ULTRIX-32
Guide to
System Shutdown and Startup

Order No. AA-ME91A-TE

ULTRIX-32 Operating System, Version 3.0

Digital Equipment Corporation

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 or its affiliated companies.

The following are trademarks of Digital Equipment Corporation:

DEC
DECnet
DECUS
MASSBUS
MicroVAX
PDP
Q-bus
RT
ULTRIX
ULTRIX-11
ULTRIX-32
UNIBUS
VAX
VAXstation
VMS
VT
ULTRIX Worksystem Software
digital

UNIX is a registered trademark of AT&T in the USA and other countries.

IBM is a registered trademark of International Business Machines Corporation.

MICOM is a registered trademark of Micom System, Inc.

This manual was written and produced by the ULTRIX Documentation Group in Nashua, New Hampshire.

Contents

About This Manual

1 System Shutdown Procedures

2 System Startup Modes

3 Processor-Specific Boot Commands

Tables

About This Manual

This guide provides information on shutting down and starting up your system and identifies the processor-specific boot commands for the processors supported by the ULTRIX operating system. It also provides a description of the Standalone ULTRIX Environment.

Audience

The *ULTRIX–32 Guide to System Shutdown and Startup* is written for the person responsible for managing and maintaining an ULTRIX system. It assumes that this individual is familiar with ULTRIX commands, the system configuration, the system's controller/drive unit number assignments and naming conventions, and an editor such as vi or ed. You do not need to be a programmer to use this guide.

Organization

This manual consists of four chapters, two appendixes, and an index. The chapters and the appendixes are:

Chapter 1:	System Shutdown Procedures Explains the various ways that you can shut down the system.
Chapter 2:	System Startup Modes Explains the three modes that you can use to start up the system: single-user, multiuser, and conversational.
Chapter 3:	Processor Specific Boot Commands Identifies and describes all of the processor-specific boot commands supported by the ULTRIX system.

Chapter 4: The Standalone ULTRIX Environment
Describes the purpose and functionality of the Standalone ULTRIX Environment and explains how to invoke it.

Appendix A: Device Mnemonics
Lists the supported device mnemonics and explains how to obtain detailed reference page information on devices.

Appendix B: General Purpose Register Use by VMB.EXE
Defines the elements of the R0 through R5 registers used by the boot software.

Related Documents

You should have the hardware documentation for your system and peripherals.

Conventions

The following conventions are used in this manual:

special — In text, each mention of a specific command, option, partition, pathname, directory, or file is presented in this type.

command(x) — In text, cross-references to the command documentation include the section number in the reference manual where the commands are documented. For example: See the cat(1) command. This indicates that you can find the material on the cat command in Section 1 of the reference pages.

literal — In syntax descriptions, this type indicates terms that are constant and must be typed just as they are presented.

italics — In syntax descriptions, this type indicates terms that are variable.

[ ] — In syntax descriptions, square brackets indicate terms that are optional.

. . . — In syntax descriptions, a horizontal ellipsis indicates that the preceding item can be repeated one or more times.

function In function definitions, the function itself is shown in this type. The function arguments are shown in italics.

UPPERCASE The ULTRIX system differentiates between lowercase and uppercase characters. Enter uppercase characters only where specifically indicated by an example or a syntax line.

example In examples, computer output text is printed in this type.

example In examples, user input is printed in this bold type.

% This is the default user prompt in multiuser mode.

This is the default superuser prompt.

>>> This is the console subsystem prompt.

.
.
. In examples, a vertical ellipsis indicates that not all of the lines of the example are shown.

<KEYNAME> In examples, a word or abbreviation in angle brackets indicates that you must press the named key on the terminal keyboard.

<CTRL/x> In examples, symbols like this indicate that you must hold down the CTRL key while you type the key that follows the slash. Use of this combination of keys may appear on your terminal screen as the letter preceded by the circumflex character. In some instances, it may not appear at all.

System Shutdown Procedures 1

On occasion, routine system maintenance may require you to shut down your system. The exact shutdown procedure that you use depends on whether you want to shut down multiuser mode and remain in single-user mode, shut down and halt the processor, or shut down multiuser mode and reboot. This chapter explains all of these procedures.

1.1 Shutting Down Multiuser Mode

There are three steps for shutting down multiuser mode and staying in single-user mode so that you can perform routine system maintenance. Steps two and three are optional and depend on the type of system maintenance that you want to perform. The steps are:

1. Use the shutdown command to bring the system to single-user mode. The following example shows what to type to shut down the system to single-user mode in fifteen minutes:

   ```
   # /etc/shutdown +15 'to install new devices'
   ```

   The shutdown command logs the specified reason in the /usr/adm/shutdownlog file. Then, it notifies current users of the impending shutdown. It also creates an /etc/nologin file five minutes before the shutdown occurs to prevent users from logging into the system. At the designated time, the shutdown command shuts down multiuser mode.

   Once the system displays the superuser prompt (#), the system is back to single-user mode. The system console is open with the superuser account active. All other terminals are disabled, but all file systems are still mounted. You may now want to unmount file systems or you may want to halt the processor.

   When you restart multiuser mode, the /etc/rc script automatically removes /etc/nologin to re-enable user logins.

2. Unmount file systems. If you want to, you can now unmount the file systems. To unmount all file systems, use the umount command with the -a option. Be sure you are in the root (/) directory before

you issue the umount command. For example:

```
# cd /
# /etc/umount -a
```

This command unmounts those file systems named in the /etc/fstab file and leaves only the root file system mounted. If you have mounted a file system that is not defined in /etc/fstab and you want to unmount it, use the umount command and specify the file system's special device name. You can tell if a file system is mounted by typing:

```
# /etc/mount
```

When specified without options, the mount command displays the currently mounted file systems. For example:

```
/dev/ra0h  on  /usr/staffs
/dev/ra2c on /usr/staff/r1 type ufs
sysname:/usr/staff/ab2 on /usr/staff/ab2 type nfs
        (rw,soft,bg,intr,nosuid)
```

To unmount the /usr/staffs file system, use the umount command as shown in the following example.

```
# /etc/umount /dev/ra0h
```

Notice that to unmount the /usr/staffs file system, you must unmount the device on which it resides. In this example, /usr/staffs resides on the h partition of the ra0 disk.

For more information on both the mount and umount commands, refer to mount(8) in the ULTRIX Reference Pages.

3. Halt the system. After you have issued the shutdown command, you can halt the system with the halt command. (Depending on your processor type, the system will either halt itself, or it will direct you to halt the system.) For example:

```
# /etc/halt
syncing disks ...
>>>
```

The halt command stops the processor and the console monitor prompt is displayed. You can now boot the system to single-user or multiuser mode as described in Chapter 2.

When your system is in single-user mode, you can proceed with the desired maintenance procedure.

1.2 Shutting Down and Halting the System

To shut down multiuser mode and halt the processor, use the shutdown command with the -h option specified. The following example shows what you would type to shut down and halt the processor in 10 minutes:

```
# /etc/shutdown -h +10 'adding a new board to the system'
```

The shutdown command logs the specified reason into the /usr/adm/shutdownlog file. Then it notifies current users of the impending shutdown. At the specified time, the shutdown command shuts down multiuser mode and halts the processor.

When you restart multiuser mode, the /etc/rc script automatically removes /etc/nologin to re-enable user logins.

The halt command provides an alternative shutdown procedure, and should only be invoked from single-user mode.

1.3 Shutting Down and Rebooting the System

To shut down multiuser mode and immediately reboot the system, use the shutdown command with the -r option specified. The following example shows what you would type to shut down and reboot the system in 20 minutes:

```
    # /etc/shutdown -r +20 'doing a quick reboot'
```

The shutdown command logs the specified reason into the /usr/adm/shutdownlog file. Then, it notifies current users of the impending shutdown. It also creates the /etc/nologin file five minutes before the shutdown occurs to prevent users from logging into the system. At the specified time, the shutdown command shuts down multiuser mode, updates the file system superblocks, halts the processor, and immediately reboots multiuser mode. When you restart multiuser mode, the /etc/rc script automatically removes the /etc/nologin file to re-enable user logins.

The reboot command provides an alternative startup and shutdown capability but is not recommended for normal operations.

1.4 Shutting Down a Diskless Client

To shutdown a diskless client, use the shutdown command at the client processor. The shutdown command works the same for diskless clients as it does for any processor. However, you should avoid using the shutdown -r command because the default boot device may not be the Ethernet device.

System Startup Modes 2

During normal operations and after system crashes, you may need to restart or boot the system. To boot any system successfully, you must know what processor you are booting and whether you want the system to come up in single-user or multiuser mode.

This chapter provides information about the available startup modes. It describes what happens when you:

- Boot the system to single-user mode
- Boot the system to multiuser mode
- Invoke multiuser mode from single-user mode
- Use conversational mode when booting to either single-user or multiuser mode

Chapter 3 describes the specific boot commands for each VAX processor.

2.1 Booting the System to Single-User Mode

When you boot the system to single-user mode:

- The system comes up with only the root file system mounted. All other file systems are unmounted and all configured terminals and networking are disabled. You have access only to those files and commands in the root file system unless you explicitly mount other file systems.
- The Bourne shell (sh) runs at the console under a partially active superuser account. Although the sh program has read the .profile file, the login utility has not been invoked, the superuser is not logged in, and a full environmental initialization for the superuser account has not occured.
- You must invoke the fsck program to check the integrity of the root file system. If the fsck program reports inconsistencies in root, you must correct them before mounting any other file system. For a description of the command and its options, see fsck(8) in the ULTRIX Reference Pages. See the Guide to System Crash Recovery for examples of how and when to use the fsck program to check for

and correct file system inconsistencies.

- If you need other file systems mounted, you must invoke the mount command to add the file systems. For a description of the command and its options, see mount(8) in the ULTRIX Reference Pages.

2.2 Booting the System to Multiuser Mode

When you boot the system to multiuser mode, the init program invokes the /etc/rc startup script. The contents of this script and the /etc/rc.local script determine what happens, but typically:

- The system comes up with the root (/) and any file systems specified in the /etc/fstab file mounted. Consequently, you have access to all files and commands in the root file system and other mounted file systems.
- All terminals listed in the /etc/ttys file are enabled. Users with accounts in the /etc/passwd file can log in to the system.
- The script automatically invokes fsck, which checks root and other file systems listed in the /etc/fstab file.
    - If fsck finds no inconsistencies, the /etc/rc script starts multiuser mode.
    - If fsck finds inconsistencies, the system stays in single-user mode, and you should run fsck on the file systems with reported inconsistencies. After correcting all reported inconsistencies, reinvoke or reboot multiuser mode.

2.2.1 Invoking Multiuser Mode from Single-User Mode

To invoke the multiuser mode from single-user without having to reboot, follow these steps:

1. Go to the root (/) directory.
2. Check for any active programs, daemons, or users on any mounted file system.
3. If you find any active processes, stop them.

4. Unmount all file systems by typing the umount command with the –a option.

   ```
   # /etc/umount -a
   ```

   The umount program checks the /etc/fstab file and unmounts all file systems listed except root.

5. Type the mount command with no options. The program lists any file systems that are still mounted. For example:

   ```
   # /etc/mount
   /dev/ra0a on / type ufs
   /dev/ra1a on /tmp type ufs
   ```

6. If any file system besides root is still mounted, type the umount command again. Specify the mounted file system by typing the device and partition on which the file system is mounted. For example, to unmount /tmp (as shown in the preceding listing), type:

   ```
   # /etc/umount /dev/ra1a
   ```

   If the unmounting is successful, the program responds by listing the root (/) file system only. This indicates that all file systems except root are now unmounted.

7. Check file systems. After unmounting all file systems, use the fsck command to check them for inconsistencies. For example, type:

   ```
   # /etc/fsck
   ```

   When you type fsck without options, the program checks the file systems listed in the /etc/fstab file, and notifies you of inconsistencies. For more information on the command and its options, see fsck(8) in the ULTRIX Reference Pages. For a description of how and when to use the fsck program to correct file system inconsistencies, see the Guide to System Crash Recovery.

8. Exit single-user mode. After running fsck and correcting any reported inconsistencies, type CTRL/D at the console. CTRL/D ends the single-user mode session.

Once single-user mode ends, the system initialization program, init, automatically invokes the multiuser start up script, /etc/rc. During execution, /etc/rc invokes /etc/rc.local. When these multiuser startup scripts successfully complete execution, the system is in multiuser mode.

Note

> You can not mount unclean file systems. If you attempt to enter multiuser mode with file systems that were not unmounted cleanly or were not checked with the fsck command, the system will not enter multiuser mode.

2.2.2 Booting Multiuser Mode from Console Mode

To boot multiuser mode directly from console mode, enter the multiuser boot command that is specific to your processor type. See Chapter 3 for a description of the processor-specific boot commands.

2.2.3 Booting the System in Conversational Mode

To boot the system in conversational mode, you enter one of the processor-specific boot commands listed in Chapter 3. In any case, when you boot in conversational mode, the program prompts you to enter an image name. For example:

```
Enter image name:
```

In response to this prompt, enter the name of the kernel image. For example:

```
Enter image name: vmunix
```

We recommend that you load the default kernel; however, you can optionally load another. If you take this option, use the following syntax:

(*device, partition*)*kernel_name*

The first variable, *device*, specifies the device where the image is located. The booted device is the default. The second variable, *partition*, specifies the partition on the device. Partition *a* of the booted device is the default. The *kernel_name* can be any kernel existing at either the default location or at the location you specify.

Some *device* and *partition* syntax rules are:

- You can specify a single number to define the device number using the default partition. For example: (3)vmunix
- You can specify a single letter from a to h to define the partition using the default boot device. For example: (g)vmunix

- You can specify a number and a letter for the device and partition. For example: (3,g)vmunix
- You can specify two numbers, the second of which corresponds to a letter from a through h for a partition, starting with 0 for a and ending with 7 for h. For example: (3,6)vmunix

The first time you enter invalid input, the boot program displays the message:

```
Syntax Error

Examples of valid input syntax are:

newvmunix      - Loads newvmunix from the booted device, partition a
(g)vmunix      - Loads vmunix from the booted device, partition g
(3)vmunix.old  - Loads vmunix.old from device unit 3, partition a
(9,g)vmunix    - Loads vmunix from device unit 9, partition g
(4,7)vmunix    - Loads vmunix from device unit 4, partition h

Note: If specified, the device unit number must be the PHYSICAL
unit number of a device connected to the SAME CONTROLLER as the
booted device.
```

If you enter another invalid entry, the boot program simply responds:

```
    Syntax Error
```

Processor-Specific Boot Commands 3

This chapter provides guidelines for booting your processor. The boot commands that you use depend on the VAX processor type and its attached hardware. The following sections describe these commands with the processors grouped by section according to their boot commands.

3.1 Booting VAXserver, MicroVAX, and VAXstation Processors (except VAXstation 3100)

This section describes the boot commands for the following processors:

- The MicroVAX 2000
- The MicroVAX 3500 and MicroVAX 3600
- The VAXstation II
- The VAXstation II/GPX
- The VAXstation 2000
- The VAXstation 3200 and VAXstation 3500
- The VAXserver 100
- The VAXserver 3500, VAXserver 3600, and VAXserver 3602

3.1.1 Booting from the Console

Follow these steps to boot your processor from the console:

1. Release the HALT button on your processor. See your Owner's Manual for the location of the HALT button.
2. Boot the default system disk by typing:

   ```
   >>>b
   ```

   The console program attempts to boot the first device it finds that contains a valid boot block. The program first searches diskette devices, other removeable disks - RA60, for example - and finally Winchester devices. Winchester devices are searched from lowest to highest unit number. Be aware that removeable disks have a higher priority than Winchester devices regardless of unit number.

3. Decide which startup mode you want, then type the corresponding entry at the prompt. For example:

Mode	Prompt and Entry
Single-user	>>> b/2 dua*n*
Multiuser	>>> b dua*n*
Conversational (single-user mode)	>>> b/3 dua*n*
Conversational (multiuser mode)	>>> b/1 dua*n*

The variable *n* specifies the device number of the system disk drive. For example, to boot vmunix (the kernel image) to single-user mode from RD53 drive 1 on a MicroVAX II, type:

```
>>> b/2 dua1
```

See Chapter 2 for additional information on startup modes.

3.1.2 Booting from a TK50 Tape

When doing an installation or booting the standalone kernel for system management tasks, you may have to boot from a TK50 tape. After installing the TK50 boot tape, type:

```
>>> b mua0
```

In response to this entry, the console subsystem boots the TK50 boot tape.

3.1.3 Booting from the Network

You boot from the network when you are:

1. Booting a diskless system
2. Initiating an installation from a remote server
3. Booting a standalone kernel from a remote server in order to perform system management tasks

To boot the system from the network, use one of the following commands:

- When booting the MicroVAX II, the VAXstation II, the VAXstation II/GPX, the VAXstation 3200, the VAXstation 3500, a MicroVAX 3000 series processor, a VAXserver 100, and a VAXserver 3000 series processor, type:

```
>>> b xqa0
```

- When booting the MicroVAX 2000 or the VAXstation 2000 from the network, type:

```
>>> b esa0
```

In response to your entry, the console subsystem boots the system and displays the memory and hardware configuration.

3.2 Booting MicroVAX 3300 and MicroVAX 3400 Processors

The following sections describe how to boot each of these processors from the console or the network.

3.2.1 Booting from the Console

Follow these steps to boot your processor from the console:

1. Release the HALT button on your processor. See your Owner's Manual for the location of the HALT button.
2. Identify which device (if any) was set as the default by typing:

```
>>> show boot
```

The console program responds with the device name. For example:

```
>>> show boot
DIA0
```

3. Boot the default system disk by typing:

```
>>> b
```

The console program boots the default device and displays the device name. For example:

```
>>> b
(BOOT/R5:0 DIA0)
```

4. Decide which startup mode you want, then type the corresponding entry at the prompt. For example:

Mode	**Prompt and Entry**
Single-user	>>> b/2 dia*n*
Multiuser	>>> b dia*n*
Conversational (single-user mode)	>>> b/3 dia*n*

Conversational (multiuser mode)	>>> b/1 dia*n*

The variable *n* specifies the device number of the system disk drive. For example, to boot vmunix (the kernel image) to single-user mode from drive 1 on a MicroVAX 3400, type:

```
>>> b/2 dia1
```

See Chapter 2 for additional information on startup modes.

3.2.2 Booting from the Network

You boot from the network when you are:

1. Booting a diskless system
2. Initiating an installation from a remote server
3. Booting a standalone kernel from a remote server in order to perform system management tasks

To boot the system from the network, use the following command:

```
>>> b esa0
```

In response to your entry, the console subsystem boots the system and displays the memory and hardware configuration.

3.3 Booting a VAXstation 3100

Your choice of a boot command for a VAXstation 3100 depends on your hardware configuration. The following sections describe the various boot commands.

3.3.1 Booting from the Console

Follow these steps to boot your processor from the console:

1. Press the HALT button on your processor. See your Owner's Manual for the location of the HALT button.

2. Find out which device (if any) was set as the default by typing:

```
>>> show boot
```

- If a default device was set, the console program responds with the name of the default device. For example:

  ```
  >>> show boot
  DKA300
  ```

- If no default device was set, the console program responds as follows:

  ```
  >>> show boot
   ....
  ```

3. Get a boot device listing by typing:

   ```
   >>> show device
   ```

   The console program displays a device listing similar to this:

   ```
   VMS/VMB   ULTRIX  ADDR          DEVTYP   NUMBYTES  RM/FX  WP  DEVNAME

   ESA0      SE0     08-00-2B-07-05-09

   DKA300    RZ23    A/3/0/00      DISK     06407E00    FX        RZ23
   MKA500    TZ5     A/5/0/00      TAPE     ........    RM
   HostID            A/6           INITR
   DKB100    RZ9     B/1/0/00      DISK     1383B200    FX        RZ55
   DKB200    RZ10    B/2/0/00      RDDISK   06407E00    RM        RZ23
   DKB300    RZ11    B/3/0/00      RDDISK   06407E00    RM        RZ23
   DKB400    RZ12    B/4/0/00      DISK     0C3B1600    RM        RRD40
   MKB500    TZ13    B/5/0/00      TAPE     ........    RM
   HostID            B/6           INITR
   ```

   In the preceding display:

- Column 1 lists the boot command name associated with a particular device configured at a specific address.
- Column 2 lists the ULTRIX device mnemonic and number associated with a particular device type.
- Column 3 lists the address of the specific device. The first character specifies the SCSI controller identification (either A or B). The second character specifies the SCSI bus identification number. The remaining characters are zeroes.
- Column 4 lists the device types.
- Column 5 lists internal addressing information needed by the system.
- Column 6 lists mnemonics that indicate whether the device is fixed or removeable.

- Column 7 lists the physical device name.

4. Boot the default system device by typing:

```
>>> b
```

The console program boots the default device. However, if no default device was set previously, the console defaults to a network boot.

5. Boot a specific device by typing:

```
>>> b boot command name
```

For example, assume you wanted to boot an RZ23 fixed disk at SCSI controller A, SCSI bus 3. To boot this device, type:

```
>>> b DKA300
```

Note

The console program is not case sensitive when accepting boot commands for the VAXstation 3100 processor. Consequently, you can use either upper-case or lower-case letters when typing the boot command name.

6. Decide which startup mode you want, then type the corresponding entry at the prompt.

- If you are booting a disk device at SCSI controller A, use the following list to determine the correct entry:

Mode	**Prompt and Entry**
Single-user	>>> b/2 dka*n*
Multiuser	>>> b dka*n*
Conversational (single-user mode)	>>> b/3 dka*n*
Conversational (multiuser mode)	>>> b/1 dka*n*

The variable *n* specifies the SCSI bus identification number of the system disk drive. For example, to boot vmunix (the kernel image) to single-user mode from the system disk at SCSI bus ID 3, type:

```
>>> b/2 dka300
```

To boot vmunix (the kernel image) to multiuser mode from the system disk at SCSI bus ID 2, type:

```
>>> b dka200
```

- If you are booting a disk device at SCSI controller B, use the following list to determine the correct entry:

Mode	Prompt and Entry
Single-user	>>> b/2 dkb*n*
Multiuser	>>> b dkb*n*
Conversational (single-user mode)	>>> b/3 dkb*n*
Conversational (multiuser mode)	>>> b/1 dkb*n*

The variable *n* specifies the SCSI bus identification number of the system disk drive. For example, to boot vmunix (the kernel image) to single-user mode from the system disk at SCSI bus ID 3, type:

```
>>> b/2 dkb300
```

To boot vmunix (the kernel image) to multiuser mode from SCSI bus ID 2, type:

```
>>> b dkb200
```

See Chapter 2 for additional information on startup modes.

3.3.2 Booting from a TZ30 or TZK50 Tape

When doing an installation or booting the standalone kernel for system management tasks, you may have to boot from tape. After installing the boot tape, use one of the following boot commands:

- If you are booting from tape at SCSI controller A, use this syntax:

```
>>> b mkan
```

The variable *n* specifies the SCSI bus identification number of the system tape. For example, to boot from tape at SCSI controller A, SCSI bus ID 3, type:

```
>>> b mka300
```

In response to this entry, the console subsystem boots the boot tape.

- If you are booting from tape at SCSI controller B, use this syntax:

  ```
  >>> b mkbn
  ```

  The variable *n* specifies the SCSI bus identification number of the system tape. For example, to boot from tape at SCSI controller B, SCSI bus ID 3, type:

  ```
  >>> b mkb300
  ```

3.3.3 Booting from the Network

You boot from the network when you are:

1. Booting a diskless system
2. Initiating an installation from a remote server
3. Booting a standalone kernel from a remote server in order to perform system management tasks

To boot the system from the network, use the following command:

```
>>> b esa0
```

In response to your entry, the console subsystem boots the system and displays the memory and hardware configuration.

3.4 Booting a VAX-11/750

Your choice of a boot command for a VAX-11/750 depends on your hardware configuration. The following sections describe the boot commands for both local disks and remote disks connected to an HSC.

3.4.1 Booting a Local Disk

The following list describes the boot commands for local disks:

1. To boot the default system disk to multiuser mode, type:

   ```
   >>> b
   ```

2. To boot the system disk to single-user mode, type:

```
>>> b/3
```

The console subsystem reads the askboo.cmd file, boots the default system disk, and displays the prompt:

```
Enter image name:
```

In response to this prompt, enter the name of the kernel.

3. To boot an alternate disk, use one of the commands listed in the following table. Be aware that RP07 drives are not supported as boot devices.

Drive Type	Single-user	Multiuser	Conversational Single-User	Conversational Multiuser
RA*xx* Disk	b/2 dua*n*	b dua*n*	b/3 dua*n*	b/1 dua*n*
RP05/06 and RM03/05/80 Disks	b/2 dba*n*	b dba*n*	b/3 dba*n*	b/1 dba*n*

The variable *xx* is the model number of the system disk drive and the variable *n* represents the unit number of the system disk drive. For example, to boot multiuser mode from an RM05 system disk, unit number one, type:

```
>>> b dba1
```

See Chapter 2 for information on each of the startup modes.

3.4.2 Booting an HSC Disk

On a VAX-11/750 processor, the system must load CI microcode contained on the console cassette. Therefore, you must ensure that a valid console cassette is in the TU58 drive and that your selector switch is at the cassette setting before attempting to boot an HSC disk.

The console cassette contains the boot command procedure files that enable the system to boot the default and alternate disks. The boot command procedure files are:

- askboo.cmd which boots the default disk to single-user mode
- defboo.cmd which boots the default disk to multiuser mode
- cira.cmd which boots an alternate disk to single-user or multiuser mode

Once the CI microcode is loaded, the software can boot either the default or an alternate HSC disk to a particular startup mode. The following list describes the boot commands:

1. To boot the default HSC system disk to multiuser mode, type:

   ```
   >>> b
   ```

   The console subsystem reads the defboo.cmd file, boots the default system device, and brings the system up in multiuser mode.

2. To boot the default HSC system disk to single-user mode, use this format:

   ```
   >>> b/800 dda0
   BOOT58> @askboo.cmd
   ```

   The console subsystem reads the askboo.cmd file, boots the default system device, and displays the prompt:

   ```
   Enter image name:
   ```

   In response to this prompt, enter the name of the kernel.

3. To boot an alternate HSC disk, use this format:

   ```
   >>> b/800 dda0
   BOOT58> D/G 2 HSC#
   BOOT58> D/G 3 unit#
   BOOT58> @cira.cmd
   ```

   The *HSC#* is the remote CI port number assigned to the specific HSC controller. The *unit#* variable is the device number of the system disk drive. The @cira.cmd string invokes the HSC boot command file.

   Note

   Both the HSC number and the unit number must be expressed in hexadecimal.

   The console subsystem reads the cira.cmd file, boots the alternate system device, and displays the prompt:

   ```
   Enter image name:
   ```

   In response to this prompt, enter the name of the kernel.

3.5 Booting a VAX-11/780 or a VAX-11/785

Your choice of a boot command for a VAX-11/780 or a VAX-11/785 depends on your hardware configuration. The following sections describe the boot commands for both local disks and remote disks connected to an HSC.

Note

The descriptions in this section assume that the front-end console storage device has been updated to reflect the proper default boot device. Refer to Section 3.10 for information on how to do this.

3.5.1 Booting a Local Disk

The VAX-11/780 and VAX-11/785 processors have front-end console storage devices that contain boot command procedure files. The command procedure files that enable you to boot the default and alternate disks are:

- defboo.cmd, which boots the default disk to multiuser mode
- askboo.cmd, which boots the default disk to single mode
- mbahp.cmd, which boots an alternate MASSBUS disk to single-user mode
- ubara.cmd, which boots an alternate UNIBUS disk connected to a UDA-50 controller to single-user mode,

The following list describes the boot commands:

1. To boot the default system disk to multiuser mode, type:

   ```
   >>> b
   ```

   The console subsystem reads the defboo.cmd file, boots the default system disk, and brings the system up in multiuser mode.

2. To boot the default system disk to single-user mode, type:

   ```
   >>> b ask
   ```

   The console subsystem reads the askboo.cmd file, boots the default system disk, and displays the prompt:

   ```
   Enter image name:
   ```

   In response to this prompt, enter the name of the kernel.

3. To boot an alternate MASSBUS disk to single-user mode, use this format:

   ```
   >>> d r1 TR#
   >>> d r3 unit#
   >>> @mbahp.cmd
   ```

The *TR#* variable is the TR level number of the MASSBUS adapter. The *unit#* variable is the device number of the system disk drive. The @mbahp.cmd string invokes the MASSBUS adapter boot command file.

Note

Both the TR level number and the unit number must be expressed in hexadecimal.

The console subsystem reads the mbahp.cmd file, boots the alternate system disk, and displays the prompt:

```
Enter image name:
```

In response to this prompt, enter the name of the kernel.

4. To boot an alternate UNIBUS disk connected to a UDA-50 controller to single-user mode, use this format:

```
>>> d r1 TR#
>>> d r3 unit#
>>> @ubara.cmd
```

The *TR#* variable is the TR level number of the UNIBUS adapter. The *unit#* variable is the device number of the system disk drive. The @ubara.cmd string invokes the UNIBUS adapter boot command file.

Note

Both the TR level number and the unit number must be expressed in hexadecimal.

The console subsystem reads the ubara.cmd file, boots the alternate system disk, and displays the prompt:

```
Enter image name:
```

In response to this prompt, enter the name of the kernel.

3.5.2 Booting an HSC Disk

The VAX-11/780 and VAX-11/785 processors have front-end console storage devices that contain boot command procedure files. The command procedure files that enable you to boot the default and alternate HSC disks are:

- defboo.cmd, which boots the default disk to multiuser mode
- askboo.cmd, which boots the default disk to single-user mode
- cira.cmd, which boots an alternate disk to single-user mode

The following list describes the boot commands:

1. To boot the default system disk to multiuser mode, type:

   ```
   >>> b
   ```

   The console subsystem reads the defboo.cmd file, boots the default system disk, and brings the system up in multiuser mode.

2. To boot the default system disk to single-user mode, type:

   ```
   >>> b ask
   ```

   The console subsystem reads the askboo.cmd file, boots the default system disk, and displays the prompt:

   ```
   Enter image name:
   ```

   In response to this prompt, enter the name of the kernel.

3. To boot an alternate HSC disk to single-user mode, use this format:

   ```
   >>> d r2 HSC#
   >>> d r3 unit#
   >>> @cira.cmd
   ```

   The *HSC#* variable is the remote CI port number assigned to the specific HSC controller. The *unit#* variable is the device number of the system disk drive. The @cira.cmd string invokes the HSC boot command file.

   Note

   Both the HSC number and the unit number must be expressed in hexadecimal.

   The console subsystem reads the cira.cmd file, boots the alternate system disk, and displays the prompt:

   ```
   Enter image name:
   ```

   In response to this prompt, enter the name of the kernel.

3.6 Booting a VAX 6210 or a VAX 6220

Your choice of a boot command for a VAX 6210 or a VAX 6220 depends on your hardware configuration. The following sections describe the boot commands for both local disks and remote disks connected to an HSC.

3.6.1 Booting a Local Disk

The following list describes the boot commands for local disks:

1. To boot the default system disk to multiuser mode, type:

   ```
   >>> b
   ```

2. To boot the system disk to single-user mode, type:

   ```
   >>> b ask
   ```

   The console subsystem reads the askboo.cmd file, boots the default system disk, and displays the prompt:

   ```
   Enter image name:
   ```

   In response to this prompt, enter the name of the kernel.

3. To boot an alternate disk, type:

   ```
   >>> b/xmi:BIA# /bi:BI# /r5:1000b duunit#
   ```

   The *BIA#* variable represents the number (0,1,2, or 3) of the BI adapter connected to the xmi. The *BI#* variable represents the BI node number of the xmi adapter. The *unit#* variable represents the device number of the system disk drive.

   Note

   The BIA number, the BI number, and the unit number must be expressed in hexadecimal.

   The console subsystem boots the alternate system device, and displays the prompt:

   ```
   Enter image name:
   ```

   In response to this prompt, enter the name of the kernel.

3.6.2 Booting an HSC Disk

On a VAX 6210 or a VAX 6220 processor, the system must load CI microcode contained on the TK50 cartridge. Therefore, you must ensure that a valid cartridge is in the TK50 drive before attempting to boot an HSC disk. See your Field Services representative for details on the correct procedure.

The following list describes the boot commands:

1. To boot the default HSC system disk to multiuser mode, type:

   ```
   >>> b
   ```

   The console subsystem reads the defboo.cmd file, boots the default system device, and brings the system up in multiuser mode.

2. To boot the default HSC system disk to single-user mode, type:

   ```
   >>> b ask
   ```

   The console subsystem reads the askboo.cmd file, boots the default system device, and displays the prompt:

   ```
   Enter image name:
   ```

   In response to this prompt, enter the name of the kernel.

3. To boot an alternate HSC disk, type:

   ```
   >>> b /xmi:BIA# /bi:BI# /node:HSC# /r5:1000b duunit#
   ```

   The *BIA#* variable represents the number (0, 1, 2, or 3) of the BI adapter connected to the xmi. The *BI#* variable represents the BI node number of the xmi adapter. The *HSC#* represents the remote CI port number assigned to the specific HSC controller. The du *unit#* variable represents the device number of the system disk drive.

   Note

   > The BIA number, the BI number, the HSC number, and the unit number must be expressed in hexadecimal.

   The console subsystem boots the alternate system device, and displays the prompt:

   ```
   Enter image name:
   ```

   In response to this prompt, enter the name of the kernel.

3.7 Booting a VAX 8200, VAX 8250, VAX 8300 or a VAX 8350

On a VAX 8200, 8250, 8300, or 8350, the boot command you use depends on your hardware configuration. The following sections describe the boot commands for both local disks and remote disks connected to an HSC.

Note

> The descriptions in this section assume that the EEPROMs have been reprogrammed to reflect the proper default boot device. Refer to Section 3.10 for information on how to do this.

3.7.1 Booting a Local Disk

On any of these processors, the default boot command boots the default device described in the EEPROM of the processor. Programming the EEPROM is described in the VAX Owner's Manual.

The following list describes the boot commands that you use to boot local disks:

1. To boot the default system disk to multiuser mode, type:

   ```
   >>> b
   ```

   The console subsystem boots the default system disk and brings the system up in multiuser mode.

2. To boot the default system disk to single-user mode, type:

   ```
   >>> b/r5:3
   ```

   The console subsystem boots the default system disk, brings the system up in single-user mode, and displays the prompt:

   ```
   Enter image name:
   ```

   In response to this prompt, enter the name of the kernel.

3. To boot an alternate disk, use one of the commands listed in the following table.

Mode	Prompt and Entry
Single-user	>>> b/r5:2 du*BI#n*
Multiuser	>>> b du*BI#n*
Conversational (In single-user mode)	>>> b/r5:3 du*BI#n*
Conversational (In multiuser mode)	>>> b/r5:1 du*BI#n*

The *BI#* variable represents the BI node number and the *n* variable represents the unit number of the desired boot device. For example, to boot in conversational mode (assuming a BI node number of 4 and a unit number of 0), type:

```
>>> b/r5:3 du40
```

The system comes up in conversational mode, signified by the prompt:

```
Enter image name:
```

In response to this prompt, enter the name of the kernel.

3.7.2 Booting an HSC Disk

On any of these processors, the default boot command boots the default device described in the EEPROM of the processor. Programming the EEPROM is described in the VAX Owner's Manual.

The following list describes the boot commands that you use to boot the default and alternate disks:

1. To boot the default system disk to multiuser mode, type:

   ```
   >>> b
   ```

   The console subsystem boots the default system disk and brings the system up in multiuser mode.

2. To boot the default system disk to single-user mode, use this format:

   ```
   >>> b/r5:800
   BOOT58> @askboo.cmd
   ```

   The console sybsystem reads the askboo.cmd file, boots the default system disk, and displays the prompt:

   ```
   Enter image name:
   ```

   In response to this prompt, enter the name of the kernel.

3. To boot an alternate HSC disk, use this format:

   ```
   >>> b/r5:800 csa1
   BOOT58> D/G 1 BI#
   BOOT58> D/G 2 HSC#
   BOOT58> D/G 3 unit#
   BOOT58> @cira.cmd
   ```

   The *BI#* variable represents the BI node number of the CI adaptor. The *HSC#* variable represents the remote CI port number assigned to the specific HSC controller. The *unit#* variable represents the device number of the system disk drive. The @cira.cmd string invokes the

HSC boot command file.

Note

> The BI number, the HSC number, and the unit number must be expressed in hexadecimal.

The console subsystem reads the cira.cmd file, boots the alternate system device, and displays the prompt:

```
Enter image name:
```

In response to this prompt, enter the name of the kernel.

3.8 Booting a VAX 8500, VAX 8530, VAX 8550, VAX 8700, VAX 8800, or a VAX 8810

On a VAX 8500, 8530, 8550, 8700, 8800, or 8810, the boot command you use depends on your hardware configuration. The following sections describe the boot commands for both local disks and remote disks connected to an HSC.

Note

> The descriptions in this section assume that the front-end console storage device has been updated to reflect the proper default boot device. Refer to Section 3.12 for information on how to do this.

3.8.1 Booting a Local Disk

All of these processors have front-end console storage devices that contain boot command procedure files. The command procedure files that enable you to boot the default and alternate disks are:

- defboo.com which boots the default system disk to multiuser mode
- askboo.com which boots the default system disk to single-user mode
- bdara.com which boots an alternate disk to single-user mode if the BI adapter is a KDB50

The following list describes the boot procedures for the various disks and modes:

1. To boot the default system device to multiuser mode, type:

```
>>> b
```

The console subsystem reads the defboo.com file, boots the default system device, and brings the system up in multiuser mode.

2. To boot the default system device to single-user mode, type:

   ```
   >>> b ask
   ```

   The console subsystem reads the askboo.com file, boots the default system disk, and displays the prompt:

   ```
   Enter image name:
   ```

   In response to this prompt, enter the name of the kernel.

3. To boot an alternate disk (where the BI adapter is a KDB50) to single-user mode, use the format:

   ```
   >>> d r1 BIA#BI#
   >>> d r3 unit#
   >>> @bdara.com
   ```

   The *BIA#* variable represents the number of the BI adapter (0, 1, 2, or 3) connected to the KDB50. The *BI#* variable represents the BI node number of the KDB50 adapter. The *unit#* variable is the device number of the system disk drive. The @bdara.com string invokes the KDB50 boot command file.

   Note

   > The BI adapter number, the BI node number, and the unit number must be expressed in hexadecimal.

   The console subsystem reads the bdara.com file, boots the alternate system disk, and displays the prompt:

   ```
   Enter image name:
   ```

   In response to this prompt, enter the name of the kernel.

3.8.2 Booting an HSC Disk

All of these processors have front-end console storage devices that contain boot command procedure files. The command procedure files that enable you to boot the default and alternate disks are:

- defboo.com, which boots the default system disk to multiuser mode
- askboo.com, which boots the default system disk to single-user mode
- bcira.com, which boots an alternate disk to single-user mode if the BI adapter is a BCA

The following list describes the boot commands for the various disks and modes:

1. To boot the default system disk to multiuser mode, type:

   ```
   >>> b
   ```

   The console subsystem reads the defboo.com file, boots the default system device, and brings the system up in multiuser mode.

2. To boot the default system disk to conversational mode, type:

   ```
   >>> b ask
   ```

   The console subsystem reads the askboo.com file, boots the default system device, and displays the prompt:

   ```
   Enter image name:
   ```

   In response to this prompt, enter the name of the kernel.

3. To boot an alternate HSC disk to single-user mode when the CI adapter is a BCA, use the format:

   ```
   >>> d r1 BIA#BI#
   >>> d r2 HSC#
   >>> d r3 unit#
   >>> @bcira.com
   ```

   The *BIA#* variable represents the number (0, 1, 2, or 3) of the BI adapter connected to the CI adapter. The *BI#* variable represents the BI node number of the CI adapter. The *HSC#* variable represents the remote CI port number assigned to the specific HSC controller. The *unit#* variable is the device number of the system disk drive. The @bcira.com string invokes the HSC boot command procedure file.

   Note

   > The BI adapter number, the BI node number, the HSC number, and the unit number must be expressed in hexadecimal.

   The console subsystem reads the bcira.com file, boots the alternate HSC disk, and displays the prompt:

```
Enter image name:
```

In response to this prompt, enter the name of the kernel.

3.9 Booting a VAX 8820

On a VAX 8820 processor, the boot command you use depends on your hardware configuration. The following sections describe the boot commands for both local disks and remote disks connected to an HSC.

Note

> The descriptions in this section assume that the front-end console storage device has been updated to reflect the proper default boot device. Refer to Section 3.12 for information on how to do this.

3.9.1 Booting a Local Disk

All of these processors have front-end console storage devices that contain boot command procedure files. The command procedure files that enable you to boot the default and alternate disks are:

- defboo.cmd which boots the default system disk to multiuser mode
- askboo.cmd which boots the default system disk to single-user mode
- bdara.cmd which boots an alternate disk to single-user mode if the BI adapter is a KDB50

The following list describes the boot procedures for the various disks and modes:

1. To boot the default system device to multiuser mode, type:

   ```
   >>> b
   ```

   The console subsystem reads the defboo.cmd file, boots the default system device, and brings the system up in multiuser mode.

2. To boot the default system device to single-user mode, type:

   ```
   >>> b ask
   ```

   The console subsystem reads the askboo.cmd file, boots the default system disk, and displays the prompt:

   ```
   Enter image name:
   ```

   In response to this prompt, enter the name of the kernel.

3. To boot an alternate disk (where the BI adapter is a KDB50) to single-user mode, use the format:

```
>>> d r1 BIA#BI#
>>> d r3 unit#
>>> @bdara.cmd
```

The *BIA#* variable represents the number of the BI adapter (0, 1, 2, 3, 4, or 5) connected to the KDB50. The *BI#* variable represents the BI node number of the KDB50 adapter. The *unit#* variable is the device number of the system disk drive. The @bdara.cmd string invokes the KDB50 boot command file.

Note

> The BI adapter number, the BI node number, and the unit number must be expressed in hexadecimal.

The console subsystem reads the bdara.cmd file, boots the alternate system disk, and displays the prompt:

```
Enter image name:
```

In response to this prompt, enter the name of the kernel.

3.9.2 Booting an HSC Disk

All of these processors have front-end console storage devices that contain boot command procedure files. The command procedure files that enable you to boot the default and alternate disks are:

- defboo.cmd, which boots the default system disk to multiuser mode
- askboo.cmd, which boots the default system disk to single-user mode
- bcara.cmd, which boots an alternate disk to single-user mode if the BI adapter is a BCA

The following list describes the boot commands for the various disks and modes:

1. To boot the default system disk to multiuser mode, type:

```
>>> b
```

The console subsystem reads the defboo.cmd file, boots the default system device, and brings the system up in multiuser mode.

2. To boot the default system disk to conversational mode, type:

   ```
   >>> b ask
   ```

   The console subsystem reads the askboo.cmd file, boots the default system device, and displays the prompt:

   ```
   Enter image name:
   ```

   In response to this prompt, enter the name of the kernel.

3. To boot an alternate HSC disk to single-user mode when the CI adapter is a BCA, use the format:

   ```
   >>> d r1 BIA#BI#
   >>> d r2 HSC#
   >>> d r3 unit#
   >>> @bcara.cmd
   ```

   The *BIA#* variable represents the number (0, 1, 2, 3, 4, or 5) of the BI adapter connected to the CI adapter. The *BI#* variable represents the BI node number of the CI adapter. The *HSC#* variable represents the remote CI port number assigned to the specific HSC controller. The *unit#* variable is the device number of the system disk drive. The @bcara.cmd string invokes the HSC boot command procedure file.

   Note

   > The BI adapter number, the BI node number, the HSC number, and the unit number must be expressed in hexadecimal.

   The console subsystem reads the bcara.cmd file, boots the alternate HSC disk, and displays the prompt:

   ```
   Enter image name:
   ```

   In response to this prompt, enter the name of the kernel.

3.10 Booting a VAX 8600 or a VAX 8650

On a VAX 8600 or VAX 8650, the boot command you use depends on your hardware configuration. The following sections describe the boot commands for both local disks and remote disks connected to an HSC.

Note

> The descriptions in this section assume that the front-end console storage device has been updated to reflect the proper default boot device. Refer to Section 3.10 for information on how to do this.

3.10.1 Booting a Local Disk

Both of these processors have front-end console storage devices that contain boot command procedure files. These files enable you to boot the default and alternate disks. They are:

- defboo.com which boots the default system device to multiuser mode
- askboo.com which boots the default system device to single-user mode
- mbahp.com which boots an alternate MASSBUS disk to single-user mode
- ubara.com which boots an alternate UNIBUS disk connected to a UDA-50 controller to single-user mode

The following list describes the boot procedures for the various disks and modes.

1. To boot the default system disk to multiuser mode, type:

   ```
   >>> b
   ```

   The console subsystem reads the defboo.com file, boots the default system device, and brings the system up in multiuser mode.

2. To boot the default system device to single-user mode, type:

   ```
   >>> b ask
   ```

   The console subsystem reads the askboo.com file, boots the default system device, and displays the prompt:

   ```
   Enter image name:
   ```

   In response to this prompt, enter the name of the kernel.

3. To boot an alternate MASSBUS disk to single-user mode, use this format:

```
>>> d r1 SBI#TR#
>>> d r3 unit#
>>> @mbahp.com
```

The *SBI#* variable represents the Synchronous Backplane Interconnect I/O adapter number (either 0 or 1). The *TR#* variable represents the TR level number of the MASSBUS adapter. The *unit#* variable is the device number of the system disk drive. The @mbahp.com string invokes the MASSBUS boot command procedure file.

Note

> The SBI number, TR level number, and the unit number must be expressed in hexadecimal.

The console subsystem reads the mbahp.com file, boots the alternate disk to single-user mode, and displays this prompt:

```
Enter image name:
```

In response to this prompt, enter the name of the kernel.

To boot an alternate UNIBUS disk connected to a UDA-50 controller to single-user mode, use this format:

```
>>> d r1 SBI#TR#
>>> d r3 unit#
>>> @ubara.com
```

The *SBI#* variable represents the SBI I/O adapter number (either 0 or 1). The *TR#* variable represents the TR level number of UNIBUS adapter. The *unit#* variable represents the device number of the system disk drive. The @ubara.com string invokes the UNIBUS boot command procedure file.

Note

> The SBI number, the TR level number, and the unit number must be expressed in hexadecimal.

The console subsystem reads the ubara.com file, boots the alternate disk to single-user mode, and displays the prompt:

```
Enter image name:
```

In response to this prompt, enter the name of the kernel.

3.10.2 Booting an HSC Disk

The VAX 8600 and VAX 8650 processors have front-end console storage devices that contain boot command procedure files. The command procedure files that enable you to boot the default and alternate HSC disks are:

- defboo.com which boots the default system disk to multiuser mode
- askboo.com which boots the default system disk to single-user mode
- cira.com which boots an alternate system disk to single-user mode

The following list describes the boot commands for for the various disks and modes.

1. To boot the default system disk to multiuser mode, type:

   ```
   >>> b
   ```

   The console subsystem reads the defboo.cmd file, boots the default system device, and brings the system up in multiuser mode.

2. To boot the default system device to conversational mode, type:

   ```
   >>> b ask
   ```

   The console subsystem reads the askboo.cmd file, boots the default system device, and displays the prompt:

   ```
   Enter image name:
   ```

   In response to this prompt, enter the name of the kernel.

3. To boot an alternate HSC disk to single-user mode, use this format:

   ```
   >>> d r1 SBI#TR#
   >>> d r2 HSC#
   >>> d r3 unit#
   >>> @cira.com
   ```

   The *SBI#* variable represents the Synchronous Backplane Interconnect I/O adapter number (either 0 or 1). The *TR#* variable represents the TR level number of the CI adapter. The *HSC#* variable represents the remote CI port number assigned to the specific HSC controller. The *unit#* variable represents the device number of the system disk drive. The @cira.com string invokes the HSC boot command procedure file.

Note

> The SBI number, the TR number, the HSC number, and the unit number must be expressed in hexadecimal.

The console subsystem reads the cira.com file, boots the alternate HSC disk, and displays the prompt:

```
Enter image name:
```

In response to this prompt, enter the name of the kernel.

3.11 Building and Updating Boot Command Files

This section describes how you can build or update processor-specific boot command files. You will have to build or update the processor-specific boot command files when you want to change the boot default system disk permanently. Some of the processors require you to build new boot command files, while others require you to update existing boot command files.

- The processors that require you to build new command files are the VAX 11/750, VAX-11/780, VAX-11/785, VAX 8600, and VAX 8650 processors.
- The processors that require you to update the existing boot command files are the VAX 6210, VAX 6220, VAX 8200, VAX 8500, VAX 8540, VAX 8550, VAX 8700, VAX 8800, and VAX 8810 processors.

The following sections contain the procedures for either building or updating a processor-specific, bootable console medium. The medium contains the necessary hardware support files and the command procedure files for booting the ULTRIX operating system.

Note

> In some cases, the procedures require you to use the file editor, EDT. While some EDT commands are provided, you should have the appropriate Console Operator's Guide available for further EDT reference information.

3.11.1 Building a VAX 11-750 Console Cassette

In general, you do not need to build a new console cassette for a VAX 11/750. However, if your hardware supports an HSC configuration, you must build a new cassette to enable the HSC remote boot commands.

To build the cassette, follow these steps:

1. Invoke the mkconsole program by typing:

   ```
   # /etc/mkconsole
   ```

   The program assumes that /vmunix is the running kernel. When you run the program, mkconsole prompts you to remove any cassette from the drive and insert a blank cassette.

2. Replace the original console cassette with a blank cassette, and press the RETURN key. The program responds with a brief message to explain its activity. At completion, the system prompt appears.

After building the new cassette, your system is able to boot a remote HSC device.

3.11.2 Building a VAX-11/780 or a VAX-11/785 Console Diskette

The procedure in this section describes how to build a boot command file that boots the current running system disk. The procedure assumes that the /usr file system is mounted.

During the creation of a bootable diskette, you may have to edit the boot command file to set either the memory starting addresses or to set interleaving if you have multiple memory controllers. Therefore, a brief description of the contents of the boot command file is also included in this section.

For either the VAX-11/780 or VAX-11/785 processors, there are several boot command files that you can use to start your system. The format of these command files is the same. Each one contains setup and initialization commands, several DEPOSIT statements, and several startup statements.

The DEPOSIT statements set the R0 through R5 General Purpose Registers (GPRs). These GPRs are evaluated by the Virtual Memory Bootstrap program VMB.EXE, to determine which device is to be booted. All DEPOSIT statements require hexadecimal values.

The GPRs and their meanings are:

- R0 – Boot device type code
- R1 – Processor-specific adapter information
- R2 – Controller number
- R3 – Unit number of the boot device
- R4 – Logical boot block number
- R5 – Boot control flags

Appendix B contains a complete listing of the values for each of these registers.

Example 3-1 shows a sample boot command file (defboo.cmd) for a VAX-11/780 processor.

Example 3-1: Sample VAX-11/780 Boot Command File

```
! RA BOOT COMMAND FILE - UNIBUS RA DISK
!
! THE UNIBUS ADAPTER TR LEVEL MUST BE DEPOSITED IN R1 AND THE
! UNIT NUMBER MUST BE DEPOSITED IN R3 BEFORE EXECUTING THIS PROCEDURE
!
HALT                        ! HALT PROCESSOR
UNJAM                       ! UNJAM SBI
INIT                        ! INIT PROCESSOR
DEPOSIT/I 11 20003800       ! SET UP SCBB
DEPOSIT R0 11               ! UDA-MSCP DISK
DEPOSIT R1 3                ! TR LEVEL OF UNIBUS
DEPOSIT R2 3F468            ! CSR ADDRESS OFFSET = 3F468
DEPOSIT R3 0                ! PLUG # OF SYSTEM DISK
DEPOSIT R4 0                ! BOOT BLOCK LBN (UNUSED)
DEPOSIT R5 10008            ! BOOT ULTRIX TO MULTI USER
DEPOSIT FP 0                ! SET NO MACHINE CHECK EXPECTED
START 20003000              ! START ROM PROGRAM
WAIT DONE                   ! WAIT FOR COMPLETION
                            !
EXAMINE SP                  ! SHOW ADDRESS OF WORKING MEMORY 0x200
LOAD VMB.EXE/START:@        ! LOAD PRIMARY BOOTSTRAP
START @                     ! AND START IT
```

There are several steps that you must follow to build an ULTRIX console diskette for the VAX-11/780 or VAX-11/785 processors.

1. Insert the RX01 console diskette into the diskette drive. This is the diskette that you use to initialize your hardware when you power up the system.

2. Run the mkconsole command:

   ```
   # /etc/mkconsole
   ```

   This command assumes that /vmunix is the running kernel. When you run the mkconsole command, it will display a number of prompts and messages.

   During this step, the mkconsole command instructs you to insert a blank diskette. Replace the RX01 console diskette in the drive with a blank diskette. When the new diskette is created, you can leave it in the diskette drive.

3. If you have multiple memory controllers, you may have to edit the defboo.cmd, askboo.cmd, and restar.cmd command files to change the

memory starting addresses or interleaving settings. Check with your field service representative to get the correct starting addresses or settings for your system.

Before you can edit any of these files, you must extract them from the console diskette by using the arff command. After you modify the command files, replace them on the console diskette using the arff command before proceeding.

The b or b ask boot command options are now available for your use, as described in Section 3.5.

3.11.3 Updating VAX 6210 or VAX 6220 Boot Command Files

On a VAX 6210 or VAX 6220 processor, the processor stores its boot data in EEPROM (Electrically Erasable Programmable Read-Only Memory). The EEPROMs contain such data as the default boot device information.

Note

> You can use the /etc/mkconsole program to get precise instructions for updating your console boot defaults.

To make changes to the EEPROM data, follow these steps:

1. Shutdown your system and halt the processor.
2. Reset the system by typing the initialize command at the console prompt. For example, type:

```
>>> initialize
```

3. Set the processor's selector switch to the "Update" position.
4. Enter the following commands to set the default multi-user boot command for a local disk and an HSC disk.

- For a local disk, use this syntax:

```
>>> set boot default /xmi:BIA# /bi:BI# /r5:10008 duunit#
```

For example, type:

```
>>> set boot default /xmi:e /bi:4 /r5:10008 du0
```

- For an HSC disk, use this syntax:

```
>>> set boot default /xmi:BIA# /bi:BI# /node:HSC# /r5:10008 duunit#
```

For example, type:

```
>>> set boot default /xmi:e /bi:4 /node:HSC# /r5:10008 du0
```

The variable numbers must be expressed in hexadecimal notation.

5. Enter the following commands to set the default single-user boot command for a local disk and an HSC disk. This allows a conversational boot to single user, using the `b ask` command.

- For a local disk, use this syntax:

```
>>> set boot ask /xmi:BIA# /bi:BI# /r5:1000b duunit#
```

For example, type:

```
>>> set boot ask /xmi:e /bi:4 /r5:1000b du0
```

- For an HSC disk, use this syntax:

```
>>> set boot ask /xmi:BIA# /bi:BI# /node:HSC# /r5:1000b duunit#
```

For example, type:

```
>>> set boot ask /xmi:e /bi:4 /node:HSC# /r5:1000b du0
```

The variable numbers must be expressed in hexadecimal notation.

6. Reset the selector switch from the "Update" setting to its original setting.

7. If you are booting a CI disk, make sure that the TK50 console tape is in the drive.

8. Boot the system to multiuser or single user mode:

- Boot to multiuser mode by typing:

```
>>> b
```

- Boot to single-user mode by typing:

```
>>> b ask
```

3.11.4 Changing the VAX 8200, VAX 8250, VAX 8300 and the VAX 8350 Boot Data

Each of these processors stores its boot data in two EEPROMs (Electrically Erasable Programmable Read-Only Memory). The EEPROMs contain such data as the default console baud rate and the default boot device. To make changes to this data, you must run the EEPROM utility, which is stored on the diskette labeled: UTIL PROG FLP.

The EEPROM utility runs under the VAX Diagnostic Supervisor (VDS) software. Therefore to run the EEPROM utility, you must boot the VDS software. The procedures for running the VDS software, as well as a complete description of the EEPROM utility's functionality, is described in your processor-specific Owner's Manual.

It may be necessary to update the EEPROMs to boot the diskette by default.

If your hardware supports an HSC configuration, you must build a new diskette to enable the HSC remote boot commands.

To build the diskette, follow these steps:

1. Invoke the mkconsole program by typing:

   ```
   # /etc/mkconsole
   ```

   The program assumes that /vmunix is the running kernel. When you run the program, mkconsole prompts you to remove the RX50 diskette from the drive and insert a blank RX50 diskette in the same drive.

2. Replace the RX50 diskette with a blank write-enabled RX50 diskette and press the RETURN key. The program responds with a brief message to explain its activity. At completion, the system prompt appears.

After building the new diskette, your system is able to boot a remote HSC device.

3.11.5 Updating the VAX 8500, VAX 8530, VAX 8550, VAX 8700, VAX 8800, and VAX 8810 Boot Command Files

For each of these processors, there are several boot command files that you can use to start your system. The format of these command files is the same. Each command file contains setup and initialization commands, several DEPOSIT statements, and several startup statements.

The DEPOSIT statements set the R0 through R5 General Purpose Registers (GPRs). These GPRs are evaluated by the Virtual Memory Bootstrap program VMB.EXE, to determine which device is to be booted.

All of the DEPOSIT statements require hexadecimal values.

The GPRs and their meanings are:

- R0 – Boot device type code
- R1 – Processor-specific adapter information
- R2 – Controller number
- R3 – Unit number of the boot device
- R4 – Logical boot block number
- R5 – Boot control flags

Appendix B contains a complete listing of the values for each of these registers. While any of the register entries in these files can be changed, the R1, R3, and R5 registers are the ones most likely to change.

Example 3–2 shows a sample boot command file (bdara.com) for a VAX 8700 processor.

Example 3-2: Sample VAX8700 Boot Command File

```
SET VERIFY
! BDARA.COM
! REV 1.0
!
! COMMAND PROCEDURE TO BOOT ULTRIX FROM A BDA DISK.
!
! NEXT_PRIMARY is expected to point to the CPU that is to be used
! as the primary CPU.
!
! The following register deposits must be done before executing this
! command procedure or must be edited to correspond to the hardware
! configuration:
!
! R1 - Bus address information
! R3 - device unit number
!
SET TERMINAL OPA0            ! Set up logging
SET DEFAULT HEXADECIMAL,PHYSICAL,LONGWORD
                        .
                        .
                        .
INITIALIZE                   ! Init primary
DEPOSIT R0 21                ! BDA boot device type code
!DEPOSIT R1 00               ! Boot device bus address:
                             ! <3:0>=BI node #, <5:4>=BI #
DEPOSIT R2 0                 ! <31:24>=optional controller letter specifier
!DEPOSIT R3 %D0              ! Unit # of drive, decimal radix
SET DEFAULT HEXADECIMAL      ! Reset radix
DEPOSIT R4 0                 ! Not applicable
DEPOSIT R5 1000B             ! bits     purpose
                             ! <0>   ask for boot image name.
                             ! <1>   boot single user
                             ! <3>   boot ultrix
                             ! <16>  ignore memory soft errors.
FIND/MEM                     ! Find 64kb of working memory; set cold
                             ! start bit
IF NOT $STATUS THEN @EXIT    ! Boot if find was successful
EXAMINE SP                   ! Show address of working memory + %X200
LOAD/MAINMEMORY/START=@ VMB.EXE ! Load VMB into good mem + %X200
START @                         ! Start executing VMB
```

The steps to update the boot command files for these processors are:

1. Exit the console mode. To do this, type a CTRL/P at the superuser prompt and type the word **exit** at the console mode prompt:

```
# <CTRL/P>
>>>exit
$
```

The $ prompt signifies that you are out of the console mode and under control of the operating system running on the PRO-380.

2. Make a copy the bdara.com file with the name defboo.com. For the VAX 8800 processor, specify the subdirectory [8800], which contains the bdara.com file. For the other VAX processors, copy the bdara.com file in the system default subdirectory:

```
$ COPY [8800]bdara.com defboo.com
           or
$ COPY bdara.com defboo.com
```

Note

Do not edit the bdara.com file. This file will be required for future ULTRIX installations, or may be needed to boot alternate system disks.

3. Edit the defboo.com file. You must use the EDT editor as described in the appropriate Console Operator's Guide. This editor is invoked with the RUN EDT command, followed by the file name that you want to edit:

```
$RUN EDT
EDT>defboo.com
```

The entries that you may have to change are the R1, R3, and R5 register entries. At a minimum, you must remove the ! signs from the beginning of the R1 and R3 lines.

The R1 register entry specifies - from the left-most bits - the following:

- Bits 0 to 3 specify the number of the BI adapter node which is connected to the BUA
- Bits 4 and 5 specify the NBIA adapter number
- Bits 6 through 31 of the R1 register must be zero.

The R3 register specifies the unit (plug) number of the system disk drive.

The R5 register entry should read 10008, which specifies booting the ULTRIX operating system to multiuser mode.

1. Exit the defboo.com file (after making the appropriate changes) and return to the $ prompt.

2. Make a copy of the defboo.com file, and name the copy askboo.com. For example, type:

   ```
   $COPY defboo.com askboo.com
   $
   ```

   The system uses the askboo.com file to boot the system in conversational mode:

3. Edit the askboo.com file, using the EDT editor:

   ```
   $RUN EDT
   EDT>askboo.com
   ```

   The only register that you change is the R5 register. The R5 register entry should read 1000B. This causes the VMB.EXE program to boot the ULTRIX operating system to conversational single-user mode as described in Chapter 2.

4. Exit the askboo.com file (after making the appropriate changes) and return to the $ prompt.

5. Return to the console monitor prompt by running the control program:

   ```
   $RUN CONTROL
   ```

   This command causes the system to redisplay the console monitor prompt >>>.

6. Return the console to the ULTRIX superuser prompt:

   ```
   >>>set term prog
   #
   ```

   The # prompt indicates that you have returned to the ULTRIX operating system and can continue normal operations. You can now use the b and b ask boot commands to boot the system, as described in Section 3.8.

3.11.6 Updating the VAX 8820 Boot Command Files

There are boot command files for the VAX 8820 processor that you use to start your system. The format of these command files is the same. Each command file contains setup and initialization commands, several DEPOSIT statements, and several startup statements.

The DEPOSIT statements set the R0 through R5 General Purpose Registers (GPRs). These GPRs are evaluated by the Virtual Memory Bootstrap program VMB.EXE, to determine which device is to be booted. All of the DEPOSIT statements require hexadecimal values.

The GPRs and their meanings are:

- R0 – Boot device type code
- R1 – Processor-specific adapter information
- R2 – Controller number
- R3 – Unit number of the boot device
- R4 – Logical boot block number
- R5 – Boot control flags

Appendix B contains a complete listing of the values for each of these registers. While any of the register entries in these files can be changed, the R1, R3, and R5 registers are the ones most likely to change.

To update the boot command files, follow these steps:

1. Exit the console mode. To do this, type a CTRL/P at the superuser prompt:

   ```
   # <CTRL/P>
   >>>
   ```

   The >>> prompt signifies that you are running under control of the console operating system.

2. Make a copy the bdara.cmd file with the name defboo.cmd. Copy the bdara.cmd file in the system default subdirectory:

   ```
   >>> COPY bdara.cmd defboo.cmd
   ```

3. Edit the defboo.cmd file. You must use the EDT editor as described in the appropriate Console Operator's Guide. This editor is invoked with the edit/edt command, followed by the file name that you want to edit:

   ```
   >>> dit/edt defboo.cmd
   ```

   The entries that you may have to change are the R1, R3, and R5 register entries. At a minimum, you must remove the ! signs from the beginning of the R1 and R3 lines.

   The R1 register entry specifies - from the left-most bits - the following:

- Bits 0 to 3 specify the number of the BI adapter node which is connected to the BUA
- Bits 4 and 5 specify the NBIA adapter number
- Bits 6 through 31 of the R1 register must be zero.

The R3 register specifies the unit (plug) number of the system disk drive.

The R5 register entry should read 10008, which specifies booting the

ULTRIX operating system to multiuser mode.

1. Exit the defboo.cmd file (after making the appropriate changes) and return to the >>> prompt.

2. Make a copy of the defboo.cmd file, and name the copy askboo.cmd. For example, type:

   ```
   >>> COPY defboo.cmd askboo.cmd
   >>>
   ```

   The system uses the askboo.cmd file to boot the system in conversational mode.

3. Edit the askboo.cmd file, using the EDT editor:

   ```
   >>> edit/edt askboo.cmd
   ```

   The only register that you change is the R5 register. The R5 register entry should read 1000B. This causes the VMB.EXE program to boot the ULTRIX operating system to conversational single-user mode as described in Chapter 2.

4. Exit the askboo.cmd file (after making the appropriate changes) and return to the >>> prompt.

5. Return the console to the ULTRIX superuser prompt:

   ```
   >>>set term prog
   #
   ```

   The # prompt indicates that you have returned to the ULTRIX operating system and can continue normal operations. You can now use the b and b ask boot commands to boot the system, as described in Section 3.9.

3.11.7 Updating the VAX 8600 and VAX 8650 Console RL02 Disk

The procedure in this section describes how to create boot command files that will boot the current running system disk. This procedure assumes that the /usr file system is mounted.

To update the VAX 8600 or the VAX 8650 console RL02 disks, run the command. This command assumes that /vmunix is the running kernel. Type:

```
# /etc/mkconsole
```

The mkconsole command writes the ULTRIX support files, which include the defboo.com and askboo.com files, to the console RL02 disk.

You can now use the b and b ask commands to boot the system, as described in Section 3.10.

The Standalone ULTRIX Environment 4

The Standalone ULTRIX Environment is a diskless environment that has its miniroot file system within the data space of the running kernel. It is used to initiate ULTRIX installations.

The primary purpose of the Standalone ULTRIX Environment is to support the initial phases of an installation, which include selecting input and output devices, as well as restoring the root file system image to the target system disk. Throughout the installation process, full ULTRIX device drivers are used.

A secondary purpose of the Standalone ULTRIX Environment is to support system management activities. These activities include:

- Restoring a damaged root file system
- Checking the consistency of the root file system
- Restoring the boot block image
- Performing disk maintenance operations

The commands included in the Standalone ULTRIX Environment are those commands that will assist you in recovering from root file system corruption, and those that will help you perform general file system and disk maintenance tasks. You should therefore consider the Standalone ULTRIX Environment a limited and intentionally small environment that does not perform like a full ULTRIX operating system environment. System management activities in the Standalone ULTRIX Environment should be performed by those individuals who have extensive ULTRIX or UNIX operating systems experience.

The sections in this chapter:

- Explain how to invoke the Standalone ULTRIX Environment
- Identify some of the more commonly used functional capabilities
- Describe how to extend the Standalone ULTRIX Environment so that additional commands can be used.

4.1 Invoking the Standalone ULTRIX Environment

The media and the commands that you use to invoke the Standalone ULTRIX Environment are dependent on the type of processor that you are using. These media and commands are identified and described in your Basic Installation Guide.

As part of the installation, the system prompts you to select one of three options:

1. BASIC Installation
2. ADVANCED Installation
3. System Management

Choose the third item, System Management, to invoke the Standalone ULTRIX Environment. The system responds by placing the system in single-user mode and by displaying the # shell prompt.

4.2 Standalone ULTRIX Environment Capabilities

The Standalone ULTRIX Environment enables you to perform all of the typical system management activities. The only difference is that in some cases, you have to use system primitives instead of the more advanced system commands. For example, to make a new file system, you must use the mkfs command instead of the newfs command. This is because of the space limitation imposed on the Standalone ULTRIX Environment.

A limitation of the Standalone ULTRIX Environment is that only peripheral devices connected to controllers that have been assigned standard, fixed, CSR addresses are accessible when making special device files. At boot time, the system does not configure controllers assigned floating CSR addresses. Once the special device files have been created with the MAKEDEV command, you will have access to the functional capabilities of the Standalone ULTRIX Environment. These functional capabilities include:

- Repair corrupted file systems with the fsck command
- Create new file systems with the mkfs command
- Restore the boot block with the dd command
- Restore file systems with the restore command
- Maintain disks with the radisk command
- Mount other disks and file systems with the mount command

An example of the Standalone ULTRIX Environment's functional capability is described in the Guide to System Backup and Restore. The description explains how to restore the root file system after a catastrophic event has occurred.

4.3 Extending the Standalone ULTRIX Environment

If you find that the commands and utilities provided by the Standalone ULTRIX Environment do not completely meet your needs, you can extend the environment to include access to other commands. To extend the Standalone ULTRIX Environment, perform the following steps:

1. Make the device special files for the device that contains the target commands. For example, if you want to have access to commands that are on an ra0 device, partition g you would type:

   ```
   #cd /dev
   #MAKEDEV ra0g
   ```

2. Mount the device. For example, to mount the /mnt file system on the ra0g device, you would type:

   ```
   #mount /dev/ra0g /mnt
   ```

   This will enable you to access any of the commands or files on that device. To see what commands and files are available, type:

   ```
   #ls /mnt
   ```

   The system will respond by displaying the contents of /mnt.

Device Mnemonics A

This appendix identifies and defines the mnemonics that are used to attach any hardware or software device to your system. The mnemonics are used by the /dev/MAKEDEV shell script to create the character or block special files that represent each of the devices. The mnemonics also appear in the system configuration file as described in the Guide to System Configuration File Maintenance.

Table A–1 lists the mnemonics in seven categories: generic, consoles, disks, tapes, terminals, modems, and printers. The generic category lists the mnemonics of a general nature and includes memory, null, trace, and tty devices. The consoles category lists the system console devices that the ULTRIX operating system uses. The disks, tapes, terminals, modems, and printers categories identify the appropriate mnemonics for those devices.

The description heading in Table A–1 identifies the corresponding device name. It does not define the mnemonic's use. For detailed information on the use of each mnemonic in relation to both the MAKEDEV script and the system configuration file, refer to the reference pages in Section 4 of the ULTRIX Reference Pages. If on-line reference pages are available, you can also use the man command. For instance, if you enter at the system prompt:

```
     # man ra
```

the system displays the reference page for the Mass Storage Control Protocol (MSCP) disk controller driver. Where appropriate, the SYNTAX section of the reference page defines the device's syntax as it appears, or should appear, in the config file. Refer to /dev/MAKEDEV for additional software device mnemonics that MAKEDEV uses. Refer to MAKEDEV(8) in the ULTRIX Reference Pages for a description of the MAKEDEV utility.

You should note that Table A–1 uses the convention of an asterisk (*) beside a mnemonic and a question mark (?) beside a device name to mean a variable number. The range of the variable number is dependent on the particular device.

Table A-1: Devices Supported by MAKEDEV

Category	Mnemonic	Description
Generic	boot*	Boot and std devices by cpu number; e.g., boot750
	mvax*	All MicroVAX setups; e.g., mvax2000
	vaxstation*	A VAXstation 2000 setup; e.g., vaxstation2000
	std	Standard devices below with all console subsystems:
	drum	Kernel drum device
	errlog	Error log device
	kUmem	Kernel Unibus/Q-bus virtual memory
	kmem	Virtual main memory
	mem	Physical memory
	null	A null device
	trace	A trace device
	tty	A tty device
	local	Customer specific devices
Consoles	console	System console interface
	crl	Console RL02 disk interface for VAX 86?0
	cs*	Console RX50 floppy interface for VAX 8??0
	ctu*	Console TU58 cassette interface for VAX 11/750
	cty*	Console extra serial line units for VAX 8??0
	cfl	Console RX01 floppy interface for 11/78?
	ttycp	Console line used as auxiliary terminal port
Disks	hp*	MASSBUS disk interface for RM?? drives
	ra*	UNIBUS/Q-bus/BI/HSC MSCP disk controller interface
	ese*	UNIBUS/Q-bus/BI/HSC MSCP electronic ESE20 disk
	rb*	UNIBUS IDC RL02 disk controller interface for RB?? drives
	rd*	VAXstation 2000 and MicroVAX 2000 RD type drives
	rz	SCSI disks (RZ22/RZ23/RZ55/RRD40)
	rk*	UNIBUS RK?? disk controller interface
	rl*	UNIBUS/Q-bus RL?? disk controller interface
	rx*	VAXstation 2000 and MicroVAX 2000 RX type drives
Tapes	mu*	TU78 MASSBUS magtape interface
	tms*	UNIBUS/Q-bus/BI/HSC TMSCP tape controller interface
	rv*	UNIBUS/Q-bus/BI/HSC TMSCP optical disk
	ts*	UNIBUS/Q-bus TS11/TS05/TU80 magtape interface
	tu*	TE16/TU45/TU77 MASSBUS magtape interface
	st*	VAXstation 2000 and MicroVAX 2000 TZK50 cartridge tape

Category	Mnemonic	Description
	tz*	SCSI tapes (TZ30/TZK50)
Terminals	cxa*	Q-bus cxa16
	cxb*	Q-bus cxb16
	cxy*	Q-bus cxt08
	dfa*	Q-bus DFA01 comm multiplexer
	dhq*	Q-bus DHQ11 comm multiplexer
	dhu*	UNIBUS DHU11 comm multiplexer
	dhv*	Q-bus DHV11 comm multiplexer
	dmb*	BI DMB32 comm multiplexer including dmbsp serial printer/plotter
	dhb*	BI DHB32 comm multiplexer
	dmf*	UNIBUS DMF32 comm multiplexer including dmfsp serial printer/plotter
	dmz*	UNIBUS DMZ32 comm multiplexer
	dz	UNIBUS DZ11 and DZ32 comm multiplexer
	sh*	MicroVAX 2000, 8 serial line expansion option
	ss*	VAXstation 2000 and MicroVAX 2000 basic 4 serial line unit
	dzq*	Q-bus DZQ11 comm multiplexer
	dzv*	Q-bus DZV11 comm multiplexer
	lta*	Sets of 16 network local area terminals (LAT)
	pty*	Sets of 16 network pseudoterminals
	qd*	Q-bus VCB02 (QDSS) graphics controller/console
	qv*	Q-bus VCB01 (QVSS) graphics controller/console
	sm*	VAXstation 2000 monochrome bitmap graphics/console
	sg*	VAXstation 2000 color bitmap graphics console
Modems	dfa*	DFA01 integral modem communications device.
Printers	dmbsp*	BI DMB32 serial printer/plotter
	dmfsp*	UNIBUS DMF32 serial printer/plotter
	lp*	UNIBUS LP11 parallel line printer
	lpv*	Q-bus LP11 parallel line printer

General Purpose Register Use by VMB.EXE B

The ULTRIX operating system uses I/O device drivers provided in the VMS Virtual Memory Bootstrap (VMB) program. The VMB program evaluates the contents of general purpose registers (GPRs) R0 through R5 to determine which device is to be booted. Where appropriate, installation procedures are set up to build default boot command files to bootstrap the system disk. If you wish to tailor the contents of boot command files, you can edit and replace them as necessary. This appendix is provided as a reference to show the use of the GPRs by the VMB program.

The following list defines the possible contents of the R0 through R5 registers. Values enclosed in < > signs define the bit positions for a particular parameter. For example: <07:00> means from bits 0 to 7. The notation MBZ means that the value must be zero.

Input Parameters : (Registers expect hex values)

R0:

- <07:00> boot device type code (RPB$B_DEVTYP)

Hex Value -------	Device ------
0	MASSBUS device (RM02/3,RP04/5/6/7,RM80)
1	RK06/7
2	RL01/2
3	IDC(almost an RA80) on 11/730
11	UDA-50 (note: values 1 - 1F are reserved for UNIBUS devices)
20	HSC on CI
21	BDA on BI
40	Console block storage device

- <15:08> reserved for future expansion

- <31:16> device class dependent (RPB$W_R0UBVEC)

UNIBUS - optional vector address; 0 implies use the default vector

MASSBUS - not used

R1: Boot device's bus address

11/780 &
11/730 - <31:04> MBZ
<03:00> TR number of adapter

11/750 - <31:24> MBZ
<23:00> address of the I/O page for the boot device's adapter

8600 - <31:06> MBZ
<05:04> A-bus Adapter number
<03:00> TR number of the adapter

8800 - <31:06> MBZ
<05:04> NBIA Adapter number
<03:00> BI node number of the adapter

R2: All controllers:

<31:24> controller letter designator (optional)

- UNIBUS:

<23:18> MBZ
<17:00> UNIBUS address of the device's CSR

- MASSBUS:

<23:04> MBZ
<03:00> adapter's controller/formatter number

- CI:
<23:08> MBZ
<07:00> HSC node number (station address)

R3: Boot device unit number

R4: <31:0> MBZ

R5: Software boot control flags. The value -1 is reserved. The following table defines the software boot control flags used by the ULTRIX operating system. The first column of the table contains a comment about the ULTRIX operating system's use of that control flag. If this column is blank, the flag is not required by the ULTRIX operating system. The second column defines the bit number of the register. The third column defines the control flag.

Comment	Bit	Meaning
OPTIONAL (RB_ASKNAME)	0	RPB$V_CONV Conversational boot. This bit will force ULTRIXBOOT to prompt the user for an image name which would presumably be different from the default vmunix. If the DIAG is also on, then the user is prompted for the diagnostic supervisor image name.
OPTIONAL (RB_SINGLE)	1	RPB$V_DEBUG If this flag is set, the ULTRIX kernel image will be booted to single-user mode.
	2	RPB$V_INIBPT. Initial breakpoint. If RPB$V_DEBUG is set, VMS executes a BPT instruction immediately after enabling mapping.
REQUIRED	3	RPB$V_BBLOCK. Secondary boot from boot block. Secondary bootstrap is a single 512-byte block, whose LBN is specified in R4. R4 must be 0 for ULTRIX.
OPTIONAL	4	RPB$V_DIAG (RB_LOADDS for ULTRIX) Diagnostic boot. Causes ULTRIXBOOT to load the appropriate diagnostic supervisor by CPU type. The default path is /field/e?saa.exe, where the partition is specified in bits <31:28> of this register.
	5	RPB$V_BOOBPT. Bootstrap breakpoint. Stops the primary and secondary bootstraps with a breakpoint instruction before testing memory.

Comment	Bit	Meaning
	6	RPB$V_HEADER. Image header. Takes the transfer address of the secondary bootstrap image from that file's image header. If RPB$V_HEADER is not set, transfers control to the first byte of the secondary boot file.
	7	RPB$V_NOTEST. Memory test inhibit. Sets a bit in the PFN bit map for each page of memory present. Does not test the memory.
	8	RPB$V_SOLICT. File name. VM@ prompts for the name of a secondary bootstrap file.
	9	RPB$V_HALT. Halt before transfer. Executes a HALT instruction before transferring control to the secondary bootstrap.
	10	RPB$V_NOPFND. No PFN deletion (not implemented; intended to tell VM@ not to read a file from the boot device that identifies bad or reserved memory pages, so that VM@ does not mark these pages as valid in the PFN bitmap).
	11	RPB$V_MPM. Specifies that multiport memory is to be used for the total executive memory requirement. No local memory is to be used. This is for tightly coupled multiprocessing.
	12	RPB$V_USEMPM. Specifies that multiport memory should be used in addition to local memory, as though both were one single pool of pages.

Comment	Bit	Meaning
	13	RPB$V_MEMTEST Specifies that a more extensive algorithm be used when testing main memory for hardware uncorrectable (RDS) errors.
	14	RPB$V_FINDMEM Requests use of MA780 memory if MS780 is insufficient for booting. Used for 11/782 installations.
	15	RPB$V_AUTOTEST Used by Diagnostic Supervisor.
REQUIRED	16	RPB$V_CRDTEST Specifies that memory pages with correctable (CRD) errors NOT be discarded at bootstrap time. By default, pages with CRD errors are removed from use during the bootstrap memory test.
	<27:17>	MBZ – Reserved for future expansion.
OPTIONAL (DIAG BOOT)	<31:28>	RPB$V_TOPSYS Redefines the default load file system partition. This field is used primarily with DIAG. The corresponding partition numbers and letters are: 0 = a 1 = b 2 = c 3 = d 4 = e 5 = f 6 = g 7 = h
SP	–	Must be set to 0x200

Index

HOW TO ORDER ADDITIONAL DOCUMENTATION

DIRECT TELEPHONE ORDERS

In Continental USA and New Hampshire, Alaska or Hawaii call **800-DIGITAL**	In Canada call **800-267-6215**

DIRECT MAIL ORDERS (U.S. and Puerto Rico*)

DIGITAL EQUIPMENT CORPORATION
P.O. Box CS2008
Nashua, New Hampshire 03061

DIRECT MAIL ORDERS (Canada)

DIGITAL EQUIPMENT OF CANADA LTD.
100 Herzberg Road
Kanata, Ontario K2K 2A6
Attn: Direct Order Desk

INTERNATIONAL

DIGITAL EQUIPMENT CORPORATION
PSG Business Manager
c/o Digital's local subsidiary
or approved distributor

Internal orders should be placed through the Software Distribution Center (SDC), Digital Equipment Corporation, Westminster, Massachusetts 01473

*Any prepaid order from Puerto Rico must be placed
with the Local Digital Subsidiary:
809–754–7575

ULTRIX - 32
Guide to System Shutdown
and Startup
AA-ME91A-TE

Reader's Comments

Note: This form is for document comments only. DIGITAL will use comments submitted on this form at the company's discretion. If you require a written reply and are eligible to receive one under Software Performance Report (SPR) service, submit your comments on an SPR form.

Did you find this manual understandable, usable, and well-organized? Please make suggestions for improvement. ______________________

____

____

____

Did you find errors in this manual? If so, specify the error and the page number.

____

____

____

Please indicate the type of user/reader that you most nearly represent.

- □ Assembly language programmer
- □ Higher-level language programmer
- □ Occasional programmer (experienced)
- □ User with little programming experience
- □ Student programmer
- □ Other (please specify) ______________________

Name ______________________ Date ______________

Organization ______________________________________

Street ____

City ______________________ State ________ Zip Code or Country ____________

------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
Documentation Manager
ULTRIX Documentation Group
ZKO3-3/X18
Spit Brook Road
Nashua, N.H.
03063

------Do Not Tear - Fold Here and Tape --

Cut Along Dotted Line

ULTRIX - 32
Guide to System Shutdown
and Startup
AA-ME91A-TE

Reader's Comments

Note: This form is for document comments only. DIGITAL will use comments submitted on this form at the company's discretion. If you require a written reply and are eligible to receive one under Software Performance Report (SPR) service, submit your comments on an SPR form.

Did you find this manual understandable, usable, and well-organized? Please make suggestions for improvement. ____________________

____

____

____

Did you find errors in this manual? If so, specify the error and the page number.

____

____

____

Please indicate the type of user/reader that you most nearly represent.

□ Assembly language programmer
□ Higher-level language programmer
□ Occasional programmer (experienced)
□ User with little programming experience
□ Student programmer
□ Other (please specify) ____________________

Name ____________________ Date ____________________

Organization ____________________

Street ____________________

City ____________________ State ________ Zip Code or Country ____________

------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
Documentation Manager
ULTRIX Documentation Group
ZKO3-3/X18
Spit Brook Road
Nashua, N.H.
03063

------Do Not Tear - Fold Here and Tape ------------------------------------

Cut Along Dotted Line