count}, /STEP:{size}, /WRONG

*Address space control:* /G, /I, /M, /P, /V, /U

*Arguments:*

- {address} A longword address that specifies the first location into which data is deposited. The address can be an actual address or a symbolic address.
- {data} The data to be deposited. If the specified data is larger than the deposit data size, the firmware ignores the command and issues an error response. If the specified data is smaller than the deposit data size, it is extended on the left with zeros.

[{data}] Additional data to be deposited (as many as can fit on the command line).

*Examples:*

```
>>>D/P/B/N:1FF 0 0      ! Clear first 512 bytes of
                        ! physical memory.

>>>D/V/L/N:3 1234 5     ! Deposit 5 into four longwords
                        ! starting at virtual memory address
                        ! 1234.

>>>D/N:8 R0 FFFFFFFF    ! Loads GPRs R0 through R8 with -1.

>>>D/L/P/N:10/ST:200 0 8 ! Deposit 8 in the first longword of
                        ! the first 17 pages in physical
                        ! memory.

>>>D/N:200 - 0          ! Starting at previous address, clear
                        ! 513 longwords or 2052 bytes.
```

### 3.4 EXAMINE

The EXAMINE command examines the contents of the memory location or register specified by the address. If no address is specified, + is assumed. The display line consists of a single character address specifier, the physical address to be examined, and the examined data.

EXAMINE uses the same qualifiers as DEPOSIT. However, the /WRONG qualifier causes EXAMINE to ignore ECC errors on reads from physical memory. The EXAMINE command also supports an /INSTRUCTION qualifier, which will disassemble the instructions at the current address.

*Format:*

**EXAMINE [qualifier-list] [address]**

*Qualifiers:*

*Data control:* /B, /W, /L, /Q, /N:{count}, /STEP:{size}, /WRONG

*Address space control:* /G, /I, /M, /P, /V, /U

*Command specific:*

/INSTRUCTION Disassembles and displays the VAX MACRO-32 instruction at the specified address.

*Arguments:*

[{address}] A longword address that specifies the first location to be examined. The address can be an actual or a symbolic address. If no address is specified, + is assumed.

*Examples:*

```
>>>EX PC                               ! Examine the PC.
    G 0000000F FFFFFFFC
>>>EX SP                               ! Examine the SP.
    G 0000000E 00000200
>>>EX PSL                              ! Examine the PSL.
    M 00000000 041F0000
>>>E/M                                 ! Examine PSL another way.
    M 00000000 041F0000
>>>E R4/N:5                             ! Examine R4 through R9.
    G 00000004 00000000
    G 00000005 00000000
    G 00000006 00000000
    G 00000007 00000000
    G 00000008 00000000
    G 00000009 801D9000

>>>EX PR$_SCBB                          !Examine the SCBB, IPR 17
    I 00000011 2004A000                ! (decimal).

>>>E/P 0                                 ! Examine local memory 0.
    P 00000000 00000000

>>>EX /INS 20040000                       ! Examine 1st byte of ROM.
    P 20040000 11 BRB                  20040019

>>>EX /INS/N:5 20040019                   ! Disassemble from branch.
    P 20040019 D0 MOVL I^#20140000,@#20140000
    P 20040024 D2 MCOML @#20140030,@#20140502
    P 2004002F D2 MCOML S^#0E,@#20140030
    P 20040036 7D MOVQ R0,@#201404B2
    P 2004003D D0 MOVL I^#201404B2,R1
    P 20040044 DB MFPR S^#2A,B^44(R1)

>>>E/INS                                  ! Look at next instruction.
    P 20040048 DB MFPR S^#2B,B^48(R1)

>>>
```

## 3.5 FIND

The FIND command searches main memory, starting at address zero for a page-aligned 128-KByte segment of good memory, or a restart parameter block (RPB). If the command finds the segment or RPB, its address plus 512 is left in SP (R14). If it does not find the segment or RPB, the console issues an error message and preserves the contents of SP. If you do not specify a qualifier, /RPB is assumed.

*Format:*

**FIND [qualifier-list]**

*Qualifiers:*

*Command specific:*

/MEMORY Searches memory for a page-aligned block of good memory, 128 KBytes in length. The search looks only at memory that is deemed usable by the bitmap. This command leaves the contents of memory unchanged.

/RPB Searches all physical memory for an RPB. The search does not use the bitmap to qualify which pages are looked at. The command leaves the contents of memory unchanged.

*Examples:*

```
>>>EX SP                ! Check the SP.
   G 0000000E 00000000
>>>FIND /MEM            ! Look for a valid 128 KBytes.
>>>EX SP                ! Note where it was found.
   G 0000000E 00000200
>>>FIND /RPB           ! Check for valid RPB.
?2C FND ERR 00C00004    ! None to be found here.
>>>
```

## 3.6 HALT

The HALT command has no effect. It is included for compatibility with other VAX consoles.

*Format:*

**HALT**

*Example:*

```
>>>HALT                ! Pretend to halt.
>>>
```

## 3.7 HELP

The HELP command provides information about command syntax and usage.

*Format:*

### HELP

*Example:*

```
>>>HELP
```

Following is a brief summary of all the commands supported by the console:

```
UPPERCASE denotes a keyword that you must type in
|         denotes an OR condition
[]        denotes optional parameters
<>       denotes a field specifying a syntactically correct value
..        denotes one of an inclusive range of integers
...       denotes that the previous item may be repeated
```

Valid qualifiers:

```
/B /W /L /Q /INSTRUCTION
/G /I /V /P /M
/STEP: /N: /NOT
/WRONG /U
```

Valid commands:

```
BOOT [[/R5:]<boot_flags>] [<boot_device>]
CONFIGURE
CONTINUE
DEPOSIT [<qualifiers>] <address> <datum> [<datum>...]
EXAMINE [<qualifiers>] [<address>]
FIND [/MEMORY | /RPB]
HALT
HELP
INITIALIZE
LOGIN
MOVE [<qualifiers>] <address> <address>
NEXT [<count>]
REPEAT <command>
SEARCH [<qualifiers>] <address> <pattern> [<mask>]
SET BFLG <boot_flags>
SET BOOT <boot_device>

SET HALT <0..4 |DEFAULT|RESTART|REBOOT|HALT|RESTART_REBOOT>
SET HOST/DUP/DSSI/BUS:<0..1> <node_number> [<task>]
SET HOST/DUP/UQSSP </DISK|/TAPE> <controller_number>[<task>]
SET HOST/DUP/UQSSP <physical_CSR_address> [<task>]
SET HOST/MAINTENANCE/UQSSP/SERVICE <controller_number>
SET HOST/MAINTENANCE/UQSSP <physical_CSR_address>
SET LANGUAGE <1..15>
```

```
SET PSE <0..1 | DISABLED | ENABLED>
SET PSWD <password>
SET RECALL <0..1 | DISABLED | ENABLED>
SET SCSI <0..7>
SET SCSI_ID

SHOW BFLG
SHOW BOOT
SHOW CONFIG

SHOW DEVICE
SHOW DSSI
SHOW ETHERNET
SHOW HALT
SHOW LANGUAGE
SHOW MEMORY [/FULL]

SHOW PSE
SHOW QBUS
SHOW RECALL
SHOW RLV12
SHOW SCSI
SHOW SCSI_ID
SHOW TRANSLATION <physical_address>
SHOW UQSSP
SHOW VERSION
START <address>
TEST [<test_code> [<parameters>]]
UNJAM
X <address> <count>

>>>
```



## 3.8 INITIALIZE

The INITIALIZE command performs a processor initialization.

*Format:*

### **INITIALIZE**

The following registers are initialized:

<b>Register</b>	<b>State at Initialization</b>
PSL	041F0000
IPL	1F
ASTLVL	4
SISR	0
ICCS	Bits <6> and <0> clear; the rest are unpredictable
RXCS	0
TXCS	80
MAPEN	0
Caches	Flushed
Instruction buffer	Unaffected
Console previous reference	Longword, physical, address 0
TODR	Unaffected
Main memory	Unaffected
General registers	Unaffected
Halt code	Unaffected
Bootstrap-in-progress flag	Unaffected
Internal restart-in-progress flag	Unaffected

The firmware clears all error status bits and initializes the following:

- CDAL bus timer
- Address decode and match registers
- Programmable timer interrupt vectors
- The QUART LPR register is set to 9600 baud
- All error status bits are cleared

*Example:*

```
>>>INIT  
>>>
```

## 3.9 LOGIN

Allows you to put the system in privileged console mode. When the console security feature is enabled and when you put the system in console mode, the system operates in unprivileged console mode. You can access only a subset of the console commands. To access the full range of console commands, you must use this command. This command may only be executed in secure console mode. The format of this command is as follows:

### **LO[GIN]**

When you enter the command, the system prompts you for a password as follows:

Password:

You must enter the current console security password. If you do not enter the correct password, the system displays the error message, **INCORRECT PASSWORD**. When you enter the console security password, the system operates in privileged console mode. In this mode, you can use all the console commands. The system exits from privileged console mode when you enter one of the following console commands:

- **BOOT**
- **CONTINUE**
- **HALT**
- **START**

## 3.10 MOVE

The MOVE command copies the block of memory starting at the source address to a block beginning at the destination address. Typically, this command has an /N qualifier so that more than one datum is transferred. The destination correctly reflects the contents of the source, regardless of the overlap between the source and the data.

The MOVE command actually performs byte, word, longword, and quadword reads and writes as needed in the process of moving the data. Moves are supported only for the physical and virtual address spaces.

*Format:*

**MOVE [qualifier-list] {src\_address} {dest\_address}**

*Qualifiers:*

*Data control:* /B, /W, /L, /Q, /N:{count}, /STEP:{size}, /WRONG

*Address space control:* /N, /U, /P

*Arguments:*

{src\_address}     A longword address that specifies the first location of the source data to be copied.

{dest\_address}    A longword address that specifies the destination of the first byte of data. These addresses may be an actual address or a symbolic address. If no address is specified, + is assumed.

*Examples:*

```
>>>EX/N:4 0           ! Observe destination.
P 00000000 00000000
P 00000004 00000000
P 00000008 00000000
P 0000000C 00000000
P 00000010 00000000

>>>EX/N:4 200           ! Observe source data.
P 00000200 58DD0520
P 00000204 585E04C1
P 00000208 00FF8FB8
P 0000020C 5208A8D0
P 00000210 540CA8DE

>>>MOV/N:4 200 0       ! Move the data.
```

```
>>>EX/N:4 0          ! Observe moved data.
P 00000000 58DD0520
P 00000004 585E04C1
P 00000008 00FF8FBB
P 0000000C 5208A8D0
P 00000010 540CA8DE
>>>
```

### 3.11 NEXT

The NEXT command executes the specified number of macro instructions. If no count is specified, 1 is assumed.

After the last macro instruction is executed, the console reenters console I/O mode.

*Format:*

#### **NEXT {count}**

The console implements the NEXT command, using the trace trap enable and trace pending bits in the PSL and the trace pending vector in the SCB.

The console enters the "Spacebar Step Mode". In this mode, subsequent spacebar strokes initiate single steps and a carriage return forces a return to the console prompt.

The following restrictions apply:

- If memory management is enabled, the NEXT command works only if the first page in SSC RAM is mapped in S0 (system) space.
- Overhead associated with the NEXT command affects execution time of an instruction.
- The NEXT command elevates the IPL to 31 for long periods of time (milliseconds) while single-stepping over several commands.
- Unpredictable results occur if the macro instruction being stepped over modifies either the SCBB or the trace trap entry. This means that you cannot use the NEXT command in conjunction with other debuggers.

*Arguments:*

{count}      A value representing the number of macro instructions to execute.

*Examples:*

```
>>>DEP 1000 50D650D4                   ! Create a simple program.
>>>DEP 1004 125005D1
>>>DEP 1008 00FE11F9
>>>EX /INSTRUCTION /N:5 1000         ! List it.
P 00001000   D4 CLRL    R0
P 00001002   D6 INCL    R0
P 00001004   D1 CMPL    S^#05,R0
P 00001007   12 BNEQ    00001002
P 00001009   11 BRB     00001009
P 0000100B   00 HALT
>>>DEP PR$ _SCBB 200                   ! Set up a user SCBB...
>>>DEP PC 1000                         ! ...and the PC.
>>>
>>>N                                   ! Single step...
P 00001002   D6 INCL    R0            ! SPACEBAR
P 00001004   D1 CMPL    S^#05,R0     ! SPACEBAR
P 00001007   12 BNEQ    00001002     ! SPACEBAR
P 00001002   D6 INCL    R0            ! CR
>>>N 5                                 ! ...or multiple step the program.
P 00001004   D1 CMPL    S^#05,R0
P 00001007   12 BNEQ    00001002
P 00001002   D6 INCL    R0
P 00001004   D1 CMPL    S^#05,R0
P 00001007   12 BNEQ    00001002
>>>N 7
P 00001002   D6 INCL    R0
P 00001004   D1 CMPL    S^#05,R0
P 00001007   12 BNEQ    00001002
P 00001002   D6 INCL    R0
P 00001004   D1 CMPL    S^#05,R0
P 00001007   12 BNEQ    00001002
P 00001009   11 BRB     00001009
>>>N
P 00001009   11 BRB     00001009
>>>
```

## 3.12 REPEAT

The REPEAT command repeatedly displays and executes the specified command. Press `CTRL/C` to stop the command. You can specify any valid console command except the REPEAT command.

*Format:*

**REPEAT {command}**

*Arguments:*

{command} A valid console command other than REPEAT.

*Examples:*

```
>>>REPEAT EX PR$ TODR !Watch the clock.  
I 0000001B 5AFE78CE  
I 0000001B 5AFE78D1  
I 0000001B 5AFE78FD  
I 0000001B 5AFE7900  
I 0000001B 5AFE7903  
I 0000001B 5AFE7907  
I 0000001B 5AFE790A  
I 0000001B 5AFE790D  
I 0000001B 5AFE7910  
I 0000001B 5AFE793C  
I 0000001B 5AFE793F  
I 0000001B 5AFE7942  
I 0000001B 5AFE7946  
I 0000001B 5AFE7949  
I 0000001B 5AFE794C  
I 0000001B 5AFE794F  
I 0000001B 5^C  
>>>
```

### 3.13 SEARCH

The SEARCH command finds all occurrences of a pattern and reports the addresses where the pattern was found. If the /NOT qualifier is present, the command reports all addresses in which the pattern did not match.

*Format:*

**SEARCH [qualifier-list] {address} {pattern} [{mask}]**

SEARCH accepts an optional mask that indicates bits to be ignored (*don't care* bits). For example, to ignore bit 0 in the comparison, specify a mask of 1. The mask, if not present, defaults to 0.

A match occurs if (pattern and not mask) = (data and not mask), where:

Pattern is the target data

Mask is the optional don't care bitmask (which defaults to 0)

Data is the data at the current address

SEARCH reports the address under the following conditions:

<b>/NOT Qualifier</b>	<b>Match Condition</b>	<b>Action</b>
Absent	True	Report address
Absent	False	No report
Present	True	No report
Present	False	Report address

The address is advanced by the size of the pattern (byte, word, longword, or quadword), unless overridden by the /STEP qualifier.

*Qualifiers:*

*Data control:* /B, /W, /L, /Q, /N:{count}, /STEP:{size}, /WRONG

*Address space control:* /P, /V, /U

*Command specific:*

/NOT            Inverts the sense of the match.

*Arguments:*

{start\_ address} A longword address that specifies the first location subject to the search. This address can be an actual address or a symbolic address. If no address is specified, + is assumed.

{pattern} The target data.

{mask} A mask of the bits desired in the comparison.

*Examples:*

```
>>>DEP /P/L/N:1000 0 0          ! Clear some memory.
>>>
>>>DEP 300 12345678             ! Deposit some search data.
>>>DEP 401 12345678
>>>DEP 502 87654321
>>>
>>>SEARCH /N:1000 /ST:1 0 12345678 ! Search for all occurrences
P 00000300 12345678             ! of 12345678 on any byte
P 00000401 12345678             ! boundary. Then try on
>>>SEARCH /N:1000 0 12345678      ! longword boundaries.
P 00000300 12345678             ! Search for all non-zero
>>>SEARCH /N:1000 /NOT 0 0        ! longwords.
P 00000300 12345678
P 00000400 34567800
P 00000404 00000012
P 00000500 43210000
P 00000504 00008765
>>>SEARCH /N:1000 /ST:1 0 1 FFFFFFFE ! Search for odd-numbered
! longwords on any boundary.
P 00000502 87654321
P 00000503 00876543
P 00000504 00008765
P 00000505 00000087
>>>SEARCH /N:1000 /B 0 12        ! Search for all occurrences
P 00000303 12                   ! of the byte 12.
P 00000404 12
>>>SEARCH /N:1000 /ST:1 /w 0 FE11 ! Search for all words that
>>>                               ! could be interpreted as
>>>                               ! a spin (10$: brb 10$).
>>>                               ! Note that none were found.
```



## 3.14 SET

The SET command sets the parameter to the value you specify.

*Format:*

**SET {parameter} {value}**

*Parameters:*

BFLAG	Sets the default R5 boot flags. The value must be a hex number of up to eight digits.
BOOT	Sets the default boot device. The value must be a valid device name or list of device names as specified in the BOOT command description in Section 3.1.
HALT	Sets the user-defined halt action. Acceptable values are the keywords "default", "restart", "reboot", "halt", "restart_reboot", or a number in the range 0 to 4 inclusive.
HOST	Makes a DUP connection to a DSSI device.
LANGUAGE	Sets console language and keyboard type. If the current console terminal does not support the multinational character set (MCS), then this command has no effect and the console message appears in English. Values are 1 through 15.
PSWD	Allows you to enable or disable the console security feature of the system. The SET PSE command accepts the following values: <ol style="list-style-type: none"><li>1. 0—Console security disabled</li><li>2. 1—Console security enabled</li></ol> <p>When the console security feature is enabled, only a subset of the console commands are available to the user. These commands are listed in Section 2.3. To enable the complete set of console commands once the console security feature is enabled, you must use the LOGIN command (see Section 3.9).</p>
PSWD	Allows you to set or change the console security password.
RECALL	Sets command recall state to either ENABLED (1) or DISABLED (0).
SCSI_ID	Sets the SCSI ID of the SCSI controller, to a number in the range 0 to 7. The SCSI ID of the SCSI controller is set to 6 by default.

*Qualifiers:* Listed in the parameter descriptions above.

*Examples:*

```
>>>
>>>SET BFLAG 220
>>>
>>>SET BOOT DUA0
>>>
>>>SET LANGUAGE 5
>>>
>>>SET HALT RESTART
>>>
```

### 3.15 SHOW

The SHOW command displays the console parameter you specify.

*Format:*

**SHOW {parameter}**

*Parameters:*

BFLAG	Displays the default R5 boot flags.
BOOT	Displays the default boot device.
CONFIG	Displays the system configuration. The command displays information about the devices that the firmware has tested. It also displays the device errors that the most recent device test detected.
DEVICE	Displays all devices in the system.
HALT	Shows the user-defined halt action.
DSSI	Shows the status of all nodes that are on the DSSI bus. For each node on the DSSI bus, the console displays the node number, the node name, and the boot name and type of the device, if available. The command does not indicate the "bootability" of the device.  The node that issues the command reports a node name of "*".  The device information is obtained from the media type field of the MSCP command GET UNIT STATUS. In the case where the node is not running or is not capable of running an MSCP server, then no device information is displayed.
ETHERNET	Displays hardware Ethernet address for all Ethernet adapters that can be found. Displays as blank if no Ethernet adapter is present.

LANGUAGE	Displays console language and keyboard type. Refer to the corresponding SET LANGUAGE command for the meaning.
MEMORY	Displays main memory configuration.
MEMORY/FULL	Additionally, displays the normally inaccessible areas of memory, such as the PFN bitmap pages, the console scratch memory pages, the Q22-bus scatter-gather map pages. Also reports the addresses of bad pages, as defined by the bitmap.
PSE	Displays the condition of the console security feature of the system.
QBUS	Displays all Q22-bus I/O addresses that respond to an aligned word read, and speculative device name information. For each address, the console displays the address in the VAX I/O space in hex, the address as it would appear in the Q22-bus I/O space in octal, and the word data that was read in hex.  This command may take several minutes to complete. Press <b>CTRL/C</b> to terminate the command. During execution, the command disables the scatter-gather map.
RECALL	Shows the current state of command recall, either ENABLED or DISABLED.
RLV12	Displays all RL01 and RL02 disks that appear on the Q22-bus.
UQSSP	Displays the status of all disks and tapes that can be found on the Q22-bus that support the UQSSP protocol. For each such disk or tape on the Q22-bus, the firmware displays the controller number, the controller CSR address, and the boot name and type of each device connected to the controller. The command does not indicate whether the device contains a bootable image.  This information is obtained from the media type field of the MSCP command GET UNIT STATUS. The console does not display device information if a node is not running (or cannot run) an MSCP server.
SCSI	Shows any SCSI devices in the system.
TRANSLATION	Shows any virtual addresses that map to the specified physical address. The firmware uses the current values of page table base and length registers to perform its search; it is assumed that page tables have been properly built.
VERSION	Displays the current firmware version.

*Qualifiers:* Listed in the parameter descriptions above.

*Examples:*

```
>>>
>>>SHOW BFLAG
00000220
>>>
>>>SHOW BOOT
DUA0
>>>SHOW CONTROLP
>>>
>>>SHOW ETHERNET
Ethernet Adapter
-EZA0 (08-00-2B-0B-29-14)
>>>
>>>SHOW HALT
restart
>>>
>>>SHOW LANGUAGE
English (United States/Canada)
>>>

>>>SHOW MEMORY

16 MB RAM, SIMM Set (0A,0B,0C,0D) present
Memory Set 0: 04000000 to 04FFFFFF, 16MB, 32768 good pages, 0 bad pages

64 MB RAM, SIMM Set (1E,1F,1G,1H) present
Memory Set 1: 00000000 to 03FFFFFF, 64MB, 131072 good pages, 0 bad pages

Total of 80MB, 163840 good pages, 0 bad pages, 136 reserved pages
>>>

>>>SHOW MEM/FULL

16 MB RAM, SIMM Set (0A,0B,0C,0D) present
Memory Set 0: 00000000 to 00FFFFFF, 16MB, 32768 good pages, 0 bad pages

Total of 16MB, 32768 good pages, 0 bad pages, 104 reserved pages

Memory Bitmap
-00FF3000 to 00FF3FFF, 8 pages

Console Scratch Area
-00FF4000 to 00FF7FFF, 32 pages

Scan of Bad Pages
>>>
```

```

>>>SHOW SCSI
SCSI Adapter 0 (761300), SCSI ID 7
-DKA100 (DEC TLZ04)
>>>
>>>SHOW TRANSLATION 1000
  V 80001000
>>>
>>>SHOW VERSION
KA52 Vn.n VMBn.n
>>>

>>>SHOW DSSI
DSSI Bus 0 Node 1 (R5WBAA)
-DIA1 (RF35)

DSSI Bus 0 Node 7 (*)

```

### 3.16 START

The START command starts instruction execution at the address you specify. If no address is given, the current PC is used. If memory mapping is enabled, macro instructions are executed from virtual memory, and the address is treated as a virtual address. The START command is equivalent to a DEPOSIT to PC, followed by a CONTINUE. It does not perform a processor initialization.

*Format:*

**START** [{address}]

*Arguments:*

[address]        The address at which to begin execution. This address is loaded into the user's PC.

*Example:*

```
>>>START 1000
```

### 3.17 TEST

The TEST command invokes a diagnostic test program specified by the test number. If you enter a test number of 0 (zero), the powerup diagnostics are executed. The console accepts an optional list of up to five additional hexadecimal arguments.

*Format:*

**TEST** [{test\_number} [{test\_arguments}]]

*Arguments:*

{test\_number}            A two-digit hex number specifying the test to be executed.  
{test\_arguments}        Up to five additional test arguments. These arguments are accepted, but they have no meaning to the console.

*Example:*

```
>>>TEST 0
72..71..70..69..68..67..66..65..64..63..62..61..60..59..58..57..
56..55..54..53..52..51..50..49..48..47..46..45..44..43..42..41..
40..39..38..37..36..35..34..33..32..31..30..29..28..27..26..25..
24..23..22..21..20..19..18..17..16..15..14..13..12..11..10..09..
08..07..06..05..04..03..
Tests completed.
>>>
```

*Example:*

```
>>>
>>>                    ! Display the CPU registers.
>>>
>>>t 9c

savpc=20048C68 savpsl=20048C68    sbr=03FA0000    slr=00003040
p0br=80000000 p0lr=00182000    plbr=00000000    pllr=00000000
sid=13001401    sie=03020801    mapen=00000000
tcr0=00000000    tir0=00000000    tnir0=00000000    tivr0=00000078
tcr1=00000001    tir1=02AF768E    tnir1=0000000F    tivr1=0000007C
bdr=3FFB08FF    sscr=00D05070    schb=20053400
DZ csr=0020        tcr=0008        msr=0F75
scr=0000D000    dser=00000000    qbear=0000000F    dear=00000000
qbmr=03FF8000    ipcr=0000
nicsr0=1FFF0003    3=00004030    4=00004050    5=8039FF00    6=83E0F000    7=00000000
nicsr9=04E204E2    10=00040000    11=00000000    12=00000000    13=00000000    15=0000FFFF
NISA=08-00-2B-29-1C-7A    intmsk=00    intreq=00    scdadr=00000000    scddir=0
SCSI_CSRS 0=00 1=00 2=00 3=00 4=00 6=05 5=05 7=00 8=16 9=5B A=5B B=00 C=04
VIC.....icsr=00000001    vmar=000007E0    ecr=000000CA
PC.....pcctl=FFFFFFC13    pcsts=FFFFFF800    pcadr=FFFFFFF8
BC_128K..cctl=00000007    bcetsts=000003E0    bcetidx=FFFFFFE0    bcetags=FFFFFFE0
.....bcedsts=00000F00    bcedidx=001FFFF8    bcedecc=00000000
.....nests=00000000    neoadr=E0055F70    neocmd=8000FF04    neicmd=000003FF
.....nedathi=FFFFFFF    nedatlo=FF7F9FFF    cefsts=00019200    cefadr=E00002C0
MEMORY...mesr=00006000    mear=08406010    Add=21018040    mmcdsr=01111000
.....memcon0=80000005    memcon1=00000007    moamr=00000000    sssr=C0CE
NCA.....cesr=00000000    cmcdsr=0000C108    cnear=00000000
.....csear1=00000000    csear2=00000000    cioear1=00000000    cioear2=000002C0
.....iccs=00000000    nicr=FFFFD8F0    icr=FFFFD8F0    todr=00000000
>>>
```

*Example:*

>>>

>>> ; list diagnostics and scripts

>>>t 9e

Test #	Address	Name	Parameters
	20053800	SCB	
	20054590	De_executive	
30	200637BC	Memory_Init_Bitmap	*** mark_Hard_SBEs *****
31	20064094	Memory_Setup_CSRs	*****
32	20064464	NMC_registers	*****
33	20064600	NMC_powerup	**
34	2005D0A4	SSC_ROM	***
35	20067394	B_Cache_diag_mode	bypass_test_mask *****
37	200681C4	Cache_w_Memory	bypass_test_mask *****
40	2006242C	Memory_count_pages	SIMM_set0 SIMM_set1 Soft_errs_allowed *****
41	200579C0	Board_Reset	*
42	2005B56C	Chk_for_Interrupts	*****
46	200670D4	P_Cache_diag_mode	bypass_test_mask *****
47	20063D7C	Memory_Refresh	start_a end incr cont_on_err time_seconds *****
48	20061558	Memory_Addr_shorts	start_add end_add * cont_on_err pat2 pat3 *****
4A	200634E0	Memory_ECC_SBEs	start_add end_add add_incr cont_on_err *****
4B	20061D78	Memory_Byte_Errors	start_add end_add add_incr cont_on_err *****
4C	20062E90	Memory_ECC_Logic	start_add end_add add_incr cont_on_err *****
4D	200613BC	Memory_Address	start_add end_add add_incr cont_on_err *****
4E	20061AF8	Memory_Byte	start_add end_add add_incr cont_on_err *****
4F	20062628	Memory_Data	start_add end_add add_incr cont_on_err *****
51	2005BA5C	FPA	*****
52	2005BED8	SSC_Prog_timers	which_timer wait_time_us ***
53	2005C1A8	SSC_TOY_Clock	repeat_test_250ms_ea Tolerance ***
54	2005B670	Virtual_Mode	*****
55	2005C360	Interval_Timer	*****
58	200602F0	SHAC_RESET	port_number time_secs not_pres
59	2005F584	SGEC_LPBACK_ASSIST	time_secs **
5C	2005FAEC	SHAC	bypass_test_mask *****
5F	2005E870	SGEC	loopback_type no_ram_tests *****
63	2005CF48	QDSS_anyc	input_csr selftest_r0 selftest_r1 *****
80	200649FC	CQBIC_memory	bypass_test_mask *****
81	2005CBA8	Qbus_MSCP	IP_csr *****
82	2005CD70	Qbus_DELQA	device_num_addr ****
83	20058C70	QZA_Intlpbck1	controller_number *****
84	2005A328	QZA_Intlpbck2	controller_number *****
85	20057EE4	QZA_memory	incr test_pattern controller_number *****
86	200583A0	QZA_DMA	Controller_number main_mem_buf *****
90	2005BE54	CQBIC_registers	*
91	2005BDE8	CQBIC_powerup	**
99	200647D0	Flush_Ena_Caches	dis_flush_VIC dis_flush_BC dis_flush_PC

```

9A 2005D1DC INTERACTION      pass_count disable_device ****
9B 20064680 Init_memory        ***
9C 2005D1A8 List_CPU_registers *
9D 2005DEC4 Utility           Modify_CPU_type *****
9E 2005C518 List_diagnostics    script_number *
9F 20060888 Create_A0_Script      *****
C1 20057B90 SSC_RAM_Data        *
C2 20057D68 SSC_RAM_Data_Addr  *
C5 2005E770 SSC_registers      *
C6 20057AD4 SSC_powerup        *****
D0 20066C98 V_Cache_diag_mode      bypass_test_mask *****
D2 20065220 O_Bit_diag_mode       bypass_test_mask *****
DA 20067FE8 PB_Flush_Cache      *****
DB 20065A18 Speed            print_speed *****
DC 200642BC NO_Memory_present  *
DD 200661FC B_Cache_Data_debug    start_add end_add add_incr *****
DE 20065DB4 B_Cache_Tag_Debug     start_add end_add add_incr *****
DF 20065614 O_BIT_DEBUG          start_add end_add add_incr seg_incr *****
E0 20068498 SCSI              environment reset_bus time_s *****
E1 20068578 SCSI_Utility      environment util_nbr target_ID lun *****
E2 20068630 SCSI_MAP        bypass_test addr_incr_data_tst *****
E4 200689D4 DZ              environment *****
E8 20068B4C SYNC            environment *****
E9 20068BF4 SYNC_Utility    environment *****
EC 20068CAC ASYNC          environment *****

```

#### Scripts

```

# Description

A0 User defined scripts
A1 Powerup tests, Functional Verify, continue on error, numeric countdown
A3 Functional Verify, stop on error, test # announcements
A4 Loop on A3 Functional Verify
A6 Memory tests, mark only multiple bit errors
A7 Memory tests
A8 Memory acceptance tests, mark single and multi-bit errors, call A7
A9 Memory tests, stop on error
B2 Extended tests plus BF
B5 Extended tests, then loop
BF DZ, SYNC, ASYNC with loopbacks
>>>
>>>

```



### 3.18 UNJAM

The UNJAM command performs an I/O bus reset, by writing a 1 (one) to IPR 55 (decimal). SHAC and SGEC are explicitly reset, EDAL\_INTREQ register error bits are cleared and SCST\_DMA map registers are cleared.

*Format:*

**UNJAM**

*Example:*

```
>>>UNJAM  
>>>
```

### 3.19 X—Binary Load and Unload

The X command is for use by automatic systems communicating with the console.

The X command loads or unloads (that is, writes to memory, or reads from memory) the specified number of data bytes through the console serial line (regardless of console type) starting at the specified address.

*Format:*

**X {address} {count} CR {line\_checksum} {data} {data\_checksum}**

If bit 31 of the count is clear, data is received by the console and deposited into memory. If bit 31 is set, data is read from memory and sent by the console. The remaining bits in the count are a positive number indicating the number of bytes to load or unload.

The console accepts the command upon receiving the carriage return. The next byte the console receives is the command checksum, which is not echoed. The command checksum is verified by adding all command characters, including the checksum and separating space (but not including the terminating carriage return, rubouts, or characters deleted by rubout), into an 8-bit register initially set to zero. If no errors occur, the result is zero. If the command checksum is correct, the console responds with the input prompt and either sends data to the requester or prepares to receive data. If the command checksum is in error, the console responds with an error message. The intent is to prevent inadvertent operator entry into a mode where the console is accepting characters from the keyboard as data, with no escape mechanism possible.

If the command is a load (bit 31 of the count is clear), the console responds with the input prompt (>>>), then accepts the specified number of bytes of data for depositing to memory, and an additional byte of received data checksum. The data is verified by adding all data characters and the checksum character into an 8-bit register initially set to zero. If the final content of the register is nonzero, the data or checksum are in error, and the console responds with an error message.

If the command is a binary unload (bit 31 of the count is set), the console responds with the input prompt (>>>), followed by the specified number of bytes of binary data. As each byte is sent, it is added to a checksum register initially set to zero. At the end of the transmission, the two's complement of the low byte of the register is sent.

If the data checksum is incorrect on a load, or if memory or line errors occur during the transmission of data, the entire transmission is completed, then the console issues an error message. If an error occurs during loading, the contents of the memory being loaded are unpredictable.

The console represses echo while it is receiving the data string and checksums.

The console terminates all flow control when it receives the carriage return at the end of the command line in order to avoid treating flow control characters from the terminal as valid command line checksums.

You can control the console serial line during a binary unload using control characters (CTRL/C, CTRL/S, CTRL/O, and so on). You cannot control the console serial line during a binary load, since all received characters are valid binary data.

The console has the following timing requirements:

- It must receive data being loaded with a binary load command at a rate of at least one byte every 60 seconds.
- It must receive the command checksum that precedes the data within 60 seconds of the carriage return that terminates the command line.
- It must receive the data checksum within 60 seconds of the last data byte.

If any of these timing requirements are not met, then the console aborts the transmission by issuing an error message and returning to the console prompt.

The entire command, including the checksum, can be sent to the console as a single burst of characters at the specified character rate of the console serial line. The console is able to receive at least 4 KBytes of data in a single X command.

### 3.20 ! (Comment)

The comment character (an exclamation point) is used to document command sequences. It can appear anywhere on the command line. All characters following the comment character are ignored.

*Format:* !

*Example:*

```
>>>! The console ignores this line.  
>>>
```

# 4

---

## Hardware Specifications

This chapter lists the hardware specifications of the following devices:

- System unit
- Internal SCSI device
- KA52-AA CPU
- Internal DSSI device

### 4.1 System Unit Specifications

The following tables list the specifications for the VAX 4000 Model 100 system.

**Table 4–1 System Specifications: VAX 400 Model 100**

<b>Subject</b>	<b>Description</b>
Processor	KA52 (NVAX).
Boot and diagnostic firmware ROM	512KB.
DRAM memory	Expandable from 16MB using one set of four 4MB SIMMs to a maximum of 128MB using two sets of four 16MB SIMMs. These are MS44 memory options.
Hard disk	RF35 and RF31T (the system supports a maximum of three disk devices in the enclosure).
Tape drive	TZ30, TZK10, TLZ06.
Compact disc drive	RRD42.
Terminals	Supports the VT series.
Interfaces	One DSSI bus, one synchronous SCSI bus, one ThinWire Ethernet port <sup>1</sup> , one standard Ethernet port <sup>1</sup> (thickwire), three MMJ ports, one modem port. Optional: 8 or 16 additional asynchronous DEC423 MMJ ports or 8 additional asynchronous modem ports, 2 synchronous ports.
Input voltage	Automatically adjusting ac input. Range: 100 V ac to 120 V ac or 220 V ac to 240 V ac.
Maximum inrush current	2.0 amperes (A) at 220 V ac.
Maximum running current	2.2 A at 110 V ac, 1.1 A at 220 V ac.
Steady state current	2.2 A at 110 V ac, 1.1 A at 220 V ac.
Maximum power consumption	200 watts (W).
Frequency	49 Hz to 61 Hz.

<sup>1</sup>Both Ethernet types cannot be used simultaneously.

**Table 4–2 System Unit Metrics**

System Unit	Weight <sup>1</sup> kg (lb)	Height cm (in)	Width cm (in)	Depth cm (in)
Model 100	18.18 (40)	14.99 (5.90)	46.38 (18.26)	40.00 (15.75)

<sup>1</sup>Depends on configuration

**Table 4–3 System Storage Conditions**

Storage Condition	Range or Value
Temperature range	5°C to 50°C (41°F to 122°F )
Relative humidity	10% to 95% at 66°C (noncondensing)
Altitude	0 m to 2400 m (0 ft to 8000 ft)
Maximum wet bulb temperature	32°C (90°F)
Minimum dew point	2°C (36°F)

**Table 4–4 System Operating Conditions and Nonoperating Conditions**

Operating Conditions	Range or Value
Temperature range	10°C (50°F) to 32°C (90°F) with TZ30 tape drive; otherwise 10°C (50°F) to 40°C (104°F)
Temperature change rate	11°C (20°F) per hour maximum
Relative humidity	10% to 90% noncondensing (20% to 80% if tape device installed)
Maximum wet bulb temperature	28°C (82°F)
Minimum dew point	2°C (36°F)
Altitude	2400 m (8000 ft) at 36°C (96°F)

(continued on next page)

**Table 4–4 (Cont.) System Operating Conditions and Nonoperating Conditions**

---

**Nonoperating Conditions**

---

Temperature range	–40°C (–40°F) to 66°C (151°F)
Relative humidity	10% to 95% at 66°C (151°F)
Altitude	4900 m (16 000 ft)
Maximum wet bulb temperature	28°C (82°F)
Minimum dew point	2°C (36°F)

---

## 4.2 Internal DSSI Device Specifications

The following tables list the specifications for the internal DSSI devices.

**Table 4–5 RF31T and RF35 Hard Disk Drive Specifications**

<b>Formatted Storage Capacity</b>	<b>RF31T</b>	<b>RF35</b>
Per drive (MB )	381	852
Blocks per track	57	57
Blocks per drive	744,534	1,664,628
Buffer size (KB)	512	512

<b>Performance</b>	<b>RF31T</b>	<b>RF35</b>
Transfer rate to or from media (MB/second)	3.3	3.3
Data transfer rate (MB/second)	4.0	4.0
Transfer rate to or from buffer, asynchronous (MB/second)	N/A (SCSI only)	N/A (SCSI only)
Transfer rate to or from buffer, synchronous (M bytes/second)	N/A (SCSI only)	N/A (SCSI only)
Average seek time (milliseconds)	5.5	9.5
Maximum seek time, full stroke (milliseconds)	–	19
Average latency (milliseconds)	5.6	5.6
Average access (milliseconds)	11.1	15.1



**Table 4–6 TZ30 Tape Drive Specifications**

<b>Subject</b>	<b>Description</b>
Mode of operation	Streaming
Media	12.77 mm (0.5 in) unformatted magnetic tape
Bit density	2624 bits/cm (6667 bits/in)
Number of tracks	22
Transfer rate (at host)	62.5Kb/s
Tape speed	190 cm/s (75 in/s)
Track format	Multiple track serpentine recording
Cartridge capacity	95MB, formatted (approx)

**Table 4–7 TZK10 QIC Tape Drive Specifications**

<b>Subject</b>	<b>Description</b>
Mode of operation	Streaming.
Media	DC6320, DC6525, or Digital approved equivalent. See the <i>VAX 4000 Model 100 Operator Information</i> manual.
Track width: write	0.1778 mm +0.0000, -0.0127 mm (0.0070 in +0.0000, -0.0005 in).
Track width: read	0.1270 mm +0.0127, -0.0000 mm (0.0050 in +0.0005, -0.0000 in).
Bit density	16Kb/in.
Number of tracks	26.
Transfer rate	200K bytes/s at average streaming mode, 1.5MB/s at SCSI maximum.
Tape speed	305 cm/s (120 in/s).
Track format	Multiple track serpentine recording.
Cartridge capacity	320MB or 525MB, formatted (approx), depending on the QIC tape used.

**Table 4–8 TLZ06 Cassette Tape Drive Specifications**

<b>Subject</b>	<b>Description</b>
Mode of operation	Streaming and start/stop.
Media	TLZ04-CA, TLZ06-CA, or Digital approved equivalent. See the <i>VAX 4000 Model 100 Operator Information</i> manual.
Bit density	114Mb/in.
Transfer rate (sustained)	183KB/s noncompression
Recording format	Digital Data Storage (DDS, DC)

**Table 4–9 RRD42 Compact Disc Drive Specifications**

<b>Subject</b>	<b>Description</b>
Acceptable discs	CD-ROM mode-1 data discs CD-ROM mode-2 data discs
Disc capacity	600MB
Rotation speed: innermost track	530 r/min at CLV = 1.4 m/s
Rotation speed: outermost track	200 r/min at CLV = 1.2 m/s
Sustained data transfer rate	150KB/s
Burst data transfer rate	1.5MB/s
Access time: full stroke	650 ms
Access time: average	380 ms

### 4.3 KA52-AA CPU Specifications

The Model 100 system uses the timesharing KA52-AA CPU (54-21797-01).

---

<b>Central Processing Unit</b>	
Addressing modes	General register: 8 Program counter: 4 Index: 9
Clock rate	286 MHz (14 ns cycle)
Data path width	64 bits
Number of data types	Hardware: 9 Software emulated: 7
Number of instructions	Hardware: 242 Microcode assisted: 21 Software emulated: 41
General purpose registers	16 (32-bit wide) Interval timer: 1 (programmable) Programmable timers: 2
I/O bus interface	One Q22-bus interface with 8192 entry scatter/gather map
Q-bus backplane termination	123 $\Omega$

---

<b>Memory Management and Control</b>	
Page size	512 bytes
Virtual address space	4GB
Physical memory space	128MB
Number of memory sets	2 sets

---

<b>Architecture</b>	
Instruction prefetch buffer size	16 bytes
Primary Cache	
Data stored	Instruction and data
Write algorithm	Write-through
Size	8KB

---

<b>Architecture</b>	
Speed	14 nanoseconds (READ)
Associativity	Two-way
<b>Back-up Cache</b>	
Data stored	Instruction stream and data
Write algorithm	Write-back
Size	128 Kbytes
Speed	42 nanoseconds
Associativity	Direct mapped
<b>Translation buffer</b>	
Size	96 entry
Associativity	Fully associative
<b>Q22-bus address translation</b>	
<b>Q22-bus map cache</b>	
Size	16 entry
Associativity	Fully associative
<b>Q22-bus I/O bus buffer size</b>	
Input	32 bytes
Output	4 bytes
<b>Q22-bus Maximum I/O bandwidth</b>	
Block mode DMA read	2.4MB/second
Block mode DMA write	3.3MB/second
<b>Ethernet Port</b>	
Supported protocols	Ethernet V2.0 (IEEE 802.3)
Supported media types	Standard or ThinWire
Data path width	1 bit
Data rate	10Mb/second
<b>Buffer size</b>	
Transmit buffer	128 bytes
Receiver buffer	128 bytes

---

**Digital Storage System Interconnect (DSSI)**

---

Number of DSSI interfaces	1
Maximum number of supported devices	7
Data path width	8 bits
Maximum bandwidth	4MB/second
Maximum queue I/O rate	1200/second
Buffer size	
Transmit buffer	256 bytes
Receiver buffer	256 bytes

---

**Console Serial Lines**

---

Interface standards	EIA RS-423-A/CCITT V.10 X.26 EIA RS-232-C/CCITT V.28 DEC-423
Data format	1 start bit, 8 data bits, 0 parity bits, 1 stop bit
Baud rates	300; 600; 1200; 2400; 4800; 9600; 19,200; 38,400

---

**Ordering Information**

---

Included as part of base system

<b>Operating System Support</b>	
VMS	Version 5.5-1HN
<b>Diagnostic Support</b>	
MicroVAX Diagnostic Monitor	Release 137A and later
Self-tests	Yes
<b>Related Documentation</b>	
EK-473AA-MG	<i>KA52 CPU System Maintenance</i>

---

# Index

## A

---

- Alternative boot devices, 2–10
- Asynchronous communications devices
  - DHW42-AA, 1–6
  - DHW42-BA, 1–6
  - DHW42-CA, 1–6
  - DHW42-UP, 1–6

## B

---

- Binary load and unload (X command), 3–25
- Boot
  - device, 2–10
    - recovery action, 2–11
- BOOT command, 2–5, 3–1
  - use of, 2–9
- Boot device
  - alternatives, 2–10
  - default, 2–10
  - device names, 2–10
  - resetting default, 2–10
- Break enable switch
  - use of, 2–4
- Break enable switch LED
  - meaning, 2–4
- Break key
  - halting the system, 2–4

## C

---

- Cables
  - synchronous option, 1–6
- Commands
  - for unprivileged users, 2–5
- Comment command (!), 3–27
- ! (comment command), 3–27
- Communications devices
  - asynchronous, 1–6
  - synchronous, 1–6
- Compact disc drives
  - RRD42, 1–5
- Console command
  - LOGIN, 3–10
- Console commands
  - binary load and unload (X), 3–25
  - BOOT, 3–1
  - ! (comment), 3–27
  - CONTINUE, 3–3
  - DEPOSIT, 3–3
  - EXAMINE, 3–4
  - FIND, 3–6
  - HALT, 3–6
  - HELP, 3–7
  - INITIALIZE, 3–9
  - MOVE, 3–11
  - NEXT, 3–12
  - REPEAT, 3–14
  - SEARCH, 3–15
  - SET, 3–17
  - SHOW, 3–18
  - START, 3–21
  - TEST, 3–21

Console commands (cont'd)  
UNJAM, 3-25  
X (binary load and unload), 3-25  
Console mode  
  privileged, 2-7  
  returning to, 2-2  
console port  
  MMJ port 3, 2-4  
  use of, 2-4  
Console port, 2-4  
Console prompt, 2-2  
Console security feature  
  description, 2-5  
  disabling, 2-8  
  enabling, 2-7  
  values, 3-17  
CONTINUE command, 3-3  
  use of, 2-9  
CPU  
  KA52, 1-1

## D

---

Defaults  
  boot device, 2-10  
  recovery action, 2-11  
DEPOSIT command, 3-3  
Device names  
  boot device, 2-10  
Dimensions  
  system unit, 4-3  
Disk drives  
  RF31T, 1-5  
  RF35, 1-5

## E

---

EXAMINE command, 3-4

## F

---

Factory installed software, 1-5, 2-10  
Features  
  Model 100, 1-1

FIND command, 3-6  
FIS, 1-5, 2-10

## H

---

Halt  
  recovery action, 2-11  
Halt button  
  use of, 2-2, 2-4  
HALT command, 3-6  
  use of, 2-9  
Halting  
  system, 2-4  
Hardware specifications  
  RF31T hard disk drive, 4-5  
  RF35 hard disk drive, 4-5  
  RRD42 compact disc drive, 4-7  
  system unit, 4-1  
  TLZK06 cassette tape drive, 4-6  
  TZ30 tape drive, 4-5  
  TZK10 QIC tape drive, 4-6  
HELP command, 3-7

## I

---

INITIALIZE command, 3-9

## K

---

KA52-AA modules  
  system names, 4-8

## L

---

LOGIN command, 2-5, 3-10  
  use of, 2-7, 2-9

## M

---

Mass storage devices  
  internal, 1-5  
MMJ port 0  
  as console port, 2-4  
MMJ port 3  
  alternative console port, 2-4



Model 100  
  features, 1-1  
  KA52 CPU, 1-1  
  system description, 1-1  
  VAX architecture support, 1-3  
MOVE command, 3-11

## N

---

NEXT command, 3-12  
Nonoperating conditions  
  system specifications, 4-4  
NVAX processor, 1-1

## O

---

Operating conditions  
  system specifications, 4-3  
Operating system  
  shutting down, 2-2

## P

---

Password  
  security password, 2-6  
Privileged console mode  
  exiting, 2-9  
  logging in to, 2-7  
  use of, 2-7  
Privileged user  
  definition, 2-5

## R

---

Recovery action  
  actions, 2-11  
  default, 2-11  
  resetting default, 2-11  
REPEAT command, 3-14  
Restart  
  recovery action, 2-11  
RF31T  
  hardware specifications, 4-5  
RF35  
  hardware specifications, 4-5

RRD42 compact disc drive  
  hardware specifications, 4-7

## S

---

SEARCH command, 3-15  
Security password  
  changing, 2-8  
  entering, 2-6, 2-7  
  setting, 2-6  
  verifying, 2-6  
SET BOOT command  
  use of, 2-10  
SET command, 3-17  
SET HALT command  
  boot, 2-11  
  halt, 2-11  
  restart, 2-11  
  use of, 2-11  
SET HOST/DUP command, 3-17  
SET PASSWORD command  
  use of, 2-6  
SET PSE command  
  use of, 2-7, 2-8  
SET PSWD command  
  use of, 2-8  
SHOW command, 3-18  
START command, 3-21  
  use of, 2-9  
Storage conditions  
  system unit, 4-3  
Storage devices  
  compact disc drives, 1-5  
  disk drives, 1-5  
  diskette drives, 1-5  
  tape drives, 1-5  
Support  
  VAX architecture, 1-3  
Synchronous communications devices  
  DSW42-AA, 1-6  
Synchronous option cables, 1-6  
System description  
  Model 100, 1-1

System disk, 1-5, 2-10  
System specifications  
  nonoperating conditions, 4-4  
  operating conditions, 4-3  
System unit  
  dimensions, 4-3  
  hardware specifications, 4-1  
  storage conditions, 4-3

## T

---

Tape drives  
  TLZ06, 1-5  
  TZ30, 1-5  
  TZK10, 1-5  
TEST command, 3-21  
TLZ06 cassette tape drive  
  hardware specifications, 4-6  
TZ30 tape drive  
  hardware specifications, 4-5  
TZK10 QIC tape drive

hardware specifications, 4-6

## U

---

UNJAM command, 3-25  
Unprivileged user  
  definition, 2-5

## V

---

VAX  
  architecture support, 1-3  
VAX architecture support  
  Model 100, 1-3  
VAX instructions  
  support of, 1-3

## X

---

X command (binary load and unload), 3-25

## Reader's Comments VAX 4000 Model 100 Customer Technical Information

EK-467AA-TI. A01

---

Your comments and suggestions help us improve the quality of our publications.

**Please rate the manual in the following categories:**

	Excellent	Good	Fair	Poor
Accuracy (product works as described)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Completeness (enough information)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Clarity (easy to understand)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Organization (structure of subject matter)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Figures (useful)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Examples (useful)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Table of contents (ability to find topic)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Index (ability to find topic)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Page design (overall appearance)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Print quality	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

What I like best about this manual: \_\_\_\_\_

\_\_\_\_\_

What I like least about this manual: \_\_\_\_\_

\_\_\_\_\_

Additional comments or suggestions: \_\_\_\_\_

\_\_\_\_\_

I found the following errors in this manual:

Page	Description
------	-------------

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

For which tasks did you use this manual?

Installation

Maintenance

Marketing

Operation/Use

Programming

System Management

Training

Other (please specify) \_\_\_\_\_

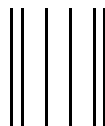
Name/Title \_\_\_\_\_

Company \_\_\_\_\_

Address \_\_\_\_\_

Do Not Tear - Fold Here and Tape

**digital**



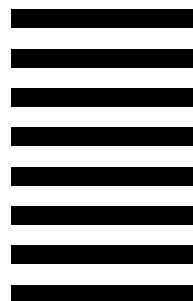
NO POSTAGE  
NECESSARY  
IF MAILED  
IN THE  
UNITED STATES

**BUSINESS REPLY MAIL**

FIRST CLASS PERMIT NO. 33 MAYNARD MASS.

POSTAGE WILL BE PAID BY ADDRESSEE

**DIGITAL EQUIPMENT CORPORATION  
INFORMATION DESIGN AND CONSULTING  
PKO3-1/D30  
129 PARKER STREET  
MAYNARD, MA 01754-9975**



Do Not Tear - Fold Here and Tape