digital

VAX-11
Software Installation Guide

Order No. AA-D021B-TE

VAX11

February 1979

This document contains detailed instructions for installing the VAX/VMS operating system and the optional software components.

VAX-11
Software Installation Guide

Order No. AA-D021B-TE

SUPERSESSION/UPDATE INFORMATION:	This revised document supersedes the VAX-11 Software Installation Guide (Order No. AA-D021A-TE)
OPERATING SYSTEM AND VERSION:	VAX/VMS V1.5
SOFTWARE VERSION:	VAX/VMS V1.5

To order additional copies of this document, contact the Software Distribution Center, Digital Equipment Corporation, Maynard, Massachusetts 01754

digital equipment corporation · maynard, massachusetts

First Printing, August 1978
Revised, February 1979

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 only be used or copied 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.

Copyright © 1978, 1979 by Digital Equipment Corporation

The postage-prepaid READER'S COMMENTS form on the last page of this document requests the user's critical evaluation to assist us in preparing future documentation.

The following are trademarks of Digital Equipment Corporation:

DIGITAL	DECsystem-10	MASSBUS
DEC	DECtape	OMNIBUS
PDP	DIBOL	OS/8
DECUS	EDUSYSTEM	PHA
UNIBUS	FLIP CHIP	RSTS
COMPUTER LABS	FOCAL	RSX
COMTEX	INDAC	TYPESET-8
DDT	LAB-8	TYPESET-11
DECCOMM	DECSYSTEM-20	TMS-11
ASSIST-11	RTS-8	ITPS-10
VAX	VMS	SBI
DECnet	IAS	PDT
DATATRIEVE	TRAX	

CONTENTS

			Page
PREFACE			vii
CHAPTER	1	INTRODUCTION	1-1
	1.1	OVERVIEW OF SYSTEM GENERATION AND START UP	1-1
	1.2	SYSGEN UTILITY	1-3
	1.3	SYSTEM GENERATION PARAMETER FILES	1-3
	1.4	SYSTEM INSTALLATION SUMMARY	1-4
CHAPTER	2	PREPARING TO BOOTSTRAP THE SYSTEM	2-1
	2.1	VAX/VMS SOFTWARE DISTRIBUTION KITS	2-1
	2.1.1	Magnetic Tape Kit	2-1
	2.1.2	RK07 Kit	2-2
	2.2	COPYING DISTRIBUTION MEDIA	2-2
	2.2.1	Loading Stand-Alone DSC	2-3
	2.2.2	Copying Tape to Disk	2-4
	2.2.3	Copying Disk to Disk	2-5
	2.2.4	Terminating Stand-Alone DSC	2-5
CHAPTER	3	BOOTSTRAPPING AND INSTALLING A VAX/VMS SYSTEM	3-1
	3.1	ENTERING COMMANDS TO THE CONSOLE PROGRAM	3-2
	3.2	BOOTING THE SYSTEM	3-2
	3.2.1	Conversational Bootstrap	3-4
	3.2.2	Non-Stop Bootstrap	3-6
	3.3	LOGGING INTO THE SYSTEM	3-8
	3.4	MODIFYING THE PARAMETER FILE YOU SELECT	3-8
	3.5	ALTERING PAGING, SWAPPING, AND DUMP FILE SIZES	3-9
	3.6	DEFAULT BOOTSTRAP COMMAND FILE	3-9
	3.6.1	Copying the Default Boot Command File	3-10
	3.6.2	Automatic Restart	3-10
	3.7	REBOOTING THE SYSTEM STOPPING IN SYSBOOT	3-10
	3.8	REBOOTING USING THE DEFAULT BOOTSTRAP	3-11
	3.9	COPYING FILES TO OR FROM THE CONSOLE FLOPPY DISKETTE	3-11
	3.10	BUILDING AND COPYING A VAX/VMS SYSTEM	3-12
	3.10.1	Using VMSKITBLD.COM	3-12
	3.10.2	Using VMSKITCPY.COM	3-13
CHAPTER	4	SYSTEM PARAMETER FILES	4-1
	4.1	MODIFYING SYSTEM PARAMETER FILES	4-1
	4.2	MAJOR PARAMETER DESCRIPTIONS	4-4
	4.2.1	BALSETCNT -- Number of Processes Allowed in the Balance Set	4-4
	4.2.2	GBLPAGES -- Global Pages	4-4
	4.2.3	GBLSECTIONS -- Global Sections	4-5
	4.2.4	IRPCOUNT -- I/O Request Packet Count	4-5
	4.2.5	MAXPROCESSCNT -- Maximum Process Count	4-5
	4.2.6	MPW_HILIM -- Modified Page List High Limit	4-6

CONTENTS

			Page
	4.2.7	MPW_LOLIMIT -- Modified Page List Low Limit	4-6
	4.2.8	MPW_WRTCLUSTER -- Modified Page Write Cluster Size	4-7
	4.2.9	NPAGEDYN -- Number of Bytes of Nonpaged Dynamic Pool	4-7
	4.2.10	PAGEDYN -- Number of Bytes of Paged Dynamic Pool	4-8
	4.2.11	PFCDEFAULT -- Page Fault Cluster Default	4-8
	4.2.12	QUANTUM -- Time Quantum	4-8
	4.2.13	SYSMWCNT -- System Maximum Working Set Count	4-9
	4.2.14	VIRTUALPAGECNT -- Maximum Virtual Page Count for any Process	4-9
	4.2.15	WSMAX -- Maximum Process Working Set Size	4-10
	4.3	DETERMINING THE SIZES OF THE PAGING, SWAPPING, AND SYSTEM DUMP FILES	4-10
	4.3.1	Paging File Size	4-11
	4.3.2	Swapping File Size	4-11
	4.3.3	System Dump File Size	4-11
CHAPTER	5	THE SYSGEN UTILITY	5-1
	5.1	INVOKING AND TERMINATING SYSGEN	5-1
	5.2	SYSGEN COMMAND SUMMARY	5-1
	5.3	SYSGEN COMMAND USAGE	5-2
	5.3.1	Parameter File SYSGEN Commands	5-3
	5.3.2	I/O Device SYSGEN Commands	5-4
	5.3.3	Swapping, Paging, Dump File SYSGEN Command	5-4
	5.4	SYSGEN COMMANDS AVAILABLE IN SYSBOOT	5-4
	5.5	AUTOCONFIGURE	5-5
	5.6	CONNECT	5-7
	5.7	CONTINUE	5-9
	5.8	CREATE	5-10
	5.9	DISABLE CHECKS	5-11
	5.10	ENABLE CHECKS	5-12
	5.11	EXIT	5-13
	5.12	HELP	5-14
	5.13	LOAD	5-16
	5.14	RELOAD	5-17
	5.15	SET (PARAMETER VALUE)	5-18
	5.16	SET (START-UP FILE)	5-19
	5.17	SHOW	5-20
	5.18	USE	5-23
	5.19	WRITE	5-24
CHAPTER	6	UPDATING THE SYSTEM AND INSTALLING OPTIONAL SOFTWARE	6-1
	6.1	GENERAL INSTALLATION PROCEDURES	6-1
	6.1.1	Distribution Kits	6-1
	6.1.2	Preparing for Installation	6-2
	6.1.3	Duration of an Installation Procedure	6-3
	6.1.4	Installation Completion	6-3
	6.2	UPDATING THE SYSTEM	6-4
	6.3	VAX-11 FORTRAN IV-PLUS	6-6
	6.4	FORTRAN IV/VAX TO RSX CROSS COMPILER	6-7
	6.4.1	Code Generation Options	6-7
	6.4.1.1	Threaded Code	6-7
	6.4.1.2	Inline Code	6-7

CONTENTS

			Page
	6.4.1.3	Code Selection Criteria	6-8
	6.4.1.4	Selecting a Code Option	6-8
	6.4.2	Speed vs. Size Optimization	6-8
	6.4.3	OTS Options	6-9
	6.4.3.1	OTS Library Option	6-9
	6.4.3.2	Hardware Configuration Options	6-10
	6.4.3.3	VIRTUAL Array Option	6-10
	6.4.4	Retaining Previous Versions of FORTRAN IV	6-10
	6.4.5	Verifying Installation of FORTRAN IV/VAX	6-11
	6.5	DECnet-VAX	6-12
	6.6	VAX-11 COBOL-74	6-13
	6.6.1	Verifying Installation of VAX-11 COBOL-74	6-14
	6.7	PDP-11 BASIC-PLUS-2/VAX	6-15
	6.8	PDP-11 DATATRIEVE/VAX	6-17
	6.8.1	Installing PDP-11 DATATRIEVE/VAX	6-17
	6.8.2	Patching PDP-11 DATATRIEVE/VAX	6-18
	6.9	VAX-11 BLISS-32	6-19
APPENDIX	A	SYSTEM GENERATION PARAMETERS	A-1
	A.1	PARAMETER DESCRIPTIONS	A-1
	A.2	SPECIAL PARAMETERS	A-14
APPENDIX	B	MESSAGES	B-1
	B.1	BOOT MESSAGES	B-1
	B.2	SYSBOOT MESSAGES	B-2
	B.3	SYSGEN MESSAGES	B-5
APPENDIX	C	FILES OF THE VAX/VMS SYSTEM	C-1
APPENDIX	D	BOOTSTRAP COMMAND FILES	D-1
APPENDIX	E	VAX/VMS AND DECnet-VAX SOURCE KITS	E-1
	E.1	THE VAX/VMS SOURCE KIT	E-1
	E.2	THE DECnet-VAX SOURCE KIT	E-1
APPENDIX	F	BOOT PROCESS	F-1
APPENDIX	G	BOOTSTRAP AND SYSGEN EXAMPLES	G-1
	G.1	FIRST-TIME BOOTSTRAP EXAMPLE	G-1
	G.2	CONVERSATIONAL BOOTSTRAP EXAMPLE	G-1
	G.3	NONSTOP BOOTSTRAP EXAMPLE	G-2
	G.4	SYSGEN EXAMPLE	G-2
INDEX			Index-1

FIGURES

FIGURE	1-1	Transferring the Distribution Medium to Disk and Booting the System	1-2
	5-1	Setting System Generation Parameters	5-3

CONTENTS

TABLES

			Page
TABLE	3-1	VAX/VMS Boot Command Procedures	3-3
	4-1	Parameter Files Provided by VAX/VMS	4-2
	4-2	Parameter Values in Parameter Files Provided by VAX/VMS	4-3
	4-3	Recommended Paging, Swapping, and Dump File Sizes	4-11
	5-1	SYSGEN Commands	5-2
	5-2	Device Names	5-6
	6-1	Code Generation Options	6-8
	C-1	Files Contained in Directory [SYSEXE]	C-2
	C-2	Files Contained in Directory [SYSHLP]	C-4
	C-3	Files Contained in Directory [SYSLIB]	C-6
	C-4	Files Contained in Directory [SYSMGR]	C-7
	C-5	Files Contained in Directory [SYSMSG]	C-7
	C-6	Files Contained in Directory [SYSTEST]	C-7
	C-7	Files Contained in Directory [SYSUPD]	C-11

PREFACE

MANUAL OBJECTIVES

The VAX/VMS Software Installation Guide describes the procedures used to install the VAX/VMS operating system on a VAX-11/780 processor and to install optional VAX-11 software in the VAX/VMS system.

INTENDED AUDIENCE

This manual is intended for VAX/VMS system managers. It contains information appropriate for two levels of experience:

- For the new VAX/VMS manager, it provides the information needed to install VAX/VMS.
- For the more experienced manager, it provides detailed information about the system parameters that can be modified to tailor the system to the needs of a particular installation.

STRUCTURE OF THIS DOCUMENT

This manual is organized as follows:

- Chapter 1 presents an overview of system installation.
- Chapter 2 describes the procedures for copying the distribution medium.
- Chapter 3 describes the steps to be taken when bootstrapping the system disk.
- Chapter 4 describes the major system parameters that can be modified during SYSGEN or SYSBOOT and describes parameter settings appropriate for various hardware configurations.
- Chapter 5 describes the commands that can be issued to the system bootstrap program (SYSBOOT) and the system generation utility (SYSGEN).
- Chapter 6 describes the steps to be taken to install optional software (for example, VAX-11 FORTRAN IV-PLUS and DECnet-VAX) in VAX/VMS.
- Appendix A lists and briefly describes all the VAX/VMS system parameters.

- Appendix B describes the error messages issued during SYSGEN and SYSBOOT and describes corrective actions.
- Appendix C lists the DIGITAL-supplied directories on the system disk and their contents.
- Appendix D provides samples of console bootstrap command files.
- Appendix E describes the steps performed to copy the source kits.
- Appendix F details the steps performed to generate and initialize the system.
- Appendix G provides examples of bootstrap and system generation.

Because this document describes a number of interrelated procedures and components, it is advisable to read the entire document once before attempting to use the procedures described in Chapters 2 and 3.

ASSOCIATED DOCUMENTS

This document has no prerequisites; however, an understanding of the information presented in the VAX/VMS Summary Description may prove helpful.

The following documents, which are referred to in this manual, provide the information needed to manage the day-to-day operation of VAX/VMS once it is installed:

- VAX/VMS System Manager's Guide
- VAX/VMS Operator's Guide

For a complete list of VAX-11 documents, including descriptions of each, see the VAX-11 Information Directory.

SUMMARY OF CHANGES

The following technical changes have been made to the VAX-11 Software Installation Guide for V1.5.

1. To Chapter 6, a general procedure to update your system has been added, along with procedures for installing the following optional software components:

   VAX-11 BLISS-32
   VAX-11 COBOL-74

2. Appendix C has been updated to reflect the addition and deletion of files provided by DIGITAL on the VAX/VMS system binary distribution medium.

CHAPTER 1

INTRODUCTION

To install a VAX/VMS operating system on a VAX-11/780 processor and to install optional VAX-11 components that are separately purchased, you use the following VAX-11/780 and VAX/VMS features and components:

- The VAX-11/780 console and console floppy diskette drive
- The stand-alone VAX-11 Disk Save and Compress utility (DSC)
- The system bootstrap program (SYSBOOT)
- The system generation utility (SYSGEN)

1.1 OVERVIEW OF SYSTEM GENERATION AND START UP

The general procedure includes the following steps:

1. Copying the distribution kit to disk
2. Bootstrapping the system
3. Modifying (if necessary) system parameters

Step 1 requires use of the stand-alone version of the Disk Save and Compress utility (DSC). Step 2 requires use of the system bootstrap program (SYSBOOT). Step 3 requires either SYSBOOT or the system generation program (SYSGEN). See Figure 1-1.

In a VAX/VMS system, system generation and start up occur automatically when the system is bootstrapped. You provide the information needed for system generation and start up by supplying the names of files that contain the necessary console commands, system generation parameter values, and start-up commands to the SYSBOOT program.

You invoke SYSBOOT by typing the name of a bootstrap command file at the console. The name of the command file indicates to SYSBOOT the following information:

- The device and unit containing the disk volume to be bootstrapped
- Whether you want SYSBOOT to prompt for commands during the bootstrap operation

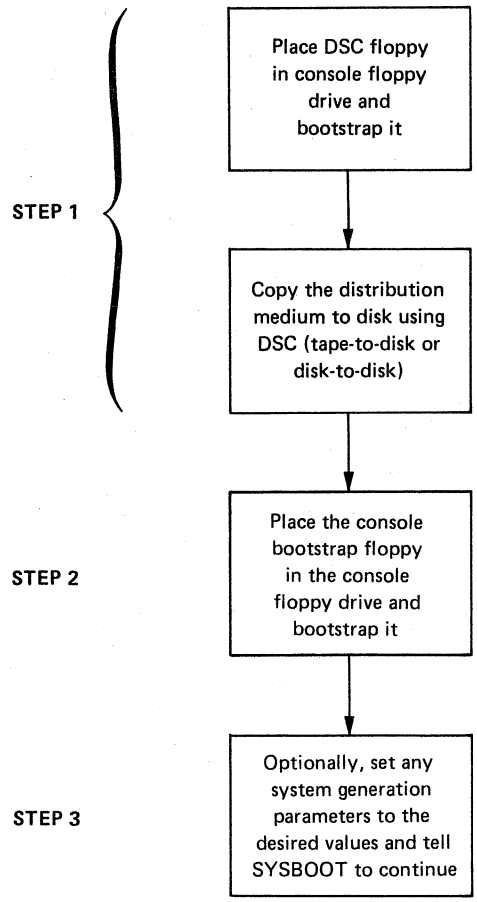

Figure 1-1 Transferring the Distribution Medium to Disk and Booting the System

If you request SYSBOOT to prompt for commands, you can perform the following functions:

- Designate the name of a file that contains system generation parameter values
- Set and show individual parameter values
- Specify the name of the start-up command file

This is referred to as a conversational bootstrap.

If you do not specify the name of a start-up command file, SYSBOOT uses SYS$SYSTEM:STARTUP.COM by default. SYS$SYSTEM:STARTUP.COM requests execution of a start-up file containing commands specified by the system manager to perform site-specific start-up functions. For example, the site-specific start-up file contains the SET TERMINAL commands that set the characteristics of the installation's terminals. By default, the name of this file is [SYSMGR]SYSTARTUP.COM. The VAX/VMS System Manager's Guide describes both these files in detail.

Once all the system initialization steps have occurred, the system creates a process to execute the two start-up files. Among the commands contained in SYS$SYSTEM:STARTUP are, by default, the following:

```
$ RUN SYS$SYSTEM:SYSGEN
AUTOCONFIGURE ALL
```

These commands request the running of the SYSGEN utility and automatically configure the system to recognize all standard (DIGITAL-supplied) devices attached to it. I/O drivers for these devices also are loaded. Keep in mind, though, that the AUTOCONFIGURE command works only when you attach the devices to the system in order described in the VAX/VMS Guide to Writing a Device Driver.

Once the start-up files have been executed, the system is ready for use.

1.2 SYSGEN UTILITY

The SYSGEN utility allows you to perform three functions:

- Modify parameter values and create a system generation parameter file that can be used in a subsequent system bootstrap operation
- Create contiguous files that can be used as swapping, paging, and dump files
- Dynamically make new devices known to the system and load their I/O drivers

SYSGEN runs as a utility program, under control of VAX/VMS. Anyone can run SYSGEN. However, you must have Change Mode to Kernel privilege to execute any device-related SYSGEN commands and Change Mode to Executive privilege to execute the SHOW/DEVICES command. See Chapter 5 for more information on using SYSGEN.

1.3 SYSTEM GENERATION PARAMETER FILES

VAX/VMS provides seven system generation parameter files that contain values suitable for the smallest through the largest VAX-11/780 configurations. Each system generation parameter has four values associated with it:

- The current value
- The default value as established by VAX/VMS
- The minimum allowable value
- The maximum allowable value

At SYSBOOT, the initial parameter values are the current values. At SYSGEN, the initial parameter values are the default values.

You can use the SET command during SYSBOOT or when running SYSGEN to set any parameter to a value in the allowable range or to its default value. When using SYSGEN, you also can set parameters to the values being used currently in the VAX/VMS system under which you are running SYSGEN.

1.4 SYSTEM INSTALLATION SUMMARY

The complete installation procedure encompasses the following steps (references to descriptions of each step are in parentheses):

1. Copy the distribution kit to disk (Chapter 2)
2. Use conversational bootstrap (Section 3.2.1)
3. Select the appropriate system generation parameter file, noting any changes to be made (Chapter 4)
4. Compute the sizes of paging, swapping, and dump files (Chapter 4)
5. Log in (Section 3.3)
6. Run SYSGEN to modify the system generation parameter file (optional; Section 3.4)
7. Alter the sizes of paging, swapping, and dump files (optional; Section 3.5)
8. Copy the bootstrap command file to the console floppy (Section 3.6)
9. Reboot the system, using a conversational bootstrap (Section 3.7)
10. Log in again, and customize the start-up command file (VAX/VMS System Manager's Guide)
11. Create the user authorization file and necessary user file directories (VAX/VMS System Manager's Guide)
12. Install any optional software (Chapter 6)
13. Reboot, with default bootstrap command file (Section 3.8)

The system is now ready for use.

CHAPTER 2

PREPARING TO BOOTSTRAP THE SYSTEM

The first step in installing the VAX/VMS operating system is to copy the distribution kit to an RK07 disk, by using the stand-alone Disk Save and Compress (DSC) utility program supplied as part of the distribution kit. You copy the distribution kit to RK07 disk for one of the following reasons:

- To create a bootable medium, if you received a magnetic tape kit
- To back up the system, if you received an RK07 kit

2.1 VAX/VMS SOFTWARE DISTRIBUTION KITS

There are two media for VAX/VMS software kits:

- Magnetic tape, in which the system binary distribution medium is a 1600 bpi, 9-track magnetic tape
- RK07 disk, in which the system binary distribution medium is an RK07 disk cartridge

The bill of materials that comes with the kit lists exactly what your VAX/VMS software distribution kit contains. After receiving a VAX/VMS software kit, you should check to be sure it contains everything listed in the bill of materials.

2.1.1 Magnetic Tape Kit

For a magnetic tape kit, the following items are needed to bootstrap the system:

1. The magnetic tape system binary distribution medium

   Part number: BB-D782B-BE

   Part description: VAX/VMS V1.5 BIN MT9

2. The floppy diskette that contains the stand-alone Disk Save and Compress (DSC) utility program

   Part number: AS-E808n-YE

   Part description: ESZCCnn 11780 S/A DSC2 FLP

3. The console floppy diskette that contains bootstrap loading functions and bootstrap command files

   Part number: AS-E633n-YE

   Part description: ESZABnn 11780 LOCAL CNSL PKG

NOTE

The part-number and part-description fields denoted by the letters n and nn depend on the ECO level of your system.

2.1.2 RK07 kit

The following items are needed to back up the RK07 distribution medium and bootstrap the system:

1. The RK07 system binary distribution medium

   Part number: AY-H020B-BE

   Part description: VAX/VMS V1.5 BIN RK07

2. The floppy diskette that contains the Disk Save and Compress (DSC) utility program

   Part number: AS-E808n-YE

   Part description: ESZCCnn 11780 S/A DSC2 FLP

3. The console floppy diskette that contains bootstrap loading functions and bootstrap command files

   Part number: AS-E633n-YE

   Part description: ESZABnn 11780 LOCAL CNSL PKG

NOTE

The part-number and part-description fields denoted by the letters n and nn depend on the ECO level of your system.

2.2 COPYING DISTRIBUTION MEDIA

The VAX/VMS system can be booted only from disk. Thus, if you receive a system on a magnetic tape, you must copy the tape to a disk before you can boot the system. You should retain the tape as a back-up copy of the distributed system.

If you receive an RK07 kit, you should back up the system by copying it either to magnetic tape or to another disk before you boot the system. When you copy a disk to another disk, you should, if possible, retain the original disk as back-up and use the newly copied disk to bootstrap the system.

To copy either a magnetic tape or an RK07 disk, use the stand-alone Disk Save and Compress (DSC) utility program. Refer to the VAX-11 Disk Save and Compress User's Guide for details on stand-alone DSC.

2.2.1 Loading Stand-Alone DSC

The procedure for loading stand-alone DSC is as follows:

1. See that the system is powered up and that the central processor is halted. The following switches should be in the positions indicated:

   a. The ON-OFF rocker switch on the LA36 console terminal: ON.

   b. The AUTO RESTART rocker switch on the console control panel: ON.

   c. The rotary key switch on the console control panel: LOCAL.

2. See that the following console control panel indicators are lit: ATTN and POWER.

3. See that needed disk drives and magnetic tape drives are turned on.

4. Place the floppy diskette that contains stand-alone DSC (ESZCCnn 11780 S/A DSC2 FLP) into the console floppy diskette drive, as follows:

   a. Unlock and open the cabinet doors of the central processor.

   b. Swing out the drive assembly until it is at a right angle to the cabinet.

      The drive assembly is a rectangular, unpainted steel box in the lower right-hand corner of the central processor cabinet. There is a black handle on the right of the drive assembly. Pull the handle to swing out the drive assembly. The diskette cannot be inserted unless the drive is swung all the way out.

   c. Insert the diskette into the drive.

      Squeeze the black pushbutton to unlock the slot cover; the cover will spring open. As you insert the diskette, its label (on the smooth side of the diskette) should be at the top and should face the right-hand cabinet door. The oval slot on the diskette should be at the bottom.

   d. Close the diskette slot cover.

   e. Swing the drive assembly back into the central processor cabinet.

5. Set the DC ON-OFF switch on the LSI-11 control panel to OFF. The LSI-11 control panel is located at the bottom left-hand corner of the central processor cabinet.

6. Set the DC ON-OFF switch on the LSI-11 control panel to ON.

   This initiates a restart, and the console program is loaded from the floppy diskette.

7. Observe the following message on the LA36 console terminal:

```
CPU HALTED, SOMM CLEAR, STEP=NONE, CLOCK=NORM
RAD=HEX,ADD=PHYS,DAT=LONG,FILL=00,REL=00000000
INIT SEQ DONE
HALTED AT 00000000

(RELOADING WCS)
LOAD DONE, 00003200 BYTES LOADED
VER:  PCS=01 WCS=08-11 FPLA=08 CON=V01-26-R
(AUTO-RESTART)
CPU HALTED
INIT SEQ DONE
```

   A message of this form indicates that the console program has been loaded. The console program then loads stand-alone DSC; this takes about 1.5 minutes. At the completion of loading, a message of the following form is displayed on the LA36 console terminal:

```
LOAD DONE, 00020400 BYTES LOADED

VAX/VMS DSC-2, VERSION A.1 30-MAY-1978

DSC>
```

   DSC is now running and ready to accept commands.

NOTE

> While DSC is being loaded, do not type on the console terminal. If you do so accidentally, either continue from the resulting halt or repeat steps 5, 6, and 7.

2.2.2 Copying Tape to Disk

To copy a distribution magnetic tape to an RK07 disk cartridge, proceed as follows:

1. Place the system binary magnetic tape on a magnetic tape drive (MTA0, for example). Remove the write-ring, to protect the contents of the tape.

2. Place a scratch RK07 disk cartridge on an RK07 disk drive (DMA0, for example).

3. At the LA36 console terminal, enter the following DSC command:

   ```
   DSC> DMA0:/VE=MTA0:/RW
   ```

   This command copies the contents of the distribution magnetic tape on drive MTA0 to an RK07 disk cartridge on drive DMA0. The qualifier /RW causes the tape to be rewound after the copying operation is complete. The qualifier /VE causes the contents of the disk and tape to be compared to make certain the copying operation was a success. Successful completion of the copying and verification is signaled by the reappearance of the prompt DSC>.

   Any messages you receive while DSC is running are explained in the VAX-11 Disk Save and Compress User's Guide.

You can also copy the system binary magnetic tape to an RM03 disk or an RP06 disk by performing suitable variations of this procedure.

2.2.3 Copying Disk to Disk

To copy a distribution RK07 disk cartridge to a back-up RK07 disk cartridge, proceed as follows:

1. Place the system binary RK07 disk cartridge on an RK07 disk drive (DMA1, for example).

2. Be sure to protect the contents of the system distribution disk by pressing the WRITE PROT pushbutton on disk drive DMA1.

3. Place a scratch RK07 disk cartridge on another RK07 disk drive (DMA0, for example).

4. At the LA36 console terminal, enter the following DSC command:

   ```
   DSC> DMA0:/VE=DMA1:
   ```

   This command copies the contents of the distribution disk on drive DMA1 to an RK07 disk cartridge on drive DMA0. The qualifier /VE causes the output and the input to be compared to make certain the copying operation was a success. Successful completion of the copying and verification is signaled by the reappearance of the prompt DSC>.

   Any messages you receive while DSC is running are explained in the VAX-11 Disk Save and Compress User's Guide.

2.2.4 Terminating Stand-Alone DSC

To terminate stand-alone DSC from the LA36 console terminal, proceed as follows:

1. In response to the prompt DSC>, enter CTRL/P. The following prompt will be displayed: >>>.

2. In response to the prompt >>>, type HALT and press RETURN.

   The following confirming message and prompt will be displayed on the console terminal.

```
      HALTED AT ...
>>>
```

   This message displays the contents of the program counter at the time the processor was halted. The prompt is a request for a console command.

For instructions on bootstrapping and installing the system, see Chapter 3.

CHAPTER 3

BOOTSTRAPPING AND INSTALLING A VAX/VMS SYSTEM

Once you have copied the distribution medium to disk, you are ready to bootstrap the VAX/VMS operating system. Proceed as follows to perform the initial bootstrapping of the system:

- Bootstrap the system, stopping in SYSBOOT to specify that the parameter file MINIMUM.PAR is to be used, as described in Sections 3.1 and 3.2.1.
- Select the appropriate parameter file and note any changes to be made, as described in Chapter 4.
- Compute the sizes of the paging file, swapping file, and system dump file, as described in Chapter 4.
- Log into the system, as described in Section 3.3.
- Run the SYSGEN utility to customize the selected parameter file, if desired, as described in Section 3.4.
- Use the command file that VAX/VMS provides to alter the paging, swapping, and dump file sizes, if desired, as described in Section 3.5.
- Copy the bootstrap command file to DEFBOO.CMD on the console floppy diskette, as described in Section 3.6.
- Reboot the system, stopping in SYSBOOT to specify the name of the selected parameter file, as described in Section 3.7.
- Log into the system again and customize the system manager's start-up command file ([SYSMGR]SYSTARTUP.COM), as described in the VAX/VMS System Manager's Guide.
- Run the AUTHORIZE program to create the user authorization file and create the necessary user file directories, as described in the VAX/VMS System Manager's Guide.
- Install any of the following optional components as described in Chapter 6:

  - VAX-11 FORTRAN IV-PLUS
  - PDP-11 BASIC-PLUS-2/VAX
  - DECnet-VAX
  - VAX-11 COBOL-74
  - PDP-11 DATATRIEVE/VAX
  - FORTRAN IV/VAX to RSX Cross Compiler
  - VAX-11 BLISS-32

- Reboot the system, as described in Section 3.8.

After you complete the final step, the system is ready for use.

3.1 ENTERING COMMANDS TO THE CONSOLE PROGRAM

To bootstrap the system, you must first invoke the console program. When the console is ready to accept commands, it prompts with three angle brackets (>>>). If this prompt does not appear on the console, perform the following steps:

- Ensure that the floppy diskette labeled ESZABnn 11780 LOCAL CNSL PKG (part number AS-E633n-YE) is in the floppy diskette drive.
- Set the AUTO RESTART switch to the OFF position.
- Turn the rotary key to the LOCAL position, if it is not already set to LOCAL. Turning the power on boots the console floppy diskette and causes the console program to prompt.
- If the power is already on, type CTRL/P to cause the console program to prompt. Type REBOOT to cause the console to be rebooted.
- Before proceeding further, ensure that the disk to be booted is write enabled.

3.2 BOOTING THE SYSTEM

You can boot the system in either of the following ways:

- Conversational: request that SYSBOOT stop and allow you to modify the system generation parameters that configure the system. This is the option to use the first time you boot the system.
- Non-stop: allow SYSBOOT to run to completion without your intervention.

Console commands are used to request the booting of the system. VAX/VMS provides a number of command files that contain the necessary console commands to boot the system from an RM03 or RP06 disk device on the first MASSBUS controller, or from an RK07 disk on the UNIBUS.

The console floppy diskette contains these command files in two sets:

- One (conversational) set boots the system from the specified device and then stops in the SYSBOOT program to accept changes to the parameters used to configure the system. See Section 3.2.1.
- The other (non-stop) set boots the system from the specified device without stopping in SYSBOOT for changes to parameter values. See Section 3.2.2.

Table 3-1 lists the names of boot command files on the console floppy diskette.

Table 3-1
VAX/VMS Boot Command Procedures

Type of Command File	Name of Command File
Conversational boot from RK07	DM0GEN DM1GEN DM2GEN DM3GEN
Conversational boot from RM03 or RP06	DB0GEN DB1GEN DB2GEN DB3GEN DB4GEN DB5GEN DB6GEN DB7GEN
Non-stop boot from RK07	DM0BOO.CMD DM1BOO.CMD DM2BOO.CMD DM3BOO.CMD
Non-stop boot from RM03 or RP06	DB0BOO.CMD DB1BOO.CMD DB2BOO.CMD DB3BOO.CMD DB4BOO.CMD DB5BOO.CMD DB6BOO.CMD DB7BOO.CMD

NOTE

Do not type on the console while the system is being bootstrapped unless prompted for input. If you do so accidentally, the following messages are printed:

SYSBOOT-W-FPLA,PCS or WCS version less than minimum required for VMS.

SYSBOOT-W-Continue from halt to proceed with boot if desired.

Either continue from the resulting halt or reboot. To continue, type: CONTINUE.

3.2.1 Conversational Bootstrap

To boot the system and request that SYSBOOT stop to allow you to change system generation parameters, type a command file name with the following format:

```
>>> @DxyGEN
```

@

Indicates that the rest of the line contains the name of a command file located on the floppy diskette.

x

Indicates the type of the device containing the system volume to be booted. Specify one of the following letters to indicate the desired boot device:

M = RK07
B = RM03 or RP06

y

Specifies the unit number of the drive containing the volume to be booted. This number is in the range 0 through 3 if you are booting from an RK07, or 0 through 7 if you are booting from an RM03 or RP06.

When SYSBOOT is ready to accept commands, it prompts as follows:

```
SYSBOOT>
```

You can now issue any of the commands listed in Section 5.4.

The first time you bootstrap the system, type the following commands:

```
SYSBOOT> USE MINIMUM.PAR

SYSBOOT> CONTINUE
```

The second time you boot the system, issue a USE command specifying the name of the parameter file that you want SYSBOOT to use to generate your system. This can be one of the file names listed in Table 4-1 (see Chapter 4), or it can be the name of a file created by means of the SYSGEN utility (see Chapter 5).

The following example shows a console printout obtained by booting the system using the bootstrap command file DM0GEN.

```
>>> @DM0GEN

!

!       DM0 CONVERSATIONAL BOOT COMMAND FILE - DM0GEN.

!       BOOT FROM DM0 AND STOP IN SYSBOOT TO ALTER PARAMETER VALUES.

!

HALT                    ! HALT PROCESSOR

        CPU HALTED

UNJAM                   ! UNJAM SBI
```

```
INIT                        ! INIT PROCESSOR

        INIT SEQ DONE
DEPOSIT/I 11 20003800       ! SET UP SCBB

DEPOSIT R0 1                ! CARTRIDGE DISK

DEPOSIT R1 3                ! UBA TR=3

DEPOSIT R2 3FF20            ! CSR ADDRESS OFFSET = 3FF20

DEPOSIT R3 0                ! CONTROLLER UNIT = 0

DEPOSIT R4 0                ! BOOT BLOCK LBN (UNUSED)

DEPOSIT R5 1                ! SOFTWARE BOOT FLAGS (CONVERSATIONAL BOOT)

DEPOSIT FP 0                ! SET NO MACHINE CHECK EXPECTED

START 20003000              ! START ROM PROGRAM

WAIT DONE                   ! WAIT FOR COMPLETION

        HALT INST EXECUTED
        HALTED AT 200034F9

                            !

EXAMINE SP                  ! SHOW ADDRESS OF WORKING MEMORY+^X200
        G  0000000E  00000200
LOAD VMB.EXE/START:@        ! LOAD PRIMARY BOOTSTRAP

                            LOAD DONE, 00001600 BYTES LOADED
START @                      ! AND START IT

<@EOF>
<@EXIT>

SYSBOOT>  HELP
Major SYSBOOT Commands are:

                            CONTINUE - Continue with boot process
                            EXIT     - Continue with boot process
                            SET      - Set parameter value
                                 SET parameter-name value
                                 SET /STARTUP file-spec
                            SHOW      - Show parameter  value (s)
                                SHOW parameter_name
                                    /ACP - Show ACP parameters
                                    /ALL - Show ALL parameters
                                    /GEN - Show  generative parameters
                                    /MAJOR - Show MAJOR parameters
                                    /NAMES - Show parameter names
                                    /PQL - Show Process Quota List values
                                    /RMS - Show RMS parameters
                                    /STARTUP - Show Startup command file name
                                    /SYS - Show SYSTEM parameters
                            USE       - Set parameter file name
                                USE file_spec.PAR
                                    Reserved filespecs are:
                             DEFAULT - Use permanent defaults
                             CURRENT - Use current values
```

```
SYSBOOT>  USE P4CONFIG.PAR
SYSBOOT>  SHOW /MAJOR

  Parameter Name     Current   Default   Minimum  Maximum  Unit
  --------------     -------   -------   -------  -------  ----
  PFCDEFAULT             127        16         0      127  PAGES
  GBLSECTIONS             80        40        20       -1  SECTIONS
  GBLPAGES              3072      2048       512       -1  PAGES
  MAXPROCESSCNT           64        64        12      256  PROCESSES
  SYSMWCNT               100        48        20    16384  PAGES
  BALSETCNT               40        24         4     1024  SLOTS
  IRPCOUNT               240        80         0    32768  PACKETS
  WSMAX                  700       256        60    16384  PAGES
  NPAGEDYN            109568     40448     16384       -1  BYTES
  PAGEDYN              32768      8192      8192       -1  BYTES
  VIRTUALPAGECNT        8192      8192       512    65536  PAGES
  QUANTUM                 30        30         2    32767  10MS
  MPW_WRTCLUSTER          64        16         0      127  PAGES
  MPW_HILIM              128        24         0    16384  PAGES
  MPW_LOLIMIT             96        12         0    16384  PAGES

SYSBOOT>  SET BALSETCNT 8

SYSBOOT>  SET VIRTUALPAGECNT 999999

%SYSBOOT-W-Value set to maximum
SYSBOOT>  SHOW VIRTUALPAGECNT
  VIRTUALPAGECNT      65536      8192       512    65536  PAGES

SYSBOOT>  CONTINUE


  VAX/VMS  Version 1.50 21-AUG-1978 15:40


   OPCOM, 25-AUG-1978 16:25:24.26 LOGFILE INITIALIZED, OPERATOR=_OPA0

$ !
$ ! VAX/VMS system startup - Version 1
$ !
$ SHOW TIME
  25-AUG-1978 15:33:30
$ SET NOVERIFY
%MOUNT-I-MOUNTED, CONSOLE      mounted on _DXA1:
  Login quotas - Interactive limit=4, Current interactive value=0
  SYSTEM       job terminated at 25-AUG-1978 15:33:51.47

  Accounting information:
  Buffered I/O count:          150      Peak working set size:   135
  Direct I/O count:             37      Peak virtual size:       111
  Page faults:                 210      Mounted volumes:           1
  Elapsed CPU time:     0 00:00:01.67   Elapsed time:     0 00:00:33.37
```

3.2.2 **Non-Stop Bootstrap**

To perform a boot operation without stopping in SYSBOOT, type a command file name with the following format:

```
>>> @DxyBOO.CMD
```

@

Indicates that the rest of the line contains the name of a command file that is located on the floppy diskette.

x

Indicates the type of the device containing the system volume to be booted. Specify one of the following letters to indicate the desired boot device:

M = RK07
B = RM03 or RP06

y

Specifies the unit number of the drive containing the volume to be booted. This number is in the range 0 through 3 if you are booting from an RK07, or 0 through 7 if you are booting from an RM03 or RP06.

Note that you can type a command in the following format to perform a non-stop bootstrap:

```
>>> BOOT Dxy
```

For example, BOOT DM0 is equivalent to @DM0BOO.CMD. If you use the long form, the contents of the command file are displayed on the console. If you use the short form, they are not displayed.

The following console printout was obtained by booting the system using the bootstrap command file DM0BOO.

```
>>> BOOT DM0

        CPU HALTED
        INIT SEQ DONE
        HALT INST EXECUTED
        HALTED AT 200034F9

        G  0000000E  00000200
        LOAD DONE, 00001600 BYTES LOADED

  VAX/VMS VERSION 1.50 21-AUG-1978 15:40

  OPCOM, 25-AUG-1978 15:42:01.87, LOGFILE INITIALIZED, OPERATOR=_OPA0:

$ !
$ ! VAX/VMS system startup - Version 1
$ !
$ SHOW TIME
  25-AUG-1978 15:42:07
$ SET NOVERIFY
%MOUNT-I-MOUNTED, CONSOLE      mounted on _DXA1:
  Login quotas - Interactive limit=64, Current interactive value=0
  SYSTEM       job terminated at 25-AUG-1978 15:42:28.91

  Accounting information:
  Buffered I/O count:          150      Peak working set size:   135
  Direct I/O count:             37      Peak virtual size:       111
  Page faults:                 210      Mounted volumes:           1
  Elapsed CPU time:     0 00:00:01.61   Elapsed time:     0 00:00:33.67
```

3.3 LOGGING INTO THE SYSTEM

Once the system is loaded into memory and initialized, it announces itself as illustrated above. At this point, you can log into the system as the system manager by performing the following steps:

- Press the RETURN key
- In response to the system's request for your user name, type SYSTEM
- In response to the system's request for your password, type MANAGER

When the DCL command interpreter prompt ($) appears on the console, you are ready to use the system. The system prints the following message and the command interpreter prompts:

```
     WELCOME TO VAX/VMS-Version 1.50
$
```

3.4 MODIFYING THE PARAMETER FILE YOU SELECT

Once you have selected the parameter file most appropriate for your configuration, you can modify the values of individual parameters and write a new file as follows:

- Set your default device and directory to SYS$SYSTEM using the following DCL command:

  ```
  $ SET DEFAULT SYS$SYSTEM
  ```

- Invoke the SYSGEN utility with the following command:

  ```
  $ RUN SYSGEN
  ```

- Enter a USE command specifying the parameter file to be used as the source of parameter values. For example:

  ```
  SYSGEN> USE 16USER.PAR
  ```

- Change the values in the parameter file. For example:

  ```
  SYSGEN> SET VIRTUALPAGECNT   8192
  ```

- Write a file that contains the modified parameter values. For example:

  ```
  SYSGEN> WRITE MYPARAM.PAR
  ```

- Exit from SYSGEN, as follows:

  ```
  SYSGEN> EXIT
  $
  ```

Chapter 5 contains a description of the SYSGEN commands.

3.5 ALTERING PAGING, SWAPPING, AND DUMP FILE SIZES

VAX/VMS provides a command file in the directory [SYSUPD] to simplify the alteration of the paging, swapping, and system dump file sizes. Perform the following steps to alter the sizes of these files:

- If necessary, log into the system as the system manager, as described in Section 3.3, and then request the command file as follows:

  ```
  $ @[SYSUPD]SWAPFILES
  ```

  The command file prints the following information:

  ```
  To leave a file size at its current value type a carriage
  return <CR> in response to its file size.  Current file
  sizes are:

  DIRECTORY DB2:[SYSEXE]
  22-AUG-78 09:59

  PAGEFILE.SYS;1     48000.    C    21-AUG-78 09:23
  SWAPFILE.SYS;1     36000.    C    21-AUG-78 09:23
  SYSDUMP.DMP;1      4104.     C    21-AUG-78 09:23

  TOTAL OF 88104./88104. BLOCKS IN 3. FILES


  Enter new size for paging file:
  Enter new size for swapping file:
  Enter new size for system dump file:
  $
  ```

- In response to each request for a file size, either press RETURN to leave the file size unchanged; or, type the new file size, then press RETURN.

3.6 DEFAULT BOOTSTRAP COMMAND FILE

Once you have selected the boot command file to be used for your system, you should copy it to the console floppy diskette, giving it a file name of DEFBOO.CMD. This establishes it as the default boot command file. The automatic boot command file is used in several situations:

- When the system automatically reboots itself; Section 3.6.2 describes the automatic bootstrap

- When you turn on the BOOT switch on the processor control panel

- When you issue the console command BOOT without providing a boot command file name

3.6.1 Copying The Default Boot Command File

VAX/VMS provides a command file named SETDEFBOO.COM that simplifies the copying of the default boot command file to the console floppy diskette. To use SETDEFBOO, first log into the system as the system manager and then type the following:

```
$ @[SYSUPD]SETDEFBOO
```

SETDEFBOO asks you to confirm that the console floppy diskette is mounted and requests the name of the boot command file that is to become the default as follows:

```
Is the console floppy mounted: YES
Enter name of default boot command file:
```

Once you enter the name of the appropriate boot command file, for example, DM0BOO.CMD, SETDEFBOO copies the specified boot command file to DEFBOO.CMD on the console floppy diskette. When it finishes the copying operation, SETDEFBOO issues the following message:

```
Default boot command file replaced with file-name
```

3.6.2 Automatic Restart

The VAX-11/780 processor is designed for unattended continuous operation. It is able to restart or reboot itself in the event of power recovery or any processor halt condition. To enable the automatic restart feature, set the AUTO RESTART rocker switch on the console panel to the ON position. Automatic restarting should be disabled during the installation procedures, but should be enabled once the installation procedure is completed.

When automatic restart is enabled and a power recovery or halt occurs, the LSI-11 console deposits the contents of PC and PSL at the time of the halt into R10 and R11 and deposits a code giving the reason for the restart into AP. The console then invokes the command file RESTAR.CMD, which is listed in Appendix D. After a power recovery, the restart ROM program checks to determine whether the contents of memory are still valid and whether a restart routine can be located. If both conditions are satisfied, the restart ROM program passes control to the restart routine; otherwise, the system is rebooted using DEFBOO.CMD.

Any condition other than power recovery results in a bugcheck and an automatic rebooting of the system using DEFBOO.CMD.

3.7 REBOOTING THE SYSTEM STOPPING IN SYSBOOT

To halt the processor when VAX/VMS is running and to reboot the system, proceed as follows:

- Halt the processor by entering CTRL/P to obtain the console prompt (>>>), and then type the HALT command

- Bootstrap the system, using the following command file:

```
>>> @DMyGEN
      or
>>> @DByGEN
```

The letter y denotes the unit number of the drive containing the volume to be booted. This number is in the range of 0 through 3 if you are booting from an RK07 or 0 through 7 if you are booting from an RM03 or RP06.

- When SYSBOOT prompts, issue a USE command specifying the name of the parameter file that you wrote and then continue; for example:

  ```
  SYSBOOT>  USE MYPARAM.PAR

  SYSBOOT>  CONTINUE
  ```

When VAX/VMS announces itself, the new parameter values and the new paging, swapping, and dump files are in use.

3.8 REBOOTING USING THE DEFAULT BOOTSTRAP

Under normal system operation, you do not need to interrupt the booting of the system to type commands to SYSBOOT; that is, parameter values have been established. Rather, you can bootstrap the system using the default bootstap command file that you copied to the file DEFBOO.CMD. To do so, proceed as follows:

- Obtain the console prompt (>>>) by entering CTRL/P
- Halt the processor by typing the HALT command
- Either type BOOT or press the BOOT button on the console

3.9 COPYING FILES TO OR FROM THE CONSOLE FLOPPY DISKETTE

Occasionally, you may want to copy files from the console floppy diskette, edit them, and place them back on the floppy diskette while the VAX/VMS operating system is running. VAX/VMS provides a command file in the directory [SYSUPD] to simplify the copying of text files (that is, ASCII files such as bootstrap command files) to or from the console floppy diskette.

Request execution of the command file by typing the following command to the VAX/VMS command interpreter:

```
$ @[SYSUPD]DXCOPY
```

The command file then asks whether the console floppy diskette is mounted, as follows:

```
Is the console floppy mounted (Y/N):
```

If it is, type Y to proceed. The command file then asks whether the copy operation is from the console floppy, as follows:

```
Copy from console floppy:
```

You can type Y to indicate a copy from diskette to the current default directory, or you can type N to indicate a copy from the current default directory to diskette.

The command file then requests the name of the file to be copied to or from the diskette, as follows:

Enter console file name:

Type the name of the file, and press RETURN.

3.10 BUILDING AND COPYING A VAX/VMS SYSTEM

Two command procedures are supplied for the purposes of building and copying a VAX/VMS system:

1. VMSKITBLD.COM
2. VMSKITCPY.COM

3.10.1 Using VMSKITBLD.COM

The command procedure VMSKITBLD.COM is used to build a VAX/VMS system binary disk. For example, if you have a mixed-disk system (with RK07 and either RP06 or RM03 disks but no magnetic tape drives), you can use VMSKITBLD.COM to transfer your VAX/VMS system from an RK07 disk cartridge to a larger, faster RP06 or RM03 disk pack.

Before you can use VMSKITBLD.COM to build a VAX/VMS system (on an RP06, for example) you must first boot your system from an RK07 disk cartridge (as described at the beginning of this chapter). With this RK07 system running, proceed as follows:

1. Log in under the system manager's account (initially, an account with the user name SYSTEM and the password MANAGER).
2. Set your default directory to SYSUPD using the following DCL command:

   SET DEFAULT [SYSUPD]

3. Place an RP06 disk pack on an RP06 drive. This will be the target disk in the system building procedure.
4. At your terminal, type the following command line:

   @VMSKITBLD

From this point, you will receive prompts for needed information and you will be kept informed of the progress of the system building procedure. When you are asked for the name of the mounted source disk, reply with the name of the RK07 system disk from which you booted the system.

In the process of building a bootable VAX/VMS system on an RP06 or RM03 disk pack, this command procedure creates a larger swap file, a larger page file, and a larger dump file than were in the original RK07 binary distribution kit. Thus, the resulting VAX/VMS system is suitable on systems containing RP06 or RM03 disks.

You are informed when the system building procedure is complete.

3.10.2 Using VMSKITCPY.COM

The command procedure VMSKITCPY.COM is used to copy the files of a source VAX/VMS system binary disk onto a target disk that contains a valid VAX/VMS system.

Before you can use VMSKITCPY.COM to copy one system disk to another, your VAX/VMS system must be running and the source disk that you intend to copy must be mounted. Often, this source disk is the system disk from which the system was booted. Proceed as follows to copy the source disk to a target disk:

1. Log in under the system manager's account (initially, an account with the user name SYSTEM and the password MANAGER).

2. Set your default directory to SYSUPD using the following DCL command:

   ```
   SET DEFAULT [SYSUPD]
   ```

3. Place a target disk on an appropriate drive.

4. At your terminal, type the following command line:

   ```
   @VMSKITCPY
   ```

From this point, you will receive prompts for needed information and you will be kept informed of the progress of the copying procedure. You will be informed when the copying procedure is complete.

CHAPTER 4

SYSTEM PARAMETER FILES

The VAX/VMS software kit contains seven parameter files, each of which generates a system appropriate for a particular hardware configuration. These files are listed in Table 4-1. The parameter values defined in each file are generally suitable for the intended configuration. By selecting the parameter file that best matches your hardware, you can produce a working system in a relatively short time. There is no need to engage in a lengthy system generation procedure.

4.1 MODIFYING SYSTEM PARAMETER FILES

After the system is installed and running, you can refine the parameter values as appropriate for your installation. The most common changes involve one of the following situations:

- Increasing the values of the WSMAX or VIRTUALPAGECNT parameters to accommodate a particular application
- Increasing NPAGEDYN to support additional devices

From Table 4-1, choose the parameter file that most closely matches your hardware configuration. Then refer to Table 4-2 for the parameter values included in that parameter file, and determine which values you want to change.

Table 4-1
Parameter Files Provided by VAX/VMS

File Name	Intended Hardware Configuration
MINIMUM.PAR	Minimum hardware configuration; includes only the console terminal, console floppy diskette drive, and the bootstrap disk
8USER.PAR	256K bytes of memory, 8 DZ11 lines, and 2 RK07 disk drives
16USER.PAR	512K bytes of memory, 16 DZ11 lines, 2 RM03 disk drives, line printer, TE16 magnetic tape unit
32USER.PAR	1024K bytes of memory, 32 DZ11 lines, 2 RP06 disk drives, line printer, 2 TE16 magnetic tape units, DMC-11 with DECnet-VAX
48USER.PAR	1536K bytes of memory, 48 DZ11 lines, 4 RP06 disk drives, 2 TE16 magnetic tape units, 2 RK07 disk drives, line printer, DMC-11 with DECnet-VAX
64USER.PAR	2048K bytes of memory, 64 DZ11 lines, 4 RP06 disk drives, 2 TE16 magnetic tape units, 2 RK07 disk drives, line printer, 2 DMC-11 with DECnet-VAX, card reader
VIRT32MB.PAR	1024K bytes of memory, 16 DZ11 lines, 2 RP06 disk drives, 2 TE16 magnetic tape units, line printer, and support of a 32-million byte virtual address space

Table 4-2
Parameter Values in Parameter Files Provided by VAX/VMS

Parameter Name	Parameter File Name						
	MINIMUM	8USER	16USER	32USER	48USER	64USER	VIRT32MB
PFCDEFAULT	16	16	32	64	127	127	127
GBLSECTIONS	20	24	32	40	80	80	32
GBLPAGES	1024	2048	2048	2048	3072	3072	2048
MAXPROCESSCNT	12	20	28	48	68	84	28
SYSMWCNT	20	20	48	64	80	96	48
BALSETCNT	4	10	16	28	40	48	8
IRPCOUNT	0	80	140	230	300	380	150
WSMAX	95	150	256	512	700	1024	1024
NPAGEDYN	16384	35840	43520	88000	109568	123904	71680
PAGEDYN	8192	16384	16384	24576	32768	32768	32768
VIRTUALPAGECNT	2048	4096	4096	8192	8192	8192	65536
QUANTUM	30	30	30	30	30	30	30
MPW_WRTCLUSTER	8	16	16	32	64	127	127
MPW_HILIM	10	24	24	48	128	192	192
MPW_LOLIMIT	2	12	12	16	96	96	96
PROCSECTCNT	10	20	20	20	20	20	20
SPTREQ	384	512	512	512	512	512	512
FREELIM	0	10	10	16	16	16	10
ACP_SHARE	0	0	0	1	1	1	0
ACP_MAPCACHE	2	2	2	4	4	4	2
ACP_HDRCACHE	4	8	8	16	16	16	8
ACP_DIRCACHE	4	6	6	8	12	12	6
ACP_WORKSET	55	55	0*	0*	0*	0*	0*
MAXPRINTSYMB	1	1	1	4	8	8	2
IJOBLIM	4	9	17	33	49	65	17
BJOBLIM	1	1	1	4	8	8	4
NJOBLIM	16	16	16	16	16	16	16

* Allows ACP to compute its own working set size.

The upper section of Table 4-2 lists the major VAX/VMS parameters; these parameters are most likely to require modification. The lower section lists parameters whose values generally increase as configurations become larger. Those parameters not listed in Table 4-2 have their default values for all seven parameter files. See Appendix A.

4.2 MAJOR PARAMETER DESCRIPTIONS

The major parameters are the ones displayed by SYSBOOT or SYSGEN in response to the SHOW/MAJOR command:

- BALSETCNT
- GBLPAGES
- GBLSECTIONS
- IRPCOUNT
- MAXPROCESSCNT
- MPW_HILIM
- MPW_LOLIMIT
- MPW_WRTCLUSTER
- NPAGEDYN
- PAGEDYN
- PFCDEFAULT
- QUANTUM
- SYSMWCNT
- VIRTUALPAGECNT
- WSMAX

4.2.1 BALSETCNT -- Number of Processes Allowed in the Balance Set

The BALSETCNT parameter determines the maximum number of processes that can have their working sets resident concurrently, even if additional physical memory is available. VAX/VMS uses the value of BALSETCNT to allocate system page table space for mapping each of the process headers for processes in the balance set.

The system page table space required depends primarily on the parameter VIRTUALPAGECNT, which specifies the maximum number of virtual pages allowed a process. Every 128 virtual pages require one longword of system page table space for each balance set slot. The default virtual page count (8192 pages) requires approximately 64 longwords. That number of longwords is then multiplied by the balance set count to determine the total requirement.

The system page table is permanently resident.

4.2.2 GBLPAGES -- Global Pages

The GBLPAGES parameter controls the total number of global pages that can be known to the system. This parameter determines the total number of global page table entries allocated. Each global section requires two global page table entries in addition to a page table entry for each page of the global section. The total is rounded up to be divisible by two.

Unused global page table entries are not costly; every 128 global page table entries add one entry, or four bytes, to the size of the system page table (SPT).

The parameter file for the minimum system (that is, the system generated using the parameter file MINIMUM.PAR) reduces the number of global sections and global pages in order to reduce resident memory requirements. This reduction is made possible through the use of a different start-up command file named STARTUP.MIN; it installs a minimum number of shared images. All parameter files for larger configurations permit the installation of images that are normally shared and provide a reasonable number of available, unused global sections and pages.

You can use the /GLOBAL option of the INSTALL utility to ascertain global page usage. Refer to the description of RUN SYS$SYSTEM:INSTALL in the VAX/VMS Operator's Guide for details.

4.2.3 GBLSECTIONS -- Global Sections

The GBLSECTIONS parameter establishes the number of global section descriptors to be allocated by SYSBOOT. Each global section created requires a global section descriptor. Each global section descriptor occupies 32 bytes of permanently resident memory in the system header.

The number of global sections initially allowed should be somewhat greater than the number actually expected. After learning about global section usage in your system by using the /GLOBAL option of the INSTALL utility, you should redefine this parameter. The VAX/VMS Operator's Guide describes the INSTALL utility.

Also see the description of GBLPAGES, in Section 4.2.2, above.

4.2.4 IRPCOUNT -- I/O Request Packet Count

To achieve low overhead, VAX/VMS preallocates a number of I/O request packets, and keeps them in a special list, which it also uses when it receives an allocation request for nonpaged dynamic memory in which the requested memory is the same size as an I/O request packet. The IRPCOUNT parameter establishes the number of preallocated I/O request packets.

Each packet requires 80 bytes of nonpaged dynamic pool memory. VAX/VMS reduces the number of preallocated I/O request packets so that no more than half of the nonpaged pool is used for preallocated I/O request packets. If the amount of memory preallocated for I/O packets is less than half the size of the nonpaged pool, IRPCOUNT remains unchanged.

In the minimum system (that is, one generated using MINIMUM.PAR), the value of IRPCOUNT is reduced to zero to minimize the size of NPAGEDYN. I/O packets can be allocated as needed from the remaining nonpaged pool. For larger systems, IRPCOUNT is approximately four times MAXPROCESSCNT.

4.2.5 MAXPROCESSCNT -- Maximum Process Count

The MAXPROCESSCNT parameter determines the maximum number of processes that can be known to the system. Each entry (slot) requires six bytes of resident memory.

To determine the correct value for MAXPROCESSCNT, add the desired number of user processes to the number of system processes, and then add two. Because of the small memory requirement for process slots, you should specify this number on the high side.

The number of processes allowed is the sum of the desired number of interactive jobs, plus the necessary system processes. The following system processes are necessary in all systems and are always present:

- Null process
- Swapper
- Job controller
- Error log process
- Operator communication process
- Files-11 ACP

The number of system processes increases as additional volumes requiring separate ACPs are mounted; for example, the magnetic tape ACP (MTAACP) is required if ANSI magnetic tapes are to be used. Network operations also require another ACP as well as provision for network jobs. The increase in MAXPROCESSCNT for larger configurations is to accommodate these additional system processes.

For additional information, see the description of WSMAX.

4.2.6 MPW_HILIM -- Modified Page List High Limit

The MPW_HILIM parameter sets the number of pages that can be accumulated on the modified page list before modified page writing is initiated.

This is a minor parameter in a system that has many more processes than can fit in the balance set, because the swapper forces the writing of modified pages if there is no other way to obtain memory for swapping in pages. In a better balanced system, the actual size of the modified page list is effectively subtracted from the pages available to contain balance set processes; in this case, the modified page list acts as an extension to the working sets of actively paging processes by serving as a cache for pages.

4.2.7 MPW_LOLIMIT -- Modified Page List Low Limit

The MPW_LOLIMIT parameter sets the size to which the modified page list must be reduced before the modified page writer stops. In normal operations, modified page writing does not start until the number of modified pages equals MPW_HILIM. The number of modified pages is then reduced by units of approximately the size of MPW_WRTCLUSTER until the modified page count is less than MPW_LOLIMIT.

For the most efficient clustering, MPW_HILIM should be approximately 1.5 times MPW_WRTCLUSTER. MPW_HILIM is assumed to be greater than MPW_LOLIMIT.

4.2.8 MPW_WRTCLUSTER -- Modified Page Write Cluster Size

The MPW_WRTCLUSTER parameter specifies the number of pages the modified page writer attempts to collect and write as a contiguous disk transfer. Each page in the cluster requires six additional bytes to be allocated in a map for the modified page writer.

In general, the cluster size should be as large as possible, but not larger than either 127 pages or the value of MPW_HILIM (see Section 4.2.6).

4.2.9 NPAGEDYN -- Number of Bytes of Nonpaged Dynamic Pool

The NPAGEDYN parameter specifies the total number of bytes to be allocated for the nonpaged dynamic pool. The value specified is rounded down to an integral number of pages.

NPAGEDYN is one of the most important parameters and one of the most difficult to set optimally. If you have enough memory, you should set the value of NPAGEDYN higher than you believe necessary. Then use the DISPLAY utility to correct the value after you have observed actual pool usage. The VAX/VMS System Manager's Guide describes the DISPLAY utility. Under conditions of a peak load, the total available pool space should be at least 150 bytes for each process.

The sum of the following numbers is helpful in estimating the nonpaged dynamic pool size required:

512 bytes for each process
WSMAX * 4 bytes
MAXPROCESSCNT * 6 bytes
IRPCNT * 80 bytes
BALSETCNT * 4 bytes
MPW_WRTCLUSTER * 6 bytes
KFILSTCNT * 68 bytes
Number of blocks in PAGEFILE.SYS divided by 8
Number of bytes to be reserved for network device (DMC11) receive buffers

The nonpaged dynamic pool is also used to allocate space for both loadable drivers and dynamically created device data base structures. The sizes of the standard drivers and associated data base requirements are as follows:

CRDRIVER - 1296 bytes
  Each unit (DDB+CRB+IDB+UCB) = 304 bytes
DBDRIVER - 1712 bytes
  First unit on MBA (DDB+CRB+IDB+UCB) = 368 bytes
  Each additional unit (UCB) = 160 bytes
DMDRIVER - 2208 bytes
  First unit (DDB+CRB+IDB+UCB) = 384 bytes
  Each additional unit (UCB) = 208 bytes
DRDRIVER - 1812 bytes
  First unit on MBA (DDB+CRB+IDB+UCB) = 368 bytes
  Each additional unit (UCB) = 160 bytes
DZDRIVER - 0 bytes because it is part of the resident executive
  Each 8-line DZ11 (DDB+CRB+IDB+8(UCB)+(type-ahead buffers)=2128 bytes
LPDRIVER - 992 bytes
  Each unit (DDB+CRB+IDB+UCB) = 304 bytes

TMDRIVER - 3968 bytes
First unit on TM03 (DDB+CRB+IDB+UCB) = 336 bytes
Each additional unit (UCB) = 160 bytes
XMDRIVER - 2864 bytes
Each unit (DDB+CRB+IDB+UCB) = 416 bytes

No matter how many device units are in the configuration, only one copy of each driver is required.

4.2.10 PAGEDYN -- Number of Bytes of Paged Dynamic Pool

The PAGEDYN parameter specifies the total number of bytes to be allocated for the paged dynamic pool. The value specified is rounded down to an integral number of pages. The paged dynamic pool is used to allocate storage for system and group logical names, image headers for known images that have resident headers, and known file list entries.

Each 512 bytes of paged pool space increases the size of the system page table by four bytes; the paged pool space has no other direct memory requirement. You can overallocate substantial amounts of paged dynamic pool space with little effect on the system.

4.2.11 PFCDEFAULT -- Page Fault Cluster Default

The PFCDEFAULT parameter sets the default page fault cluster size that is used for any process or global section that does not explicitly specify its own page fault cluster size. The parameter is also used to cluster pages read from the paging file.

This parameter is not very sensitive; however, it should have a value of at least 16 to ensure reasonable performance. As a general rule, the page fault cluster factor should not exceed one-fourth of the normal working set quota.

The default page fault cluster size increases with increased physical memory. As the default cluster size approaches its maximum value, the actual size of clusters read tends to be limited by other factors, such as the size of sections and the pattern of references. When physical memory is small, the default cluster size is reduced to prevent a single page fault from replacing the entire working set.

4.2.12 QUANTUM -- Time Quantum

Among processes of equal priority, processor time is divided on a round-robin basis: the QUANTUM parameter specifies the time quantum used. The quantum is also used to guarantee that processes have enough time to perform useful work before they are swapped to secondary storage. That is, a process newly swapped into memory is usually ineligible to be swapped out until its first quantum expires.

Throughput in a heavily swapping system can be improved at the expense of response time by increasing the quantum to a maximum of 200 (2000 milliseconds). However, decreasing the quantum below the default value will not necessarily improve response time. In fact, such a decrease is likely to have the opposite effect. The smallest quantum value should be larger than the time required to complete the typical interaction.

4.2.13 SYSMWCNT -- System Maximum Working Set Count

The SYSMWCNT parameter sets the number of pages to be used for the pageable executive code and data, VAX-11 RMS code, and the system message file.

Values above the minimum for SYSMWCNT provide better performance at the expense of additional pages of memory occupied by the system. This value generally should be approximately 4 or 5 percent of the total physical memory.

The sizes of paged system components are listed below:

Component	Number of Pages
EXECUTIVE	49
RMS	61
SYSMSG.MPF	123
	233
Paged Pool	16 to 64
Total	249 to 297

SYSMWCNT should not be larger than the total number of pages of paged system components contained in system address space.

The system working set values are quite small in the three smallest configurations to provide the maximum number of pages available to users. If the majority of the page fault activity in these systems is for system pages, the value of SYSMWCNT should be increased.

The system working set values are sufficient, but not generous, in the larger configurations. Some system paging from the free list does not impair performance significantly and the smaller guaranteed working set size allows the system to adapt readily to changing memory demands.

4.2.14 VIRTUALPAGECNT -- Maximum Virtual Page Count for any Process

The VIRTUALPAGECNT parameter specifies the maximum number of virtual pages that can be mapped for any process. VAX/VMS uses VIRTUALPAGECNT to determine the amount of space to allocate in a process header for the P0 and P1 page tables. The total number of virtual pages can be divided between the two page tables in any proportion; however, the minimum size for the P1 page table is large enough to map approximately 320 pages. Every 128 virtual pages require an additional page table page in the process header. Each additional page table page increases the space required for each balance set slot in the system page table by 4 bytes and increases the size of process headers by 8 bytes; process headers are not paged but can be swapped.

For general purpose use, the virtual page count should be at least 2048 pages. The default value provides 4,194,204 bytes of virtual address space for processes, which should be adequate.

The system generated using MINIMUM.PAR provides a 1-million byte address space. Systems generated using 8USER.PAR and 16USER.PAR provide a 2-million byte address space. Systems generated using 32USER.PAR, 48USER.PAR, and 64USER.PAR provide a 4-million byte address space. While these address spaces should be sufficient for their intended configurations, you may need to increase them if

virtual-space-full errors occur. The parameter file VIRT32MB.PAR provides a system configured for running very large, 32-million byte programs.

Note also that as processes use larger virtual address spaces, the size of the paging file should be increased.

4.2.15 WSMAX -- Maximum Process Working Set Size

The WSMAX parameter specifies the maximum number of pages allowed for any process's working set. In addition, VAX/VMS uses WSMAX to allocate space in both the swapper map and the process header. Each page of working set requires one longword in the swapper map, and every 128 pages of working set require one additional longword in the system page table for each balance set slot.

The value of WSMAX significantly affects the size of the swapping file (SYS$SYSTEM:SWAPFILE.SYS) required. That size is equal to WSMAX multiplied by MAXPROCESSCNT. If the swapping file is not large enough, the number of processes allowed is reduced.

After the swapper map and system page table are allocated and the system is fully initialized, VAX/VMS compares the value of WSMAX with the number of available pages (p) minus the system working set size (SYSMWCNT) and the minimum number of free pages (FREELIM). That is:

n=p-(SYSMWCNT+FREELIM)

VAX/VMS selects the lower of the two values (WSMAX and n) and uses it as the divisor to divide the swap file into swap slots. The number of processes that can be created (MAXPROCESSCNT) is compared with the number of swap slots; the lower of the two is used as the value of MAXPROCESSCNT. However, the nonpaged pool used for process slots is based on the original value of MAXPROCESSCNT.

For the configuration generated using MINIMUM.PAR, WSMAX is reduced to the smallest value that permits necessary system functions to be performed by facilities that must, at least temporarily, lock all or part of their pages in the working set. In larger configurations, the value of WSMAX is approximately one fourth of the size of physical memory.

4.3 DETERMINING THE SIZES OF THE PAGING, SWAPPING, AND SYSTEM DUMP FILES

The VAX/VMS system as distributed comes with paging, swapping, and system dump file sizes that are suitable for small configurations, and are appropriate for the parameter files 8USER.PAR and MINIMUM.PAR only. Any larger configuration requires an increase in the sizes of these files.

Table 4-3 contains the recommended paging file, swapping file, and system dump file sizes in blocks for the parameter files that VAX/VMS provides.

Table 4-3
Recommended Paging, Swapping, and Dump File Sizes

File Type	Parameter File Name						
	MINIMUM	8USER	16USER	32USER	48USER	64USER	VIRT32MB
PAGEFILE.SYS	8192	8192	16384	32768	61440	98304	98304
SWAPFILE.SYS	3080	3080	7168	24576	47600	86016	28672
SYSDUMP.DMP	516	516	1028	2052	3076	4100	2052

Section 3.5 describes the procedures for modifying these values.

4.3.1 Paging File Size

The size of the paging file required is the size of the average program in pages, multiplied by the number of processes. You can determine the size of an average program by looking at the peak virtual size displayed when a user logs off the system using the following command:

```
$ LOGOUT /FULL
```

The accounting record for each user also contains the peak virtual size.

You can limit the number of paging file pages required by any process with the paging file limit specified in the user authorization file. However, under normal circumstances, you should allow at least 1024 pages for each process. A process that does not exceed its working set size requires a paging file limit but does not actually use paging file space. Refer to Table 4-3 for suggested paging file sizes.

4.3.2 Swapping File Size

The swapping file size is the product of the values of the parameters MAXPROCESSCNT and WSMAX. Refer to Table 4-3 for recommended sizes.

4.3.3 System Dump File Size

The system dump file serves two purposes:

- It provides continuity of the error log when the system is shut down or crashes.
- It saves the contents of physical memory at the time of a crash, for subsequent analysis.

The first function requires three disk blocks, and the second function requires one additional block for each page of physical memory. Refer to Table 4-3 for recommended sizes.

S.L.

CHAPTER 5

THE SYSGEN UTILITY

The SYSGEN utility lets you modify certain aspects of your system configuration.

SYSGEN performs the following functions:

- Modifies the values of parameters and creates parameter files to be used in subsequent system boot operations
- Identifies devices to the system and loads their I/O drivers
- Creates contiguous files that can be used as swapping, paging, and dump files

Sections 5.2 and 5.3 summarize SYSGEN commands. Section 5.4 summarizes the commands that can be used with SYSBOOT.

5.1 INVOKING AND TERMINATING SYSGEN

To invoke SYSGEN, type the following command:

```
$ RUN SYS$SYSTEM:SYSGEN
```

When SYSGEN is invoked, the values of the parameters are set to the default values, not the current values. Refer to the USE command in Section 5.18 for more information on the default values.

To terminate SYSGEN, type either of the following commands: CONTINUE or EXIT. These commands are described in Sections 5.7 and 5.11, respectively.

5.2 SYSGEN COMMAND SUMMARY

Table 5-1 lists the complete set of SYSGEN commands, in alphabetical order. Detailed descriptions of these commands are in Sections 5.5 through 5.19.

Table 5-1
SYSGEN Commands

Command	Function
AUTOCONFIGURE	Makes all standard devices attached to the system known to VAX/VMS, and loads their device drivers
CONNECT	Makes a single device known to VAX/VMS, and loads its device driver
CONTINUE	Terminates SYSGEN
CREATE	Creates a contiguous file for use as a paging file, swapping file, or dump file
DISABLE CHECKS	Inhibits checking of parameter values specified with a SET command
ENABLE CHECKS	Permits checking of parameter values specified with a SET command
EXIT	Terminates SYSGEN
HELP	Displays a summary of SYSGEN commands at your terminal
LOAD	Loads an I/O driver
RELOAD	Loads an I/O driver, and removes a previously loaded version of the I/O driver
SET (parameter-value)	Specifies a system parameter value
SET (start-up file)	Specifies the start-up file to be executed by SYSBOOT
SHOW (/qualifier)	Displays parameters specified by the qualifier
USE	Specifies the source of values to be written in a parameter file
WRITE	Creates a parameter file

5.3 SYSGEN COMMAND USAGE

SYSGEN commands are used for three general purposes:

- To modify and create parameter files
- To specify I/O devices and drivers as part of the system
- To create a swapping, paging, or dump file

MCR SYSGEN
SYSGEN > CONNECT CONSOLE
— — > ^Z

5.3.1 Parameter File SYSGEN Commands

The SYSGEN commands used with parameter files are:

DISABLE CHECKS
ENABLE CHECKS
HELP
SET
SHOW
USE
WRITE

THE SET and USE commands establish parameter values to be used in a subsequent system bootstrap operation. The WRITE command places the selected values in a parameter file that is supplied to SYSBOOT. When you run SYSGEN online, these commands have no effect on the running system; for example, changing the value of the GBLSECTIONS parameter does not affect the number of global sections that can be used in the system at that time.

Figure 5-1 shows an example of a SYSGEN session.

```
$ RUN SYS$SYSTEM:SYSGEN
SYSGEN> USE CURRENT
SYSGEN> SHOW /JOB

  Parameter Name     Current   Default   Minimum   Maximum  Unit
  --------------     -------   -------   -------   -------  ----
  MAXPRINTSYMB             8         8         1       255  PROCESSES
  DEFPRI                   4         4         1        31
  IJOBLIM                 64        64         1      1024  JOBS
  BJOBLIM                 16        16         0      1024  JOBS
  NJOBLIM                 16        16         0      1024  JOBS

SYSGEN> SET DEFPRI 6
SYSGEN> WRITE SYS$SYSTEM:NEWVAL.PAR
SYSGEN> EXIT
$
```

Figure 5-1 Setting System Generation Parameters

In this example, the user invokes SYSGEN online, sets all parameter values to their defaults, displays the job controller parameters, increases the value of DEFPRI to 6, writes a new parameter file named NEWVAL.PAR, and then terminates SYSGEN.

5.3.2 I/O Device SYSGEN Commands

The SYSGEN commands used for I/O devices and drivers are:

AUTOCONFIGURE
CONNECT
LOAD
RELOAD
SHOW /DEVICES

The most frequently used device command is AUTOCONFIGURE; it is executed by default in the command procedure SYS$SYSTEM:STARTUP.COM. Other device commands are used for nonstandard configurations and to incorporate nonstandard devices and user-written I/O drivers into the system.

With the exception of SHOW /DEVICES, all I/O device and driver commands require Change Mode to Kernel privilege; SHOW /DEVICES requires Change Mode to Executive privilege.

The VAX/VMS Guide to Writing a Device Driver provides supplemental information on all I/O device and driver commands.

5.3.3 Swapping, Paging, Dump File SYSGEN Command

The CREATE command creates a contiguous file that is to be used as the system's swapping, paging, or dump file. See Section 5.8.

5.4 SYSGEN COMMANDS AVAILABLE IN SYSBOOT

You can use a subset of the SYSGEN commands with the SYSBOOT program, which actually allocates the necessary system structures based on the parameter values. The commands that can be used with SYSBOOT are:

- HELP
- SHOW
- SET
- DISABLE CHECKS
- ENABLE CHECKS
- USE
- CONTINUE and EXIT

5.5 AUTOCONFIGURE

The AUTOCONFIGURE command is the primary means of making standard devices and their drivers known to a VAX/VMS system. AUTOCONFIGURE locates each device unit physically attached to the system, performs a connect for it, and loads the driver. Table 5-2 lists standard device names.

The VAX/VMS Guide to Writing a Device Driver describes the vector and CSR requirements for AUTOCONFIGURE.

Formats

```
AUTOCONFIGURE  ALL  [/SELECT=(device-name,...)]

AUTOCONFIGURE tr-number [/SELECT=(device-name,...)]
```

Parameters

ALL

Indicates that all standard devices attached to the system are to be included in the configuration.

If you specify ALL /SELECT, devices are included in the configuration according to the device names specified.

tr-number

Specifies the SBI arbitration line to which the UNIBUS adapter is attached. If you specify a TR number as a parameter to the AUTOCONFIGURE command, AUTOCONFIGURE includes all devices on that TR number in the configuration.

If you specify a TR number with /SELECT, the devices using that TR number are included according to the device names specified.

Command Qualifier

/SELECT=(device-name,...)

Specifies devices that are to be included in the configuration. The device name can consist only of a device type and controller designation; it cannot include a unit number. If you omit the controller designation, all devices with the specified device type are included in the configuration.

If you specify multiple device names, separate them with commas and enclose them in parentheses.

Example

```
SYSGEN>  AUTOCONFIGURE ALL /SELECT=(TT,MTA,DR,LP)
```

This command configures the system for all terminals, magnetic tape units on controller A, all RM03 disks, and all line printers.

Table 5-2
Device Names

Mnemonic	Device Type
CR	Card Reader
DB	RP04, RP05, RP06 Disk
DM	RK06, RK07 Disk
DR	RM03 Disk
DX	Floppy Disk
LP	Line Printer
MB	Mailbox
MT	TE16 Magnetic Tape
NET	Network logical link device
TT	Interactive terminal
XM	DMC-11

5.6 CONNECT

The CONNECT command is used to add a nonstandard device and user-written I/O driver to a VAX/VMS system. The CONNECT command creates I/O data base control blocks for additional devices.

For more information on the CONNECT command, see the VAX/VMS Guide to Writing a Device Driver.

Format

CONNECT device-name required-quals [optional-quals]

Command Qualifiers

/ADAPTER=tr-value
/CSR=csr-address
/DRIVERNAME=driver-name (optional)
/NUMVEC=vector-address (optional)
/VECTOR=vector-address

Parameter

device-name
Specifies the name of the device for which control blocks are to be added to the I/O data base. Specify the device name in the following format:

devcu

dev = device type
c = controller designation
u = unit number

For example, LPA0 specifies the line printer (dev) on controller A (c) at unit 0 (u).

Required Qualifiers

/ADAPTER=tr-value
Specifies the number of the SBI arbitration line to which the UNIBUS adapter is attached. The tr-value must be in the range 0 through 15.

/CSR=csr-address
Specifies the UNIBUS address of the controller status register for the device.

/VECTOR=vector-address
Specifies the UNIBUS address of the interrupt vector for the device.

Optional Qualifiers

/NUMVEC=number
Specifies the number of interrupt vectors for the device. If you omit this qualifier, the number of vectors defaults to 1. The number specified by the /VECTOR qualifier is the address of the lowest vector.

/DRIVERNAME=driver-name

Specifies the name of the driver as recorded in the driver prologue table. If the driver is not loaded when you issue the CONNECT command, CONNECT assumes the driver name is also the file name of an executable image that is located in SYS$SYSTEM and contains a driver for the device type.

For DIGITAL-supplied devices, the driver name and its image file name both have the following format:

aaDRIVER

The letters aa represent the device type of the device. For DIGITAL-supplied devices, the following device types are valid: CR, DB, DM, DX, DR, LP, MT, NET, TT, XM.

If you do not specify /DRIVERNAME, CONNECT uses a driver name of aaDRIVER, by default. For example, if the device name specified is LP, CONNECT uses a driver name of LPDRIVER.

Example

```
SYSGEN>  CONNECT LPA0 /ADAPTER=3       /CSR=%O777514   /VECTOR=%O200
```

This command loads the driver (default name LPDRIVER), if it is not already loaded, and creates the device data base (DDB, CRB, IDB, and UCB) needed to describe LPA0. The CSR and vector values are for the first line printer.

Note

You should use extreme caution when entering the CONNECT command. The driver and data base loading procedure does little error checking. If you specify an incorrect vector or misspell a device name, for example, you will damage the I/O data base. Such damage may cause a system crash.

5.7 CONTINUE

The CONTINUE command terminates the SYSGEN utility. If you are running SYSGEN as part of the system booting process, issuing the CONTINUE command allows SYSBOOT to proceed. If you invoked SYSGEN through the command interpreter, the CONTINUE command causes the command interpreter to prompt.

Format

CONTINUE

Example

```
    .
    .
    .
SYSBOOT> CONTINUE
```

This command causes SYSBOOT to resume booting the system.

CREATE

5.8 CREATE

The CREATE command creates a contiguous file that can be used as a paging file, a swapping file, or a dump file. VAX/VMS provides these files as part of the distribution kit. It also provides a command file for changing their sizes; therefore, you need not issue CREATE commands directly.

Format

```
CREATE file-spec/SIZE=block-count
```

There must be no spaces between the file specification and the /SIZE qualifier.

file-spec
: Provides the file specification of the file to be created.

/SIZE=block-size
: Indicates the number of contiguous blocks to be allocated for the file on disk.

Note

See Chapter 4 for information on calculating the size of paging, swapping, and dump files.

5.9 DISABLE CHECKS

The DISABLE CHECKS command inhibits the range checking SYSGEN performs on parameter values. By default, range checking is enabled for both SYSGEN and SYSBOOT. If you attempt to set a parameter to a value that is above the maximum, SYSGEN sets it to the maximum and issues a warning message to inform you. If you attempt to set a parameter to a value that is below the minimum, SYSGEN sets it to the minimum and issues a warning message.

Format

```
DISABLE CHECKS
```

Example

```
SYSGEN> SET WSMAX 20
%SYSGEN-W-Value set to minimum
SYSGEN> DISABLE CHECKS
SYSGEN> SET WSMAX 20
SYSGEN> SHOW WSMAX
WSMAX                20        256        60        16384     PAGES
```

In this sequence of commands, you first attempt to set WSMAX to 20 while range checking is enabled; SYSGEN issues an error message. You then issue the second command to disable range checking. Subsequent commands set the current value of WSMAX to 20 and display WSMAX values, respectively.

Note

If you create a parameter file containing values that are outside the normal range, and wish to use the parameter file with SYSBOOT, you must issue a DISABLE CHECKS command to SYSBOOT. Otherwise, SYSBOOT does not allow you to use parameter values that are not in the standard range.

ENABLE CHECKS

5.10 ENABLE CHECKS

The ENABLE CHECKS command requests that SYSGEN ensure that parameter values changed using the SET command remain in the allowable range. By default, range checking is enabled. If you attempt to set a parameter to a value that is above the maximum, SYSGEN sets it to the maximum and issues a warning message to inform you. If you attempt to set a parameter to a value that is below the minimum, SYSGEN sets it to the minimum and issues a warning message.

Format

ENABLE CHECKS

Example

```
SYSGEN>  ENABLE CHECKS
SYSGEN>  SET GBLSECTIONS 18
%SYSGEN-W-Value set to minimum
SYSGEN>
```

In the sequence above, the first command enables range checking. The second command attempts to limit the number of global sections in the system to 18; the minimum value for that parameter is 20. SYSGEN sets the parameter value to 20 and issues the warning message.

5.11 EXIT

The EXIT command terminates the SYSGEN utility. If you are running SYSGEN as part of the system booting process, issuing the EXIT command allows SYSBOOT to proceed. If you invoked SYSGEN by issuing a command to the command interpreter, the EXIT command causes the command interpreter to prompt.

Format

```
EXIT
```

Example

```
SYSGEN> EXIT
$
```

Because SYSGEN was invoked from the command level, this command causes SYSGEN to exit and the command interpreter to prompt.

HELP

5.12 HELP

The HELP command displays a summary of the SYSGEN commands at your terminal. Because only a subset of command are available during SYSBOOT, the HELP displays for SYSGEN and SYSBOOT differ.

Format

HELP

Examples

The following command displays the HELP information available during SYSGEN.

```
SYSGEN> HELP
Major SYSGEN Commands are:

        CREATE      - Create contiguous file
             CREATE file_spec/SIZE=block_count
        EXIT        - Exit SYSGEN utility
        SET         - Set parameter value
             SET parameter_name value
             SET /STARTUP file_spec
        SHOW        -Show parameter value(s)
             SHOW parameter_name
                    /ACP     - Show ACP parameters
                    /ALL     - Show ALL parameters
                    /DEVICES - Show device data base
                    /GEN     - Show generative parameters
                    /MAJOR   - Show MAJOR parameters
                    /NAMES   - Show parameter names
                    /PQL     - Show Process Quota List values
                    /RMS     - Show RMS parameters
                    /STARTUP - Show startup command file name
                    /SYS     - Show SYSTEM parameters
        USE         - Set parameter file name
             USE file_spec
                  Reserved filespecs are:
                            DEFAULT - Use permanent defaults
                            CURRENT - Use current values
        WRITE       - Write parameter file
             WRITE file_spec
```

The following command displays the HELP information available during SYSBOOT.

```
SYSBOOT> HELP
Major SYSBOOT Commands are:

        CONTINUE    - Continue with boot process
        EXIT        - Continue with boot process
        SET         - Set parameter value
             SET parameter_name value
             SET /STARTUP file_spec
```

```
SHOW        - Show parameter value(s)
     SHOW parameter_name
            /ACP - Show ACP parameters
            /ALL - Show ALL parameters
            /GEN - Show generative parameters
            /MAJOR - Show MAJOR parameters
            /NAMES - Show parameter names
            /PQL - Show Process Quota List values
            /RMS - Show RMS parameters
            /STARTUP - Show Startup command file name
            /SYS - Show SYSTEM parameters
USE         - Set parameter file name
     USE file_spec.PAR
          Reserved filespecs are:
                    DEFAULT - Use permanent defaults
                    CURRENT - Use current values
```

LOAD

5.13 LOAD

The LOAD command loads an I/O driver into the system. For more information on this command, see the VAX/VMS Guide to Writing a Device Driver.

Format

LOAD driver-file-spec

driver-file-spec
Provides the file specification of the image file containing the I/O driver to be loaded. If the driver file specification is the same as the driver name of a loaded driver, the LOAD command has no effect.

If you do not specify a file type, EXE is used by default.

The driver file specifications for devices supported by VAX/VMS are:

Driver-File-Spec	Device
CRDRIVER.EXE	Card reader
DBDRIVER.EXE	RP05 and RP06 disk
DMDRIVER.EXE	RK06 and RK07
DRDRIVER.EXE	RM03 disk
LPDRIVER.EXE	Line printer
TMDRIVER.EXE	TE16 magnetic tape
XMDRIVER.EXE	DMC-11

Example

```
SYSGEN>  LOAD CRDRIVER
```

This command loads the card reader driver.

Notes

- If the name (not file specification) of the driver to be loaded is the same as the name of a driver that is currently loaded, the new driver replaces the existing one.
- The AUTOCONFIGURE and CONNECT commands automatically load the I/O drivers for standard devices.

5.14 RELOAD

The RELOAD command loads a driver and removes a previously-loaded version of that driver. The RELOAD command provides all the functions of LOAD, except that it loads the driver regardless of whether it is already loaded.

If any unit associated with the driver is busy, the driver cannot be reloaded; SYSGEN issues an error message.

For more information on the RELOAD command, see the VAX/VMS Guide to Writing a Device Driver.

Format

RELOAD driver-file-spec

driver-file-spec
Provides the file specification of the driver to be loaded.

SET (PARAMETER VALUE)

5.15 SET (PARAMETER VALUE)

This form of the SET command allows you to establish the value of a system generation parameter.

Format

SET parameter-name value

parameter-name
Specifies the name of the parameter for which the new value is to be established.

value
Specifies the value of the parameter. The value can be either a decimal number in the allowable range for that parameter or the keyword DEFAULT.

If you specify DEFAULT, SYSGEN uses the default value established by VAX/VMS.

Examples

```
SYSGEN>  SHOW PFCDEFAULT
  PFCDEFAULT              16        16        0        127     PAGES

SYSGEN>  SET PFCDEFAULT  20
```

The first command above displays the current, default, minimum, and maximum values for the page fault cluster default size. The SET command raises the cluster size to 20.

```
SYSGEN>  SET GBLSECTIONS DEFAULT
```

This command sets the number of global section descriptors to be allocated by SYSBOOT to the default value of 40.

SET (START-UP FILE)

5.16 SET (START-UP FILE)

This form of the SET command specifies the name of a start-up command procedure to be executed the next time the system is bootstrapped.

Format

SET /STARTUP file-spec

/STARTUP
Indicates that the SET command is to designate a start-up file.

file-spec
Specifies the name of the start-up command procedure to be used. The file must be located on the system disk.

Example

```
SYSGEN>  SET /STARTUP SYS$SYSTEM:NEWSTART.COM
```

This command establishes the start-up command procedure as SYS$SYSTEM:NEWSTART.COM.

Note

If you do not specify a start-up file, the start-up process executes SYS$SYSTEM:STARTUP.COM.

SHOW

5.17 SHOW

The SHOW command displays the names of system parameters, or the values associated with system parameters. The SHOW command displays four values for each parameter and indicates the unit of measure associated with the values:

- Current value
- Default value
- Minimum allowable value
- Maximum allowable value

A -1 in the minimum or maximum value column indicates that no minimum or maximum value exists for the parameter.

Format

SHOW parameter-name

Command Qualifiers

/ACP
/ALL
/DEVICES
/GEN
/JOB
/MAJOR
/NAMES
/PQL
/RMS
/SPECIAL
/STARTUP
/SYS

parameter-name

Specifies the name of a single parameter for which the value is to be displayed. If you specify a parameter name, you cannot include a qualifier in the command.

You can issue the following command to display the names of all parameters:

```
SHOW /NAMES
```

Appendix A lists the names of all parameters and describes them. The most frequently used parameter names (that is, the names of parameters in the major category) follow:

BALSETCNT -- balance set count
GBLPAGES -- global pages
GBLSECTIONS -- global section
IRPCOUNT -- I/O request packet count
MAXPROCESSCNT -- maximum process count
MPW_HILIM -- modified page list high limit
MPW_LOLIMIT -- modified page list low limit
MPW_WRTCLUSTER -- modified page write cluster size

NPAGEDYN -- nonpaged dynamic memory
PAGEDYN -- paged dynamic memory
PFCDEFAULT -- page fault cluster default size
QUANTUM -- time quantum
SYSMWCNT -- system maximum working set count
VIRTUALPAGECNT -- virtual page count
WSMAX -- working set maximum

These parameters are described in detail in Chapter 4.

Command Qualifiers

/ALL
Displays the values for all parameters except those displayed by specifying /SPECIAL.

/ACP
Displays all parameters associated with Files-11 ancillary control processes (ACPs).

/DEVICES
Displays the names of all I/O drivers, their starting and ending virtual addresses, and the virtual addresses of their I/O data base control blocks. You must have Change Mode to Executive privilege to use this qualifier; otherwise, you will receive a syntax error message.

For more information on the SHOW /DEVICES command, see the VAX/VMS Guide to Writing a Device Driver.

/GEN
Displays structure-generating parameters and other parameters that have effect only when the system is bootstrapped.

/JOB
Displays all job controller parameters.

/MAJOR
Displays the major parameters. These are the parameters that are most likely to require adjustment for individual installations. Chapter 4 provides additional information about these parameters.

/NAMES
Displays the names of all parameters.

/PQL
Displays the values associated with process creation limits and quotas.

/RMS
Displays the values associated with VAX-11 RMS.

/SPECIAL
Displays the values associated with special parameters used by DIGITAL.

/STARTUP
Displays the name of the start-up file.

/SYS
Displays parameters associated with overall system operation.

Examples

```
SYSGEN>  SHOW /JOB

Parameter Name  Current  Default  Minimum     Maximum        Unit
--------- ----  -------  -------  -------     -------        ----

MAXPRINTSYMB         8        8        1         255        PROCESSES
DEFPRI               4        4        1          31
IJOBLIM             64       64        1        1024        JOBS
BJOBLIM             16       16        0        1024        JOBS
NJOBLIM             16       16        0        1024        JOBS
```

This command causes the parameters associated with the job controller to be displayed.

```
SYSGEN>  SHOW WSMAX
  WSMAX            256      256       60       16384        PAGES
```

This command shows the values associated with the parameter that controls the maximum working set size of all processes in the system.

5.18 USE

The USE command specifies the source of system generation parameter values. You can specify the source as a file that you created, or you can indicate that either the current parameter values or the default parameter values are to be used. Having established the source, you can modify individual parameters by means of the SET command.

Formats

```
USE file-spec

USE CURRENT

USE DEFAULT
```

file-spec
: Indicates the name of the parameter file whose values are to be used. You must have created the file previously by issuing a WRITE command. The default for file type is PAR.

CURRENT
: Indicates that the values used for the currently running system are to be used. When the system is first brought in from the distribution medium, the CURRENT and DEFAULT values are the same.

DEFAULT
: Indicates that the default values established by VAX/VMS are to be used.

Example

```
SYSGEN>  USE DEFAULT
SYSGEN>  SET IJOBLIM 80
SYSGEN>  SET BJOBLIM 10
SYSGEN>  WRITE NEWVALS.PAR
```

The first command sets the values of the system generation parameters to the defaults supplied by VAX/VMS. The two SET commands establish new values for the limits on the number of interactive and batch jobs, respectively. The WRITE command creates a new file (NEWVALS.PAR) to contain the parameter values.

WRITE

5.19 WRITE

The WRITE command creates a file that contains system parameter values. The values normally are a combination of VAX/VMS-supplied defaults and parameter values that you establish by means of the SET command. Once created, the file can be specified in a USE command.

Format

WRITE file-spec

file-spec
Indicates the file specification for the parameter file to be created. The default for file type is PAR.

Example

```
SYSGEN> WRITE [SYS$SYSTEM]OURVAL
```

This command creates a parameter file named [SYS$SYSTEM]OURVAL.PAR.

Note

By default, the file is placed in your current default device and directory.

CHAPTER 6

UPDATING THE SYSTEM AND INSTALLING OPTIONAL SOFTWARE

This chapter describes the procedures for updating a VAX/VMS system and installing optional software components available for VAX-11. Optional software components include:

- VAX-11 FORTRAN IV-PLUS
- FORTRAN IV/VAX to RSX Cross Compiler
- DECnet-VAX
- VAX-11 COBOL-74
- PDP-11 BASIC-PLUS-2/VAX
- PDP-11 DATATRIEVE/VAX
- VAX-11 BLISS-32

The procedures for installing an update or optional software component are automated, and thus, require little involvement on your part beyond (1) setting up the proper conditions for the installation and (2) responding to queries and prompting messages displayed as the installation proceeds. Most queries are simple "Yes" or "No" questions. You respond with Y or N, as appropriate.

6.1 GENERAL INSTALLATION PROCEDURES

This section describes the general procedures for installing optional software components and the steps you take to prepare for their installation. These procedures also apply to updates to system software.

6.1.1 Distribution Kits

Updates and optional software components are distributed on floppy diskettes, usually two or more depending on the component.

Each diskette is labeled with both a serial number corresponding to the software component and a name that differentiates that diskette from others in the distribution kit.

The contents of the diskettes comprise files, including command procedures that copy the components to the system disk. These command procedures direct the installation procedure by means of queries and instructions sent to the terminal. (The term "floppy" or "floppy

disk" is used in some command procedure queries as a synonym for floppy diskette.)

6.1.2 Preparing for Installation

To prepare for installation of an update or optional software component, proceed as follows:

1. Log in under the privileged system manager's account.

2. Be sure that logical name SYS$DISK is assigned to the disk that contains the current version of VAX/VMS, as distributed, and containing all updates. This disk contains the command procedure that initiates the new installation/update procedure. Note that SYS$DISK need not be (and if possible should not be) SYS$SYSTEM.

3. Set defaults as:

   UIC [1,4]

   directory [SYSUPD]

   Set these defaults in this order.

4. At the console terminal, type:

   @VMSUPDATE

You will see the following message text at the terminal:

> This command procedure performs VAX/VMS software updates and unbundled software installations....
>
> During this sequence, the standard console floppy will not be present in the console floppy drive.
>
> Therefore, the system is vulnerable to power failure or other fatal crash. If a system crash should occur during this time the update sequence can be restarted at the beginning of the first incomplete update.
>
> Dismount the current console floppy.
>
> Please place the first floppy of the kit in the console drive.

Note that you will receive a device-not-mounted message, if there is no floppy diskette mounted. Ignore the message, and place your first diskette in the diskette drive.

Consult the pertinent section below to determine which diskette should be used first. As you remove the standard diskette, note the direction it is facing: diskettes from the distribution kit must be inserted in the drive so they face the same direction. (The label is on the front side of the diskette.)

Next you will receive the following query:

ARE YOU READY TO CONTINUE?

If you type Y, the installation proceeds as described in the section pertinent to the software component being installed. See Sections 6.2 through 6.9.

If you type N, the request to put the first kit diskette in the console floppy drive and the query (ARE YOU READY TO CONTINUE?) are repeated.

6.1.3 Duration of an Installation Procedure

The time required to install an update or software component varies from one component to another. It is not unusual for an installation to take 20 minutes or more. In some cases, an interval of this magnitude may occur for individual phases of an installation.

6.1.4 Installation Completion

When the installation of an update or software component is finished, control is returned to the system's command procedure (VMSUPDATE), which sends the following messages to the terminal:

ARE THERE MORE KITS TO PROCESS? [Y/N]

If you type Y, you will receive the following request, and the installation procedure begins again.

Please place the first floppy of the kit in the console drive.

If you have no further installations or updates, type N. You then receive the message:

Please place the system console floppy in the console drive.

You should immediately restore the standard console diskette to the console drive.

Next, you will receive the following query:

ARE YOU READY TO CONTINUE?

If you type Y, the console floppy diskette is automatically mounted and you receive the following message:

Requested update sequence is complete.

Finally, after installing a software component or an update, you should back up the volume SYS$DISK and save the original for future updates.

6.2 UPDATING THE SYSTEM

A distribution kit to update the VAX/VMS system consists of one or more floppy diskettes.

To update your system, follow the procedure described in Section 6.1.2 and perform the following modifications before you invoke the VMSUPDATE command file:

1. Prevent users from accessing the system by running the SHUTDOWN command file. Then reboot the system. Running the SHUTDOWN command file and rebooting the system are described in the VAX/VMS Operator's Guide.

2. Set the login count to 0 by typing:

   ```
   $ SET LOGIN/INTERACTIVE = 0
   ```

3. As a precautionary measure, copy the contents of the system disk to another disk. Several procedures for backing up the system disk are described in the VAX/VMS Operator's Guide.

When directed to place the kit floppy diskette in the console drive, replace the console floppy diskette with the first diskette. Then enter Y to the query:

```
ARE YOU READY TO CONTINUE?
```

Continuation of the update is indicated by the display of the announcement:

```
VAX/VMS Version 1.50 Update
```

This message is followed by three questions.

The first question is:

```
Do you want all the updates applied?  (Y/N, YES RECOMMENDED):
```

A YES to this question is recommended, because it is the option supported by DIGITAL. Answering with a NO lets you choose or reject each patch. Rejecting a patch, however, may make it difficult for DIGITAL to answer problem reports and difficult for you to apply future patches supplied by DIGITAL.

The second question is:

```
Do you want an explanation of each update displayed during the
update?  (Y/N):
```

If you give a YES answer, brief descriptions of the patches will be displayed on your terminal.

The third question is:

```
Do you want previous versions of updated file purged?  (Y/N):
```

Answer with a NO if there is enough disk space to preserve previous versions of the updated files.

When updating has been completed, the completion messages described in Section 6.1.4 are displayed.

To transfer control of the system to the updated version of VAX/VMS (VAX/VMS Version 1.5), halt the processor and reboot the system as described in Chapter 3.

6.3 VAX-11 FORTRAN IV-PLUS

The distribution kit for VAX-11 FORTRAN IV-PLUS consists of two floppy diskettes:

1. VAXFORTV1 BIN RX01

2. VAXFORTBB BIN RX01

These diskettes are to be used in the order shown. When you have completed the steps described in Section 6.1.2, replace the console diskette with the diskette labeled VAXFORTV1.

Respond with Y to the inquiry:

ARE YOU READY TO CONTINUE?

The command procedure on VAXFORTV1 assumes control, and the files needed to build the compiler are copied from VAXFORTV1 to the system disk. When this is done, you receive the following messages:

Please put the second floppy for the FORTRAN kit (VAXFORTBB) in the drive.

Are you ready to continue?

Replace diskette VAXFORTV1 with VAXFORTBB, and type Y to indicate that the installation can continue.

When this stage is complete you will receive the message:

This procedure created new versions of the following files:

```
SYS$DISK:[SYSMSG]F4PLUS.MSG
SYS$DISK:[SYSEXE]F4V.EXE
```

You can keep or delete any older versions.

Do you want to delete older versions?

To delete older versions, type Y. To retain them, type N. This is the final step in installing VAX-11 FORTRAN IV-PLUS. You will receive the completion messages described in Section 6.1.4.

6.4 FORTRAN IV/VAX TO RSX CROSS COMPILER

The distribution kit for the FORTRAN IV/VAX to RSX cross compiler consists of three floppy diskettes:

1. VMS11MR1 BIN RX01
2. VMS11FORA BIN RX01
3. VMS11FORB BIN RX01

These diskettes are to be used in the order shown. When you have completed the steps described in Section 6.1.2, replace the console diskette with the diskette labeled F4/VAX VMS11MR1.

Respond with Y to the query:

ARE YOU READY TO CONTINUE?

The first phase of the installation procedure consists of copying the compiler build files and selecting compiler options. These options are described in the following sections.

6.4.1 Code Generation Options

You can select either of two types of object code: threaded code or inline code.

6.4.1.1 **Threaded Code** - Threaded code is hardware independent. Common FORTRAN operations are performed by reference to library routines that provide functions such as the end-of-DO-loop sequence, locating an element of a multidimensional array, initializing I/O operations, and simulating floating-point operations. When one of these functions is required, the compiler generates a reference to the appropriate library function. At run time, these references are threaded together to ensure that the library routines are invoked correctly to perform the specified operations.

6.4.1.2 **Inline Code** - One of three forms of inline code can be selected, corresponding to three hardware options:

1. EAE (Extended Arithmetic Element)
2. EIS (Extended Instruction Set)
3. FIS (Floating Instruction Set)

In VAX-11 compatibility mode, the only form of inline code supported is EIS.

6.4.1.3 **Code Selection Criteria** - When deciding between threaded and inline code, keep the following points in mind:

- For programs that do not require REAL*4, REAL*8, or COMPLEX*8 operations, inline code executes faster than threaded code, and there is little difference in their size.
- For programs that do require REAL*4, REAL*8, or COMPLEX*8 operations, threaded code is much smaller than inline code and executes nearly as fast.

6.4.1.4 **Selecting a Code Option** - You select a code option by responding to the query:

What kind of code generation is desired?

Enter one of the codes shown in Table 6-1.

Table 6-1
Code Generation Options

Code	Option
THR	Threaded
EAE	Inline (Extended Arithmetic Element)
EIS	Inline (Extended Instruction Set)
FIS	Inline (Floating Instruction Set)

Any other response causes the following message to appear:

Invalid value for this option, please reenter one of (THR,EAE,EIS,FIS).

6.4.2 Speed vs. Size Optimization

Next, you are asked to choose between speed and size, as indicated by the message:

The inline code generation can optimize for SPEED or for SIZE. These options are mutually exclusive.

Should the compiler optimize for SPEED?

If you respond with Y, your programs will run faster at the expense of memory. A response of N specifies that speed is to be traded off in favor of smaller executable programs.

After this phase is completed, you receive the message:

Please put the second floppy disk (VMS11FORA) in the drive.

Are you ready to continue?

Place diskette VMS11FORA in the drive. When you are ready to proceed, type Y. The following message then appears:

> The compiler object library must be copied and the compiler task-built. This will take approximately 20 minutes.

Completion of these procedures is indicated by the message:

> Please put the third floppy disk (VMS11FORB) in the drive.
>
> Are you ready to continue?

Place diskette VMS11FORB in the drive. When you are ready to continue, type Y. You will now be queried about OTS options.

6.4.3 OTS Options

OTS options include:

- Generating the OTS as a separate library or making it part of the standard system library
- Hardware configuration
- Virtual array support

6.4.3.1 **OTS Library Option** - The first OTS query is:

> The FORTRAN IV OTS can either be generated as a separate library:
>
> [SYSLIB]FOROTS.OLB
>
> or it can be included in the standard VAX/VMS RSX-11M system library:
>
> [SYSLIB]SYSLIB.OLB
>
> Do you want the OTS included in SYSLIB.OLB?

If you want the OTS to be in the standard system library, type Y. If you want it to be a separate library, type N.

Installations that want to conserve main memory, and which have many FORTRAN programs, will find it useful to build a FORTRAN IV OTS resident library. A MACRO-11 source file (FORRES.MAC) is included as part of the distribution kit to help in this regard. FORRES.MAC is located on the diskette labeled VMS11MR1, in directory [SYSUPD]. This file contains global references to all modules of the OTS and documentation on logical groups of OTS modules.

Refer to the RSX-11M Task Builder Reference Manual (Order No. AA-2588D-TC) for information on building a FORTRAN OTS shareable library.

6.4.3.2 **Hardware Configuration Options** - The following messages indicate that you must choose a hardware configuration from among those listed:

> The FORTRAN IV OTS can be configured for any of the following hardware options:
>
> No specific hardware: NHD
> EAE hardware: EAE
> EIS hardware: EIS
> FIS hardware: FIS
> FP11 hardware: FPU
>
> Please enter the three-character mnemonic for the desired hardware. What OTS hardware option do you want?

If you respond with anything except one of these codes you receive the message:

> Invalid value for this option, please reenter one of (NHD,EAE,EIS,FIS,FPU).

VAX-11 compatibility mode supports NHD, EIS, or FPU configurations.

As the OTS is being configured, you will receive the message:

> The entry point $ERTXT will be deleted as part of the OTS build.

This message requires no response. It is informational only.

6.4.3.3 **VIRTUAL Array Option** - VIRTUAL arrays are supported on systems that support memory management directives; in other words, on target RSX-11M systems. You can include VIRTUAL array support by responding Y to the query:

> Do you want VIRTUAL array support in the OTS?

Respond N if you do not want this option.

6.4.4 Retaining Previous Versions of FORTRAN IV

Finally, you are asked to indicate whether you want to delete any previous versions of FORTRAN IV, as follows:

> This procedure created new versions of the FORTRAN IV compiler and OTS. You can keep or delete older versions.
>
> Do you want to delete older versions?

To retain previous versions, type N. To delete them, type Y.

This completes the installation of FORTRAN IV. You will next receive the completion message described in Section 6.1.4.

6.4.5 Verifying Installation of FORTRAN IV/VAX

A sample test program is provided to allow you to ensure that FORTRAN IV has been installed correctly. This program resides on the diskette labeled VMS11MR1. The program name is FORTST.FTN, and it is in directory [SYSUPD].

6.5 DECnet-VAX

The DECnet-VAX distribution kit consists of a single floppy diskette: VAXDNETV1 BIN RX01. To install DECnet-VAX, follow the steps described in Section 6.1.2.

After you place the distribution kit diskette (VAXDNETV1) into the console diskette drive, type Y at the console terminal. The contents of VAXDNETV1 are copied to SYS$DISK. There are no queries generated during this process.

When the installation is complete, you will receive the messages described in Section 6.1.4. Approximately 3 to 5 minutes are required to install DECnet-VAX.

6.6 VAX-11 COBOL-74

The distribution kit for the VAX-11 COBOL-74 includes three floppy diskettes:

1. VAXC741 Bin RX01
2. VAXC742 Bin RX01
3. VAXC743 Bin RX01

These are to be used in the order shown.

Perform the steps described in Section 6.1.2, with the following modification:

> Be sure that logical name SYS$LIBRARY is assigned to the disk and directory that contains the current version of VAX/VMS, as distributed, and contains all the updates.

The installation procedure runs under the control of the command file copied from the first diskette (VAXC741 Bin RX01). The command file copies the files from the floppy diskette, and upon completion, dismounts the device and requests the next floppy:

> Please put the second COBOL-74 kit floppy in the drive (label=VAXC742).
>
> Are you ready to continue?

Remove the floppy diskette and replace it with the next one. Then type Y to indicate that you are ready to continue.

After the command file has copied the files from the second floppy diskette, the following message is displayed on your terminal:

> Please put the third COBOL-74 kit floppy in the drive (label=VAXC743).
>
> Are you ready to continue?

Remove the floppy diskette and replace it with the next one. Then type Y to indicate that you are ready to continue.

The installation procedure continues to completion without further intervention. When the installation procedure is complete, the following message is printed:

> Installation of COBOL-74 has completed successfully.

The time required to install VAX-11 COBOL-74 is approximately 30 minutes. When the installation is complete, you will receive the messages described in Section 6.1.4.

You are now ready to run and use VAX-11 COBOL-74.

Note that as a result of the installations procedure, the file C74LIB.EXE is placed in SYS$SYSTEM. You can use this shared image as an alternative to SYS$LIBRARY:C74LIB.OLB. Linking with a shared image is described in the VAX/VMS Cobol User's Guide. Installing a shared image is described in the VAX/VMS System Manager's Guide.

6.6.1 Verifying Installation of VAX-11 COBOL-74

After the installation of the VAX-11 COBOL-74 compiler is complete, the following sample test programs should be run to confirm that the compiler is operating correctly.

There are three sample test programs in the distribution package:

1. IO001.COB
2. IO002.COB
3. IO003.COB

IO001.COB creates a sequential disk file named PD1101 on the system disk. IO002.COB creates a relative file named PD1102 on the system disk. IO003.COB reads PD1101 from the system disk and lists the file in formatted form on the system disk.

There is also a file named UETPC74.COM, which is the command file for the sample test programs. All four files are copied into the directory [SYSTEST] during the installation procedure.

To use the sample test programs, perform the following steps:

1. Log in under the privileged system manager's account.
2. Issue the following SET commands to set the UIC and directory to the proper defaults:

```
SET UIC [1,4]

SET DEF [SYSTEST]
```

3. At the terminal, type:

```
@UETPC74
```

Each sample test program should run to successful completion.

The following ECO message may appear on your terminal:

```
CVTTP ECO not installed.
```

If this message appears on your terminal, it is recommended that you contact your local DIGITAL field service representative.

6.7 PDP-11 BASIC-PLUS-2/VAX

The distribution kit for PDP-11 BASIC PLUS-2/VAX includes four floppy diskettes:

1. BP2VAX BIN RX01
2. BP2VAX2 BIN RX01
3. BP2VAX3 BIN RX01
4. BP2VAX4 BIN RX01

These are to be used in the order shown. After you have performed the steps described in Section 6.1.2, the installation runs under the control of the command procedure copied from diskette BP2VAX.

Respond with Y to the message:

```
ARE YOU READY TO CONTINUE?
```

The following message then appears:

```
This procedure copies in all necessary files to build and install
the Basic Plus 2 (Version 1.5) compiler and OTS system.

...copying first floppy...
```

When the first phase is finished, the following message is displayed:

```
Please put the second Basic Plus 2 kit floppy disk in the drive.

Are you ready to continue?
```

Replace diskette BP2VAX with BP2VAX2 in the console drive, and respond with Y to the query.

The following messages then appear:

```
%MOUNT-I-MOUNTED,  BP2VAX2   mounted on _DXA1:

     ...copying second floppy...
```

When all the files on BP2VAX2 have been copied, you receive the prompting message:

```
Please put the third Basic Plus 2 kit floppy disk in the drive.

Are you ready to continue?
```

Replace diskette BP2VAX2 with BP2VAX3, and respond with Y to the query.

The following messages then appear:

```
%MOUNT-I-MOUNTED,  BP2VAX3   mounted on _DXA1:

     ...copying third floppy...
```

When the contents of BP2VAX3 have been copied, you receive the prompting message:

```
Please put the fourth Basic Plus 2 kit floppy disk in the drive.

Are you ready to continue?
```

Replace diskette BP2VAX3 with BP2VAX4, and respond with Y to the query.

The following messages then appear:

```
%MOUNT-I-MOUNTED, BP2VAX4   mounted on _DXA1:

     ...copying fourth floppy...
```

These are followed by a message indicating that you must select the default precision for floating-point numbers. The text of this message is:

```
Before the compiler is rebuilt, you must answer a question about
a compiler option:  the default precision for floating point
numbers.  The normal default is SINGLE  (2-word).  If, however,
you wish the default to be double, you must answer YES to the
next question:

     Do you want DOUBLE precision as default?
```

If you type Y (or YES), the default will be double precision. If you type N (or NO), the default is single precision. After you respond to this question, you receive the messages:

```
...building compiler...

...copying files to system accounts...

...cleaning up [SYSUPD]

...Basic Plus 2 installation completed...
```

At this point, PDP-11 BASIC PLUS-2/VAX is installed, and you receive the standard completion messages described in Section 6.1.4. The time required for installing PDP-11 BASIC PLUS-2/VAX is approximately 25 to 30 minutes.

6.8 PDP-11 DATATRIEVE/VAX

The Datatrieve software kit consists of two floppy diskettes:

1. DTRVAX BIN RX01

2. DTRVAX2 BIN RX01

These diskettes are to be used in the order shown. Approximate installation time is 20 minutes.

Follow the procedure described in Section 6.1.2, with the following modifications:

1. Assign the device on which PDP-11 DATATRIEVE/VAX is to be built to the logical name LB:. For example:

   ```
   $ASSIGN DBB2:  LB:
   ```

   Note that when you invoke PDP-11 DATATRIEVE/VAX, logical name LB: is used by default. Therefore, it must be assigned to the device on which PDP-11 DATATRIEVE/VAX was built.

2. Specify LB: when you invoke the command procedure to start the installation. For example:

   ```
   $ @LB:[SYSUPD]VMSUPDATE
   ```

When directed to place the first kit floppy diskette in the console drive, replace the console floppy diskette with floppy diskette DTRVAX. Then enter Y to the query:

ARE YOU READY TO CONTINUE?

You will receive the following message:

The answer to the following question will either cause Datatrieve V1.1 to be installed or cause Datatrieve V1.1 to be patched.

Do you wish to install or patch Datatrieve? (answer I or P)

Installation is described in Section 6.8.1; patching is described in Section 6.8.2.

6.8.1 Installing PDP-11 DATATRIEVE/VAX

To install PDP-11 DATATRIEVE/VAX, enter an I in response to the query:

Do you wish to install or patch Datatrieve? (answer I or P)

Continuation of the installation is indicated by the message:

...copying first floppy...

This phase of the installation is complete when you receive the message:

Please put the second Datatrieve kit floppy disk in the drive.

Remove diskette DTRVAX from the console drive and replace it with diskette DTRVAX2. Then type Y in response to the query:

Are you ready to continue?

You will receive the following message to indicate that the installation is continuing:

...copying second floppy...

You will receive a series of messages telling you that the query dictionary and message file are being created. When this phase is complete, the PDP-11 DATATRIEVE/VAX installation verification test is run, as indicated by the message:

...running Datatrieve acceptance test...

You will receive a listing of the test at the terminal, including prompts for input. Follow the directions displayed at the terminal, and the installation verification test will confirm whether PDP-11 DATATRIEVE/VAX has been properly installed.

Completion of this phase is signaled by the messages:

...cleaning up [SYSUPD]...

...Datatrieve installation/update completed...

You will receive the completion messages described in Section 6.1.4.

6.8.2 Patching PDP-11 DATATRIEVE/VAX

To patch PDP-11 DATATRIEVE/VAX, respond by typing P to the query:

Do you wish to install or patch Datatrieve? (answer I or P)

The following message is displayed:

...copying floppy...

This is followed by the queries:

What is the name of the module to be patched?
What are the device and directory of the macro source patch file?
What is the filename of the macro source patch file?
What is the check-sum for the object module xx?
What is the check-sum for the object module yy?

(The value of xx and yy depend on your responses to the preceding queries.)

Each patch kit will include documentation indicating the correct responses. When you have finished entering patches, respond Y to the query:

Is this the last patch to be made?

If you type N, the preceding queries are repeated.

Type Y if there are no more patches to be made.

6.9 VAX-11 BLISS-32

The VAX-11 BLISS-32 software kit includes four floppy diskettes:

1. VAXBLISV1 BIN RX01
2. VAXBLISOBS1 BIN RX01
3. VAXBLISOBS2 BIN RX01
4. VAXBLISREQ BIN RX01

These are to be used in the order shown. After you have completed the steps shown in Section 6.1.2, the installation runs under the control of the command procedure copied from diskette VAXBLSV1.

Enter Y to the prompt:

ARE YOU READY TO CONTINUE?

The following message then appears:

You must edit HELP.HLP to include PRETTY.HLP and BLSCRF.HLP in the "Help is available for..." message.

To make the files PRETTY.HLP and BLSCRF.HLP accessible by use of the HELP command, you must add the following lines, in the proper alphabetical position, to the [SYSHLP]HELP.HLP file:

1. *PRETTY
   @_PRETTY
2. *BLSCRF
   @_BLSCRF

When this phase is completed, you receive the message:

Load second Bliss kit floppy [VAXBLISOBS1] in drive.

Replace diskette VAXBLISV1 in the console floppy drive with diskette VAXBLISOBS1. Type Y in response to the prompt:

Are you ready to continue?

When the second diskette has finished loading, you receive the message:

Load third Bliss kit floppy [VAXBLISOBS2] in drive.

Replace VAXBLISOBS1 with VAXBLISOBS2 in the console floppy drive, and type Y in response to the prompt:

Are you ready to continue?

When the third diskette has finished loading, you receive the message:

Load fourth Bliss kit floppy [VAXBLISREQ] in drive.

Replace VAXBLISOBS2 with VAXBLISREQ in the console floppy drive, and type Y in response to the prompt:

Are you ready to continue?

Completion of the installation is signaled by the following messages:

```
Installation Procedure Verification Test.

Running the program EZTEST should print "Hello There, Bliss-User"
on SYS$OUTPUT:
```

These should be followed by the words

```
Hello There, Bliss-User
```

on the console. If you do not receive this message, retry the installation from the beginning. If the problem persists, report it to your DIGITAL representative.

APPENDIX A

SYSTEM GENERATION PARAMETERS

This appendix lists and briefly describes all of the system generation parameters defined for VAX/VMS, in alphabetic order. Chapter 4 describes the major parameters in greater detail; these are the parameters most likely to be modified for each installation.

A.1 PARAMETER DESCRIPTIONS

ACP_BASEPRIO -- Base Priority

The ACP_BASEPRIO parameter specifies the base priority for all file ACP processes. You can alter the priority of a file ACP by using the DCL command SET PROCESS/PRIORITY=n.

ACP_DIRCACHE -- Directory Cache

The ACP_DIRCACHE parameter specifies the number of pages that each file ACP is to allocate and use for caching directory blocks.

ACP_EXTCACHE -- Extent Cache Size

Not implemented currently.

ACP_EXTLIMIT -- Extent Limit

Not implemented currently.

ACP_FIDCACHE -- File Identification Cache

Not implemented currently.

ACP_HDRCACHE -- File Header Cache

The ACP_HDRCACHE parameter specifies the number of pages that each file ACP is to allocate and use for caching file header blocks.

ACP_MAPCACHE -- Bit Map Cache

The ACP_MAPCACHE parameter specifies the number of pages that each file ACP is to allocate and use for caching bit map blocks.

ACP_MAXREAD -- Maximum Blocks to Read

The ACP_MAXREAD parameter specifies the maximum number of blocks to read for directories. This parameter does not refer to user I/O.

ACP_MULTIPLE -- Create Multiple ACPs

The ACP_MULTIPLE parameter indicates whether a separate ACP is to be created (by default) whenever a volume is mounted. If ACP_MULTIPLE is set (= 1), the default is to create a separate ACP.

Because each ACP requires its own process, having separate ACPs can be costly; the default is to share the existing ACP unless the MOUNT command specifically requests a separate ACP.

ACP_SHARE -- Enable Sharing of Initial File ACP

The ACP_SHARE parameter, when set, causes the image for the initial file ACP to be installed as a global section (that is, as shareable). Normally, ACP_SHARE is set.

It is worthwhile to disable this feature only in a configuration that has the minimum number of global sections and pages and that never has more than one ACP.

ACP_SWAPFLAGS -- Swapping Flags

The ACP_SWAPFLAGS parameter specifies the value of a 4-bit number that contains four flags to indicate whether different types of ACPs can be swapped. If the bit is set, the corresponding class of ACP can be swapped.

Bit	ACP Class
0	Disks mounted /SYSTEM
1	Disks mounted /GROUP
2	Private disks
3	Magnetic tape ACP

ACP_SYSACC -- Directory Access Cache

The ACP_SYSACC parameter specifies, for disks mounted /SYSTEM, the approximate number of directory file control blocks (FCBs) that will be cached. You can change the value of this parameter at mount time with the /ACCESSED qualifier. Typically, the ACP_SYSACC parameter should be set roughly equivalent to the number of directories that will be in use concurrently.

The ACP maintains an FCB for each open file. Directory FCB's may be cached when not in use. In addition, each directory FCB has 16 bytes associated with it and each byte contains the starting letter of the last entry in each block of the directory (or group of blocks if the directory is greater than 16 blocks). Since directories are alphabetical, these 16 bytes facilitate quick access to the correct directory block.

ACP_WINDOW -- Default Number of Window Pointers

The ACP_WINDOW parameter specifies, for disks mounted /SYSTEM, the default number of window pointers for which space is to be allocated in a window for a default file access.

ACP_WORKSET -- File ACP Working Set Size

The ACP_WORKSET parameter, if nonzero, specifies the size of the working set for each of the file ACPs. A value of zero causes each ACP to compute its proper working set size and use the resulting value.

BALSETCNT -- Number of Processes Allowed in the Balance Set

The BALSETCNT parameter determines the maximum number of processes that can have their working sets resident concurrently even if additional physical memory is available.

BALSETCNT is a major parameter; for further information, see Chapter 4.

BJOBLIM -- Batch Job Limit

The BJOBLIM parameter specifies the maximum number of batch jobs allowed. The number of processes allowed in the system may actually limit the number of batch jobs to a smaller number.

BUGCHECKFATAL -- Makes All Bugchecks Fatal

The BUGCHECKFATAL parameter, when set, causes all bugchecks to be fatal bugchecks. Fatal bugchecks interrupt operation of the system and cause it to be rebooted.

When this flag is clear, a nonfatal bugcheck causes an error log entry, but does not interrupt system operation.

BUGREBOOT -- Automatic Reboot on Bugcheck

The BUGREBOOT parameter enables or disables automatic rebooting of the system if a fatal bugcheck occurs. Normally, this feature is disabled only when the executive is being debugged.

CRDENABLE -- Corrected Read Data Error Enable

The CRDENABLE parameter, when set, enables detection and logging of memory-corrected read data errors (ECC errors). By default, CRDENABLE is set and, normally, should remain set to gather information about memory errors.

DEFMBXBUFQUO -- Default Mailbox Buffer Quota

The DEFMBXBUFQUO parameter specifies the number of bytes to be used as the default buffer quota if none is specified in a Create Mailbox system service call.

DEFMBXMXMSG -- Default Mailbox Maximum Message Size

The DEFMBXMXMSG parameter specifies the number of bytes to be used as the default value for the maximum message size if none is specified in a Create Mailbox system service call.

DEFMBXNUMMSG -- Default Mailbox Number of Messages

The DEFMBXNUMMSG parameter specifies the default value to be used for the number of messages allowed to be queued to a mailbox if the Create Mailbox system service call does not include that argument.

DIALTYPE -- Dial-up/Hang-up Algorithm Selector

The DIALTYPE parameter is set to 0 for United States style networks and to 1 for United Kingdom style networks. A United Kingdom style network requires hang-up at first carrier loss; a United States style network waits for nine consecutive scan periods without carrier before hang-up.

DUMPBUG -- Enable Dump on Fatal Bugcheck

The DUMPBUG parameter, when set, enables the writing of error log buffers and memory content to the file SYS$SYSTEM:SYSDUMP.DMP when a fatal bugcheck occurs. By default, DUMPBUG is set (enabled). Normally, this feature is disabled only when the executive is being debugged.

FREELIM -- Free Page List Lower Limit

The FREELIM parameter sets the number of pages that must be readily available as fluid memory. If the actual number of free pages falls below FREELIM, the swapper is activated to acquire additional free pages by either swapping out processes or writing modified pages.

For most systems, this value should be small but greater than zero.

The maximum process working set limit is limited to the total number of pages available after all permanently resident pages have been allocated and the values of FREELIM and SYSMWCNT have been subtracted.

GBLPAGES -- Global Pages

The GBLPAGES parameter controls the total number of global pages that can be known to the system. This parameter determines the total number of global page table entries allocated.

GBLPAGES is a major parameter; for further information, see Chapter 4.

GBLSECTIONS -- Global Sections

The GBLSECTIONS parameter establishes the number of global section descriptors to be allocated by SYSBOOT.

GBLSECTIONS is a major parameter; for further information, see Chapter 4.

IJOBLIM -- Interactive Job Limit

The IJOBLIM parameter specifies the maximum number of interactive jobs allowed. The total number of processes allowed in the system may actually limit the number of interactive jobs to a smaller number.

INTSTKPAGES -- Number of Interrupt Stack Pages

The INTSTKPAGES parameter specifies the number of pages to be allocated for the interrupt stack. Each page of the interrupt stack requires one page of physical memory.

Use of the default value is recommended unless there is a case of interrupt-stack-not-valid exceptions caused by either of the following:

- An unusually large number of devices
- A driver that requires an exceptionally large amount of stack space

IRPCOUNT -- I/O Request Packet Count

The IRPCOUNT parameter establishes the number of I/O request packets to preallocate and keep in a special list for low overhead during packet allocation and deallocation.

IRPCOUNT is a major parameter; for further information, see Chapter 4.

KFILSTCNT -- Number of Known File List Heads

The KFILSTCNT parameter specifies the number of known file list heads that can be present in the system. A known file list head is needed for each combination of device, directory, and file type for which known file entries are made using the INSTALL utility. The VAX/VMS Operator's Guide describes the INSTALL utility.

Each active list head requires 68 bytes of nonpaged pool. Each extra, never-used list head pointer requires only 4 bytes of nonpaged dynamic pool. The actual known file entries are allocated from paged dynamic pool.

MAXBUF -- Maximum Buffered I/O Transfer Size

The MAXBUF parameter limits the size of buffered I/O transfers. Buffered I/O devices include the card reader, console floppy, line printers, mailboxes, and terminals.

This parameter value should be raised above the default if larger buffers are required. However, larger buffers use additional nonpaged pool space.

MAXPRINTSYMB -- Maximum Number of Print Symbionts

The MAXPRINTSYMB parameter specifies the maximum number of print symbionts to be created by the job controller. The actual number allowed is limited by the value of PQL_DPRCLM, the default process creation limit.

MAXPROCESSCNT -- Maximum Process Count

The MAXPROCESSCNT parameter determines the maximum number of processes that can be known to the system.

MAXPROCESSCNT is a major parameter; for further information, see Chapter 4 and the description of WSMAX in this section.

MAXSYSGROUP -- Maximum System Group Number

The MAXSYSGROUP parameter establishes the highest value (in decimal) that a group number can have and still be classified as a system UIC group number for the purposes of file protection and other UIC-based protection.

MINWSCNT -- Minimum Fluid Working Set Count

The MINWSCNT parameter determines the minimum number of fluid pages required for a process to execute. Fluid pages are pages that are not locked in the working set. VAX/VMS uses MINWSCNT to establish the minimum working set size, which is a larger number because it includes the size of the process header.

The value of MINWSCNT is really limited by the number of pages required to complete any arbitrarily selected instruction. VAX-11 architecture theoretically requires 52 pages to complete the worst-case instruction in a very large address space. However, all VAX/VMS code can run with approximately 20 fluid pages in a working set. If the value of MINWSCNT is too small, a process continues to incur page faults on the same instruction for different pages but never completes the instruction. Slightly larger values allow the process to make progress but may not provide satisfactory performance.

MPW_HILIM -- Modified Page List High Limit

The MPW_HILIM parameter sets the number of pages that can be accumulated on the modified page list before initiating any modified page writing.

MPW_HILIM is a major parameter; for further information, see Chapter 4. Also refer to MPW_LOLIMIT in this section.

MPW_LOLIMIT -- Modified Page List Low Limit

The MPW_LOLIMIT parameter sets the size to which the modified page list must be reduced before the modified page writer stops.

MPW_LOLIMIT is a major parameter; for further information, see Chapter 4.

MPW_WRTCLUSTER -- Modified Page Write Cluster Size

The MPW_WRTCLUSTER parameter specifies the number of pages that the modified page writer attempts to collect and write as a contiguous disk transfer.

MPW_WRTCLUSTER is a major parameter; for further information, see Chapter 4.

NJOBLIM -- Network Job Limit

The NJOBLIM parameter specifies the maximum number of network jobs allowed. The total number of processes allowed in the system may actually limit the number of network jobs to a smaller number.

NPAGEDYN -- Number of Bytes of Nonpaged Dynamic Pool

The NPAGEDYN parameter specifies the total number of bytes to be allocated for the nonpaged dynamic pool. The value specified is rounded down to an integral number of pages.

NPAGEDYN is a major parameter; for further information, see Chapter 4.

PAGEDYN -- Number of Bytes of Paged Dynamic Pool

The PAGEDYN parameter specifies the total number of bytes to be allocated for the paged dynamic pool. The value specified is rounded down to an integral number of pages.

PAGEDYN is a major parameter; for further information, see Chapter 4.

PFCDEFAULT -- Page Fault Cluster Default

The PFCDEFAULT parameter sets the default page fault cluster size that is used for any process or global section that does not explicitly specify its own page fault cluster size, as well as for pages read from the paging file.

PFCDEFAULT is a major parameter; for further information, see Chapter 4.

POOL_PAGING -- Enable Paging of Pageable Dynamic Pool

The POOL_PAGING parameter, when set, enables the paging of the pageable dynamic pool. Normally, POOL_PAGING is set to conserve use of physical memory.

It might be desirable to disable pool paging for performance reasons or for debugging purposes. If pool paging is disabled, the paged pool is allocated and initialized in the same manner as the nonpaged pool.

PQL_DASTLM -- Default AST Limit

The PQL_DASTLM parameter specifies the default AST limit to be used.

PQL_DBIOLM -- Default Buffered I/O Limit

The PQL_DBIOLM parameter specifies the default limit on the number of buffered I/O operations allowed a created process.

PQL_DBYTLM -- Default Buffered I/O Byte Limit

The PQL_DBYTLM parameter specifies the default limit on the number of bytes a created process can have in use for buffered I/O operations.

PQL_DCPULM -- Default Central Processor Limit

The PQL_DCPULM parameter specifies the default amount of central processor time that a created process can use.

PQL_DDIOLM -- Default Direct I/O Limit

The PQL_DDIOLM parameter specifies the default limit on the number of direct I/O operations a created process can have active at one time.

PQL_DFILLM -- Default Open File Limit

The PQL_DFILLM parameter specifies the default limit on the number of files that a created process can have open concurrently.

PQL_DPRCLM -- Default Subprocess Limit

The PQL_DPRCLM parameter specifies the default limit on the number of subprocesses that a created process can create.

PQL_DTQELM -- Default Timer Queue Entries

The PQL_DTQELM parameter specifies the default limit on the number of timer queue entries that a created process can have pending.

PQL_DWSDEFAULT -- Default for Working Set Default

The PQL_DWSDEFAULT parameter specifies the default working set size for a created process.

PQL_DWSQUOTA -- Default Working Set Quota

The PQL_DWSQUOTA parameter specifies the default working set quota for a created process.

PQL_MASTLM -- Minimum AST Limit

The PQL_MASTLM parameter specifies the minimum number of pending ASTs that a created process can be allowed.

PQL_MBIOLM -- Minimum Buffered I/O Limit

The PQL_MBIOLM parameter specifies the minimum number of buffered I/O operations that a created process can be allowed.

PQL_MBYTLM -- Minimum Buffered I/O Byte Count

The PQL_MBYTLM parameter specifies the minimum number of bytes for buffered I/O operations that a created process can be allowed.

PQL_MCPULM -- Minimum Central Processor Limit

The PQL_MCPULM parameter specifies the minimum amount of central processor time that a created process can be allowed.

PQL_MDIOLM -- Minimum Direct I/O Limit

The PQL_MDIOLM parameter specifies the minimum number of direct I/O operations that a created process can be allowed.

PQL_MFILLM -- Minimum Open File Limit

The PQL_MFILLM parameter specifies the minimum number of concurrently open files that a created process can be allowed.

PQL_MPRCLM -- Minimum Subprocess Creation Limit

The PQL_MPRCLM parameter specifies the minimum number of subprocesses that a created process can be allowed to create.

PQL_MTQELM -- Minimum Timer Queue Limit

The PQL_MTQELM parameter specifies the minimum number of time queue entries that a created process can be allowed.

PQL_MWSDEFAULT -- Minimum Working Set Default

The PQL_MWSDEFAULT parameter specifies the minimum value of the default working set for a created process.

PQL_MWSQUOTA -- Minimum Working Set Quota

The PQL_MWSQUOTA parameter specifies the minimum working set quota that can be specified for a created process.

PROCSECTCNT -- Process Section Count

The PROCSECTCNT parameter reserves space in the process header for the specified number of process sections. Each process section descriptor increases the size of the fixed portion of the process header by 32 bytes.

The correct setting of this parameter is a value greater than the number of image sections in any image to be run in the system. The number of sections needed for an image to run can be obtained from the memory allocation map produced when the image was linked.

QUANTUM -- Time Quantum

The QUANTUM parameter specifies the time quantum used to divide the processor, on a round-robin basis, among processes of equal priority.

QUANTUM is a major parameter; for further information, see Chapter 4.

RMS_DFMBC -- VAX-11 RMS Default Multiblock Count

The RMS_DFMBC parameter specifies the default multiblock count for VAX-11 RMS operations.

RMS_DFMBFHSH -- VAX-11 RMS Default Multibuffer Count (Hashed)

Not implemented currently.

RMS_DMBFIDX -- VAX-11 RMS Default Multibuffer Count (Indexed)

The RMS_DMBFIDX parameter specifies the default multibuffer count for VAX-11 RMS indexed file operations.

RMS_DMBFREL -- VAX-11 RMS Default Multibuffer Count (Relative)

The RMS_DMBFREL parameter specifies the default multibuffer count for VAX-11 RMS relative file operations.

RMS_DMBFSDK -- VAX-11 RMS Default Multibuffer Count (Sequential Disk)

The RMS_DMBFSDK parameter specifies the default multibuffer count for VAX-11 RMS sequential disk operations.

RMS_DMBFSMT -- VAX-11 RMS Default Multibuffer Count (Magnetic Tape)

The RMS_DMBFSMT parameter specifies the default multibuffer count for VAX-11 RMS operations on magnetic tape.

RMS_DMBFSUR -- VAX-11 RMS Default Multibuffer Count (Unit Record)

Not implemented currently.

SBIERRENABLE -- SBI Error Interrupt Enable

The SBIERRENABLE parameter enables the SBI error interrupt to permit error logging for certain classes of SBI errors. SBI error logging should be disabled only under extraordinary debugging situations.

SETTIME -- Prompt for Time at Boot

The SETTIME parameter, when set, causes the system to solicit the time of day each time it is bootstrapped.

Normally, the system uses the time of day provided in the time-of-day processor register, unless it is not a reasonable value. To force the system to prompt for the time once and then use the normal automatic mode, use the following console command before invoking the appropriate bootstrap command file:

```
DEPOSIT/I 1B 0
```

This command clears the time-of-day register and causes the time to be solicited once. The system correctly sets the time-of-day register.

SPTREQ -- Additional System Page Table Entries Required

The SPTREQ parameter specifies the number of system page table (SPT) entries to be allocated for mapping the executive image, the VAX-11 RMS image, the system message file, SBI configuration registers and I/O adapters, and windows for I/O drivers that require buffer overmapping for ECC correction. The number of SPT entries required for all other purposes is automatically computed and added to SPTREQ to yield the actual SPT size.

The system page table entries required for each of the items listed above follow.

Component	SPT Entries Required
Executive image	170
VAX-11 RMS image	107
SYSMSG.MPF	139
Each MASSBUS adapter	8
Each memory controller	1
Each UNIBUS adapter	24

SYSMWCNT -- System Maximum Working Set Count

The SYSMWCNT parameter sets the number of pages to be used for the pageable executive code and data, VAX-11 RMS code, and the system message file.

SYSMWCNT is a major parameter; for further information, see Chapter 4.

SYSPAGING -- Enable System Code Paging

The SYSPAGING parameter, when set, allows paging of executive code. Normally, SYSPAGING is set. In a system with sufficient memory and a stringent real-time response requirement or while debugging, it might be desirable to keep all executive code resident.

Clearing SYSPAGING does not disable the paging of VAX-11 RMS code; however, if the system working set is large enough, RMS code is preloaded after it is mapped.

TTY_BUF -- Default Line Width for Terminal

The TTY_BUF parameter sets the default line width for terminal devices; SYSGEN uses this value when it creates the terminal data base.

TTY_DEFCHAR -- Default Terminal Characteristics

The TTY_DEFCHAR parameter sets the default terminal characteristics. SYSGEN uses this value when it creates the terminal data base. Useful values for common terminals are:

Terminal	Value
LA36	251658912
VT52	268440224

These values are derived as follows:

```
<Page length>*2**24 +
<If SCOPE then 1 else 0>*2**12 +
<If WRAPAROUND then 1 else 0>*2**9 +
<If TERMINAL_SYNCH then 1 else 0>*2**5 +
<If LOWER_CASE then 1 else 0>*2**7 +
<If MECH_TAB then 1 else 0>*2**8
```

TTYSCANDELTA -- Terminal Dial-up/Hang-up Scan Interval

The TTYSCANDELTA parameter sets the interval for polling terminal devices for dial-up or hang-up events. Shorter intervals use more processor time in polling; longer intervals may result in missing a hang-up.

TTY_SPEED -- Default Speed for Terminal

The TTY_SPEED parameter establishes the default speed for terminal devices; SYSGEN uses this value when it creates the terminal data base. Values for this parameter are specified as follows:

Baud Rate	Value
50	1
75	2
110	3
134.5	4
150	5
300	6
600	7
1200	8
1800	9
2000	10
2400	11
3600	12
4800	13
7200	14
9600	15

UAFALTERNATE -- Use Alternate Authorization File

The UAFALTERNATE parameter, when set, causes SYSINIT to assign SYSUAF as the logical name for SYSUAFALT. This assignment causes all references to the user authorization file (SYSUAF) to be translated to the file SYS$SYSTEM:SYSUAFALT. You can thus bring up a system for a restricted set of users. Normal use of the user authorization file can be restored by deassigning the system logical name SYSUAF.

The system manager is responsible for placing the desired content in SYSUAFALT before bringing up a system that uses it. If it does not exist, the system accepts any user name.

VIRTUALPAGECNT -- Maximum Virtual Page Count for Any Process

The VIRTUALPAGECNT parameter specifies the maximum number of virtual pages that can be mapped for any process.

VIRTUALPAGECNT is a major parameter; for further information, see Chapter 4.

WSMAX -- Maximum Process Working Set Size

The WSMAX parameter specifies the maximum number of pages allowed for any process's working set.

WSMAX is a major parameter; for further information, see Chapter 4.

A.2 SPECIAL PARAMETERS

The special parameters described below are a subset of the special parameters defined in VAX/VMS for use by DIGITAL.

PAGTBLPFC -- Page Table Fault Cluster Size

The PAGTBLPFC parameter specifies the page fault cluster size to be used when faults occur for page table pages. Conceptually, this parameter is similar to PFCDEFAULT; however, it should be set only to very small values.

PHYSICALPAGES -- Maximum Physical Pages

The PHYSICALPAGES parameter specifies the maximum number of physical pages to be used by the system. This parameter permits the testing of smaller memory configurations without actually changing the hardware configuration. This value is compared with the actual number of physical pages and the smaller of the two values is selected before any pages are allocated.

SYSPFC -- System Page Fault Cluster Size

The SYSPFC parameter specifies the page fault cluster size to be used when faults occur for pages of the system. This parameter should be kept to a small value.

APPENDIX B

MESSAGES

This appendix lists the messages issued by SYSGEN and SYSBOOT. Each message consists of a prefix followed by message text, in the form:

%component-x-text

component — Is the component issuing the message, that is, BOOT, SYSBOOT, or SYSGEN.

x — Indicates the severity level of the error:

- E (error)
- F (fatal error)
- W (warning)

text — Is the message text. Within the three sections that follow, messages are described in alphabetic order by message text.

A warning message (W) indicates that BOOT, SYSBOOT or SYSGEN has altered a user-specified parameter value because that value was not within the allowable range. An error message (E) indicates that a command contains an error or that an I/O error occurred during a command. Commands in which an error occurs have no effect. A fatal error message (F) indicates that the current attempt to boot the system has been terminated.

B.1 BOOT MESSAGES

%BOOT-F-Bootfile not contiguous

Explanation: The file [SYSEXE]SYSBOOT.EXE was located but is not contiguous.

User Action: The disk you are attempting to boot cannot be booted. Obtain another copy of the system disk.

%BOOT-F-I/O error reading boot file

Explanation: An uncorrectable read error occurred while the file [SYSEXE]SYSBOOT.EXE was being read.

User Action: Try booting the disk again. If subsequent attempts fail, obtain another copy of the system disk.

%BOOT-F-Nonexistent drive

Explanation: The specified drive number does not exist.

User Action: Specify an appropriate device unit.

%BOOT-F-Unable to locate boot file

Explanation: The file [SYSEXE]SYSBOOT.EXE could not be found.

User Action: You are attempting to boot a volume that does not contain a VAX/VMS binary system. Attempt to boot again using an appropriate disk volume.

%BOOT-F-Unexpected Exception

Explanation: An unexpected exception occurred while the primary bootstrap (VMB.EXE) was being executed. This condition probably indicates a corrupted SYSBOOT.EXE file or a hardware failure.

User Action: Determine the source of the error, and either obtain a new copy of the system disk or call your local field service engineer.

%BOOT-F-Unexpected Machine Check

Explanation: An unexpected machine check occurred while the primary bootstrap (VMB.EXE) was being executed. This condition probably indicates a corrupted SYSBOOT.EXE file or a hardware failure.

User Action: Determine the source of the error, and either obtain a new copy of the system disk or call your local field service engineer.

B.2 SYSBOOT MESSAGES

%SYSBOOT-E-File not contiguous

Explanation: The file specified by a USE command is not contiguous.

User Action: Create a contiguous parameter file.

%SYSBOOT-E-I/O error reading file

Explanation: An unrecoverable I/O error occurred while SYSBOOT was reading a parameter file or the system image. If the error occurred while a parameter file was being read, SYSBOOT does not perform the requested action. If the error occurred while the system image was being read, SYSBOOT terminates the boot operation.

User Action: Attempt to reboot the system or try a different drive.

%SYSBOOT-W-Maximum WS raised to PHD+MINWSCNT

Explanation: SYSBOOT has raised the maximum working set size you specified, to accommodate the minimum working set size allowed by VAX/VMS. To execute, every process requires a minimum fluid working set (MINWSCNT) plus space for its process header (PHD).

User Action: None.

%SYSBOOT-E-No such parameter

Explanation: A parameter name specified in a SET or SHOW command is not a recognized parameter name.

User Action: Reissue the command specifying correct parameter names.

%SYSBOOT-E-Syntax error

Explanation: You issued a command that was syntactically incorrect or you executed the SHOW /DEVICES command without the Change Mode to Executive privilege.

User Action: Reissue the command using proper syntax or contact your system manager to obtain the Change Mode to Executive privilege and reissue the command.

%SYSBOOT-F-Unable to allocate physical memory

Explanation: The sum of pages required for the system page table, nonpaged dynamic pool, interrupt stack, and resident executive exceeds available physical memory.

User Action: Reconfigure the system to require less physical memory and reboot the system. Reduce one or more parameters controlling the allocation of physical memory: NPAGEDYN, BALSETCNT, and VIRTUALPAGECNT.

%SYSBOOT-F-Unable to allocate SPT

Explanation: SYSBOOT cannot find enough contiguous pages of physical memory to contain the required system page table (SPT).

User Action: Reduce parameters controlling SPT size: VIRTUALPAGECNT, BALSETCNT, and SPTREQ.

%SYSBOOT-F-Unable to locate driver for boot device

Explanation: SYSBOOT cannot locate the driver image file for the boot device in directory [SYSEXE] on the volume to be booted.

User Action: Put the correct driver image file in the directory SYSEXE.

Device Type	Driver Name
RK06/RK07	DMDRIVER.EXE
RM03	DRDRIVER.EXE
RP05/RP06	DBDRIVER.EXE
Other	UNKDRIVER.EXE

%SYSBOOT-E-Unable to locate file

Explanation: SYSBOOT cannot locate a file specified in a USE command on the volume to be booted.

User Action: Reissue the command specifying the correct file name.

%SYSBOOT-F-Unable to locate SYS.EXE

Explanation: SYSBOOT cannot locate the system image file in directory [SYSEXE] on the boot volume.

User Action: The system disk is either defective or corrupted. Obtain a new copy of the VAX/VMS system disk.

%SYSBOOT-F-Unexpected exception

Explanation: An unexpected exception has occurred indicating a probable hardware error or SYSBOOT logic error.

User Action: Call your local field service engineer.

%SYSBOOT-F-Unexpected machine check

Explanation: An unexpected machine check occurred indicating a probable hardware error or SYSBOOT logic error.

User Action: Call your local field service engineer.

%SYSBOOT-W-Value set to maximum

Explanation: You attempted to set a parameter to a value greater than the maximum allowable value. SYSBOOT has set the parameter to the maximum permissible value.

User Action: None.

%SYSBOOT-W-Value set to minimum

Explanation: You attempted to set a parameter to a value that is less than the allowable minimum value. SYSBOOT has set the parameter to the minimum permissible value.

User Action: None.

%SYSBOOT-W-WS default and quota raised to PDH+MINWSCNT

Explanation: SYSBOOT has raised the default working set size you specified to accommodate the sizes of the process header plus the minimum fluid working set size.

User Action: None.

B.3 SYSGEN MESSAGES

%SYSGEN-E-DEVACTIVE - device is active

Explanation: You attempted to reload a driver that had I/O in progress.

User Action: Stop all I/O operations on the device and reissue the RELOAD command.

%SYSGEN-E-Error reading parameter file

Explanation: An unrecoverable I/O error occurred while SYSGEN was reading a parameter file.

User Action: Reissue the command.

%SYSGEN-E-FNF - file not found

Explanation: SYSGEN was unable to locate the driver image file. The default directory and disk device for driver images is SYS$SYSTEM. This message is also produced during an AUTOCONFIGURE operation if a UNIBUS device does not have a corresponding driver in SYS$SYSTEM.

%SYSGEN-F-INSFMEM - insufficient dynamic memory

Explanation: There is not enough room in the nonpaged pool to load a driver or create a control block for it.

User Action: Reboot the system, stopping in SYSBOOT, and increase the value of the parameter NPAGEDYN.

%SYSGEN-E-INVDPT - invalid driver image - DPT$_TYPE, VBN 2 .NEQ. DYN$C_DPT

Explanation: The driver image file specified in a LOAD, RELOAD, or CONNECT command did not pass a basic validity check. This error can be caused by the following conditions:

- Not having the driver prologue table (DPT) as the first code-generating component of the driver
- Not linking the driver as an executable image based at 0
- Having PSECT statements in the driver program
- Any other problem that causes the driver prologue table not to be in the second block (VBN 2) of the image file

User Action: Correct the driver code and reissue the command.

%SYSGEN-E-INVDPTINI - invalid driver prologue init table value

Explanation: One of the following conditions exists:

- The initialization or reinitialization entry in the driver prologue table has an invalid control block type code; that is, other than DDB, UCD, CRB, or IDB

- The initialization or reinitialization entry in the driver prologue table has an invalid store function; that is, other than B, W, L, R, V, @B, @W, @L, @R, or @V

User Action: Correct the DPT_STORE macro in the driver prologue table.

%SYSGEN-E-INVVEC - invalid or unspecified interrupt vector

Explanation: Either no interrupt vector was specified in a CONNECT commmand, or the one that was specified is not within the UNIBUS interrupt vector address range.

User Action: Reissue the command specifying a valid vector number.

%SYSGEN-E-No adapter TR number specified

Explanation: An adapter TR number must be specified in a CONNECT command.

User Action: Reissue the command specifying the /ADAPTER qualifier.

%SYSGEN-F-NOPRIV - No privilege for attempted operation

Explanation: You are attempting to perform an operation that requires Change Mode to Kernel or Executive privilege.

User Action: Ask the system manager to give you the privilege.

%SYSGEN-E-No such parameter

Explanation: The parameter name specified in a SET or SHOW command is not valid.

User Action: Reissue the command specifying a valid parameter name.

%SYSGEN-E-NOUNLOAD - driver is not unloadable

Explanation: You attempted to reload a driver that is marked unloadable (DPT$M_NOUNLOAD) in the driver prologue table, or the unload routine of the driver indicates that the driver should not be unloaded.

User Action: None.

%SYSGEN-E-SPTFULL - system page table is full

Explanation: No more room is available in the system page table to allocate a system page for the device (requested by DPT$M_SVP in the driver prologue table).

User Action: Reboot the system, stopping in SYSBOOT, and increase the value of the system page table parameter (SPTREQ).

%SYSGEN-E-Syntax error

Explanation: The command does not follow the correct syntax rules.

User Action: Reissue the command using the proper syntax.

%SYSGEN-E-Unable to create file

Explanation: The file specified in a CREATE command cannot be created for one of the following reasons:

- The device or directory does not exist
- The volume is write-locked or the user does not have write access to it
- The volume is full

User Action: Correct the situation and reissue the command.

%SYSGEN-E-Unable to create parameter file

Explanation: The file specified in a WRITE command cannot be created for one of the following reasons:

- The device or directory does not exist
- The volume is write-locked or the user does not have write access to it
- The volume is full

User Action: Correct the situation and reissue the command.

%SYSGEN-E-Unable to open file

Explanation: The file specified in a USE command cannot be found or opened.

User Action: Reissue the command specifying the correct file specification.

%SYSGEN-W-Value set to maximum

Explanation: You issued a SET command to set a parameter to a value that is greater than the allowable maximum. SYSGEN reduces the value to the permissible maximum.

User Action: None.

%SYSGEN-W-Value set to minimum

Explanation: You issued a SET command to set a parameter to a value that is less than the allowable minimum. SYSGEN increases the value to the permissible minimum.

User Action: None.

%SYSGEN-E-VECINUSE - vector in use

Explanation: The interrupt vector address that was specified in a CONNECT command is already being used by another device.

User Action: Change the vector address in the device driver and reissue the CONNECT command.

See 3.0 Rel. 2-46

APPENDIX C

FILES OF THE VAX/VMS SYSTEM

This appendix contains the names and brief descriptions of the files provided by DIGITAL on the VAX/VMS system binary distribution medium. The files on this medium are cataloged in seven directories; two other directories on the medium are provided for later use by the system. The names of all nine directories and descriptions of their contents follow.

1. [SYSERR]

   This directory is reserved for the error log file (ERRLOG.SYS).

2. [SYSEXE]

   As shown in Table C-1, this directory contains commonly used executable images of the VAX/VMS operating system and installation.

3. [SYSHLP]

   As shown in Table C-2, this directory contains text files for the HELP utility.

4. [SYSLIB]

   As shown in Table C-3, this directory contains various macro and object libraries as well as other files used by reference.

5. [SYSMAINT]

   This directory is reserved for system hardware diagnostic programs.

6. [SYSMGR]

   As shown in Table C-4, this directory contains files used in managing the operating system. This directory is the default directory of the system manager's account.

7. [SYSMSG]

   As shown in Table C-5, this directory contains system message text files.

8. [SYSTEST]

   As shown in Table C-6, this directory contains files used in testing the functions of the operating system.

9. [SYSUPD]

As shown in Table C-7, this directory contains files used in applying system updates.

Table C-1
Files Contained in Directory [SYSEXE]

File Name	Description
16USER.PAR	VAX/VMS system parameter file
32USER.PAR	VAX/VMS system parameter file
48USER.PAR	VAX/VMS system parameter file
64USER.PAR	VAX/VMS system parameter file
8USER.PAR	VAX/VMS system parameter file
ANALYZ.EXE*	VAX/VMS object module analyzer
AUTHORIZE.EXE	User authorization utility
BACKTRANS.EXE	Back translator of DCL into MCR commands
BAD.EXE	Bad block disk utility
BADBLOCK.EXE	Dynamic bad blocks utility
BCK.EXE	RMS-11 back-up utility
CANCEL.EXE	CANCEL command
CHMK.EXE*	Operating system debugging facility
CNV.EXE	RMS-11 file conversion utility
COPY.EXE	File copying utility
CRDRIVER.EXE	Card reader driver
DBDRIVER.EXE	RP05 and RP06 disk pack driver
DCL.EXE	Command language interpreter
DEF.EXE	RMS-11 interactive file definition utility
DELETE.EXE	File deletion utility
DFN.EXE	RMS-11 noninteractive file definition utility
DIF.EXE	File compare utility
DISMOUNT.EXE	Device dismount utility
DISPLAY.EXE	Utility that displays system performance statistics
DMDRIVER.EXE	RK07 disk cartridge driver
DMP.EXE	File dump utility
DRDRIVER.EXE	RM03 disk pack driver
DSC1.EXE	Files-11 Structure Level 1 disk back-up facility
DSC2.EXE	Files-11 Structure Level 2 disk back-up facility
DSP.EXE	RMS-11 file attribute display utility
EDI.EXE	RSX-11M text editor
EDT.EXE	Text editor
ERRFMT.EXE	Error logging facility
F11AACP.EXE	Files-11 Structure Level 1 ancillary control process image
F11BACP.EXE	Files-11 Structure Level 2 ancillary control process image
FLX.EXE	File exchange utility
HELP.EXE	Help utility
HEXZAP.EXE*	Hexadecimal image patching utility
INIT.EXE	Disk device initialization utility
INPSMB.EXE	Input symbiont

* Not supported by DIGITAL.

(continued on next page)

Table C-1 (Cont.)
Files Contained in Directory [SYSEXE]

File Name	Description
INSTALL.EXE	Utility that installs known images
JBCSYSQUE.DAT	Data file for queuing facility
JBCSYSQUE.EXE	Queuing facility
JOBCTL.EXE	Job controller
LADRIVER.EXE	LPA-11 driver
LALOAD.EXE	Accepts commands from or sends requests to LALOADER to load LPA-11 microcode
LALOADER.EXE	Loads LPA-11 microcode upon power recovery or upon request from LALOAD
LBR.EXE	RSX-11M librarian
LIB.EXE	VAX/VMS librarian
LINK.EXE	Linker
LOGINOUT.EXE	Login/logout utility
LPDRIVER.EXE	Line printer driver
MAC.EXE	MACRO-11 assembler
MACRO32.EXE	VAX/VMS assembler
MAR.EXE	Compatibility mode cross assembler
MCR.EXE	MCR command interpreter
MINIMUM.PAR	VAX/VMS system parameter file for minimum hardware configuration
MTAAACP.EXE	Magnetic tape ancillary control process image
NOTICE.TXT	ASCII file that can contain announcements to system users
OPCCRASH.EXE	System shutdown utility
OPCOM.EXE	Operator communications utility
PAGEFILE.SYS	System paging file
PAT.EXE	RSX-11M object module patch utility
PATCH.EXE	VAX/VMS image file patch utility
PAX.EXE*	VAX/VMS object module patch utility
PIP.EXE	RSX-11M peripheral interchange utility
PRTSMB.EXE	Print symbiont
QUEMAN.EXE	Queue managing facility
REPLY.EXE	Message broadcasting facility
REQUEST.EXE	Operator request facility
RMS.EXE	Record management services image
RMS.MAP	Release 1 Link map for RMS.EXE
RMSSHARE.EXE	File sharing utility program
RNO.EXE*	Text formatting utility
RST.EXE	RMS-11 file restoration utility
RSX.EXE	RSX-11M application migration executive main program
RTB.EXE	Utility that writes an RT-11 bootstrap on disk
RUNDET.EXE	Facility that runs detached images
SDA.EXE*	System dump analyzer program
SET.EXE	SET command processor
SHOW.EXE	SHOW command processor
SHUTDOWN.COM	System shutdown command file
SLP.EXE	RSX-11M source file editing utility
SORT32.EXE	SORT command processor
SOS.EXE	SOS text editor
SRT.EXE	Sort-11 utility

* Not supported by DIGITAL.

(continued on next page)

Table C-1 (Cont.)
Files Contained in Directory [SYSEXE]

File Name	Description
STARTUP.COM	System start-up command file
STARTUP.MIN	Alternate system start-up command file for system with minimum hardware configuration
SUBMIT.EXE	Batch job submission facility
SWAPFILE.SYS	System swap file
SYE.EXE	Utility that formats the error log file
SYS.EXE	Operating system image file
SYS.MAP	Map of the operating system image
SYS.STB	Global symbol table of operating system
SYSBOOT.EXE	System bootstrap utility
SYSDUMP.DMP	Crash dump of system image
SYSGEN.EXE	System generation and configuration utility
SYSINIT.EXE	Operating system initialization image
SYSUAF.DAT	User authorization data file
SYSUAF.RT1	Release 1 supplied SYSUAF file
TEC.EXE*	TECO text editor
TKB.EXE	RSX-11M task builder
TMDRIVER.EXE	Magnetic tape driver
TYPE.EXE	TYPE command processor
UFD.EXE	Utility that creates user directories
VFY1.EXE	Files-11 Structure Level 1 file structure verification utility
VFY2.EXE	Files-11 Structure Level 2 file structure verification utility
VIRT32MB.PAR	VAX/VMS system parameter file that supports system with maximum virtual address space
VMOUNT.EXE	Volume mount utility
XMDRIVER.EXE	DMC-11 Synchronous Communications Line Interface driver
ZAP.EXE	RSX-11M task file octal patch utility

* Not supported by DIGITAL.

Table C-2
Files Contained in Directory [SYSHLP]

File Name	Description
APPEND.HLP	Help file for APPEND command
ASSIGN.HLP	Help file for ASSIGN command
BLISS.HLP	Help file for BLISS command
CANCEL.HLP	Help file for CANCEL command
CLOSE.HLP	Help file for CLOSE command
COBOL.HLP	Help file for COBOL compiler
COBOLC74.HLP	Help file for VAX-11 COBOL-74 compiler
COBOLRSX.HLP	Help file for PDP-11 COBOL-74/VAX compiler
COPY.HLP	Help file for COPY command
CREATE.HLP	Help file for CREATE command
DEALLOCAT.HLP	Help file for DEALLOCATE command
DEASSIGN.HLP	Help file for DEASSIGN command
DECK.HLP	Help file for DECK command

(continued on next page)

Table C-2 (Cont.)
Files Contained in Directory [SYSHLP]

File Name	Description
DEFINE.HLP	Help file for DEFINE command
DELESYMBO.HLP	Help file for DELETE/SYMBOL command
DELETE.HLP	Help file for file deletion command
DEPOSIT.HLP	Help file for DEPOSIT command
DIFFERENC.HLP	Help file for file compare utility
DIRECTORY.HLP	Help file for DIRECTORY command
DISMOUNT.HLP	Help file for DISMOUNT command
DUMP.HLP	Help file for file dumping utility
EDIT.HLP	Help file for text editors SOS and SLP
EDITSLP.HLP	Help file for EDIT/SLP command
EDITSOS.HLP	Help file for EDIT/SOS command
ERROR.HLP	Run-Time Library error codes
EXAMINE.HLP	Help file for EXAMINE command
EXPRESSIO.HLP	Rules for forming an expression
FORTRAN.HLP	Help file for VAX-11 FORTRAN IV-PLUS compiler
HELP.HLP	Help file for HELP, ALLOCATE, BASIC, CONTINUE, DEBUG, DISMOUNT, EOD, EOJ, EXIT, GOTO, MCR, ON, PASSWORD, and UNLOCK commands
IF.HLP	Help file for IF command
INITIALIZ.HLP	Help file for device initialization utility
INQUIRE.HLP	Help file for INQUIRE command
JOB.HLP	Help file for JOB command
LEXICAL.HLP	Descriptions of the lexical functions of the command language
LIBRARY.HLP	Help file for library utility
LINK.HLP	Help file for LINK command
LINKRSX11.HLP	Help file for RSX-11M link command
LOGIN.HLP	Help file for login procedure
LOGOUT.HLP	Help file for logout procedure
MACRO.HLP	Help file for VAX-11 MACRO assembler
MOUNT.HLP	Help file for MOUNT command
OPEN.HLP	Help file for OPEN command
PRINT.HLP	Help file for PRINT command
PRIVILEGE.HLP	Description of privileges
PROCEDURE.HLP	Explanation of command procedures
PURGE.HLP	Help file for file purging command
READ.HLP	Help file for READ command
RENAME.HLP	Help file for RENAME command
REQUEST.HLP	Help file for REQUEST command
RUN.HLP	Help file for RUN command
RUNIMAGE.HLP	Explanation of how to run an image
RUNPROCES.HLP	Explanation of how to run a process
SET.HLP	Help file for SET command: DEFAULT, ON, PASSWORD, and VERIFY options
SETCARDRE.HLP	Help file for SET CARD_READER command
SETMAGTAP.HLP	Help file for SET MAGTAPE command
SETPROCES.HLP	Help file for SET PROCESS command
SETPROTEC.HLP	Help file for SET PROTECTION command
SETQUEUE.HLP	Help file for SET QUEUE command
SETRMSDEF.HLP	Help file for SET RMS_DEFAULT command
SETTERMIN.HLP	Help file for SET TERMINAL command
SETWORKIN.HLP	Help file for SET WORKING_SET command
SHODEVICE.HLP	Help file for SHOW DEVICES command
SHOLOGICA.HLP	Help file for SHOW LOGICAL command

(continued on next page)

Table C-2 (Cont.)
Files Contained in Directory [SYSHLP]

File Name	Description
SHOPROCES.HLP	Help file for SHOW PROCESS command
SHOQUEUE.HLP	Help file for SHOW QUEUE command
SHOSYMBOL.HLP	Help file for SHOW SYMBOLS command
SHOW.HLP	Help file for SHOW command: DAYTIME, DEFAULT, NETWORK, PROTECTION, STATUS, SYSTEM, TERMINAL, TRANSLATION, and WORKING_SET options
SORT.HLP	Help file for sort utility
SORTRSX11.HLP	Help file for RSX-11M compatibility mode sort utility
SPECIFY.HLP	Specification rules for absolute time, delta time, protection codes, privileges
STOP.HLP	Help file for STOP command
SUBMIT.HLP	Help file for SUBMIT command
SYMBOLS.HLP	Description of symbolic names
SYNCHRONI.HLP	Help file for SYNCHRONIZE command
TYPE.HLP	Help file for TYPE command
WRITE.HLP	Help file for WRITE command

Table C-3
Files Contained in Directory [SYSLIB]

File Name	Description
CRFSHR.EXE	Shareable image for CREF
DEBUG.EXE	VAX/VMS debugging facility
DEBUG.OBJ	VAX/VMS debugger bootstrap
DELTA.OBJ	Alternate VAX/VMS debugging tool
FORDEF.FOR	FORTRAN program utility INCLUDE files
LIB.MLB	Operating system macro library
LIBDEF.FOR	FORTRAN program utility INCLUDE files
MTHDEF.FOR	FORTRAN program utility INCLUDE files
ODT.OBJ	RSX-11M debugging tool
RMS11.ODL	RMS-11 sample overlay description
RMSLIB.OLB	RMS-11 object library
RMSMAC.MLB	RMS-11 macro library
RMSRES.ODL	RMS-11 shared resident library sample build file
RMSVECTOR.OBJ	Reserved for future use
RSXMAC.SML	RSX-11M compatibility mode macro library
RSXSHR.EXE	Shareable image of RSX-11M application migration executive
SIGDEF.FOR	FORTRAN program utility INCLUDE files
STARLET.MLB	System macro library
STARLET.OLB	System object library and Run-Time Library
SYSLIB.OLB	RSX-11M object library
TRACE.EXE	VAX/VMS error traceback facility
VMLIB.OLB	RSX-11M work file support routines
VMSRTL.EXE	Shareable image of Run-Time Library
VMSRTL.MAP	Release 1 Link map of VMSRTL.EXE
VT52.TEC	TECO macro that provides a keypad-scope TECO editor

Table C-4
Files Contained in Directory [SYSMGR]

File Name	Description
FORMSTYPE.DAT	Line printer forms description for print symbiont
LPA11MREG.COM	Command file to change LPA11 map register preallocation
LPA11STRT.COM	LPA11 site-specific start-up command file
SYSHUTDWN.COM	Site-specific system shutdown command file
SYSTARTUP.COM	Site-specific system start-up command file

Table C-5
Files Contained in Directory [SYSMSG]

File Name	Description
EDTCOM.MSG	EDT editor message file
QIOSYM.MSG	RSX-11M compatibility mode QIO message file
SOS.HLP	Help file for SOS text editor
SYSMSG.MPF	System message file

Table C-6
Files Contained in Directory [SYSTEST]

File Name	Description
APPEND.PIP	Tests PIP append function
BLISUB.OBJ	BLISS compiled code to call SORT test
COBSUB.OBJ	COBOL compiled code to call SORT test
DMPASRC.DMP	Master file to test ASCII mode in DMP
DMPBYOC.DMP	Master file to test byte octal format in DMP
DMPDATA.DMP	Good input file for DMP test
DMPDCWD.DMP	Master file to test decimal word format in DMP
FORSUB.OBJ	FORTRAN compiled code to call SORT test
GLOBALS1.COM	Command procedure that contains global symbols
GLOBALS2.COM	Command procedure that contains global symbols
ISAM.IDX	Empty ISAM file for SORT output
MCLBR1.MAC	Macro to test LBR utility
MCLBR2.MAC	Macro to test LBR utility
MCLBR3.MAC	Macro to test LBR utility
OBJLBR1.OBJ	Object module to test LBR utility
OBJLBR2.OBJ	Object module to test LBR utility
OBJLBR3.OBJ	Object module to test LBR utility
PATCHED.PAT	Tests PAT utility
PATCHPAT.OBJ	Patch file to test PAT utility
PIPDATA.PIP	Good input file for PIP test
R010SQ.DAT	Input data file for SORT test
RANDOM.FLX	Tests FLX utility
SATSSF01.EXE	Tests for event flag services and $SETEXV service

(continued on next page)

Table C-6 (Cont.)
Files Contained in Directory [SYSTEST]

File Name	Description
SATSSF02.EXE	Tests for event flag services
SATSSF03.EXE	Tests for logical name services
SATSSF04.EXE	Tests for time services
SATSSF05.EXE	Tests for process control services
SATSSF06.EXE	Tests for process control services
SATSSF07.EXE	Tests for process control services
SATSSF08.EXE	Tests for change mode services
SATSSF09.EXE	Tests for I/O services
SATSSF10.EXE	Tests for send message services
SATSSF11.EXE	Tests for I/O services
SATSSF12.EXE	Tests for memory management services
SATSSF13.EXE	Tests for memory management services
SATSSF14.EXE	Tests for memory management services
SATSSF15.EXE	Tests for handler services
SATSSF16.EXE	Tests for $FAO services
SATSSS01.EXE	Tests for I/O services
SATSSS05.EXE	Tests for send message services
SATSSS07.EXE	Tests for $CREMBX, $DELMBX services
SATSSS08.EXE	Tests for $BRDCST service
SATSSS09.EXE	Tests for $FAO, $GETJPI, $GETMSG services
SATSSS22.EXE	Tests for condition handling services
SATSSS26.EXE	Tests for timer and AST services
SATSSS30.EXE	Tests for logical name services
SATSSS35.EXE	Tests for $CREPRC service
SATSSS36.EXE	Tests for $DELPRC service
SATSSS37.EXE	Tests for $SUSPND service
SATSSS38.EXE	Tests for $RESUME service
SATSSS39.EXE	Tests for $HIBER service
SATSSS40.EXE	Tests for $WAKE service
SATSSS41.EXE	Tests for $EXIT service
SATSSS42.EXE	Tests for $FORCEX service
SATSSS43.EXE	Tests for exit and change mode handling services
SATSSS44.EXE	Tests for $SETPRN service
SATSSS45.EXE	Tests for $SETPRI service
SATSSS46.EXE	Tests for $SETRWM service
SATSSS50.EXE	Tests for $ASCEFC, $DACEFC services
SATSSS52.EXE	Tests for $DLCEFC service
SATSSS53.EXE	Tests for $SETEF service
SATSSS54.EXE	Tests for $CLREF service
SATSSS55.EXE	Tests for $READEF service
SATSSS56.EXE	Tests for wait services
SATSSS60.EXE	Tests for time conversion services
SATSSS61.EXE	Tests for $SCHDWK, $CANWAK services
SATSSS70.EXE	Tests for $EXPREG service
SATSSS71.EXE	Tests for $CNTREG service
SATSSS72.EXE	Tests for $CRETVA service
SATSSS73.EXE	Tests for $DELTVA service
SATSSS74.EXE	Tests for global section services
SATSSS78.EXE	Tests for $LKWSET, $ULWSET services
SATSSS79.EXE	Tests for $LCKPAG, $ULKPAG services
SATSSS81.EXE	Tests for $ADJWSL service
SATSSS82.EXE	Tests for $SETPRT service
SATSSS83.EXE	Tests for $SETSWM service
SATSSS90.EXE	Tests for change mode services

(continued on next page)

Table C-6 (Cont.)
Files Contained in Directory [SYSTEST]

File Name	Description
SATSSS91.EXE	Tests for $ADJSTK service
SATSUT01.EXE	Utility module for SATSSF05
SATSUT02.EXE	Utility module for SATSSF06
SATSUT03.EXE	Utility module for SATSSF07
SATSUT04.EXE	Utility module for SATSSS50
SATSUT05.EXE	Utility module for SATSSS38
SATSUT06.EXE	Utility module for SATSSS40
SATSUT07.EXE	Utility module for SATSSS61
SATSUT08.EXE	Utility module for SATSSS37, SATSSS45
SATSUT09.EXE	Utility module for SATSSS41
SATSUT10.EXE	Utility module for SATSSS52
SATSUT11.EXE	Utility module for SATSSS56
SATSUT12.EXE	Utility module for abort feature
SATSUT13.EXE	Utility module for SATSSS42
SATSUT14.EXE	Utility module for SATSSS36
SCAN.COM	Command procedure that scans service list
SETEXEC.COM	Command procedure that executes test modules
SLPDATA.SLP	Input data to test SLP utility
SLPLIST.SLP	Good output file for SLP test
SLPOUT.SLP	Good output file for SLP test
SORTED.FLX	Tests FLX utility
SORTIN.DAT	Input data file for COBOL SORT
SORTUETP.COM	Master script for VAX-11 SORT test
SPECFILE.TST	File specs for SORT test
SSHELP.COM	Command procedure that contains help information for system service tests
SSTEST.COM	Master command procedure for system service tests
TST01A.EXE	Test for QIO, QIOW (11M directive)
TST01B.EXE	Test for QIO, QIOW (11M directive)
TST02A.EXE	Test for event flags (11M directive)
TST03A.EXE	Test for GET TIME, MARK TIME, RUN (11M directive)
TST03B.EXE	Test for GET TIME, MARK TIME, RUN (11M directive)
TST06A.EXE	Test for MCR command line, specify SST vector table (11M directive)
TST07A.EXE	Test for AST services (11M directive)
TST10A.EXE	Test for SEND DATA, RECEIVE DATA (11M directive)**Currently Disabled**
TST10B.EXE	Test for SEND DATA, RECEIVE DATA (11M directive)**Currently Disabled**
TST11A.EXE	Test for SUSPEND, RESUME (11M directive)
TST11B.EXE	Test for SUSPEND, RESUME (11M directive)
TST12A.EXE	Test for ABORT, EXIT-IF (11M directive)
TST12B.EXE	Test for ABORT, EXIT-IF (11M directive)
TST15A.EXE	Test for ASSIGN LUN, GET LUN (11M directive)
TST16A.EXE	Test for CANCEL SCHEDULED requests (11M directive)
TST16B.EXE	Test for CANCEL SCHEDULED requests (11M directive)
TST16C.EXE	Test for CANCEL SCHEDULED requests (11M directive)
TST17A.EXE	Test for CANCEL MARK TIME request (11M directive)

(continued on next page)

Table C-6 (Cont.)
Files Contained in Directory [SYSTEST]

File Name	Description
TST17B.EXE	Test for CANCEL MARK TIME request (11M directive)
TST20A.EXE	Test for GET TASK PARAMS, REQUEST, RUN (11M directive)
TST20B.EXE	Test for GET TASK PARAMS, REQUEST, RUN (11M directive)
UETCOMP00.COM	Main script for compatibility mode utility tests
UETCOMP03.COM	Main script for RSX11M executive directive test
UETCSOS01.DAT	Data file for SOS test
UETCSOS02.CMD	Commands for SOS test
UETCSOS03.DAT	Known good data after SOS editing
UETDISK00.COM	Logs in and runs the disk test
UETDISK00.EXE	Disk device test for RP and RK drives
UETFORT00.COM	Control file for FORTRAN test
UETFORT01.DAT	FORTRAN data file used by UETFORT01
UETFORT01.FOR	FORTRAN source for FORTRAN test
UETFORT02.FOR	FORTRAN source for FORTRAN test
UETFORT97.EXE	Compiled version of UETFORT01 for load test
UETFORT98.EXE	Compiled version of UETFORT02 for load test
UETFORT99.EXE	Compiled program for load test
UETINIT00.EXE	Gets VAX/VMS configuration and builds UETINIDEV.DAT
UETINIT01.EXE	Quick checks all devices for testability
UETLANG00.COM	Unbundled languages user inquiry script
UETLOAD01.EXE	Controls load test with various numbers of users
UETLOAD02.COM	User script for load test
UETLOAD03.COM	User script for load test
UETLOAD04.COM	User script for load test
UETLOAD05.COM	User script for load test
UETLOAD06.COM	User script for load test
UETLOAD07.COM	User script for load test
UETLOAD08.COM	User script for load test
UETLOAD09.COM	User script for load test
UETLOAD10.COM	User script for load test
UETLOAD11.COM	User script for load test
UETMEMY01.EXE	Tests high page faulting rates in load test
UETNATV00.COM	Script that runs VAX/VMS system service tests
UETNATV01.EXE	Creates a process to run VAX/VMS system service tests
UETNATV02.COM	Main script for native-mode utility tests
UETNRMS00.COM	Main script for VAX-11 RMS tests
UETNRMS01.EXE	Test program to exercise RMS functions
UETP.COM	Main command file for entire UETP
UETPDEV01.EXE	Creates detached process to run I/O device tests
UETPRIN00.COM	Logs printer test
UETPRIN00.EXE	Tests printer
UETTAPE00.COM	Logs magnetic tape test, including MOUNT and DISMOUNT commands

(continued on next page)

Table C-6 (Cont.)
Files Contained in Directory [SYSTEST]

File Name	Description
UETTAPE00.EXE	Tests magnetic tapes on one controller
UETTAPE01.EXE	Creates logical names used for mounting magnetic tapes
UETTAPE02.COM	Delivers exit status from tape test to UETPDEV01
UETTTYS00.COM	Logs in and runs terminal test
UETTTYS00.EXE	Tests terminals for one controller

Table C-7
Files Contained in Directory [SYSUPD]

File Name	Description
BOOTUPD.COM	Command file that updates VMS bootstrap file on console floppy diskette
CONSCOPY.COM	Command file that copies console floppy diskette
DXCOPY.COM	Command file that copies files from console floppy diskette and restores files to floppy diskette
LINEPAGE.COM	Command file that controls printer lines/page for system utilities
SETDEFBOO.COM	Command file that sets default boot command file
SWAPFILES.COM	Command file that creates swapping, paging, and dump files of appropriate size for system being installed
VMSKITBLD.COM	Command file that builds VAX/VMS distribution disk
VMSKITCPY.COM	Command file that copies VAX/VMS distribution disk
VMSUPDATE.COM	System update command file

APPENDIX D

BOOTSTRAP COMMAND FILES

This appendix contains samples of the command files used to bootstrap the system.

```
   50   !
   55   !        DB3 CONVERSATIONAL BOOT COMMAND FILE - DB3GEN.
   57   !        BOOT FROM DB3 AND STOP IN SYSBOOT TO ALTER PARAMETERS
   60   !
  100   HALT                      ! HALT PROCESSOR
  200   UNJAM                     ! UNJAM SBI
  300   INIT                      ! INIT PROCESSOR
  400   DEPOSIT/I 11 20003800     ! SET UP SCBB
  500   DEPOSIT R0 0              ! DISK PACK DEVICE TYPE
  600   DEPOSIT R1 8              ! MBA TR=8
  700   DEPOSIT R2 3              ! ADAPTER UNIT = 3
  800   DEPOSIT R3 3              ! CONTROLLER UNIT = 3
  900   DEPOSIT R4 0              ! BOOT BLOCK LBN (UNUSED)
 1000   DEPOSIT R5 1              ! SOFTWARE BOOT FLAGS (CONVERSATIONAL BOOT)
 1100   DEPOSIT FP 0              ! SET NO MACHINE CHECK EXPECTED
 1200   START 20003000            ! START ROM PROGRAM
 1300   WAIT DONE                 ! WAIT FOR COMPLETION
 1400                             !
 1500   EXAMINE SP                ! SHOW ADDRESS OF WORKING MEMORY+^X200
 1600   LOAD VMB.EXE/START:0      ! LOAD PRIMARY BOOTSTRAP
 1700   START 0                   ! AND START IT
```

```
   50   !
   55   !        DB6 CONVERSATIONAL BOOT COMMAND FILE - DB6GEN.
   57   !        BOOT FROM DB6 AND STOP IN SYSBOOT TO ALTER PARAMETERS
   60   !
  100   HALT                        ! HALT PROCESSOR
  200   UNJAM                       ! UNJAM SBI
  300   INIT                        ! INIT PROCESSOR
  400   DEPOSIT/I 11 20003800       ! SET UP SCBB
  500   DEPOSIT R0 0                ! DISK PACK DEVICE TYPE
  600   DEPOSIT R1 8                ! MBA TR=8
  700   DEPOSIT R2 6                ! ADAPTER UNIT = 6
  800   DEPOSIT R3 6                ! CONTROLLER UNIT = 6
  900   DEPOSIT R4 0                ! BOOT BLOCK LBN (UNUSED)
 1000   DEPOSIT R5 1                ! SOFTWARE BOOT FLAGS (CONVERSATIONAL BOOT)
 1100   DEPOSIT FP 0                ! SET NO MACHINE CHECK EXPECTED
 1200   START 20003000              ! START ROM PROGRAM
 1300   WAIT DONE                   ! WAIT FOR COMPLETION
 1400                               !
 1500   EXAMINE SP                  ! SHOW ADDRESS OF WORKING MEMORY+^X200
 1600   LOAD VMB.EXE/START:0        ! LOAD PRIMARY BOOTSTRAP
 1700   START 0                     ! AND START IT
```

```
   50   !
   55   !        DB1 BOOT COMMAND FILE - DB1BOO.CMD
   60   !
  100   HALT                        ! HALT PROCESSOR
  200   UNJAM                       ! UNJAM SBI
  300   INIT                        ! INIT PROCESSOR
  400   DEPOSIT/I 11 20003800       ! SET UP SCBB
  500   DEPOSIT R0 0                ! DISK PACK DEVICE TYPE
  600   DEPOSIT R1 8                ! MBA TR=8
  700   DEPOSIT R2 1                ! ADAPTER UNIT = 1
  800   DEPOSIT R3 1                ! CONTROLLER UNIT = 1
  900   DEPOSIT R4 0                ! BOOT BLOCK LBN (UNUSED)
 1000   DEPOSIT R5 0                ! SOFTWARE BOOT FLAGS
 1100   DEPOSIT FP 0                ! SET NO MACHINE CHECK EXPECTED
 1200   START 20003000              ! START ROM PROGRAM
 1300   WAIT DONE                   ! WAIT FOR COMPLETION
 1400                               !
 1500   EXAMINE SP                  ! SHOW ADDRESS OF WORKING MEMORY+^X200
 1600   LOAD VMB.EXE/START:0        ! LOAD PRIMARY BOOTSTRAP
 1700   START 0                     ! AND START IT
```

```
   50   !
   55   !        DB0 BOOT COMMAND FILE - DB0B00.CMD
   60   !
  100   HALT                        ! HALT PROCESSOR
  200   UNJAM                       ! UNJAM SBI
  300   INIT                        ! INIT PROCESSOR
  400   DEPOSIT/I 11 20003800       ! SET UP SCBB
  500   DEPOSIT R0 0                ! DISK PACK DEVICE TYPE
  600   DEPOSIT R1 8                ! MBA TR=8
  700   DEPOSIT R2 0                ! ADAPTER UNIT = 0
  800   DEPOSIT R3 0                ! CONTROLLER UNIT = 0
  900   DEPOSIT R4 0                ! BOOT BLOCK LBN (UNUSED)
 1000   DEPOSIT R5 0                ! SOFTWARE BOOT FLAGS
 1100   DEPOSIT FP 0                ! SET NO MACHINE CHECK EXPECTED
 1200   START 20003000              ! START ROM PROGRAM
 1300   WAIT DONE                   ! WAIT FOR COMPLETION
 1400                               !
 1500   EXAMINE SP                  ! SHOW ADDRESS OF WORKING MEMORY+^X200
 1600   LOAD VMB.EXE/START:0        ! LOAD PRIMARY BOOTSTRAP
 1700   START 0                     ! AND START IT
```

```
   50   !
   55   !        DBB2 BOOT COMMAND FILE - DEFBOO.CMD
   60   !
  100   HALT                        ! HALT PROCESSOR
  200   UNJAM                       ! UNJAM SBI
  300   INIT                        ! INIT PROCESSOR
  400   DEPOSIT/I 11 20003800       ! SET UP SCBB
  500   DEPOSIT R0 0                ! DISK PACK DEVICE TYPE
  600   DEPOSIT R1 9                ! MBA TR=9 ; SECOND MBA
  700   DEPOSIT R2 2                ! ADAPTER UNIT = 2
  800   DEPOSIT R3 2                ! CONTROLLER UNIT = 2
  900   DEPOSIT R4 0                ! BOOT BLOCK LBN (UNUSED)
 1000   DEPOSIT R5 2                ! SOFTWARE BOOT FLAGS (KEEP DEBUG CODE)
 1100   DEPOSIT FP 0                ! SET NO MACHINE CHECK EXPECTED
 1200   START 20003000              ! START ROM PROGRAM
 1300   WAIT DONE                   ! WAIT FOR COMPLETION
 1400                               !
 1500   EXAMINE SP                  ! SHOW ADDRESS OF WORKING MEMORY+^X200
 1600   LOAD VMB.EXE/START:0        ! LOAD PRIMARY BOOTSTRAP
 1700   START 0                     ! AND START IT
```

```
RESTAR.CMD;2                    18-AUG-1978 15:58:23.95          Page 1

   50   !
   55   !        RESTART COMMMAND FILE - RESTAR.CMD
   60   !
   65   !        THIS COMMAND FILE IS INVOKED IN THE EVENT OF POWER RECOVERY AND
   70   !        OTHER CONSOLE DETECTED RESTART CONDITIONS IF THE AUTO RESTART SWITCH
   75   !        IS SET.  IT CAN ALSO BE INVOKED MANUALLY WITH THE COMMAND:
   80   !                                  *RESTAR.CMD
   85   !
  100   HALT                        ! HALT PROCESSOR
  200   UNJAM                       ! UNJAM SBI
  300   INIT                        ! INITIALIZE PROCESSOR
  400   DEPOSIT/I 11 20003800       ! SET ADDRESS OF SCB BASE
  500   DEPOSIT R0 0                ! CLEAR UNUSED REGISTERS
  600   DEPOSIT R1 3                ! UBA TR=3
  700   DEPOSIT R2 0                ! CLEAR UNUSED REGISTER
  800   DEPOSIT R3 0                ! CLEAR UNUSED REGISTER
  900   DEPOSIT R4 0                ! CLEAR UNUSED REGISTER
 1000   DEPOSIT R5 0                ! CLEAR UNUSED REGISTER
 1100   DEPOSIT FP 0                ! NO MACHINE CHECK EXPECTED
 1200   START 20003004              ! START RESTART REFEREE
```

```
DM0B00.ILV;1                    18-AUG-1978 15:57:50.81          Page 1

  100   !
  200   !       DM0 BOOT COMMAND FILE FOR INTERLEAVED MEMORIES - DM0B00.ILV
  300   !       THIS IS A TEMPLATE COMMAND FILE FOR BOOTING FROM RK06/RK07 DISKS
  400   !       IN SYSTEMS WITH TWO INTERLEAVED MEMORY CONTROLLERS.  THE MEMORY
  500   !       CONTROLLERS ARE ASSUMED TO BE AT TR NUMBERS 0 AND 1.
  550   !
  555   !       THIS COMMAND FILE SHOULD BE EDITED TO CHANGE THE UNIT NUMBER IF
  560   !       NECCESSARY AND USED TO REPLACE THE DEFAULT BOOT COMMAND FILE.
  600   !
  700   HALT                      ! HALT PROCESSOR
  800   UNJAM                     ! UNJAM SBI
  900   INIT                      ! INIT PROCESSOR
 1000   DEPOSIT/I 11 20003800     ! SET UP SCBB
 1100   DEPOSIT R0 1              ! CARTRIDGE DISK
 1200   DEPOSIT R1 3              ! UBA TR=3
 1300   DEPOSIT R2 3FF20          ! CSR ADDRESS OFFSET = 3FF20
 1400   DEPOSIT R3 0              ! CONTROLLER UNIT = 0
 1500   DEPOSIT R4 0              ! BOOT BLOCK LBN (UNUSED)
 1600   DEPOSIT R5 0              ! SOFTWARE BOOT FLAGS
 1700   DEPOSIT FP 0              ! SET NO MACHINE CHECK EXPECTED
 1705   DEPOSIT 20002000 101      ! ENABLE INTERLEAVE FOR TR #1 MEMORY CONTROLLER
 1710   DEPOSIT 20002004 4000     ! FORCE 0 STARTING ADDRESS
 1715   DEPOSIT 20004000 101      ! ENABLE INTERLEAVE FOR TR #2 MEMORY CONTROLLER
 1720   DEPOSIT 20004004 4000     ! FORCE 0 STARTING ADDRESS
 1800   START 20003000            ! START ROM PROGRAM
 1900   WAIT DONE                 ! WAIT FOR COMPLETION
 2000                             !
 2100   EXAMINE SP                ! SHOW ADDRESS OF WORKING MEMORY+^X200
 2200   LOAD VMB.EXE/START:0      ! LOAD PRIMARY BOOTSTRAP
 2300   START 0                   ! AND START IT
```

```
 100   !
 200   !        DB0 BOOT COMMAND FILE FOR INTERLEAVED MEMORIES - DB0B00.ILV
 212   !        THIS IS A TEMPLATE COMMAND FILE FOR BOOTING FROM RM03/RP06 DISKS
 224   !        IN SYSTEMS WITH TWO INTERLEAVED MEMORY CONTROLLERS.  THE MEMORY
 236   !        CONTROLLERS ARE ASSUMED TO BE AT TR NUMBERS 0 AND 1.
 248   !
 260   !        THIS COMMAND FILE SHOULD BE EDITED TO CHANGE THE UNIT NUMBER IF
 272   !        NECCESSARY AND USED TO REPLACE THE DEFAULT BOOT COMMAND FILE.
 284   !
 300   !
 400   HALT                      ! HALT PROCESSOR
 500   UNJAM                     ! UNJAM SBI
 600   INIT                      ! INIT PROCESSOR
 700   DEPOSIT/I 11 20003800     ! SET UP SCBB
 800   DEPOSIT R0 0              ! DISK PACK DEVICE TYPE
 900   DEPOSIT R1 8              ! MBA TR=8
1000   DEPOSIT R2 0              ! ADAPTER UNIT = 0
1100   DEPOSIT R3 0              ! CONTROLLER UNIT = 0
1200   DEPOSIT R4 0              ! BOOT BLOCK LBN (UNUSED)
1300   DEPOSIT R5 0              ! SOFTWARE BOOT FLAGS
1400   DEPOSIT FP 0              ! SET NO MACHINE CHECK EXPECTED
1420   DEPOSIT 20002000 101      ! ENABLE INTERLEAVE FOR TR #1 MEMORY CONTROLLER
1440   DEPOSIT 20002004 4000     ! FORCE 0 STARTING ADDRESS
1460   DEPOSIT 20004000 101      ! ENABLE INTERLEAVE FOR TR #2 MEMORY CONTROLLER
1480   DEPOSIT 20004004 4000     ! FORCE 0 STARTING ADDRESS
1500   START 20003000            ! START ROM PROGRAM
1600   WAIT DONE                 ! WAIT FOR COMPLETION
1700                             !
1800   EXAMINE SP                ! SHOW ADDRESS OF WORKING MEMORY+^X200
1900   LOAD VMB.EXE/START:0      ! LOAD PRIMARY BOOTSTRAP
2000   START 0                   ! AND START IT
```

```
RESTAR.ILV;1                    18-AUG-1978 15:57:52.58          Page 1

  100   !
  200   !       RESTART COMMMAND FILE FOR INTERLEAVED MEMORIES - RESTAR.ILV
  300   !       THIS COMMAND FILE SHOULD REPLACE RESTAR.CMD FOR SYSTEMS WITH TWO
  400   !       INTERLEAVED MEMORY CONTROLLERS.  MEMORY CONTROLLERS ARE ASSUMED
  500   !       TO BE AT TR NUMBERS 1 AND 2.
  600   !
  700   !       THIS COMMAND FILE IS INVOKED IN THE EVENT OF POWER RECOVERY AND
  800   !       OTHER CONSOLE DETECTED RESTART CONDITIONS IF THE AUTO RESTART SWITCH
  900   !       IS SET.  IT CAN ALSO BE INVOKED MANUALLY WITH THE COMMAND:
 1000   !                                 @RESTAR.CMD
 1100   !
 1200   HALT                      ! HALT PROCESSOR
 1300   UNJAM                     ! UNJAM SBI
 1400   INIT                      ! INITIALIZE PROCESSOR
 1500   DEPOSIT/I 11 20003800     ! SET ADDRESS OF SCB BASE
 1600   DEPOSIT R0 0              ! CLEAR UNUSED REGISTERS
 1700   DEPOSIT R1 3              ! UBA TR=3
 1800   DEPOSIT R2 0              ! CLEAR UNUSED REGISTER
 1900   DEPOSIT R3 0              ! CLEAR UNUSED REGISTER
 2000   DEPOSIT R4 0              ! CLEAR UNUSED REGISTER
 2100   DEPOSIT R5 0              ! CLEAR UNUSED REGISTER
 2200   DEPOSIT FP 0              ! NO MACHINE CHECK EXPECTED
 2300   DEPOSIT 20002000 101      ! ENABLE INTERLEAVE FOR TR #1 MEMORY CONTROLLER
 2400   DEPOSIT 20002004 4000     ! FORCE 0 STARTING ADDRESS
 2500   DEPOSIT 20004000 101      ! ENABLE INTERLEAVE FOR TR #2 MEMORY CONTROLLER
 2600   DEPOSIT 20004004 4000     ! FORCE 0 STARTING ADDRESS
 2700   START 20003004            ! START RESTART REFEREE
```

```
!               Bootstrap Help File - BOOT.HLP
!
!
!       This file describes the input parameters to the bootstrap program
!       VMB.EXE .  Normally the bootstrap will lookup the file [SYSEXE]SYSBOOT.EXE
!       on the specified device, load it into memory and transfer control
!       to it.
!
!       Two sets of command files are provided on the VAX/VMS console floppy
!       to perform the necessary bootstrap operations.  One set of these command
!       files will boot selecting an option to stop in SYSBOOT to alter system
!       parameters.  They are invoked as console indirect command files.
!
!               @DM0GEN                 ! Boot from RK07 unit 0
!               @DM1GEN                 !                unit 1
!               @DM2GEN                 !                unit 2
!               @DM3GEN                 !                unit 3
!               @DB0GEN                 ! Boot from RM03/RP06 unit 0
!               @DB1GEN                 !                     unit 1
!               @DB2GEN                 !                     unit 2
!               @DB3GEN                 !                     unit 3
!               @DB4GEN                 !                     unit 4
!               @DB5GEN                 !                     unit 5
!               @DB6GEN                 !                     unit 6
!               @DB7GEN                 !                     unit 7
!
!       The other set of these command files is normally invoked only via
!       the BOOT command but may be invoked explicitly as indirect command files.
!       These command files perform a normal, non-interactive boot without any
!       stop in SYSBOOT to change parameters.
!
!       BOOT DM0    or    @DM0BOO.CMD   ! Boot RK07 unit 0
!       BOOT DM1                        !           unit 1
!       BOOT DM2                        !           unit 2
!       BOOT DM3                        !           unit 3
!       BOOT DB0                        ! Boot RM03 or RP06 unit 0
!       BOOT DB1                        !                   unit 1
!       BOOT DB2                        !                   unit 2
!       BOOT DB3                        !                   unit 3
!       BOOT DB4                        !                   unit 4
!       BOOT DB5                        !                   unit 5
!       BOOT DB6                        !                   unit 6
!       BOOT DB7                        !                   unit 7
!
!       The bootstrap is loaded into memory at least one page above the first
!       available working memory to allow space for the Restart Parameter
!       Block.  The address of the base of the bootstrap is passed through
!       SP, the stack pointer, where it also serves as a temporary stack pointer.
!
! Input Parameters:
!       R0  -  <31:4>=MBZ; <3:0>=Device Type Code
!                     0 => Disk Pack          (RM03/RP04/RP05/RP06/RP07)
!                     1 => Cartridge Disk     (RK06/RK07)
!
!       R1  -  <31:4>=MBZ; <3:0>=System Bus Address("TR" Number)
!              For most configurations the following convention has been
!              used:
!                     TR Number       Adapter / Controller
!                     ---------       --------------------
!                         3           UNIBUS adapter
!                         8           MASSBUS adapter number 1
```

```
!                         9          MASSBUS adapter number 2
!
!       R2  -  For UBA:
!                      <31:18>=MBZ; <17:3>=UNIBUS Address of Control Register!
!                      <2:0>=MBZ
!                      RK06/RK07 CSR = 3FF20
!              FOR MBA:
!                      <31:4>=MBZ; <3:0>=Controller/Formatter Number
!       R3  -  <31:4>=MBZ; <3:0>=Unit Number
!       R4  -  <31:0>=Logical Block Number to read as boot block
!       R5  -  <31:0>=Software Boot Control flags
!               Bit     Meaning
!               ---     -------
!                0      Conversational boot. At various points in the system
!                       boot procedure, parameter and other input will be
!                       solicited from the console.
!
!                1      Debug.  This flag is passed through to VMS and causes
!                       the code for the exec debugger to be included in
!                       the running system.
!
!                2      Initial breakpoint. If this flag is set, and the exec
!                       debugger code is included (flag bit 1) then a breakpoint
!                       will occur immediately after the exec enables mapping.
!
!                3      Boot block.  If this flag is set then the boot block
!                       will be read and control transfered to it.
!
!                4      Diagnostic boot.  This flag causes a boot by file
!                       name for the diagnostic supervisor.
!
!                5      Bootstrap breakpoint. This flag causes the bootstrap
!                       to stop a breakpoint after performing necessary init-
!                       ialization if it has been built with debug code.
!
!                6      Image header. If this flag is set the transfer address
!                       from the image header of the boot file will be used.
!                       Otherwise control will transfer to the first byt of the
!                       boot file.
!
!                7      Memory test inhibit. This flag inhibits the testing
!                       of memory during bootstrapping.
!
!                8      File name. Causes the bootstrap to solicit the name
!                       of the boot file.
!
!                9      Halt before transfer.  Causes a HALT instruction
!                       to be executed prior to the transfer to the bootfile.
!                       This option is useful for debugging purposes.
!
!
!       SP  -  ADDRESS+(^X200) of first working 64Kb memory region
!              usable as both stack pointer and pointer to good memory.
!
!
! Output Parameters:
!       R10     -      Base address of region containing secondary bootstrap
!       R11     -      Pointer to Restart Parameter Block (RPB)
!       SP      -      Stack pointer
!       PR$_SCBB -     System Control Block base register
!
```

```
!       Memory layout at start of secondary bootstrap!
!
!       +-----------------------------------------+ !BASE
!       !                                         !
!       !       Restart Parameter Block (RPB)     !
!       !                                         !
!       +-----------------------------------------+ !BASE+^X200
!       !                                         !
!       !       Primary Bootstrap Code            !
!       !                                         !
!       !                                         !
!       +-----------------------------------------+ !PR$_SCBB
!       !                                         !
!       !          System Control Block           !
!       !                                         !
!       +-----------------------------------------+ !PFNMAP
!       !                                         !
!       !               PFN Bitmap                !
!       !                                         !
!       +-----------------------------------------+ !PFNMAP+^X800
!       !                                         !
!       !             Bootstrap Stack             !
!       !                                         !
!       +-----------------------------------------+ !(SP)
!       !                                         !
!       !        Secondary Bootstrap Code         !
!       !                                         !
!       +-----------------------------------------+
!
```

APPENDIX E

VAX/VMS AND DECnet-VAX SOURCE KITS

This appendix contains instructions for using both the VAX/VMS and the DECnet-VAX source kits.

E.1 THE VAX/VMS SOURCE KIT

The VAX/VMS source kit is a multivolume magnetic tape set that contains source files for selected components of the VAX/VMS system and object libraries and command files for all standard components of the VAX/VMS system.

To use the VAX/VMS source kit, you must mount the first volume of the kit (the tape labeled VMSRC1) and copy the files VMSRCKIT.DOC and VMSRCRST.COM from the tape to your disk. To do this copying, use the following sequence of DCL commands:

```
$ ALLOCATE MTcn:
$ MOUNT MTcn:  VMSRC1
$ COPY MTcn:  VMSRCRST.COM, VMSRCKIT.DOC  SYS$DISK:
```

When the copying operation is complete, dismount the tape. Then, display (by use of the TYPE command) or list (by use of the PRINT command) the file VMSRCKIT.DOC. This file will guide you in restoring the remaining files on the VAX/VMS source kit to your disk.

E.2 THE DECnet-VAX SOURCE KIT

The DECnet-VAX source kit is a single magnetic tape that contains source files, object libraries, and command files.

To use the DECnet-VAX source kit, you must mount the tape (labeled NTSRC1) and copy the files NETSRCKIT.DOC and NETSRCRST.COM from the tape to your disk. To do this copying, use the following sequence of DCL commands:

```
$ ALLOCATE MTcn:
$ MOUNT MTcn:  NTSRC1
$ COPY MTcn:  NETSRCRST.COM,  NETSRCKIT.DOC   SYS$DISK:
```

When the copying operation is complete, dismount the tape. Then, display (by use of the TYPE command) or list (by use of the PRINT command) the file NETSRCKIT.DOC. This file will guide you in restoring the remaining files on the DECnet-VAX source kit to your disk.

APPENDIX F

BOOT PROCESS

This appendix details the steps required to produce a running VAX/VMS syste.

1. Power up occurs.

2. Code in LSI ROM reads block 0 from the floppy into LSI memory.

3. This code reads CONSOL.SYS (the console program) from the floppy into LSI memory.

4. CONSOL.SYS now has control. This program performs the following steps:

   a. Loads WCS from the floppy file WCSxxx.PAT.

   b. Accepts commands interactively or from a command file on the floppy to:

      1. Set up SCBB for memory controller ROM code.

      2. Set up registers that indicate boot device type, TR, unit, CSR, and boot flags.

      3. Initiate execution of memory controller ROM code; find the first good contiguous 64k bytes of VAX memory. (Use adapter mapping register space for scratch space.) Return base address + ^X200 in SP.

      4. Load VMB.EXE (the primary bootstrap) from the floppy into base +^X200 and start it. At the base address is a restart parameter block (RPB) which is filled in at different stages of the boot process.

5. VMB is the primary bootstrap. It contains a generalized I/O driver capable of reading and writing all bootstrap devices. The driver will be available to the secondary bootstrap. VMB performs the following steps:

   a. Saves the register values and some values calculated from the register values in the RPB.

   b. Determines the amount and pattern of memory. A PFN bitmap is constructed. Unless inhibited by a boot flag, memory is tested for gross, uncorrectable parity errors. VMB constructs, in the RPB, a table indexed by TR number of all memory controller and I/O adapter types.

c. Based on register values, one of the following occurs:

   1. A boot block at the designated LBN will be read into VAX memory and given control.

   2. A file named [SYSEXE]SYSBOOT.EXE will be read and given control.

   3. A file named [SYSMAINT]DIAG.EXE will be read and given control.

   4. A file specified by the user in response to a prompt will be read and given control.

6. SYSBOOT is the standard secondary bootstrap. It performs initialization suitable for the unmapped environment. SYSBOOT performs the following steps:

   a. Reads current parameter settings from SYS.EXE.

   b. Looks up the boot device driver file and stores information about it.

   c. If register values so indicate, prompts the user to modify current system parameter settings. The user may change the start-up command file name and modify system parameters using SET or a previously created parameter file. New parameters become the "current" parameters on the next boot.

   d. Sets up SPT, SYSPHD, and PFN data structures.

   e. Reads resident executive into high physical memory. The PFN data structure can be smaller because it need not contain data about the resident executive.

   f. Locates and transfers to INIT code.

7. INIT is part of SYS.EXE. It performs the following:

   a. Enables mapping and sets the PC to system space.

   b. Announces the system.

   c. Initializes map for I/O adapters. For MASSBUS, allocates and initializes ADP, CRB, IDB. For UNIBUS, allocates and initializes ADP. Initializes hardware registers.

   d. Loads the boot disk driver into nonpaged pool and, based on driver prologue data, finishes allocating and initializing the data structures for the boot device.

   e. Creates SYS$DISK logical name.

   f. Moves a piece of INIT code to the pool and REIs to it. This segment of code releases INIT pages to the free list and jumps to the scheduler.

8. SYSINIT process:

   a. If necessary or requested, prompts for time of day.

   b. Writes back system parameters to SYS.EXE.

c. Creates some logical names.

d. Sets up swap and page files.

e. Installs VAX-11 RMS image and system message file as a pageable system section.

f. Mounts the disk (ACP process created).

g. Creates job controller, OPCOM, and ERRFMT.

h. Creates STARTUP process.

9. STARTUP reads start-up file, which causes it to:

a. Mount console floppy.

b. Create logical names.

c. Install images.

d. Run SYS$SYSTEM:SYSGEN.

e. Invoke [SYSMGR]SYSTARTUP.COM.

f. Logout.

10. SYSGEN is run by STARTUP or at any other time. SYSGEN:

a. Provides for dynamic loading of and connecting to drivers. (The DZ, operator, null, network, and mailbox drivers are permanently part of the executive image.)

b. Provides for creation of new parameter files (which have an encoded format).

APPENDIX G

BOOTSTRAP AND SYSGEN EXAMPLES

G.1 FIRST-TIME BOOTSTRAP EXAMPLE

The following example shows how a typical first-time bootstrap is performed. The distribution kit is an RK07 cartridge disk, on drive DMA1. The stand-alone Disk Save and Compress utility program is already loaded.

```
DSC>  DMA0:/VE=DMA1:

DSC>  ^P

>>>   HALT

      HALTED AT PC value

>>>   @DM0GEN

SYSBOOT>  USE MINIMUM.PAR

SYSBOOT>  CONTINUE
```

From this point, the bootstrap loads VAX/VMS, with the parameter settings defined in the parameter file MINIMUM.PAR.

G.2 CONVERSATIONAL BOOTSTRAP EXAMPLE

The following example shows how to use SYSBOOT to modify parameter values while booting the system, subsequent to the first-time bootstrap. Note that the file specified with the USE command (NEWPAR.PAR) refers to a file created by the SYSGEN utility.

```
>>>  @DM0GEN

SYSBOOT>  USE NEWPAR.PAR

SYSBOOT>  SHOW BALSETCNT

BALSETCNT    24  24  4   1024 SLOTS

SYSBOOT>  SET BALSETCNT 32

SYSBOOT>  CONTINUE
```

In this example, the parameter file NEWPAR.PAR is specified, but prior to booting the system, the system manager decides to change the value of the BALSETCNT parameter. The system is then booted.

G.3 NONSTOP BOOTSTRAP EXAMPLE

The following example shows the steps involved in bootstrapping the system without modifying any system parameter values. In this example, SYSBOOT does no prompting for commands.

```
>>>  BOOT  DM0

CPU HALTED
INIT SEQ DONE
HALT INST EXECUTED
HALTED AT 200034F9

G  0000000E  00000200
LOAD DONE, 00001600 BYTES LOADED
```

At completion of the bootstrap the system displays its announcement messages.

G.4 SYSGEN EXAMPLE

This example illustrates how to use the SYSGEN utility to create a new parameter file that can be used as input for subsequent system bootstraps. By changing the values of system parameters, you can generate a version of VAX/VMS tailored to your particular requirements.

```
$ RUN SYS$SYSTEM:SYSGEN
SYSGEN> USE CURRENT
SYSGEN> SHOW /JOB

  Parameter Name     Current   Default   Minimum   Maximum  Unit
  --------------     -------   -------   -------   -------  ----
  MAXPRINTSYMB             8         8         1       255  PROCESSES
  DEFPRI                   4         4         1        31
  IJOBLIM                 64        64         1      1024  JOBS
  BJOBLIM                 16        16         0      1024  JOBS
  NJOBLIM                 16        16         0      1024  JOBS

SYSGEN> SET DEFPRI 6
SYSGEN> WRITE SYS$SYSTEM:NEWVAL.PAR
SYSGEN> EXIT
$
```

8USER.PAR parameter file, 4-2, 4-9, 4-11
16USER.PAR parameter file, 4-2, 4-9, 4-11
32USER.PAR parameter file, 4-2, 4-9, 4-11
48USER.PAR parameter file, 4-2, 4-9, 4-11
64USER.PAR parameter file, 4-2, 4-9, 4-11

A

ACP_BASEPRIO parameter, A-1
ACP_DIRCACHE parameter, 4-3, A-1
ACP_EXTCACHE parameter, A-1
ACP_EXTLIMIT parameter, A-1
ACP_FIDCACHE parameter, A-1
ACP_HDRCACHE parameter, 4-3, A-1
ACP_MAPCACHE parameter, 4-3, A-1
ACP_MAXREAD parameter, A-2
ACP_MULTIPLE parameter, A-2
ACP_SHARE parameter, 4-3, A-2
ACP_SWAPFILES parameter, A-2
ACP_SYSACC parameter, A-2
ACP_WINDOW parameter, A-3
ACP_WORKSET parameter, 4-3, A-3
AUTHORIZE program, 3-1
AUTOCONFIGURE command, 1-3, 5-5
Automatic restart, 3-10

B

BALSETCNT parameter, 4-3, 4-4, A-3
BJOBLIM parameter, 4-3, A-3
BOOT error messages, B-1
Booting the system,
  See Bootstrapping the system
Bootstrap,
  examples of, 3-4, 3-6, G-1
Bootstrap command files, 3-3, D-1
Bootstrapping, 3-11
  internal operation of, F-1
Bootstrapping the system, 3-2
  conversational, 3-2, 3-4, G-1
  non-stop, 3-2, 3-6, G-2
BUGCHECKFATAL parameter, A-3
BUGREBOOT parameter, A-3
Building and copying the system, 3-12

C

Command files,
  DEFBOO.CMD, 3-1, 3-9
  SETDEFBOO.COM, 3-10
  [SYSMGR]SYSTARTUP.COM, 1-2, 3-1
  SYS$SYSTEM:STARTUP.COM, 1-2, 5-4
Command files (Cont.)
  [SYSUPD]DXCOPY, 3-11
  [SYSUPD]SWAPFILES, 3-9
  used to bootstrap the system, 3-3, D-1
CONNECT command, 5-7
Console,
  getting the attention of, 3-2
Console floppy diskette,
  copying files to, 3-11
CONTINUE command, 5-9
Conversational bootstrap, 3-2, 3-4, G-1
Copying the distribution media, 2-2
  disk to disk, 2-5
  tape to disk, 2-4
CRDENABLE parameter, A-3
CREATE command, 5-10

D

DECnet-VAX, 6-1
  installing, 6-12
DECnet-VAX source kit, E-1
Default bootstrap,
  command file for, 3-10
DEFBOO.CMD command file, 3-1, 3-9
DEFMBXBUFQUO parameter, A-3
DEFMBXMXMSG parameter, A-4
DEFMBXNUMMSG parameter, A-4
DIALTYPE parameter, A-4
Directories of VAX/VMS system, C-1, C-2
DISABLE CHECKS command, 5-11
Distribution kit,
  See VAX/VMS software distribution kit
DSC,
  See VAX-11 Disk Save and Compress
Dump file,
  calculating the size of, 4-11
  changing the size of, 3-1, 3-9
DUMPBUG parameter, A-4

E

ENABLE CHECKS command, 5-12
Error messages, B-1
  from BOOT, B-1
  from SYSBOOT, B-2
  from SYSGEN, B-5
EXIT command, 5-13

F

Files of VAX/VMS system,
  in directory [SYSERR], C-1
  in directory [SYSEXE], C-1, C-2
  in directory [SYSHLP], C-1, C-4

Files of VAX/VMS system (Cont.)
  in directory [SYSLIB], C-1, C-6
  in directory [SYSMAINT], C-1
  in directory [SYSMGR], C-1, C-7
  in directory [SYSMSG], C-1, C-7
  in directory [SYSTEST], C-1, C-7
  in directory [SYSUPD], C-2, C-11
FORTRAN IV/VAX to RSX Cross Compiler, 6-1
  code installation options, 6-7
  installing, 6-7
  OTS options, 6-9
  retaining previous versions, 6-10
  speed vs. size optimization, 6-8
FREELIM parameter, 4-3, 4-10, A-4

G

GBLPAGES parameter, 4-3, 4-4, A-4
GBLSECTIONS parameter, 4-3, 4-5, A-5

H

HELP command, 5-14

I

IJOBLIM parameter, 4-3, A-5
Installing optional software,
  completion of, 6-3
  DECnet-VAX, 6-12
  distribution kits for, 6-1
  FORTRAN IV/VAX to RSX Cross Compiler, 6-7
  general procedures for, 6-1
  PDP-11 BASIC-PLUS-2/VAX, 6-15
  PDP-11 DATATRIEVE/VAX, 6-17
  VAX-11 BLISS-32, 6-19
  VAX-11 COBOL-74, 6-13
  VAX-11 FORTRAN IV-PLUS, 6-6
Installing VAX/VMS, 1-4, 3-1
INTSTKPAGES parameter, A-5
IRPCOUNT parameter, 4-3, 4-5, A-5

K

KFILSTCNT parameter, A-5

L

LOAD command, 5-16
Loading VAX-11 Disk Save and Compress, 2-3
Logging into the system, 3-1, 3-8

M

Magnetic tape kit, 2-1
Major parameters, 4-4
MAXBUF parameter, A-6
Maximum process working set size, calculation of, 4-10
MAXPRINTSYMB parameter, A-6
MAXPROCESSCNT parameter, 4-3, 4-5, 4-10, 4-11, A-6
MAXSYSGROUP parameter, A-6
MINIMUM.PAR parameter file, 3-1, 3-4, 4-2, 4-9, 4-11
MINWSCNT parameter, A-6
Modifying parameter files, 4-1
MPW_HILIM parameter, 4-3, 4-6, 4-7, A-6
MPW_LOLIMIT parameter, 4-3, 4-6, A-7
MPW_WRTCLUSTER parameter, 4-3, 4-6, 4-7, A-7

N

NJOBLIM parameter, 4-3, A-7
Nonpaged dynamic pool, calculating the size of, 4-7
Non-stop bootstrap, 3-2, 3-6, G-2
NPAGEDYN parameter, 4-1, 4-3, 4-5, 4-7, A-7

O

Optional software,
  DECnet-VAX, 6-12
  distribution kits, 6-1
  FORTRAN IV/VAX to RSX Cross Compiler, 6-7
  general procedures for installing, 6-1
  list of, 6-1
  PDP-11 BASIC-PLUS-2/VAX, 6-15
  PDP-11 DATATRIEVE/VAX, 6-17
  VAX-11 BLISS-32, 6-19
  VAX-11 COBOL-74, 6-13
  VAX-11 FORTRAN IV-PLUS, 6-6

P

Page fault cluster factor, 4-8
Paged dynamic pool, calculating the size of, 4-8
PAGEDYN parameter, 4-3, 4-8, A-7
PAGEFILE.SYS file, 4-11
Paging file,
  calculating the size of, 4-11
  changing the size of, 3-1, 3-9
PAGTBLPFC parameter, A-14
Parameter files, 1-2, 1-3, 4-1
  list of, 4-2
  MINIMUM.PAR, 3-1, 3-4
  modification of, 3-1, 3-8
  relationship to VIRTUALPAGECNT parameter, 4-9
Parameter values, 4-3
  discussion of major parameters, 4-4 through 4-10

Parameter values (Cont.)
  list of major parameter values, 4-4
Parameters,
  description of all, A-1
Patching PDP-11 DATATRIEVE/VAX, 6-18
PDP-11 BASIC-PLUS-2/VAX, 6-1
  installing, 6-15
PDP-11 DATATRIEVE/VAX, 6-1
  installing, 6-17
  patching, 6-18
PFCDEFAULT parameter, 4-3, 4-8, A-7
PHYSICALPAGES parameter, A-14
POOL_PAGING parameter, A-8
PQL_DASTLM parameter, A-8
PQL_DBIOLM parameter, A-8
PQL_DBYTLM parameter, A-8
PQL_DCPULM parameter, A-8
PQL_DDIOLM parameter, A-8
PQL_DFILLM parameter, A-8
PQL_DPRCLM parameter, A-8
PQL_DTQELM parameter, A-8
PQL_DWSDEFAULT parameter, A-9
PQL_DWSQUOTA parameter, A-9
PQL_MASTLM parameter, A-9
PQL_MBIOLM parameter, A-9
PQL_MBYTLM parameter, A-9
PQL_MCPULM parameter, A-9
PQL_MDIOLM parameter, A-9
PQL_MFILLM parameter, A-9
PQL_MPRCLM parameter, A-9
PQL_MTQELM parameter, A-9
PQL_MWSDEFAULT parameter, A-10
PQL_MWSQUOTA parameter, A-10
PROCSECTCNT parameter, 4-3, A-10

Q

QUANTUM parameter, 4-3, 4-8, A-10

R

Rebooting the system, 1-4, 3-1, 3-2, 3-10
RELOAD command, 5-17
Restart,
  automatic, 3-10
RK07 kit, 2-2
RMS_DFMBC parameter, A-10
RMS_DFMBFHSH parameter, A-10
RMS_DMBFIDX parameter, A-10
RMS_DMBFREL parameter, A-10
RMS_DMBFSDK parameter, A-10
RMS_DMBFSMT parameter, A-11
RMS_DMBFSUR parameter, A-11

S

SBIERRENABLE parameter, A-11
SET parameter value command, 5-18
SET start-up file command, 5-19
SETDEFBOO.COM command file, 3-10
SETTIME parameter, A-11
SHOW command, 5-20
Software distribution kit,
  See VAX/VMS software distribution kit
Source kits,
  DECnet-VAX, E-1
  VAX/VMS, E-1
Special parameters, A-14
  PAGTBLPFC parameter, A-14
  PHYSICALPAGES parameter, A-14
  SYSPFC parameter, A-14
SPTREQ parameter, 4-3, A-11
Stand-alone DSC,
  See VAX-11 Disk Save and Compress
SWAPFILES.SYS file, 4-10, 4-11
Swapping file,
  calculating the size of, 4-11
  changing the size of, 3-1, 3-9
SYSBOOT commands, 5-4
  CONTINUE, 5-9
  DISABLE CHECKS, 5-11
  ENABLE CHECKS, 5-12
  EXIT, 5-13
  HELP, 5-14
  SET parameter value, 5-18
  SET start-up file, 5-19
  SHOW, 5-20
  USE, 5-23
SYSBOOT error messages, B-2
SYSBOOT program, 1-1
SYSDUMP.DMP file, 4-11
[SYSERR], C-1
[SYSEXE], C-1, C-2
SYSGEN commands, 5-2
  AUTOCONFIGURE, 5-5
  Available in SYSBOOT, 5-4
  CONNECT, 5-7
  CONTINUE, 5-9
  CREATE, 5-10
  DISABLE CHECKS, 5-11
  ENABLE CHECKS, 5-12
  EXIT, 5-13
  HELP, 5-14
  LOAD, 5-16
  RELOAD, 5-17
  SET parameter value, 5-18
  SET start-up file, 5-19
  SHOW, 5-20
  Summary of, 5-1 through 5-4
  USE, 5-23
  Used for I/O devices, 5-4
  Used with parameter files, 5-3
  WRITE, 5-24
SYSGEN error messages, B-5
SYSGEN examples, G-2
SYSGEN utility, 1-1, 1-3, 5-1
  invoking, 5-1

SYSGEN utility (Cont.)
  terminating, 5-1
[SYSHLP], C-1, C-4
[SYSLIB], C-1, C-6
[SYSMAINT], C-1
[SYSMGR], C-1, C-7
[SYSMGR]SYSTARTUP.COM command file, 1-2, 3-1, F-3
[SYSMSG], C-1, C-7
SYSMWCNT parameter, 4-3, 4-9, A-12
SYSPAGING parameter, A-12
SYSPFC parameter, A-14
SYS$SYSTEM:STARTUP.COM command file, 1-2, 5-4
System generation,
  details of, A-1
  overview of, 1-1
System generation parameters,
  list of, A-1 through A-14
System minimum working set,
  calculating the size of, 4-9
System software,
  installing optional software, 6-1
  updating, 6-1
[SYSTEST], C-1, C-7
[SYSUPD], C-2, C-11
[SYSUPD]DXCOPY command file, 3-11
[SYSUPD]SWAPFILES command file, 3-9

T

Terminating VAX-11 Disk Save and Compress, 2-5
Time quantum, 4-8
TTY_BUF parameter, A-12
TTY_DEFCHAR parameter, A-12
TTY_SCANDELTA parameter, A-12
TTY_SPEED parameter, A-13

U

UAFALTERNATE parameter, A-13
Updating system software, 6-4
  distribution kit for, 6-1
  general procedure for, 6-1
USE command, 5-23

V

VAX-11 BLISS-32, 6-1
  installing, 6-19
VAX-11 COBOL-74, 6-1
  installing, 6-13
VAX-11 Disk Save and Compress, 1-1, 2-1
  loading, 2-3
  terminating, 2-5
  used to copy magnetic tape, 2-4
  used to copy RK07 disk cartridge, 2-5
VAX-11 FORTRAN IV-PLUS, 6-1
  installing, 6-6
VAX/VMS software distribution kit, 2-1
  contents of, 2-1
  magnetic tape kit, 2-1
  RK07 kit, 2-2
VAX/VMS source kit, E-1
VIRT32MB.PAR parameter file, 4-2, 4-10, 4-11
VIRTUALPAGECNT parameter, 4-1 4-3, 4-9, A-13
VMSKITBLD.COM command file, 3-12
VMSKITCPY.COM command file, 3-13

W

WRITE command, 5-24
WSMAX parameter, 4-1, 4-3, 4-10, 4-11, A-13

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 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_____

Please cut along this line.

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

RT/C SOFTWARE PUBLICATIONS TW/A14
DIGITAL EQUIPMENT CORPORATION
1925 ANDOVER STREET
TEWKSBURY, MASSACHUSETTS 01876

Do Not Tear - Fold Here

Cut Along Dotted Line

digital

PRINTED IN USA