


pdp11



**RSX-11D
System Generation
Reference Manual**

Order No. DEC-11-OXDIA-E-D

digital

RSX-11D
System Generation
Reference Manual

Order No. DEC-11-OXDIA-E-D

RSX-11D Version 6.2

First Printing, June 1975
Revised: November 1975
February 1977

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

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

Digital Equipment Corporation assumes no responsibility for the use or reliability of its software on equipment that is not supplied by DIGITAL.

Copyright © 1975, 1977 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-10
DECCOMM	DECsystem-20	TYPESET-11

CONTENTS

		Page
PREFACE		vii
	0.1 MANUAL OBJECTIVES AND READER ASSUMPTIONS	vii
	0.2 STRUCTURE OF THE DOCUMENT	vii
	0.3 ASSOCIATED DOCUMENTATION	vii
CHAPTER 1	OVERVIEW OF SYSTEM GENERATION	1-1
	1.1 CONCEPTS AND TERMINOLOGY	1-2
CHAPTER 2	GETTING ON THE AIR	2-1
	2.1 THE DISTRIBUTION SYSTEM	2-1
	2.1.1 Magnetic Tape Distribution	2-1
	2.1.1.1 Procedures for Using the System Tape	2-1
	2.1.1.2 Procedures for Using Auxiliary Tapes	2-3
	2.1.2 RK05 Cartridge Distribution	2-4
	2.1.3 RK06 Cartridge Distribution	2-4
	2.2 ORGANIZATION OF DIRECTORIES ON THE DISTRIBUTION MEDIA	2-4
	2.2.1 System Disk Directories	2-5
	2.2.2 Auxiliary Media Directory Contents	2-6
	2.3 HARDWARE BOOTSTRAPS	2-8
	2.3.1 MR11-DB Bootstrap	2-8
	2.3.2 BM792-YB Bootstrap	2-9
	2.3.3 BM873-YA Bootstrap	2-10
	2.3.4 BM873-YB Bootstrap	2-11
	2.3.5 M9301-YC Bootstrap	2-12
	2.4 TERMINAL HANDLER GENERATION	2-13
	2.4.1 Introduction	2-13
	2.4.2 Editing PARAMS.MAC	2-13
	2.4.3 Editing CONFIG.MAC	2-13
	2.4.4 Building the Handler	2-15
	2.4.5 Terminal Handler Error Messages	2-21
CHAPTER 3	SYSTEM GENERATION PROCEDURES	3-1
	3.1 PHASE 1 OF SYSTEM GENERATION	3-2
	3.2 PHASE 2 OF SYSTEM GENERATION	3-5
CHAPTER 4	SYSTEM GENERATION DIRECTIVES	4-1
	4.1 TARGET DIRECTIVE	4-2
	4.2 PDP11 DIRECTIVE	4-3
	4.3 EXEC DIRECTIVE	4-4
	4.4 SCOM DIRECTIVE	4-5
	4.4.1 SCOM Size	4-5
	4.5 PAR DIRECTIVE	4-7
	4.6 DEV DIRECTIVE	4-8
	4.6.1 Pseudo-Devices	4-11
	4.7 SY DIRECTIVE	4-12
	4.8 DPAR DIRECTIVE	4-13
	4.9 DUIC DIRECTIVE	4-14
	4.10 CKPNT DIRECTIVE	4-15
	4.11 PSWD DIRECTIVE	4-16
	4.12 INS DIRECTIVE	4-17
	4.13 SYSBLD DIRECTIVE	4-18

	4.14	PHASE 2 DIRECTIVE (*DELAY)	4-18
CHAPTER	5	ADDITIONAL SYSTEM GENERATION INFORMATION	5-1
	5.1	EDITING SYSTEM GENERATION FILES	5-1
	5.2	PHASE 1 OF SYSTEM GENERATION	5-2
	5.2.1	Target Specification	5-2
	5.2.2	CPU Specification	5-3
	5.2.3	Memory Allocation	5-3
	5.2.4	Device Specifications	5-3
	5.2.5	System Default Specifications	5-3
	5.2.6	Required Task Installations	5-4
	5.2.7	Specification of SYSBLD Command Files	5-5
	5.2.8	Pool Requirements of SGN1	5-5
	5.3	PHASE 2 OF SYSTEM GENERATION	5-7
	5.4	RSX-11D BOOTSTRAPPING	5-7
	5.4.1	System Generation Phase 1 (SGN1)	5-8
	5.4.2	System Generation Phase 2 (SGN2)	5-9
	5.4.3	Saving a Generated System	5-10
	5.4.4	Subsequent Rebooting of a Saved Image	5-12
	5.5	MEMORY ALLOCATION AND USAGE	5-12
CHAPTER	6	TAILORING THE SYSTEM	6-1
	6.1	EXECUTIVE CHANGES	6-1
	6.2	UTILITY PROGRAMS	6-1
APPENDIX	A	EXAMPLES OF SYSTEM GENERATION FILES	A-1
	A.1	TERMINAL AND INTERFACE CONFIGURATION FILE, [311,114]CONFIG.MAC	A-1
	A.2	SYSTEM GENERATION PHASE 1, [11,17]RK05L48K.CMD	A-3
	A.3	SYSTEM GENERATION PHASE 2, [11,17]SYSBLD.CMD	A-4
APPENDIX	B	DEVICE CHARACTERISTICS WORDS	B-1
ERROR MESSAGES			Messages-1
INDEX			Index-1

FIGURES

			Page
FIGURE	5-1	SGN1 Pool Usage	5-6

TABLES

			Page
TABLE	2-1	Contents of System Disk	2-5
	2-2	Contents of Auxiliary Media	2-6
	2-3	Device Addresses for the MR11-8DB Bootstrap	2-8
	2-4	Device Addresses for the BM792-YB Bootstrap	2-9
	2-5	Device Addresses for BM873-YA Bootstrap	2-10
	2-6	Device Addresses for BM873-YB Bootstrap	2-11
	2-7	Device Codes for M9301-YC Bootstrap -- Switches 3 through 6	2-12
	2-8	Terminal Handler Group 1 Assembly Parameters	2-17
	2-9	Terminal Handler Group 2 Assembly Parameters	2-18
	2-10	Parameters Dependent on B\$SAW in the Handler as Distributed	2-20
	4-1	Device Information	4-10

PREFACE

0.1 MANUAL OBJECTIVES AND READER ASSUMPTIONS

This manual describes the procedures to be followed by the system manager to perform an RSX-11D system generation. Understanding of the introductory material in the RSX-11D User's Guide and a general knowledge of RSX-11D is a prerequisite for the use of this document.

0.2 STRUCTURE OF THE DOCUMENT

Chapters 1, 2, and 3 introduce system generation. Chapter 2 describes the process of transferring the system from the distribution medium to the system disk. Chapter 3 contains the basic steps that must be followed to perform any type of RSX-11D system generation.

Chapter 4 describes the directives used to define the hardware configuration and allocate memory.

Chapter 5 contains further system generation information.

Chapters 6 summarizes the procedures available for tailoring individual components within the system.

0.3 ASSOCIATED DOCUMENTATION

Associated RSX-11D documents are described and their readerships are defined in the RSX-11D Documentation Directory, Order Number DEC-11-OXUGA-D-D.

CHAPTER 1

OVERVIEW OF SYSTEM GENERATION

System generation is the process of building a system that is tailored to both a particular hardware configuration and the requirements of the application programs that execute in the system. An RSX-11D system is distributed to the user on one of the following media:

- Magnetic tape (7-track or 9-track),
- RK05, RK06 cartridge disk.

A magnetic tape distribution is used to form a bootable disk system. A disk distribution comes as a bootable disk system.

Among the system programs provided by RSX-11D is a system generation program that executes in two phases to create an expanded or modified version of the system. This program is designed to accept user-provided definitions of the system to be created.

System generation is performed in two phases. The first phase can execute in a normal production environment without disturbing system operation. The second phase must be performed on the target hardware configuration.

System generation normally is performed for one of the following reasons:

1. Creating a system for task development, as described in Chapter 2, "Getting on the Air,"
2. Revising a system to reflect an altered physical configuration,
3. Revising a system to tailor it to its applications in order to improve performance.

The information needed for reasons 2 and 3, above, is described in Chapters 3, 4 and 5.

OVERVIEW OF SYSTEM GENERATION

1.1 CONCEPTS AND TERMINOLOGY

RSX-11D is a partitioned multiprogramming system. Partitions are named, contiguous blocks of memory, the number of which is fixed during system generation. All tasks in all partitions can execute in parallel. Partitions can be either user-controlled or system-controlled.

A user-controlled partition can accommodate only one task at a time. A system-controlled partition can accommodate as many tasks as can fit in the defined physical space. All tasks in a system-controlled partition can run in parallel. System-controlled partitions can schedule tasks on a priority basis or on a time-slice basis.

An active task is one in main memory that is competing for system resources. In a priority-oriented partition, a task can be checkpointed to make room for a higher priority task to execute in that partition if the first task is designated as checkpointable. Likewise, in a time-scheduled partition, a checkpointable task can be checkpointed if its time-slice expires.

Before a task can execute, it must be installed. More than one task can be installed to run in a partition. The main purpose of the installation procedure is to record disk retrieval pointers in the Executive's main memory so that the task can be made ready to execute with minimum delay when a request is issued for it. The task can be either explicitly installed using the INS command to MCR or implicitly installed as a result of a RUN command issued by a nonprivileged user. Tasks installed as a result of RUN are removed automatically when the task exits. Tasks explicitly installed can be removed using the REM command.

The system task directory (STD) establishes the maximum number of tasks that can be installed at one time. Normally the number of installed tasks is greater than the number of executing tasks. The number of simultaneously installed tasks is limited by the number of system task directory entries specified during system generation. The number of executing tasks is limited by the number of user-controlled partitions plus the number of tasks that can fit into system-controlled partitions. The number of STD entries for tasks should be greater than the number of available partitions so that a maximum number of tasks can execute simultaneously. Installed tasks can be removed when no longer needed to free additional STD entries.

In RSX-11D, some of the dynamic memory requirements are satisfied from a pool of nodes. Nodes are variable-size memory blocks that are a multiple of 8 words. The size of the node pool is established during system generation.

CHAPTER 2

GETTING ON THE AIR

This chapter describes the organization of the media on which RSX-11D is distributed and the procedures to be followed in preparing these media for the initial system generation. Section 2.3 contains the instructions for use of the various bootstraps available on the PDP-11. Section 2.4 describes how to use the basic system as distributed to generate the Terminal Handler task for the desired configuration. This must be done before the main system generation described in Chapter 3.

2.1 THE DISTRIBUTION SYSTEM

RSX-11D systems are distributed on three basic media: magnetic tape (7- or 9-track), RK05 cartridges, and RK06 cartridges.

2.1.1 Magnetic Tape Distribution

The system distribution tape contains a number of bootable system images. These are run in turn to create a single-terminal RSX-11D system from the files also contained in the system distribution tape.

2.1.1.1 Procedures for Using the System Distribution Tape

1. Using the procedures for the particular ROM bootstrap described in Section 2.3, boot the system tape into memory. The system prints the following on the console.

```
RSX-11D V06.2 SYSTEM DISTRIBUTION TAPE  
SYSTEM DISK?
```

2. Respond with the device name to indicate which device is the system disk. For example:

```
RK05 for RK05 system disk  
RP06 for RP06 system disk
```

3. Once the name of the system device has been typed, the system prints the following message or its equivalent:

```
LOAD DISK ON xy0 WRITE ENABLED  
TYPE 'CR' WHEN READY
```

where xy is the device type. For example, xy is

```
DK for RK05 system disk  
DB for RP04, RP05, or RP06 system disk
```

GETTING ON THE AIR

DM for RK06 system disk
DP for RP02 or RP03 system disk

4. When the disk is ready, type carriage return, and if the disk can be formatted, the system prints the following:

FORMAT DISK?

NOTE

Formatting a volume destroys all information on the disk as does running BADBLOCKS (step 6) and initializing (step 8).

5. Respond with one of the following:

YES to format the system disk
NO if disk formatting is not required.

6. The system then prints the following message:

RUN 'BADBLOCKS'?

7. Respond with one of the following:

YES to run the BAD BLOCKS utility (see RSX-11D User's Guide) on the system disk
NO if this operation is not required.

8. The system then prints the following message:

INITIALIZE SYSTEM DISK?

9. Respond with one of the following:

YES to initialize the disk
NO if the disk does not require initialization.

NOTE

This question will not be asked if either or both of the previous two replies were YES. Initialization is mandatory in these circumstances.

If initialization is not required then the system continues from step 12 onwards.

10. if the disk requires initialization, the system types:

STANDARD VOLUME INITIALIZATION?

11. Respond with one of the following:

YES if standard volume initialization required
NO to make the initialization task prompt for the initialization parameters.

See the RSX-11D User's Guide for a description of the MCR command INITIALIZE VOLUME. Standard volume initialization

GETTING ON THE AIR

gives the volume a label of RSXSYS

With standard volume initialization, if the reply to the BADBLOCKS question was YES, the volume will be initialized with the /BAD=[AUTO] option, otherwise without.

12. The system is then created.

The system prints the command file used during system creation on the console.

During this process, all of the files required on an RSX-11D system disk are loaded from the tape and a complete basic system is generated on the target disk. If all the replies to the three questions in 4, 6, 8 above were YES, the process will take approximately 15 minutes for an RK05 system or approximately 30 minutes for an RP04 system. When finished, the system prints the following.

*** END OF SYSTEM GENERATION PHASE 2 ***

13. Enter the time and date, dismount the system disk and save the new system as follows:

Press CTRL/C to obtain MCR

```
MCR>TIM dd-mmm-yy hh:mm:ss
MCR>DMO SY:
MCR>FIX FllACP
MCR>SAV
```

NOTE

In this manual, all command lines are terminated by pressing the RETURN key unless otherwise specified.

14. The system then responds with a sign-on message giving the actual memory size and the version of the RSX-11D Executive that is in use.

All of the system memory is available. If the hardware has more than 48K, a message in the following format is printed on the terminal.

```
SAV -- PARTITION GEN EXPANDED BY nnnn*32 (DEC) WORD BLOCKS
```

In addition, if the central processor is a PDP-11/70, it has been enabled as such, thereby overriding any previous specification.

15. The new system disk is now usable as a base for tailoring the desired RSX-11D system. Every time this disk is booted, the message in step 14 above is printed. If, however, the system is saved again, the size of the expanded partition will be recorded and the amount of expansion will not be printed.

2.1.1.2 Procedures for Using Auxiliary Tapes

The directory organization of RSX-11D is described in Section 2.2. This section describes the procedure for obtaining files from the object tapes, which are the other two tapes in the magnetic tape kit.

The files are written on tape in DOS format. The file interchange utility (FLX) is used to obtain files from the tape. The FLX commands are described in the RSX-11D Utility Programs Procedures Manual.

2.1.2 RK05 Cartridge Distribution

The first of the three RK05 cartridges is a single-user 48K RSX-11D system. The other two cartridges contain object files and command files for rebuilding system components. All three are Files-11 volumes with a directory organization that is described in Section 2.2.

It is advised that all three disks be copied for backup purposes before any other use is made of them. Preserve (PRE) is the utility to be used for this operation. Refer to the Preserve Manual.

To use the system disk, it need only be bootstrapped from device 0. See Section 2.3 for the appropriate bootstrap operation. When the disk is booted, the system displays the sign-on message for this version of the RSX-11D Executive. All of the system memory is made available. If the hardware has more than 48K, a message in the following format is printed on the terminal.

```
SAV -- PARTITION GEN EXPANDED BY nnnn*32 (DEC) WORD BLOCKS
```

In addition, if the central processor is a PDP-11/70, it has been enabled as such, thereby overriding any previous specification.

2.1.3 RK06 Cartridge Distribution

The RK06 distribution cartridge contains a single-user 48K RSX-11D system, object files and command files for rebuilding system components. The directory organization is as described in Section 2.2.

It is advised that the disk be copied for backup purposes before any other use is made of it. Disk Save and Compress (DSC) is the utility to be used for this operation. Refer to the Disk Save and Compress Manual. To use the system disk, it need only be bootstrapped from device 0. When the disk is booted, the system displays the sign-on message for this version of the Executive. All of the system memory is made available. If the hardware has more than 48K, a message in the following format is printed on the terminal.

```
SAV -- PARTITION GEN EXPANDED BY nnnn*32(DEC) WORD BLOCKS
```

In addition, if the central processor is a PDP-11/70, it has been enabled as such, thereby overriding any previous specification.

2.2 ORGANIZATION OF DIRECTORIES ON THE DISTRIBUTION MEDIA

Whether the system distribution is on magnetic tape or RK05 cartridges, the organization of directories is the same:

- [311,n] contain source files, and assembly command files,
- [211,n] contain listing files,
- [111,n] contain task map listings,
- [11,n], (excluding [11,1]) contain all object files, build (overlay description) command files, and command files for the librarian, PIP and system generation.

GETTING ON THE AIR

2.2.1 System Disk Directories

An RSX-11D system disk has the directories listed in Table 2-1.

Table 2-1
Contents of System Disk

UFD	CONTENTS
[1,1]	Executive symbol table System macro library System resident and relocatable object library
[1,2]	Error message files
[1,3]	Temporary files used and created by the Files-11 verify utility -- initially empty
[1,4]	Output spooler queue and spooled files -- initially empty
[1,5]	Task accounting data files -- initially empty
[1,6]	Error logging data files -- initially empty
[11,1]	All system task images
[11,17]	System generation command files Bootstraps System generation phase 2 task Executive task image

2.2.2 Auxiliary Media Directory Contents

The auxiliary media contain the directories listed in Table 2-2.

Table 2-2
Contents of Auxiliary Media

UFD	CONTENTS
[11,2]	All modules that compose SYSLIB, SYSRES, and the Files-11 system
[11,4]	FLX object modules
[11,5]	Object modules of PIP, DMP, VFY, ZAP and PAT
[11,6]	SLP object modules
[11,7]	Librarian object modules
[11,10]	MACRO-11 Assembler object modules (cross-reference version)
[11,11]	Task builder object modules
[11,12]	Spooler object modules
[11,13]	Object modules for all the MCR functions except BOO and SAV
[11,14]	Handler object modules
[11,15]	Executive and SCOM
[11,16]	File control primitives and file system utilities such as UFD and INITVOL.
[11,17]	System generation Bootstraps, BOO and SAV MCR functions Virtual install INV build command files
[11,20]	EDI object modules
[11,21] and [11,22]	System test tasks
[11,23]	Batch
[11,25]	Unsupported software
[11,26]	Accounting system
[11,27]	Error logging and diagnostics modules
[11,30]	Online Preserve, Disk Save and Compress (DSC)
[11,32]	Cross Reference
[11,41]	FORTRAN IV compiler
[11,42]	FORTRAN IV OTS

GETTING ON THE AIR

Table 2-2 (Cont.)
Contents of Auxiliary Media

UFD	CONTENTS
[11,114]	Terminal Handler
[311,13]	Sources for DEMO
[311,25]	Sources for unsupported modules
[311,114]	Terminal handler sources

2.3 HARDWARE BOOTSTRAPS

Five models of hardware bootstraps are available on systems used for RSX-11D: MR11-DB and BM792-YB and the Massbus bootstraps BM873-YA, BM873-YB, and M9301-YC. The type of bootstrap for a particular PDP-11 can be determined by consulting the equipment order.

Whenever a request to bootstrap the system is encountered in the following text, refer to one of the five sections that follow to perform the appropriate bootstrap.

2.3.1 MR11-DB Bootstrap

Perform the following steps to use an MR11-DB Bootstrap.

1. On the console switches, set HALT.
2. Set ENABLE.
3. Enter the address of the device from which the bootstrap is to occur into the console switches. Table 2-3 provides the device addresses.
4. Press LOAD ADDR.
5. Press START.

Table 2-3
Device Addresses for the MR11-DB Bootstrap

DEVICE	ADDRESS
RP03 Disk	173154
RK Disk	173110
RF Disk	173100
TU10 Magnetic Tape	173136 (7 or 9 Channel)
DECtape	173120

2.3.2 BM792-YB Bootstrap

Perform the following steps to use a BM792-YB Bootstrap.

1. On the console switches, set HALT.
2. Set ENABLE
3. Enter 173100 into the display switches.
4. Press LOAD ADDR.
5. Enter the address of the device from which the bootstrap is to occur into the console switches. Table 2-4 provides the device addresses.
6. Press START.

Table 2-4
Device Addresses for the BM792-YB Bootstrap

DEVICE	ADDRESS
RP03 Disk	176716
RK Disk	177406
RF Disk	177462
DEctape	177344
NOTE	
Magnetic tape cannot be booted with the BM792-YB Bootstrap.	

2.3.3 BM873-YA Bootstrap

Perform the following steps to use the BM873-YA Bootstrap.

1. On the console switches, set HALT.
2. Set ENABLE.
3. Enter the address of the device from which the bootstrap is to occur into the console switches. Table 2-5 provides the device addresses.
4. Press LOAD ADDR.

NOTE

If a unit other than 0 contains the device to be booted, set the switch register to the unit number of the device to be booted before pressing START.

5. Press START.

Table 2-5
Device Addresses for BM873-YA Bootstrap

DEVICE	ADDRESS
RF11 (RFn disk)	773000
RP11 (RP03 disk)	773100
RK11 (RK05 disk) -- Unit 0	773010
Unit specified in switch register	773020
TC11 (DECTape)	773030
TM11 (TU10 magnetic tape)	773050

GETTING ON THE AIR

2.3.4 BM873-YB Bootstrap

Perform the following steps to use the BM873-YB bootstrap.

1. On the console switches, set HALT.
2. Set ENABLE.
3. Enter the address of the device from which the bootstrap is to occur into the console switches. Table 2-6 provides the device addresses.
4. Press LOAD ADDR.

NOTE

If a unit other than 0 contains the device to be booted, set the switch register to the unit number of the device to be booted before pressing START.

5. Press START.

Table 2-6
Device Addresses for BM873-YB Bootstrap

DEVICE	ADDRESS
RH11 (RS03,4 disk) -- unit 0	773000
Unit specified in switch register	773002
RK11 (RK05 disk) -- unit 0	773030
Unit specified in switch register	773032
RH11 (RP04 disk) -- unit 0	773320
Unit specified in switch register	773322
RP11 (RP03 disk) -- unit 0	773350
Unit specified in switch register	773352
RF11 (RS11 disk)	773136
TC11 (DECTape)	773070
TM11 (TU10 magnetic tape)	773110
RH11 (TU16/TM02 magnetic tape)	773150

2.3.5 M9301-YC Bootstrap (PDP-11/70 Only)

The M9301-YC bootstrap operates only on the PDP-11/70. Perform the following steps to use this bootstrap.

1. Press the HALT switch and set back to the ENABLE position.
2. Set the start address of 177765000 in the console switches.
3. Press LOAD ADDR.
4. Set the device unit number in switches 0 through 2.
5. Set the device code in switches 3 through 6. Refer to Table 2-7 for device codes.
6. Ensure that switches 7 through 21 are off (down).
7. Press START.

NOTE

Before the M9301-YC bootstrap actually boots the system, it performs CPU tests, instruction and addressing tests, and memory and cache tests. If a hardware failure is detected, the diagnostic program halts. The lights contain the ROM address of the halt. If this occurs, call the DIGITAL field service engineer.

It may, however, be possible to continue with the bootstrap operation if the lights contain the address 17773764, which indicates a cache failure. To continue in this case, press CONT. This is the ONLY case in which it is possible to continue bootstrapping after the diagnostic detects an error.

Table 2-7
Device codes for M9301-YC Bootstrap -- Switches 3 through 6

DEVICE	CODE
TM11/TU10 Magnetic Tape	1
TC11/TU56 DECTape	2
RK11/RK05 DECpack Disk Cartridge	3
RP11/RP03 Disk Pack	4
RH70/TU16 Magnetic Tape	6
RH70/RP04 Disk Pack	7
RH70/RS04/RS03 Fixed Head Disk	10

NOTE

The M9301-YC bootstrap is capable of relocating its input to various banks of memory. However, it is not possible to do this with RSX-11D. An RSX-11D system must be located as defined during system generation.

2.4 TERMINAL HANDLER GENERATION

2.4.1 Introduction

Before phase 1 of system generation is performed it is necessary to build the terminal handler to reflect the configuration of terminals in the system, unless the single terminal handler (TT01) is to be used. There are two files which must be edited. Of these [311,114]PARAMS.MAC, describes in general terms the facilities required, the number of each type of interface in the system etc. [311,114]CONFIG.MAC describes each interface in detail and how every terminal is connected. Once these files have been set up the handler must be assembled and built as described in Section 2.4.4.

2.4.2 Editing PARAMS.MAC

This file is divided into three parts:

1. Assembly parameters which must be set up for every system (see Table 2-8).
2. Assembly parameters which will not normally need to be changed. The values in the second column of Table 2-9 are those in the software distributed by DIGITAL. They may need to be changed to suit the particular needs of some installations.
3. Assembly parameters whose definitions must not be changed.

2.4.3 Editing CONFIG.MAC

This file is in two parts, the interface descriptions and the terminal descriptions. Each interface is described by a line of the form:

```
INTF          number,type,epa,vec[,extra]
```

number is the interface number. The first interface must be number 0, and the remainder must follow in consecutive ascending order. Interface 0 must be the console DL11.

type is the interface type, selected from DC, DH, DJ, DL, DM, DZ.

epa is the external page address of the first register of the interface; e.g., 177560 for the console.

vec is the address of the first interrupt vector

extra may be absent. It is used to contain interface dependent information, as follows:

```
DC   -   indicates the type of DC11
      0 - DC11AX
      1 - DC11AA
      2 - DC11AB
      3 - DC11AC
      4 - DC11AD
      5 - DC11AE
      6 - DC11AG
      7 - DC11AH
```

GETTING ON THE AIR

- DH - if the DH11 has a corresponding DM11 for dialup line control, this must be the number of the DM11 interface. The DM11 must precede the DH11 in this file.
- DL - If 'extra' is present and non-zero, the interface is a DL11E.

Each terminal is described by a line of the form:

TERM number,intnum,subline,type,speed,dialup,char

number is the terminal number. The first terminal must be number 0, and the remainder must be in consecutive ascending order.

intnum is the number of the interface (in the previous section) to which this terminal is connected.

subline for terminals connected to multiplexor interfaces this is the subline number on the interface. For single-line interfaces it may be left blank.

type is the type of the terminal, selected from: AS33 (ASR33), KS33 (KSR33), AS35 (ASR35), L30S (LA30S), L30P (LA30P), LA36, VT05, VT50, VT52, VT55, VT61. If the terminal is not one of these DEC-supported models, the type should be given as 'USR0' and the 'char' parameter (below) used to describe its characteristics.

speed If the default speed for the terminal is to be used (see Device Handlers Reference Manual, Table 2-3), this may be left blank. Otherwise, it should be a number which is the baud rate of terminal or, for a split-speed line, two numbers separated by a comma and enclosed in angle brackets, with the keyboard (lower) speed first, e.g. <150,2400>.

dialup if the terminal is connected to a dialup line this parameter must be the string 'DIALUP' otherwise it should be blank.

char If any of the terminal characteristics are to be non-standard they may be specified here. The parameter is a list of pairs of characteristic names and values; e.g., <<ch1,0>>, <ch2,3>, <ch3,1>>. The names are those specified in the Device Handlers Reference Manual, Chapter 2, Table 2-2. The value may be any acceptable value as specified in that table. The number of pairs allowed in this list is limited only by the length of the line. If still more are needed an alternative form of the TERM macro may be used:

```
TERMBG number,intnum,subline,type,speed,dialup
SETFLD ch1,vall
...     ...
...     ...
SETFLD chn,valn
TERMED
```

GETTING ON THE AIR

A separate line must be used for each characteristic.

An example CONFIG file is given as Appendix A.1. This example applies to a system with:

1. LA36 as a console.
2. DH11 interface with five VT50s and three LA30Ss.
3. DH11 interface with associated DM11 for dialup lines, with six dialup lines set up as ASR33s and two local VT05s, one running at a non-standard low speed.
4. A DL11E dialup line set up as a LA36.
5. A DJ11 with two low speed hardcopy terminals and two high-speed terminals, connected to different four-group sublines.
6. A DL11 line with a special, non-standard terminal running at 150/1200 baud, split-speed.

2.4.4 Building the Handler

The following commands should be used to build the handler:

```
MCR>SET /UIC=[1,1]
MCR>MAC @[311,114]TTMAC.CMD
MCR>TKB @[11,114]TTTKB.CMD
```

NOTE

The TKB operation will result in four 'undefined symbols' messages, for the segments TT, INIT, IDLE and LINES. The exact number of undefined symbols in each message will depend on the assembly parameter files.

If the assembly parameter files have been set up incorrectly it is possible that some assembly errors will be reported. It is possible to check for these first, to minimise wasted time, using the command file [311,114]TTMACCHK.CMD. This will assemble only the modules which can produce errors. If the CONFIG file is changed but not PARAMS, for example because some interfaces have been relocated in the external page, CONFIG may be assembled using the command file [311,114]CONFIGMAC.CMD.

The normal version of the terminal handler is overlaid, one overlay segment containing the code to deal with I/O requests and the other containing the initialization and powerfail code. This means that the handler has to access the disk on which it is resident when the system is booted or during powerfail recovery. For some applications this may not be practical in which case it is possible to build a non-overlaid version. This is used, for example, in the distribution kit where there is no system disk to read overlays from. The non-overlaid handler may be built using the TKB command file [11,114]TTUNOVR.CMD. In this case the symbol O\$\$VR in PARAMS.MAC must be 0. It is recommended that I\$\$ERM also be 0 so that space is not used to store the initialization error messages.

Table 2-8
Terminal Handler Group 1 Assembly Parameters (2.4.2)

See section 2.4.2

Name	B if Binary	Meaning
U\$\$NTS	9	Maximum number of terminals to be serviced.
B\$\$22	B=1	=1 if 22-bit support required; must be set on PDP-11/70 configurations with more than 124K words of memory.
B\$\$AW	B=1	'Bells and Whistles'. This should =1 unless space is at a premium. If it =0 the features in Table 2-10 will all be omitted.
B\$\$LCK	B=0	=1 for Block-Mode terminal support.
D\$\$C11	0	Number of DC11 single-line interfaces in the configuration.
D\$\$CDU	B=0	=1 if any DC11 interfaces are connected to modems for dial-up support.
D\$\$H11	0	Number of DH11 16-line NPR multiplexor interfaces.
D\$\$IAL	B=0	Dialup Support. =1 if any dialup lines required in the system. =0 causes system to ignore dialup lines.
D\$\$J11	0	Number of DJ11 16-line interfaces in the configuration.
D\$\$L11	0	Number of DL11 single-line interfaces in the configuration.
D\$\$LDU	B=0	=1 if any DL11(DL11E) interfaces are connected to modems for dialup support.
D\$\$M11	0	Number of DM11BB 16-line modem control interfaces in the configuration. These interfaces provide dialup support for lines interfaced through DH11s.
D\$\$Z11	2	Number of DZ11 8-line interfaces in the configuration.
D\$\$ZDU	B=0	=1 if any DZ11 interfaces are connected to modems for dialup support.
E\$\$EQ	B=1	=1 if handler is to recognize and parse escape sequences. (see <u>IAS/RSX-11D Device Handlers Reference Manual</u> , Section 2.6).
H\$\$RTZ	60.	Clock frequency (50., 60. or 100.). The . is necessary and indicates a decimal number.
T\$\$APE	B=0	=1 if low speed paper tape is required. =0 if no low paper tape is required.

GETTING ON THE AIR

Table 2-9
Terminal Handler Group 2 Assembly Parameters

See Section 2.4.2.

Name	Normal Setting	Meaning
B\$\$SCN	=B\$\$AW	=1 to allow typeahead CTRL/R and Backspace Over Tab.
B\$\$SP	=B\$\$AW	=1 for support of terminals with hardware backspace. If it = 0, backspace will be output but the handler will lose track of terminal's horizontal position.
D\$\$HSF	=B\$\$AW&D\$\$Hll&B\$\$LCK	=1 If support of DHll Silo Fill is required. This should be used if block mode terminals are present to reduce the number of interrupts.
D\$\$KIL	=0	=0 for notification of termination of requests by IO.KIL.
D\$\$RAT	=2	Default Read-ahead Processing Type, viz. 0 Type-ahead is illegal and will be ignored. 1 'Deferred Processing' Mode. Characters typed ahead will only be processed when they are read. Thus RUBOUT, CTRL/U etc. will not be seen until a read is performed. 2 'Immediate Processing, Deferred Echo' Mode. Characters are processed as they are typed but not echoed till they are read. Echo is therefore a 'clean copy'. 3 'Immediate Echo' Mode. Characters will be echoed as they are typed, and not echoed when they are read. See also the <u>IAS/R SX Device Handlers Reference Manual</u> , Section 2.2.2
NOTE		
Read-ahead type can be changed in the configuration file for particular lines.		
E\$\$ALT	=1	=1 if ALTMODE is to be echoed as '\$' followed by carriage return, =0 if ALTMODE is not to be echoed.
H\$\$OLD	=B\$\$AW	=1 for support of VT5x hold-screen mode. This support is provided automatically if E\$\$EQ=1.
I\$\$ERM	=B\$\$AW	=1 to generate initialization Error Messages (see Section 2.4.5).
L\$\$30S	=1 = 0	=1 to support LA30S Serial DECwriters.

GETTING ON THE AIR

Table 2-9 (Cont.)
Terminal Handler Group 2 Assembly Parameters

Name	Normal Setting	Meaning
M\$\$ANS	=2	Time, in seconds, to wait after sensing first ring before answering the phone. The parameters M\$\$ANS through M\$\$WCR are described more fully in the <u>IAS/RSX-11D Device Handlers Reference Manual</u> , Table 2-5.
M\$\$CAR	=2	Time in seconds, to wait after answering the telephone before applying carrier.
M\$\$RNG	=4	Time, in seconds, to wait for a confirming ring signal. If a ring signal is not detected in this interval the caller is assumed to have hung up. This time must be greater than the period of the ring cycle (3 seconds in USA).
M\$\$SUK	=1	=1 if UK-type Short Modem Timeouts required. Some countries, including the UK, require a line to be dropped within a very short time after detecting carrier loss, because their modems and some acoustic couplers do not distinguish carrier from dial tone. This facility is not available for modems interfaced via a DZ11.
M\$\$WIC	=20.	Time, in seconds, to wait for caller to apply carrier after the system has applied carrier. This allows the caller time to place the phone in its bed on an acoustic coupler, without tying up the line for too long when a wrong number is dialled and then cancelled.
M\$\$WCR	=3	Time to wait after carrier loss before assuming caller has hung up. If M\$\$SUK (above) =0, this time is M\$\$WCR seconds. It should allow for the phone being disturbed on the bed of an acoustic coupler or for dropouts on the line. If M\$\$SUK =1, this time is M\$\$WCR clock ticks. It should be such that the line is guaranteed to be dropped in the required time.
M\$\$PRI	=5 = 4	Priority of Interrupt Service Routines. =5 unless there are no multiplexor interfaces in the system, in which case it can =4. It may be useful for configurations with no block-mode terminals to use a BR4 Jumper in DH11/DJ11/DZ11 interfaces to improve the service to disks etc, when M\$\$PRI should be changed accordingly.
N\$\$L	=B\$\$SAW = 0 ?	'Newline' terminal support should =1 for support of terminals which treat code 12 (octal) as 'Newline' rather than 'Line Feed' and have a 'newline' key instead of 'carriage return'.
N\$\$ODS	(see text) 54	Total number of <u>16-word</u> nodes in handler's internal node pool. Nodes are picked from the pool for all internal buffering. This includes type-ahead, reads,

GETTING ON THE AIR

Table 2-9 (Cont.)
Terminal Handler Group 2 Assembly Parameters

Name	Normal Setting	Meaning
		writes and the saving of default values when terminal characteristics are reset. The distributed handler calculates the value from U\$\$NTS (Table 2-8) thus: For U\$\$NTS from 0 to 5 allow 5 nodes per terminal. For each terminal from 6 to 32 allow 3 more nodes. For each terminal beyond 32 allow 2 more nodes.
O\$\$VR	=1	=1 to build overlaid version of handler. The handler should normally be overlaid (see section 2.4.4).
S\$\$CHR	=B\$\$AW	=1 to enable 'set characteristics' function (see <u>IAS/RSX-11D Device Handlers Reference Manual</u>). The MCR TER command relies on this function. The function uses over 1/2K words of memory and may be omitted for small systems.
G\$\$CHR	=S\$\$CHR	=1 to enable 'get characteristics' subfunctions of 'set characteristics' (SF.GSC, SF.GMC, see <u>Device Handlers Reference Manual</u> , Sections 2.4.3.5, 2.4.3.6). Allows applications programs to determine characteristics of terminal on which they are running.
D\$\$CHR	=S\$\$CHR	=1 to enable dump characteristics subfunction (SF.GAC, SF.SAC in <u>IAS/RSX-11D Device Handlers Reference Manual</u> , Sections 2.4.3.7, 2.4.3.8). Allows application programs to change terminal characteristics and restore them on exit.
R\$\$BTB	=B\$\$AW&B\$\$SCN	=1 to enable scope tab rubout. Rubout over tab then moves cursor to where it was before tab was typed. =0 saves about 60 words but rubout over tab then echoes as backspace-space-backspace.
S\$\$FF	=B\$\$AW	=1 for software simulation of form feed and vertical tab. On terminals with characteristics set correctly this will replace a token output of a fixed number of line feeds. Used for example when a slow high-quality printer is connected via a terminal interface.
V\$\$FIL	=1 ?	=1 sets VT05-type vertical fill (see <u>Device Handlers Reference Manual</u>). =1 is required if any VT05 is configured in the system.

GETTING ON THE AIR

Table 2-10
Parameters Dependent on B\$\$AW in the Handler as Distributed

Parameter	Meaning	Parameter	Meaning
B\$\$SCN		H\$\$OLD	
B\$\$SP	see	I\$\$ERM	see
D\$\$CHR	Table	N\$\$L	Table
D\$\$HSF	2-9	R\$\$BTB	2-9
G\$\$CHR		S\$\$CHR	
		S\$\$FF	

2.4.5 Terminal Handler Error Messages

The terminal handler performs some checking of the consistency and sense of the parameters used to build it. Some of these errors are detected at assembly time, as described above. Others cannot be detected until the handler becomes active, at the beginning of System Generation Phase 2. These are printed when the system is first bootstrapped, and have the form:

TT.... *FATAL* error message

If the error is related to a particular terminal or interface, its number is printed after the message. After this message, SGN2 will print:

UNABLE TO FIND ATL FOR TERMINAL HANDLER(TT....)

The system is not usable, and the terminal handler must be rebuilt correctly and System Generation repeated.

The handler abandons initialization as soon as an error is found. For this reason if more than one error condition is present, only the first is reported.

Exceptionally, the message 'FAILED TO ALLOCATE UNIBUS MAP REGISTER' may be output when a system is booted after System Generation, if it is moved to a system with more than 124K of memory after running on a smaller system.

Error message texts:

DH11 MODEM INTERFACE IS NOT A DM11

The interface specified as the 'extra' parameter for a DH11 is not a DM11.

DH11 MUST FOLLOW CORRESPONDING DM11

The interface specified as the 'extra' parameter for a DH11 has a number greater than that of the DM11. It is essential that the DM11 precede the DH11 so that initialization can be performed in the right order.

FAILED TO ALLOCATE UNIBUS-MAP REGISTER

On a PDP11-70 with more than 124K of memory, the handler needs to allocate a Unibus Map Register (UMR) to service DH11 lines. If all of the 31 available registers are in use by other handlers, this message will occur. Normally, this will only occur if the system is moved from a small (<124K) machine to a larger one, if a large number of Unibus NPR devices (e.g. RK, TU10, RP03) are on the system.

FAILED TO CONNECT TO INTERRUPT VECTOR

Two conditions may cause this error:

1. There is an error in the CONFIG file such that more than one interface is specified as having the same interrupt vector address.

GETTING ON THE AIR

2. Another handler is connected to one of the interrupt vectors specified in the CONFIG file. This indicates an error either in CONFIG or in the parameters supplied to Sysgen 1.

The message indicates an error either in CONFIG or in the parameters supplied to phase 1 of system generation.

FAILED TO DECLARE HANDLER

Another version of the handler, or another terminal handler, is already active.

FAILED TO DECLARE POWER FAIL AST

This error should not occur. It may do if the terminal handler build file is changed to reduce the node pool limit of the handler.

ILLEGAL DC11 SPEED TYPE

The speed type of a DC11 interface, specified as the 'extra' parameter, is not in the range 0-7.

LINE SPEED NOT VALID FOR THIS INTERFACE

The speed specified for a terminal, either implicitly by the terminal type or explicitly as the 'speed' parameter to the TERM macro, is not available on the specified interface. This will occur, for example, if a VT50 is connected to a DC11 without an explicit speed specification suitable for the interface.

MORE THAN ONE DH11 REFERS TO ONE DM11

More than one DH11 interface has specified the same speed specification suitable for the interface.

MORE THAN ONE DH11 REFERS TO ONE DM11

More than one DH11 interface has specified the same DM11 modem control interface as the 'extra' parameter.

MORE THAN ONE TERMINAL ON SINGLE-LINE INTERFACE

More than one terminal has specified the same single-line interface.

NO MODEM INTERFACE FOR DIALUP LINE

A terminal has the 'dialup' parameter set but the interface to which it is connected has no dialup capability.

SLAVE INTERFACE USED AS PRIMARY INTERFACE

The interface specified for a terminal in the TERM macro is a DM11.

TWO TERMINALS ON SAME SUBLINE

More than one terminal has specified the same subline of the same multiplexor interface.

CHAPTER 3

SYSTEM GENERATION PROCEDURES

RSX-11D provides a system generation capability that consists of two tasks (phase 1 and phase 2) and a set of command files. The first phase can be performed online independently of the target hardware or software configuration. Its purpose is to create a file on disk that is an RSX-11D system capable of running on the target hardware configuration.

Phase 2 of system generation is designed to perform those functions that require the target configuration. Phase 2 is a task that is activated in the system created by phase 1.

If the generated system is being written onto the current system disk, i.e., if a target disk is not being used, system generation must not be performed online with other system operations.

Both phases accept input from command files. Phase 1 also accepts input from a terminal. Chapter 5 details the procedures for phase 1 input from a terminal. The distributed system includes several system generation command files under UFD [11,17]. The command files that can be used as input to phase 1 all have the following format for the filename and type.

diskznnK.CMD

disk indicates the disk that is to be the system device, i.e., RK05, RK06, RP02, RP03, RP04, RP05, RP06.

z indicates the programmable clock (P) or the line frequency clock (L).

nn indicates the size of the save image in units of 1K words

Any one of these files can be used as input to phase 1.

Appendix A provides a sample listing of the files.

The phase 2 input file is SYSBLD.CMD unless an alternative is specified by the SYSBLD directive in phase 1. It is located under UFD [11,17]. Phase 2 automatically uses this file to create a running system on the target hardware. SYSBLD.CMD can be edited prior to running phase 2 to reflect local requirements. Appendix A contains a listing of SYSBLD.CMD.

SYSTEM GENERATION PROCEDURES

5. Prepare the required phase 1 input command file. Several files are provided with the distributed system.
6. Depending on the available pool size of the host system, may be necessary to remove some tasks from the host system. Section 5.2.7 provides guidelines for calculating node pool usage during system generation.

A file under [11,17] called SYSGENREM.CMD can be used to remove tasks. It is advisable to use this file. However, if others are using the system, care should be taken not to remove tasks they require. Enter the following command to remove installed tasks using SYSGENREM.CMD.

```
MCR>REM @[11,17]SYSGENREM
```

7. Run phase 1 ensuring that the command is terminated by pressing ALTMODE.

```
MCR>RUN SGN1          Depress ALTMODE.
```

8. The system responds as follows.

```
SYSGEN PHASE 1
SPECIFY TARGET DEVICE AND FILENAME
SGN>
```

9. At this point, the user has the option of naming the device UFD, and filename to be assigned to the file produced in phase 1. The directive has the following format.

```
SGN>TARGET=td:[ufd]filename
```

td indicates the target disk. If it is omitted, SY is used by default.

[ufd] is the UFD of the system image file. If it is omitted, [11,17] is used.

filename is the name to be assigned to the file. If it is omitted, RSX.SAV is used.

If the TARGET directive is not specified, the default is SY:[11,17]RSX.SAV. This directive may have been included in the command file to phase 1. In this case, proceed to the next step.

SYSTEM GENERATION PROCEDURES

10. Name the command file to be processed by phase 1 of system generation (SGN1). The command file can be any one of those provided with the system, e.g., RP03L48K.CMD, or it can be a user-created file. The following command to SGN1 is used.

SGN>@command file specification

All of the commands read from the command file are printed on the terminal. Upon successful completion of the processing of the command file, the following message appears on the terminal.

SGN>END OF PHASE 1

Depress CTRL C to obtain MCR.

The host system remains usable in the normal way even if fatal errors occurred during execution of phase 1.

11. If any errors occurred during phase 1, correct them (consult the Messages Appendix) and repeat steps 7 through 10.

NOTE

Fatal errors must be corrected, but diagnostic messages imply that the resulting system may be usable.

12. Phase 1 of system generation does not produce a target system that is hardware bootable; i.e., it does not write physical block 0 of the disk.

Do not log off the host system. Consult step 1 of the phase 2 procedures (Section 3.2) to be able to boot the target system.

3.2 PHASE 2 OF SYSTEM GENERATION

Phase 1 of system generation has created a file on the target system disk. This file contains a memory image of a running RSX-11D system.

In this system, two tasks are active. One is the system disk handler, and the other is phase 2 of system generation. Bootstrapping this file into memory causes phase 2 to run.

NOTE

For phase 2 to execute successfully, it must be booted on a machine with as much or more memory than that specified during phase 1.

Phase 2 of system generation processes by default a command file named [11,17]SYSBLD.CMD. If any modifications need to be made to this file or its equivalent to reflect system requirements, they must be made prior to execution of phase 2. If an alternative file has been specified by the SYSBLD directive in phase 1 this file will be used in place of SYSBLD.CMD.

SYSTEM GENERATION PROCEDURES

The steps necessary to run phase 2 follow.

1. Bootstrap phase 2. There are two ways to bootstrap phase 2 into memory on the target system. Whichever way is used, it must be booted from the unit specified as SY during phase 1. Methods a and b follow.

- No a. Create a bootstrap block 0 on the target disk and use the hardware bootstrap. Issue the following command on the host system to create bootstrap block 0.

```
MCR>BOO target disk:filename/WB
```

filename is the name previously specified in the TARGET directive.

This command takes virtual block 1 of the file created by phase 1 and writes it on physical block 0 of the target device. The disk is now hardware bootable (see Section 2.1). It can be taken to the target hardware and booted.

- b. Use the software BOOT function. The MCR software bootstrap function is described in the RSX-11D User's Guide. The software bootstrap boots the file created by phase 1 into memory in the host system. Block 0 of the target system is not modified.

When the software bootstrap function without the /WB switch is executed, no other users should be online to the host system. The host system is overlaid in memory by the target system. The host system disk remains unaltered.

Before issuing the BOOT command, ensure that the target system disk is mounted on the unit specified as SY during phase 1. Then issue the following command to the host system.

```
→ MCR>BOO target disk:filename
```

Notice that the MCR BOOT command can be used either to create the bootstrap block on a system disk by including the /WB switch or to perform the bootstrapping from any disk by omitting the /WB switch. The two functions are mutually exclusive.

If the bootstrap is currently written for a system image on the disk (by means of a previous /WB) it is often advisable not to write the bootstrap at this point. Not doing so retains the ability to obtain the old system on hardware booting.

Writing the bootstrap here means that such a hardware boot would initiate the System Generation Phase 2 image. This can be useful if Phase 2 fails for any reason.

The function BOO is discussed in the privileged user section of the RSX-11D User's Guide. Chapter 5 of this manual contains a detailed discussion of bootstrapping.

SYSTEM GENERATION PROCEDURES

2. Once the system has been bootstrapped, phase 2 prints the following message.

```
*** SYSTEM GENERATION PHASE 2 ***
```

Phase 2 proceeds to mount the new system disk and then open and read the file SYSBLD.COMD. All commands encountered in SYSBLD.COMD are executed and echoed on the terminal.

NOTE

If the terminal handler has been built incorrectly it may print an error message at this point. In this case System Generation phase 2 will fail and the system will not be usable. See Section 2.4.5 for a list of these error messages and their meanings.

3. Upon successful completion, phase 2 prints the following message.

```
*** END OF SYSTEM GENERATION PHASE 2 ***
```

4. Set the time and date, redirect the console logging device (CL), load the message output handler, and perform any other similar functions; e.g., loading other handlers.

```
MCR>TIM hh:mm:ss dd-mmm-yy
MCR>RED LP:=CL:
MCR>LOA MO
```

Additionally, to prevent the fragmentation of the GEN partition, fix the file primitives using the following procedure.

```
MCR>DMO SY:      Also dismount any other mounted volumes.
MCR>FIX FllACP
MCR>MOU SY:/OVR
```

5. If you are satisfied at this point that the system generation was successful, dismount any mounted devices and perform a save using the MCR SAV command. The following is an example.

```
MCR>DMO SY:
MCR>SAV
```

Once the system is saved, the following message is printed on the console.

```
nnK WORDS RSX-11D Vxxxx
```

where:

nn is the memory size.

xxxx is the Executive version number.

At this point, SAV may issue messages describing memory size adjustments; see Section 5.5 and the discussion of SAV in the RSX-11D User's Guide. The MCR prompt follows the messages.

SYSTEM GENERATION PROCEDURES

NOTE

If at any time before performing the SAV, a reboot of the new system disk is performed, as described in step 1 above, phase 2 of system generation is executed again.

A reboot after a SAV causes the same response as seen after performing the SAV above, i.e., nnK WORDS RSX-11D. The SAV function is discussed in the privileged user section of the RSX-11D User's Guide.

6. If the user is satisfied with the system generation and wants the disk to be hardware bootable, the functions of step 1a in this section should be performed if they have not been done already.

NOTE

It may be desirable to have the system automatically mount the system disk and request the time whenever it is booted. This can be done using type ahead after dismounting all volumes. It is recommended that the type ahead procedure be used after a test run of Step 6 above. The following is an example.

```
MCR>DMO SY:           Also dismount any other mounted
                       volumes.
MCR>↵or ($)          Type either RETURN or ALTMODE
SAV                   Type ahead
MOU SY:/OVR           Type ahead
TIM                   Type ahead
```

At this point, press CTRL/C to cause the save, the sign-on messages, and the execution of those functions typed ahead. Typing only a single space after TIM leaves MCR ready for the operator to insert the current time and date followed by RETURN. The type ahead feature also can be used for comments by prefixing the comment with a semicolon (;) or an exclamation mark (!), to a maximum total of 80 characters.

Type ahead is available only on the multiterminal teletype handler, not on the single-terminal handler TT01.

CHAPTER 4

SYSTEM GENERATION DIRECTIVES

Phase 1 and phase 2 of system generation are accomplished by issuing a series of commands to the system. The commands issued during phase 1 are called directives and are described in this chapter. Most commands issued during phase 2 are MCR commands. One non-MCR phase 2 directive *DELAY also is described in this chapter.

Many system generation directives can have multiple parameter sets for a single occurrence of the directive name. Multiple sets are separated from each other by backslash (\) characters. The examples below illustrate two PAR directives first and then the same two directives using the convention for multiple parameter sets.

```
PAR=JIM,,,U  
PAR=JOHN,,,U
```

or

```
PAR=JIM,,,U\JOHN,,,U
```

The method used to specify these directives to system generation is detailed in Chapter 5. Examples of the directives are provided in Appendix A. These examples should be used as guidelines in tailoring individual systems.

SYSTEM GENERATION DIRECTIVES

4.1 TARGET DIRECTIVE

The TARGET directive is used to define the target device, i.e., the device on which phase 1 is to create its output file and obtain all required tasks for installation. The TARGET directive has the following format.

TARGET=xyn:[ufd]filename.ext

xyn: is the device mnemonic and unit number. The default is SY0.

[ufd] is the UFD under which phase 1 is to store the system file. The default is [11,17].

filename.ext is the filename and extension used to name the file created by phase 1. The default is RSX.SAV.

If the TARGET directive is omitted, phase 1 uses the following file specification.

SY0:[11,17]RSX.SAV

Only one target specified is permitted.

SYSTEM GENERATION DIRECTIVES

4.2 PDP11 DIRECTIVE

The PDP11 directive is required to define the target system processor configuration to system generation. The directive has the following format

PDP11=cpu-type,[mem-size],[fpt-optn],[clock-freq]

cpu-type = nn to specify a PDP-11/nn where nn is 34,35,40,45,50,55 or 70. These are the only accepted values.

mem-size = physical memory size (to be saved and booted) specified as nnnnK words. nnnn must be a decimal integer. If this parameter is omitted, 32K is used as the memory size. The mem-size specified must be equal to or less than the amount of memory on the target hardware. The maximum memory size permitted is 124K for a PDP-11/34,35,40,45,50 or 55 and 1920K for a PDP-11/70. (See Section 5.5.)

fpt-optn = FP to indicate that the PDP-11/34,45,50,55 or 70 floating point processor option is to be used; otherwise, the parameter is omitted. This option is illegal for a PDP-11/35 or 40.

clock-freq = system clock ticks per second for the standard clock, or interrupts per second, clock identification, and clock ticks per interrupt for the programmable clock.

For the standard clock, specify a decimal number that is used as the line frequency (normally 50 or 60). The system runs with the line frequency clock at the specified frequency. If the parameter is omitted, 60 Hz is used.

For a programmable clock, three decimal numbers are specified.

1. Frequency of clock interrupts per second; e.g., 1000.

2. Clock identification

- 0 = 100 KHz
- 1 = 10 KHz
- 2 = line frequency
- 3 = external

3. Number of clock ticks per interrupt

The three numbers must be enclosed in angle brackets and separated by commas as in the following example:

```
<100,0,1000>  
60,2,1
```

The above example indicates that the programmable clock is to be used as the system clock. It is interrupting 100 times per second using the 100 KHz clock with a count of 1000 loaded into the count register.

4.3 EXEC DIRECTIVE

The EXEC directive is used to load the RSX-11D Executive into memory starting at a specified 32-word boundary. An address higher than the bootstrap program must be used. See Section 5.4.1 for bootstrap sizes.

The EXEC directive has the following format.

EXEC=base

base = a 32-word boundary at which the Executive is to start. The octal number specified in the EXEC directive is multiplied by 100 (octal) by system generation to determine the starting address. For example, if base is specified as 540, the starting address is memory location 54000 (octal).

The EXEC directive is optional. If the EXEC directive is not included, the Executive starts at the next 32-word boundary after the bootstrap program and base addresses specified in subsequent directives are ignored.

If the EXEC directive is included, a base address must also be specified in every directive that contains base as an optional parameter.

If a PDP-11/70 system in excess of 124K is being generated, the Executive, SCOM, and system disk partition must be positioned completely below the 124K boundary. Furthermore, the partition into which SGN2 is installed must have sufficient space below the 124K boundary for SGN2 to fit completely. Phase 1 checks for all the above conditions and issues an error message if any is violated.

SYSTEM GENERATION DIRECTIVES

4.4 SCOM DIRECTIVE

The SCOM directive allocates space for the system common area. The system common area comprises the system subroutines; a communication region; the system tables, including the system task directory; and the node pool. The system common area cannot exceed 12K words.

The SCOM directive has the following format:

SCOM=[base],size,std-entries

base = starting address for the system common area. This location is specified using the same conventions as for the base address in the EXEC directive. If the EXEC directive is included in system generation, the base parameter must be part of the SCOM directive. If this parameter and the EXEC directive are omitted, the system common area follows the Executive in memory.

size 12K = total length of the system common area. The size can be specified as an octal number of 32-word blocks or as nnK. When the form nnK is used, the number (nn) is multiplied by 1024 words to determine the desired size. The factors involved in determining the size of SCOM are detailed below in Section 4.4.1.

std-entries = decimal number of system task directory entries. This parameter establishes the maximum number of simultaneously-installed tasks (user and system tasks) that the system is to support.

One SCOM directive is required.

4.4.1 SCOM Size

The size of the system common area (see Section 4.4) is the sum of the following elements:

1. The length in words of the system subroutines and the communication area. This is calculated as follows:
$$\text{Subroutines Size} = (160000 - .CRTSK)/2 \text{ words (octal)}$$
2. The number of tasks installed at any given time multiplied by 16 words,
3. The number of system task directory (STD) entries specified by std-size, above,
4. The number of DEV directives multiplied by 25 words,
5. The number of PAR directives multiplied by 10 words,
6. Size of the variable-length node pool.

To determine the approximate amount of remaining space in the pool, subtract items 1 through 5 from the SCOM size parameter as follows.

Pool size = SCOM size - (sum of items 1 through 5)

SYSTEM GENERATION DIRECTIVES

The node pool is used by the Executive and many of the system utilities as dynamic storage. Each node consists of a variable number of 8-word blocks. If the system runs out of pool space, its performance degrades and results may be unpredictable. Until experience is gained with the system, leave as much space as possible for the pool.

SYSTEM GENERATION DIRECTIVES

4.5 PAR DIRECTIVE

The partition directive establishes the name, base address, size, and type of every partition in the target system. Every partition in the system must be defined during system generation.

The PAR directive has the following format.

PAR=par-name,[base],size,[par-type],[par-pri]

par-name = up to 6 character ASCII name for the partition. It is converted to a radix 50 name by the system.

base = an optional starting address for the partition. This parameter is specified only if a base address is included in the EXEC directive and follows the same rules as those for the EXEC directive.

If base is omitted, the system places the partition optimally in memory.

size = the size of the partition specified as an octal number of 32-word blocks or as nnK. When nnK is used, nn is multiplied by 1024 to determine the desired size.

One partition in the PAR specifications can contain an asterisk (*) instead of an actual size. The size of that partition is the remainder of memory after the bootstrap, Executive, SCOM, and all specified partition sizes have been subtracted from the memory size stated in the PDP11 directive.

par-type = partition type. U indicates user-controlled; S indicates system-controlled; and T indicates that the time-based scheduler is to operate in this partition. This algorithm is described in the RSX-11D System Manager's Guide. If par-type is omitted, the partition is user-controlled.

par-pri = is the priority at or above which the time-based scheduler declares a task to be runnable. It is a decimal value in the range 2 through 200 (PR.A for the minimum and PR.C for the maximum, respectively). This parameter is valid only for T-type partitions. If par-pri is not specified for T-type partitions, a default value specified by the Executive global symbol PR.DFL is used. PR.A, PR.C and PR.DFL are described in the RSX-11D System Manager's Guide.

Multiple parameter sets are allowed.

Unless base addresses are specified, the partitions are placed in memory in the order specified by the set of PAR directives.

If PDP-11/70 generation is being performed, the system disk handler and SGN2 must be completely below the 124K boundary. Refer to Section 5.5 for a discussion of partitioning under RSX-11D.

Only one partition can use the asterisk (*) size specification. Any number of U, S, and T partitions is permitted. At least two partitions are required: one for the system disk handler and one for system generation phase 2.

4.6 DEV DIRECTIVE

The DEV directive is used to characterize completely each device to be controlled. Each use of DEV creates an entry in the system's Physical Unit Directory (PUD). Any device not defined by a DEV directive does not exist as far as RSX-11D is concerned.

Often with DEV it is enough to specify the device mnemonic with its number and the device type. See Appendix A.2 for examples.

If a device is to have non-standard characteristics, the latter can be specified in place of a standard type. See below under 'type' and Appendix B for details of device characteristics.

A device's interrupt trap vector, software priority and external page address may also be taken by default. The default values are listed in Table 4-1.

With a device there can be associated a file primitives task or ancillary control processor task. See below under 'devacp'.

The format of the DEV directive is as follows.

```
DEV=xyn,type[, [vect],[pri],[ext-page][,devacp]]
```

xy a 2-character ASCII mnemonic to be used to refer to the device.

n a 1-or 2-digit octal unit number.

type either a device type that phase 1 uses to determine the device characteristics or the actual device characteristics. Table 4-1 lists the device types recognized by phase 1. Appendix B lists the associated characteristics.

If the four device characteristics are to be specified they must appear between angle brackets as in the following example.

```
DEV=XY0,<3,17,21,1000>,210,5,177000
```

vect the device interrupt trap vector. See Table 4-1 for the default values.

pri the software priority at which the device interrupts are to be serviced. The user should ensure that the software priority is equal to or higher than the hardware level at which the device interrupts unless the interrupt service routine of the handler is re-entrant. See Table 4-1 for the default values.

ext-page the external page address of the device controller. In most cases, ext-page is the lowest address when a device uses several external page addresses. See Table 4-1 for the default values.

devacp is an optional file primitives task name or ACP name for file-oriented devices.

An ACP (ancillary control processor) is a task that assists a device handler in managing a file-structured volume. An ACP also is referred to as a file primitives task under RSX-11D.

SYSTEM GENERATION DIRECTIVES

If an ACP is not specified, phase 1 defaults to F11ACP for directory devices and MTAACP for magnetic tape.

If an ACP task is specified, its name must be 6 characters ending in ACP. The first three characters are stored in the PUD and used at mount time. The ACP task must be installed before the device can be mounted.

If the vector, priority and external page address are not specified, they will be defaulted to the values shown in Table 4-1.

For examples of the DEV directive see Appendix A.2.

SYSTEM GENERATION DIRECTIVES

Table 4-1
Device Information

Mnemonic	Type	Normal Vector	Priority	External Page Address
DK	RK03 RK05 (5)	220	5	177400
DF	RFn (1)	204	5	177460
DP	RP03B,RP02	254	5	176710
DS	RS03	204	5	172040
DS	RS04 (2)	204	5	172040
✓DB	RP04,RP05,RP06 (2)	254	5	176700
DM	RK06	210	5	177440
DX	RX01	264	5	177170
✓TT	CONSOL (3)	60	4	177560
✓TT	TERM	Float	4	Float
LP	LP11A (80Col)	200	4	177514
✓LP	LP11B (132Col)	200	4	177514
LP	Lx11yz (4)	200	4	177514
✓CR	CR11	230	6	177160
CR	CD11	230	4	172460
MT	TU10,TS03	224	5	172520
✓MM	TU16	224	5	172440
✓PR	PTR	70	4	177550
✓PP	PTP	74	4	177554
DT	DT11	214	6	177340
CT	TAl1	260	6	177500
AF	AF11	134	4	172570
AD	AD01	130	6	176770
LS	LPS11	Float (6)	5	170400
UD	UDC11	234	6	171000
✓MO	None	None	None	None

- (1) n = number of RFl1 platters
- (2) These are disks on the RH Massbus controller.
- The DB unit number must correspond to the number assigned by physical connection to the RH.
- (3) CONSOL must be used for the console terminal TT0. All other terminals must be specified as TERM.
- (4) The line printer type is Lx11yz where
- x = P for LP11 line printer
 - = S for LP05 and LS11 type printers
 - y = W for wide (132 column) printers
 - = N for narrow (80 column) printers
 - z = U translate lower case characters to upper case (ASCII codes 140-176 are converted to 100-136)
 - = L for no translation of characters.
- (5) An RK05F disk is specified as two RK05 units. The unit numbers must be paired as follows for each RK05F unit: (DK0 and DK1), (DK2 and DK3), (DK4 and DK5) and (DK6 and DK7).
- (6) For an LPS11 the vector, priority and external page address must all be specified.

SYSTEM GENERATION DIRECTIVES

NOTE

For various combinations of the above devices, vector addresses and external page addresses may differ from those in Table 4-1. The correct vector and external page address for any configuration can be determined by Field Service Engineers. Useful information can also be found in PDP-11 Peripherals Handbook.

4.6.1 Pseudo-Devices

The following pseudo devices are used in RSX-11D (see Chapter 3 of RSX-11D User's Guide):

SY system disk
TI terminal interface
CI console input
CO console output
CL console log
MO message output
BP batch
SP spooling output

SY, TI, CI, CO, CL are automatically provided. MO, BP and SP if required must be declared in phase 1 by the DEV directive, thus

```
DEV = MO
DEV = SP
DEV = BP
```

and can not be added at a later stage. All except BP may be redirected to particular physical devices after phase 2 by the MCR>RED[IRECT] command, for example

```
MCR>RED DPl:=SP:
```

Both BP and SP are required for BATCH processing. SP must be redirected to a disk. BP is automatically redirected to the terminal invoking batch. This invoking terminal is the one from which batch commands are submitted or, in the case of card input, the one from which the card-reader handler was loaded. See the RSX-11D BATCH Reference Manual, Chapter 4.

SYSTEM GENERATION DIRECTIVES

4.7 SY DIRECTIVE

The system directive establishes the initial SY of the target system, i.e., the system disk to be used when phase 2 of system generation runs. The SY directive has the following format.

SY = xyn = ~~DB0~~

xy = the 2-character mnemonic of the device.

n = the unit number of the device.

The default is DK0.

Phase 1 of system generation checks that the handler task for the device designated as SY is installed. For example, if DP0 is to be the system device, the user must install the handler DP.... during phase 1 of system generation.

Care must be taken to ensure that the correct device is assigned to SY. Phase 1 of system generation assumes that all tasks that phase 1 installs are to be resident on the SY of the target system and creates STDs accordingly.

Since some of these tasks are required during execution of Phase 2, it is necessary to bootstrap phase 2 from the target SY device. See Section 5.4 on software bootstrapping of RSX-11D.

SYSTEM GENERATION DIRECTIVES

4.8 DPAR DIRECTIVE

The default partition directive specifies the partition to be used when a partition is not indicated for a task during task building or when the /PAR portion of an MCR command is omitted.

The DPAR directive has the following format.

DPAR=par-name

par-name = the name of the default partition. It must be a name used in a PAR directive.

The DPAR directive is required.

4.9 DUIC DIRECTIVE

RSX-11D provides the user with an MCR function referred to as RUN \$. The purpose of this function is to allow a nonprivileged user to run any of a set of multiuser tasks that reside under the same UFD on the system disk. The DUIC directive is used to designate the UFD that is to contain these multiuser tasks; normally, the nonprivileged user would not have access to the designated UFD.

The DUIC directive has the following format.

```
DUIC=g,m ←  
g   = the group number  
m   = the member number
```

For example, if the directive DUIC=200,201 is issued during system generation, any user can run a task having the file specification SY:[200,201]task.tsk by entering the following command.

```
MCR>RUN $task
```

If the DUIC directive is omitted, UFD [11,1] is used by default when a RUN \$ command is issued.

It is the system manager's responsibility to ensure that all tasks that are to be available to the nonprivileged user by means of RUN \$ are stored under the UFD designated by the DUIC directive.

SYSTEM GENERATION DIRECTIVES

4.10 CKPNT DIRECTIVE

The checkpoint directive indicates that checkpointing is to be enabled in the system and specifies the disk and disk storage area to be allocated for checkpointing.

The CKPNT directive has the following format.

CKPNT=dev,size

dev = ~~020~~ = name of the device on which the checkpoint area is to reside. The device must be named in a DEV directive.

size = ~~100K~~ = size of the checkpoint area on the disk 'dev'. The size can be specified either as nnK (i.e. words), with nn in decimal, or as an octal number of 32-word blocks.

For example:

CKPNT=DK1,50K

specifies 50K words on DK1, and

CKPNT=DK1,1600

specifies 1600(octal) 32-word blocks on DK1, that is 160000 (octal) bytes.

The device named for checkpointing must contain a Files-11 structured volume and be mounted during phase 2 of system generation. It may also be necessary to install and load the appropriate disk handler before creating the checkpoint file. To create a Files-11 structured volume, use the INI command to MCR in the SYSBLD.CMD command file that is executed during phase 2 of system generation or have a previously created volume on the device. Allocation of the checkpoint file is the last function performed by SGN2 after processing SYSBLD.

The use of checkpointing is optional. However, if it is used, real memory space is allocated to support it. The amount of memory used depends on the checkpoint area of the disk. The relationship is one 32-word memory block for every 512K words of disk space specified for checkpointing. The checkpoint bitmap is at the end of the Executive. This memory allocation must be taken into account when specifying the size of partitions.

SYSTEM GENERATION DIRECTIVES

4.11 PSWD DIRECTIVE

The password directive specifies a password that may be used in the generated system. The PSWD directive has the following format.

PSWD = password of up to 6 alphanumeric characters

NOTE

None of the software provided with RSX-11D uses this password. Passwords are assigned to UICs by the system manager using the PWD command to MCR. This directive is provided so that a user password facility can be developed independently of the RSX-11D log-on procedure. The 6 characters specified in a PSWD directive are stored in the system communication area at location `'MCRPW'`.

SYSTEM GENERATION DIRECTIVES

4.12 INS DIRECTIVE

The install directive installs the system tasks needed to provide the target system with the capability to do useful work. The INS directive has the following format.

INS=par-name,file-specifier,file-specifier,...,file-specifier

par-name = partition name.

file-specifier = An RSX-11D file containing a task to be installed in the named partition. Up to five files can be specified in one INS directive. The file specifier must not include a device mnemonic and must be fewer than 32 (decimal) characters.

Minimally, the disk driver (DK...., DP...., DF...., DB.... or DM....), the terminal handler, F11ACP, and phase 2 of system generation (SGN2) must be installed. See the INS directives in the system generation command files in Appendix A for examples. Refer to the PAR directive (Section 4.6).

To obtain a list of system tasks that can be installed, print a directory of UFD [11,1]. Further discussion can be found in Sections 5.2.6 and 5.5.

All tasks installed during phase 1 are assumed to be on the device specified in the TARGET directive. It is further assumed that they are to be on the target SY when phase 2 executes.

The following limitations apply to phase 1 installation of tasks and Shareable Global Areas (SGAs)

- No more than 15 tasks can be installed in phase 1.
- No more than 15 tasks and SGAs can be installed into a single partition.
- The system disk handler and System Generation phase-2 can not bind to an SGA or have a read-only segment.

4.13 SYSBLD DIRECTIVE

The SYSBLD directive is used to define the file that will be read by SGN2 in processing MCR commands. The SYSBLD directive has the following format.

```
SYSBLD=[ufd]filename.ext
```

where:

[ufd] is the UFD containing the command file. The default is [11,17]

filename.ext is the filename and extension used to name the file to be read. The default is SYSBLD.CMD.

If the SYSBLD directive is omitted Sysgen Phase 2 will read the file [11,17]SYSBLD.CMD. Only one SYSBLD directive is permitted. A device specification is not allowed. The file should exist on the system disk by the time it is booted to perform phase 2 of Sysgen.

4.14 PHASE 2 DIRECTIVE (*DELAY)

The *DELAY directive causes a 1-second delay in the processing of Phase 2 commands. Phase 2 sequentially retrieves and executes MCR commands from the file named by the SYSBLD directive, or, if none was specified, from [11,17] SYSBLD.CMD. Each command is allowed to finish before the next command is executed.

The *DELAY directive does not have any parameters.

CHAPTER 5

ADDITIONAL SYSTEM GENERATION INFORMATION

Once the system is placed on disk from the distribution medium as explained in Chapter 2, system generation can begin. RSX-11D system generation proceeds in two phases.

Phase 1 defines the hardware configuration, specifies system defaults, and installs the system tasks required for execution of phase 2. System generation directives are used to supply the required information.

Phase 2 performs a series of actions requested in the indirect file specified by the SYSBLD directive, or, if none was specified, [11,17]SYSBLD.CMD. Phase 2 information cannot be typed at a terminal. Once all the requests in the SYSBLD file are performed, the newly-generated system is operational.

5.1 EDITING SYSTEM GENERATION FILES

Several indirect command files for system generation are provided in RSX-11D.

They can be used to define the system to phase 1 instead of typing directives in response to phase 1 requests. A SYSBLD file is always used to perform phase 2. If the user wishes to tailor the system to a particular installation, he can either edit the existing files or create new ones. See the RSX-11D Utility Programs Procedures Manual for a description of the editor.

All commands files are stored under UFD [11,17]. Type the following command to request the editor for modification of existing generation files or the creation of new ones.

```
MCR>SET /UIC=[11,17]
MCR>EDI command file
```


5.2 PHASE 1 OF SYSTEM GENERATION

Phase 1 system generation directives are divided into seven categories:

1. The TARGET directive that defines the target system device.
2. The PDP11 directive that defines the computer to be used.
3. The EXEC, SCOM, and PAR, directives that divide the computer memory into the Executive, the system communication area, and partitions, respectively.
4. The DEV directives that define the system peripherals.
5. The DPAR, DUIC, CKPNT, PSWD, and SY directives that supply system default information.
6. The INS directives that install tasks needed for phase 2.
7. The SYSBLD directive which defines the file which is to be processed by Phase 2.

Regardless of whether the directives are to be typed in response to system generation requests or an indirect file is to be used, each category of directives is separated from the preceding category by a record containing only a slash (/). The end of input to phase 1 is indicated by a record containing two slashes (//). See the examples in Appendix A.

To reach the point where the directives are entered, perform the appropriate steps as detailed in Chapter 3 through step 8 of Section 3.1. At this point, the following message is printed on the terminal.

```
SYSGEN PHASE1
SPECIFY TARGET DEVICE AND FILENAME
```

5.2.1 Target Specification

When the message SPECIFY TARGET DEVICE AND FILENAME is printed on the console, the user can type the TARGET directive or the name of the indirect command file to be used. Alternatively, the user can type both; see Section 3.1, steps 9 and 10.

If the TARGET directive is used, it establishes a complete description of where the output file of SGN1 is to be created and with what name. The following example creates a file named 48KV62.SAV on DK1 under UFD [11,17].

```
SGN>TARGET=DK1:[11,17]48KV62.SAV
```

As with any other directive, the TARGET directive can be included in the command file, in which case the procedure is as described in Section 3.1.

ADDITIONAL SYSTEM GENERATION INFORMATION

After entering the TARGET directive, SGN1 again prompts. The response at this point is either a command file name or a slash (/) to cause the CPU specification prompt as in the following example.

```
SGN>TARGET=DK1:[11,17]48KV62
SGN>/
```

ENTER CPU SPECIFICATION

```
SGN>
```

5.2.2 CPU Specification

When the message ENTER CPU SPECIFICATION is printed on the console, the user types the PDP11 directive. Enter the PDP11 directive followed by a record containing only a slash (/) as in this example.

```
SGN>PDP11=40,48K
SGN>/
```

5.2.3 Memory Allocation

The memory allocation directives make it possible for the user to place the Executive and system common area in memory and to establish partitions and shared regions. The allocation of memory is accomplished using the following directives:

- EXEC (optional),
- SCOM (required),
- PAR (required).

These directives can be entered in any order in response to the following request that is printed on the console. The order of the PAR directives specifies the order of partitions in memory, unless base addresses are specified.

SPECIFY DIVISION OF MEMORY

The memory allocation directives must be followed by a record containing only a slash (/).

5.2.4 Device Specifications

Type the DEV directives in response to the following message.

SPECIFY DEVICES

When all directives are entered, type a record containing a slash (/).

5.2.5 System Default Specifications

Once the device specifications are entered, the system requests the default directives as follows.

SPECIFY DEFAULTS

ADDITIONAL SYSTEM GENERATION INFORMATION

Any of the following directives can be typed at this point:

- DPAR,
- DUIC
- CKPNT,
- PSWD,
- SY.

The directives can be in any order. The last directive must be followed by a record containing a slash (/).

5.2.6 Required Task Installations

At the completion of phase 1, a target system exists that can operate as an independent RSX-11D system. It must contain an RSX-11D Executive that can either satisfy its intended end-use requirements or be capable of expanding itself to do so. Therefore, the final portion of phase 1 installs the tasks required to accomplish this end.

When a task is installed, it is given an entry in the system task directory (STD). This entry allows the system to load the task without making use of the file system.

Selection of tasks for installation depends on the following two factors:

1. The capabilities defined for the target system,
2. The procedures to be used in creating the operational target system.

If the target system is to operate as an independent RSX-11D system following bootstrap, it must have tasks installed in it that are capable of doing the required work.

While processing the INS directives, phase 1 of system generation checks for the names of two system tasks: the disk handler for the system disk and phase 2 of system generation. These tasks are installed and, upon bootstrap, are running. The device handler is either DK...., DP...., DF...., DB...., DM.... or DS.... depending on the unit assigned SY. Phase 2 of system generation is called SGN2. If these tasks are not named in the INS directive, the system is not operational when bootstrapped.

Normally other tasks must be installed in addition to the disk handler and SGN2. Typically the following additional tasks are installed:

- Install (INS),
- Mount (MOU),
- File system primitives,
- Terminal handler (TT),
- MCR.

A critical decision centers on the inclusion of the INS MCR function. Unless INS is installed during phase 1, the target system is unable to install tasks. Such systems may not be uncommon; for example, a remote data collector involves few tasks, is rigidly defined, and is generated to occupy minimum core, thus having no need for INS.

Since phase 2 of system generation automatically mounts the system device, the appropriate file primitive tasks must be installed during phase 1. Two versions of the directory device file primitives are

ADDITIONAL SYSTEM GENERATION INFORMATION

provided. Both have the task name F11ACP. One has the filename FCP.TSK. It is a small, heavily overlaid version of the task. The other is named BIGFCP.TSK and is minimally overlaid.

If the user has his own file primitives, he should set the default ACP name in the DEV directive and install that task during phase 1.

With the minimum number of required tasks installed, the installation of any additional tasks can occur during phase 2. The decision to install additional tasks during phase 2 is partly procedural and partly due to the 15-task installation limitation of phase 1.

The system prints the following message to request the INS directives.

SPECIFY INSTALLS

THE INS directives should be followed by a record containing one slash(/) or if no SYSBLD directive is to be specified then two slashes (/ /).

5.2.7 Specification of SYSBLD Command Files

The system will request the SYSBLD command file by typing

SPECIFY SYSBLD COMMAND FILE SPECIFICATION

The name of the appropriate command file should be entered. After the SYSBLD directive has been typed enter a record consisting of two slashes (/ /).

5.2.8 Pool Requirements of SGN1

Phase 1 of system generation uses the dynamic pool for storage of input data and for communication with the version of install (...INV) that it requests. SGN1 returns nodes to the pool as soon as it has finished with them. If a fatal error occurs, it returns all nodes using its own internal accounting system.

As a result, pool usage is quite dynamic. The important consideration, however, is the maximum usage at any time. To compute the maximum pool usage consider the phase 1 command input while referring to Figure 5-1.

ADDITIONAL SYSTEM GENERATION INFORMATION

NODE USAGE

1 node = 16 contiguous words.

This graph does not include MCR buffers or nodes used by the file system.

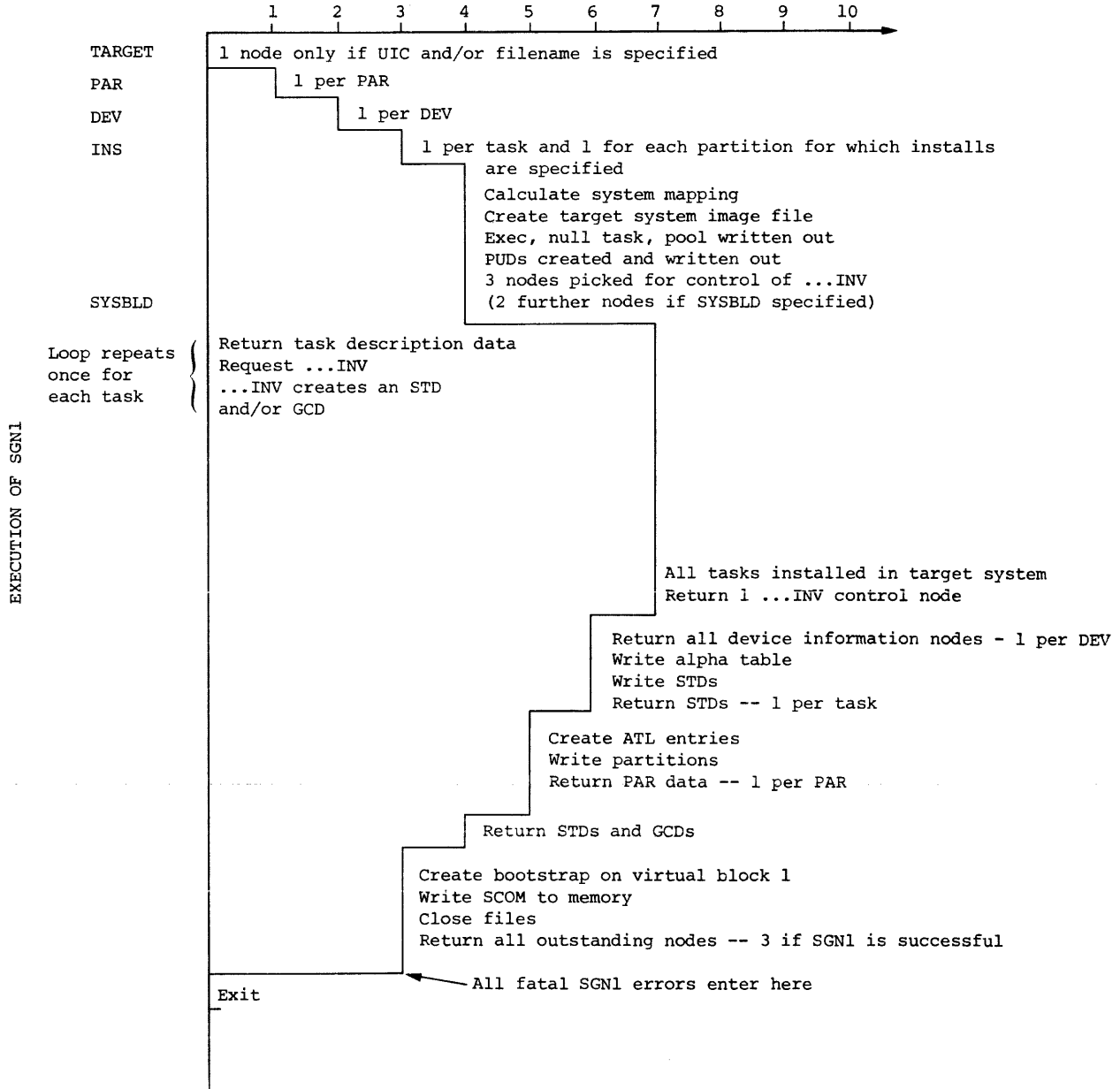


Figure 5-1 SGN1 Pool Usage

5.3 PHASE 2 OF SYSTEM GENERATION

Phase 2 of system generation is installed during phase 1. On bootstrap from a phase 1 target disk as described in Section 2.3, phase 2 is activated and proceeds as follows:

1. Loads the terminal handler,
2. Issues a MOUNT command for the system device (SY),
3. Opens the file SY:[11,17]SYSBLD.CMD, or other user specified file and if the open is successful, begins to process the file and print it on the console.

The file SYSBLD.CMD is provided in the RSX-11D system. If the user wishes to modify it, he should do so before bootstrapping phase 2.

SYSBLD.CMD can contain the phase 2 directive described in Chapter 3, *DELAY, and any MCR request except SAVE or BOOT.

A task must be installed before it can be used. The task can be installed either during phase 2 of system generation or by the INS MCR request. If the task refers to a shared region, the region referred to must be installed before the task is installed.

When phase 2 is complete, it prints the following message on the console.

END OF SYSTEM GENERATION PHASE 2

At this point, perform the steps in Section 3.2 starting with step 4. This process properly saves the system for continued use.

5.4 RSX-11D BOOTSTRAPPING

The PDP-11 family of computers is supplied with a hardware bootstrapping program in read only memory. The function of this program is to read one block (physical block 0) of a specified device into main memory starting at real location zero. On successful completion of the read, the processor jumps to real location zero with memory management disabled and starts to execute the contents.

Under RSX-11D, block zero of the booted device must contain a program to read in the whole operating system and user-defined partitions, enable memory management, and start the Executive.

During the generation and subsequent management of an RSX-11D system, the operations related to the bootstrap go through several phases. Although these operations are relatively transparent to the user, the system manager should be aware of the actions taken and their implications. The operations can be divided into the following four categories.

1. Phase 1 of system generation
2. Phase 2 of system generation
3. Saving the generated system
4. Subsequent bootstrapping and saving of a stable system

5.4.1 System Generation Phase 1 (SGN1)

The purpose of the first phase of system generation is to create a file; e.g., RSX.SAV, on a target disk. This file is a bootable RSX-11D system image configured for the target hardware; i.e., the contents of RSX.SAV on completion of phase 1 are independent of the hardware configuration on which it was created. The only bootstrap-related function performed by SGN1 is to include the bootstrap at the beginning of the RSX-11D image file.

Due to differences among mass storage device controllers, the bootstraps for various devices differ in their control logic. Therefore, RSX-11D provides a bootstrap program for each different device supported as a system device. The bootstrap programs are named using the convention xxxxBOOT.TSK. xxxx is the commonly-used name for the device. At present, bootstraps are supplied for RK05, RK06, RF11, RP03, RP04, RS04 disk devices.

The RK03 uses RK05BOOT, the RS03 uses RS04BOOT, the RP02 uses RP03BOOT and the RP05 and RP06 use RP04BOOT.

Using the SY directive and the related device specification, SGN1 determines which bootstrap task is to be written into the beginning of the memory image file that it is creating.

Before it writes this task into the image file, however, SGN1 must insert the following information into the program.

1. The amount of memory to be filled (upper limit 124K words).
2. The disk address of the RSX.SAV file.
3. The real base address of the Executive (to load kernel APR0 when memory management is enabled).
4. The external page address of the disk controller as specified by the corresponding DEV directive.
5. The kernel virtual address of the power recovery routine within the Executive. This routine is used to start the Executive as well as to recover from a power failure.

The first four items are available to SGN1. The last (power recovery routine address) is obtained by building the appropriate bootstrap task with the Executive symbol table, EXEC.STB. SGN1, therefore, inserts items one through five above into the bootstrap task before writing it to the RSX.SAV file.

To insert the required information, SGN1 must have access to the symbol table of the bootstrap task. Consequently, SGN1 also reads xxxxBOOT.STB to determine the bootstrap program's data locations. Since the bootstrap is linked with the EXEC.STB, SGN1 is able to locate all other symbols for the target Executive from the bootstrap task symbol file.

During phase 1 of system generation, the user has the option of defining base addresses for the Executive, the system communication area, and partitions. If base addresses are not specified, SGN1 performs a memory allocation and places the Executive on the next 32 word boundary after the bootstrap program. SGN1 determines the length of each bootstrap from the corresponding .STB file.

ADDITIONAL SYSTEM GENERATION INFORMATION

If the user specifies the base address, he must take care to leave sufficient space for the bootstrap program. The user can determine the bootstrap size by obtaining a task builder listing of the appropriate xxxxBOOT.STB. The following command is used.

```
TKB>,LP:=[11,17]xxxxBOOT.STB
```

The value of .BO.ND (end bootstrap) rounded up to the next 32 word (100 octal) boundary provides the lowest real address at which the Executive can be placed.

Whenever the executive is rebuilt using the [11,15]TKB15.CMD command file the bootstraps must also be rebuilt using the command file [11,17]BOOTSBLD.CMD. This is because system generation phase 1 obtains all executive symbols from the appropriate xxxxBOOT.STB symbol table file.

It is important to note that unlike earlier versions of RSX-11D, version 6.2 and subsequent versions do not write block 0 of the target system device. Rather, it places the bootstrap task at the beginning of the memory image file created by phase 1.

5.4.2 System Generation Phase 2 (SGN2)

Some of the functions necessary for a complete system generation must be performed on the target hardware configuration (e.g., creation of the checkpoint file). Others are more conveniently performed under the target software configuration (e.g., installation of a large number of tasks). Therefore, phase 1 of system generation creates a runnable RSX-11D system with SGN2 installed and loaded into its partition. When the Executive starts, it activates SGN2.

SGN2 makes no modifications to the bootstrap program either in memory or on disk. The only bootstrap-related aspect of SGN2 is the initial bootstrapping of the output file from SGN1. This bootstrapping causes the system disk handler to be activated on the first ATL scan and SGN2 to be activated on the second ATL scan.

There are two methods of booting the output of SGN1:

1. By using the MCR BOOT function,
2. By using a combination of the MCR BOOT function and the hardware bootstrap (ROM).

Whichever method is used, the output of SGN1 always must be booted from the device that was specified as SY during phase 1.

When the first method is used, all devices except the bootstrap device must be dismantled to ensure that all activity within the system is stopped. Then the MCR BOOT function is requested and given the file specifier of the image file to be booted. The BOOT task reads the first block of the image file into memory starting at real zero and begins to execute it. This process simulates the ROM bootstrap. BOOT does not require block zero of the device to have any special content. It uses the first virtual block of the image file specified.

With the second method, BOOT is used to create a bootstrap block 0 on the device. Then the device can be booted with the appropriate hardware ROM. BOOT accepts the switch /WB following the file specifier for the RSX-11D image file to be booted. If this switch is present, the first block of the file is copied to block 0 of the same volume. This procedure does not perform a system reload, but does

make the volume hardware bootable.

The inclusion of the BOOT MCR function allows the complete generation and testing of a new RSX-11D system without changing the target device substantially. The only modification to the target device is the creation of a new memory image file and installation of tasks into that system. Note that task headers are impure, i.e., they are modified when a task is installed. Therefore, even though it is possible to reboot a disk on which a new RSX-11D image file has been created, care must be taken with such tasks as INS, MOU, and F11ACP, if they are not fixed, because their headers will reflect the new system. This problem can be circumvented by making separate copies of such tasks before running SGN1.

It is possible to have more than one complete running system on the same disk. However very great care is needed. All systems must have the same number of partitions and devices. Devices which are common to one or more systems must be in the same relative position in the list of DEV directives. This is because the task headers contain pointers to system data structures. If these precautions are not observed a system crash will almost certainly occur whenever any of the systems is booted.

For further information about BOOT, refer to the RSX-11D User's Guide.

5.4.3 Saving a Generated System

When SGN2 runs successfully, all the facilities of the generated RSX-11D system are available. Once the user is satisfied that the system does perform as intended, it is necessary to dismount all devices and save the updated memory-resident system on the bootstrap device. If extensive testing of the new system is intended, it is advisable to save the system first.

The purpose of the SAVE MCR function is to rewrite the file that was originally booted. SAVE uses the bootstrap program that is permanently resident in low memory to perform this function. This approach insures the following:

1. Only as much memory as specified during SGN1 is written,
2. Memory is saved at the disk address (i.e., in the image file) from which it is to be booted.

NOTE

The booting of a previous save image should only be done if the system environment is known to be the same as it was when the save image was created. Serious errors can occur if, for example, tasks which were installed at the time the image was created have been moved or deleted.

3. The device on which the file is saved is independent of any redirection of SY or other devices. That is, the save takes place on the device from which it was booted.

ADDITIONAL SYSTEM GENERATION INFORMATION

Because the output of SAVE is an exact replica of memory, the RSX-11D image file appears as a system with the SAVE task running. In order for SAVE to exit, it makes several modifications to the bootstrap in low memory after copying it into its own buffer to perform the disk write and before performing the actual save. These modifications are as follows.

1. Store the content of user APR0.
2. Store the re-entry address in SAVE which is relative to user APR0.
3. Modify an internal branch within the bootstrap so that a sequence of instructions to return to SAVE in user mode is executed after memory management is enabled.

At this point, two versions of the bootstrap program exist in memory: the original within SAVE and a modified version beginning at real location 0.

SAVE now changes the I/O function code of its version from a read to a write, inhibits task switching and interrupts (processor priority 7), and executes its version of the bootstrap. Upon completion of the save to disk, SAVE executes the same code that it executes when rebooting the saved memory image, as described in the next section.

Before a save of the system can be attempted, the system must be quiescent. The SAV function ensures that no activity is taking place by searching the system data base for any of the following conditions:

1. Mounted devices,
2. A user logged onto any terminal except the one from which SAV was initiated,
3. Tasks with I/O in progress,
4. Tasks being loaded or checkpointed,
5. Shareable global areas being loaded or, in the case of read/write common areas, being written to disk,
6. Tasks loaded or fixed beyond the end of the system image file; that is, the SAV file (possible only on extended memory PDP-11/70).
7. Shareable global areas, including the read-only portion of a multiuser task, loaded beyond the end of the save file (possible only on extended memory PDP-11/70).
8. Tasks with send/receive data queued.
9. Tasks or SGAs installed from devices other than the system disk.

If any of these conditions is detected, SAV prints an appropriate message and exits.

Before writing the system to disk, SAV translates the absolute disk addresses of task image files, shareable global areas and the checkpoint file into the corresponding file-id. Since this is independent of absolute disk addresses these files may be relocated on the disk, for example by the DSC utility. When the system is booted the file-ids are translated back into absolute disk address using the file structure on the disk, as described in the next section.

5.4.4 Subsequent Rebooting of a Saved Image

Regardless of which method is used to boot a saved RSX-11D system, the bootstrap program that performs the read originally came from virtual block 1 of the file being booted. This bootstrap overlays itself with the beginning of the file it is reading. Since the only differences between bootstraps is in data, the instructions continue to execute.

On completion of the read of memory, which is performed in 1K increments, the overlay copy of the bootstrap executes a sequence of instructions that returns to SAVE in user mode with task switching and interrupts inhibited. SAVE then restores all volatile registers, restores the power recovery vector which was used to direct the bootstrap to SAVE, and performs a number of other functions not related to the bootstrap. Upon completion, task switching and interrupts are enabled and the Executive is started by simulating a power fail AST. Finally, SAVE prints the RSX-11D sign-on message and exits.

One of the other functions performed by SAVE is to check the existence of the system clock. Since RSX-11D can use either a KW11-L or a KW11-P clock, SAVE first checks for the clock for which the system was generated. If this test fails, SAVE attempts to start the other type of clock. Therefore, it is possible to boot an RSX-11D system that was generated and saved on a configuration with a different type of clock. As described in the next section, SAVE performs a number of memory size checks and adjustments before it exits. This checking can be inhibited by applying the /NOXT switch to the SAVE command.

As mentioned in section 5.4.3, when the system is booted SAV has to translate the file-id stored in the saved image for each installed task and shareable global area and the checkpoint file into an absolute disk address. Normally this process is not apparent to the user, except that it may take up to several minutes. If hardware errors are detected while reading the disk or if task files have been deleted SAV recovers the system as far as possible.

Unless the offending block or file is one of a small number of critical ones the system will still be runnable. The error messages which may be produced are described in Appendix A of the RSX-11D User's Guide.

5.5 MEMORY ALLOCATION AND USAGE

This section provides additional information about the way in which RSX-11D allocates and uses memory. It also provides some guidelines for the choice of partitions and their sizes.

The initial memory size is specified to phase 1 of system generation by means of the PDPl1 directive. This quantity establishes the size of the save file and, consequently, the amount of system memory that can be saved and booted.

The maximum length of the save file is 124K words. If the hardware is a PDP-11/70, the system can be generated for up to 1920K words. Phase 1 of system generation accounts for the fact that the hardware is a PDP-11/70 with more than 124K. Regardless of the hardware, however, the save file cannot exceed 124K words.

The size of the system components (bootstrap, Executive, SCOM communication region and SCOM subroutines) is determined by phase 1 of system generation from the bootstrap symbol table file on the target disk. The STB file contains all global symbols of the Executive and

ADDITIONAL SYSTEM GENERATION INFORMATION

the bootstrap. If the user intends to specify base addresses for all partitions, this STB file should be examined by obtaining a task build map with only the STB file as input. The following symbols provide the necessary sizes.

.BO.ND	Bootstrap size
.SG.EX	Size of Executive
.SG.BE minus .SG.BC	Total size of SCOM communication region and subroutines excluding the node pool

Once these sizes are obtained, base addresses can be assigned without wasting memory.

If the user does not specify base addresses, phase 1 performs all size calculations and positions all components optimally to avoid wasting memory. Memory is allocated from location zero upward. The amount of memory available for partitions is the initial specification in the PDP11 directive minus the total lengths of the bootstrap, Executive, and SCOM. The length of SCOM is user-defined and consists of the node pool, communications region, and subroutines.

When base addresses are not specified, the partitions are allocated by phase 1 in the order that they are specified to phase 1. The size of all partitions except one must be specified. One partition can be given a partition size of asterisk (*). Phase 1 computes that partition's size from the memory remaining after memory for all other partitions is allocated. The asterisk has no effect on the base address of the partition. It only affects its size and, consequently, the base address of all succeeding partitions.

The asterisk can be used with the specification of base addresses. The resulting partition size is the difference between the base address of this partition and the next.

During phase 1, the bootstrap, Executive, SCOM, system disk handler, and phase 2 of system generation are written into the save file on the target disk. Since the save file has a maximum length of 124K words, phase 1 ensures that all are positioned below the 124K boundary when generating a PDP-11/70 system with more memory. RSX-11D requires a minimum of two partitions: one for the system disk handler and one for the phase 2 task. Depending upon the application for which the user has chosen RSX-11D, these two partitions may be sufficient. Then the system consists of a user-controlled system disk partition and a system-controlled general (GEN) partition using all the remaining memory.

Partitions are used to provide areas of memory that can guarantee an acceptable response time to requests for tasks. However, the existence of too many infrequently-used partitions can result in the wasting of memory for long periods of time. Partition design involves trade-offs among the following desirable goals:

- Fast response to task initialization,
- Maintaining a high level of memory usage,
- Minimizing memory fragmentation in multitask partitions,
- Minimizing the possibility of frequent checkpointing of low priority tasks that are compute bound in a partition where high priority tasks must run,

ADDITIONAL SYSTEM GENERATION INFORMATION

- Maintaining a fast response for all terminals,
- Maintaining high throughput for all file activity.

The choice of relative partition sizes depends on the application environment. In particular, different partition sizes are chosen in a system with many terminals than in one with a predominantly process control environment.

After an RSX-11D system generation, the system is saved in the image file as described in Section 5.4.3. SAVE is the task that is resident in a saved system and is the task that initiates a system restart.

SAVE has the capability of adapting the system size to match that of the machine on which the system is run. If the machine is a PDP-11/70, SAVE interrogates a register that defines the amount of memory available. On all other machines SAVE determines the system size by actually testing memory in 1K increments, starting at 32K. SAVE attempts to expand or truncate one or more partitions to reflect the available memory. Automatic expansion and truncation can be inhibited by using the /NOXT switch in the SAVE command.

Memory expansion is the simpler process. In this case, the size of the last partition in memory, that is, the highest addressed partition, is increased by the value of the expansion. If the highest partition is a system-controlled partition, the last hole pointer in that partition is updated to reflect the new size of that hole.

To make effective use of the expansion facility, the last partition should be a system-controlled partition. If the expanded partition is a user-controlled partition, only one task can be active in it at a time regardless of the expanded size.

If the memory size has decreased, SAVE attempts to truncate one or more partitions starting from the highest end of memory. It is impossible for this process to be successful in the following circumstances:

1. Any occupied partition reduces to zero size,
2. Any truncation occurs in an occupied user-controlled partition,
3. The truncation of an occupied system-controlled partition is so extensive that a previously allocated area of memory no longer exists.

In summary, any amount of truncation can occur in the unoccupied part of a system-controlled partition and in an unoccupied user-controlled partition, including truncation of the whole partition.

CHAPTER 6

TAILORING THE SYSTEM

Several system components can be modified to suit the particular needs of an installation. The tasks which can be modified, and the ways of modifying them, vary from release to release and are described in the Release Notes. This chapter provides an outline to the subject.

6.1 EXECUTIVE CHANGES

In addition to the tailoring implicit in the system generation process, it is possible to build the executive task EXEC.TSK in different ways:

1. A crash dump module may be included, to dump all of the memory to a scratch device automatically if it crashes. The dump may be analyzed using the Core Dump Analyzer (CDA), which is documented in Chapter 6 of the RSX-11D System Manager's Guide.
2. Some executive modules may be replaced by null modules if their function is not required.

These changes are described in the file [11,15]TKB15.CMD, which appears as an appendix to the RSX-11D System Manager's Guide.

6.2 UTILITY PROGRAMS

Many utility programs use the /INC facility of INSTALL to increase the available space for symbol tables, buffer areas, etc. In general giving a larger value for the /INC option will make the utility run faster or provide more space for data structures. There is a list in the Release Notes of the tasks affected.

APPENDIX A

EXAMPLES OF SYSTEM GENERATION FILES

A.1 TERMINAL AND INTERFACE CONFIGURATION FILE, [311,114]CONFIG.MAC

See Sections 2.4.1, 2.4.3

```
.SBTTL TERMINAL AND INTERFACE CONFIGURATION FILE
;+
; THIS FILE CONTAINS THE MACRO CALLS TO DEFINE THE TERMINALS AND
; INTERFACES PRESENT IN THIS SYSTEM.
;-

;
; COPYRIGHT (C) 1976
; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS
;
;
;
;          AUTHOR :      JOHN HARPER
;          DATE   :      26-JUL-76
;-
; FIRST DEFINE EACH INTERFACE PRESENT IN THE SYSTEM, IN THE FORM:
;
;      INTF    NUMBER,TYPE,ADDRESS,VECTOR[,EXTRA]
;
;      NUMBER  INTERFACE NUMBER, STARTING AT ZERO.  INTERFACE ZERO MUST
;              ALWAYS BE THE CONSOLE DL11.
;      TYPE    IS THE INTERFACE TYPE (E.G. 'DL', 'DH')
;      ADDRESS IS THE EXTERNAL PAGE ADDRESS OF THE INTERFACE
;      VECTOR  IS THE INTERRUPT VECTOR ADDRESS
;      EXTRA   IF PRESENT CONTAINS INTERFACE DEPENDENT INFORMATION:
;              DL - NON-ZERO IF DL11E
;              DH - INTERFACE NUMBER OF ASSOCIATED DH11, IF ANY, OTHERWISE
;                  BLANK.  THE DM MUST PRECEDE THE DH, AND MUST NOT
;                  NOT BE INTERFACE ZERO.
;              DC - 1 FOR DC11AA, 2 FOR DC11AB, ETC, 0 FOR DC11AX
;-

INTF    0,DL,177560,060
INTF    1,DH,160020,340
INTF    2,DM,170500,310
INTF    3,DH,160040,350,2
INTF    4,DL,175610,320,1
INTF    5,DJ,160010,330
INTF    6,DL,176500,300
```


EXAMPLE OF SYSTEM GENERATION FILES

A.3 SYSTEM GENERATION PHASE 2

See the beginning of Chapter 3 and Chapters 4.13, 5.3 on [11,17]SYSBLD.CMD.

```
;  
;      [ 1 1 , 1 7 ] R K 0 5 L 4 8 K . C M D  
;  
;      COMMAND FILE FOR GENERATING A 48K RSX11D SYSTEM ON AN 11/40  
;      WITH A LINE CLOCK, WITH DK0: AS SYSTEM DISK  
;  
PDP11=40,48K  
/  
SCOM=,300,96  
PAR=SYDISK,,37,U  
PAR=MCR,,41,S  
PAR=GEN,,*,S  
/  
DEV=DK0,RK05  
DEV=DK1,RK05  
DEV=DP0,RP03B  
DEV=DB0,RP04  
DEV=DM0,RK06  
DEV=TT0,CONSOL  
DEV=LP0,LP11B  
DEV=DT0,DT11  
DEV=DT1,DT11  
DEV=MT0,TU10  
DEV=MM0,TU16  
DEV=MQ  
DEV=SP  
/  
DPAR=GEN  
SY=DK0  
/  
INS=SYDISK,[11,1]DK  
INS=GEN,[1,1]SYSRES/LI/ACC=RO  
INS=GEN,[1,1]TTLIBA/LI  
INS=GEN,[1,1]TTLIBB/LI  
INS=GEN,[11,1]TT  
INS=GEN,[11,17]SGN2,[11,1]MOU,[11,1]INS/UIC=[11,1],[11,1]FCP  
//
```

EXAMPLE OF SYSTEM GENERATION FILES

```

)
)
) [ 1 1 , 1 7 ] S Y S B L D . C M D
) -----
)
) INPUT FILE TO SYSGEN PHASE 2. ANY MCR FUNCTION IS ALLOWED
) EXCEPT SAV AND BOO. CARE MUST BE TAKEN THAT ANY TASK
) REQUIRED BY SG2 HAS BEEN INSTALLED BEFORE SG2 TRIES TO
) USE IT.
)
) NOTE:
) -----
) SYSTEM GENERATION PHASE2 ("SGN2") IS INSTALLED AND
) LOADED BY SYSTEM GENERATION PHASE 1 ("SGN1") SUCH THAT
) BOOTING THE NEW SYSTEM DISK RESULTS IN THE RUNNING OF
) SGN2.
)
) SGN2 PERFORMS THE FOLLOWING FUNCTIONS AUTOMATICALLY:
)
) (1). CHECK THERE IS SUFFICIENT MEMORY
)
) (2). CHECK THAT SYSTEM DISK HANDLER IS ACTIVE
)
) (3). PICK THE SPECIFIED NODES FOR TERMINAL ISR
)
) (4). REQUEST THE TERMINAL HANDLER ("TT....")
)
) (5). CHECK THAT TT.... BECOMES ACTIVE
)
) (6). MOUNT THE SYSTEM DISK
)
) (7). OPEN FOR READ (THIS) FILE, SY:[11,17]SYSBLD,CMD
)
) SGN2 ASSUMES THAT THE FILES=11 ANCILIARY CONTROL PROCESSOR
) ("F11ACP") HAS BEEN INSTALLED AS WELL AS TT...., MOUNT ("...MOU")
)
)
) INSTALL THE TASK TERMINATION NOTIFIER (".TKTN.")
)
) INS [11,1]TKTN
)
) INSTALL MCR, THE COMBINED MCR FUNCTION TASK ("...MFT")
) AND THE MCR ERROR TASK
) FOR SYSTEMS WHICH RUN THE LARGE TERMINAL HANDLER (TT) INSTEAD
) OF TT01, MCR SHOULD BE INSTALLED USING:
) INS [11,1]TTMCR/UIC=[1,1]
)
) INS [11,1]MCR/UIC=[1,1]
) INS [11,1]MFT
) INS [11,1]MCRERR
)
)
)
) NOTE:
) -----
) ALL THE ABOVE FUNCTIONS MUST BE PERFORMED AS SHOWN, ANY
) MODIFICATIONS SHOULD BE MADE BELOW.
)
) THE TASK ACCOUNTING UTILITY TASKS
)
) INS [11,1]ACCLOG
) INS [11,1]ACCRPT
) INS [11,1]ACCOFF
) INS [11,1]ALCABT

```

EXAMPLE OF SYSTEM GENERATION FILES

```

)
)   MCR FUNCTION TO OUTPUT THE ACTIVE TASK LIST
)
INS [11,1]ACT
)
)   THE BAD BLOCK DETECTION UTILITY "...BAD"
)
INS [11,1]BAD
)
)   NOW THE BATCH SYSTEM
)
INS [11,1]BAT
INS [11,1]BPR
)
)   MCR FUNCTION FOR BOOTING SYSTEMS AND WRITING BOOTSTRAP
)   BLOCKS
)
INS [11,1]BOO
)
)   THE LOGGING OFF MCR FUNCTION
)
INS [11,1]BYE
)
)   THE CRASH DUMP ANALYSER
)
INS [11,1]CDA
)
)   THE FILE COMPARE UTILITY
)
INS [11,1]CMP
)
)   THE GLOBAL CROSS REFERENCE TASK ("CRF...") REQUESTED BY
)   ...TKB WHENEVER /CR IS SPECIFIED ON A MAP
)   OR   ...MAC WHENEVER /CR IS SPECIFIED ON A LISTING
)
)   THERE ARE TWO VERSIONS OF CROSS REFERENCE --
)   1) OVERLAYED   INSTALLED BY   INS [11,1]CRF
)   2) UNOVERLAYED THIS HAS TO BE BUILT USING COMMAND
)   FILES SUPPLIED ON THE AUXILIARY MEDIA
)   UNDER UFD [11,23], WHEN BUILT, IT IS
)   INSTALLED BY   INS [11,1]UNOVRCRF
)
)   BOTH VERSIONS HAVE TASK NAME "CRF..."
)
)   THE EXECUTION TIME OF THE BOTH VERSIONS MAY BE
)   DECREASED BY ALLOCATING MORE CORE SPACE AT INSTALL
)   TIME USING THE /INC=NNNNN SWITCH, NNNNN MAY BE ANY VALUE
)   FROM 0 TO 19000 APPROX FOR [11,1]CRF   (DEFAULT=0)
)   FROM 0 TO 20000 APPROX FOR [11,1]UNOVRCRF (DEFAULT=16000)
)   HIGHER VALUES GIVE FASTER EXECUTION
)
INS [11,1]CRF
)
)   THE MEMORY USAGE DISPLAY (VT05 ONLY) TASK "...DEM"
)
INS [11,1]DEMO
)
)   THE AVAILABLE DISK HANDLERS ARE AS FOLLOWS:
)
)   FOR RK-11 (RK03,RK05) DEVICES:
INS [11,1]DK
)   FOR RK-11 WITH OVERLAPPED SEEKS ON
)   MULTIPLE DRIVES:
)   INS [11,1]DKOVL
)   FOR RF-11 MULTI PLATTER FIXED HEAD DISKS:
)   INS [11,1]DF
)   FOR RP-11C (RP02,RP03) DEVICES:

```

EXAMPLE OF SYSTEM GENERATION FILES

```

;     INS [11,1]DP
;           FOR RP04 MASSBUS DISKS;
;     INS [11,1]DB
;           FOR RS03, RS04 MASSBUS FIXED HEAD DISKS;
;     INS [11,1]DS
;           FOR RK611 (RK06) DISKS
;     INS [11,1]DM
;
;     AND THE DIAGNOSTIC EQUIVALENTS:
;
;     INS [11,27]DKD
;
;     INS [11,27]DPD
;
;     INS [11,27]DFD
;
;     INS [11,27]DBD
;
;     INS [11,27]DSD
;
;     INS [11,27]DMD
;
;     ONE OF THESE WILL BE THE SYSTEM DISK HANDLER ALREADY INSTALLED
;     AND LOADED BY SYSGEN PHASE 1. IT IS INAPPROPRIATE TO
;     ASSUME HERE WHICH IS THE SYSTEM DISK OR WHICH ADDITIONAL DISK
;     DEVICES A USER WISHES TO USE, THEREFORE SELECTION OF FURTHER
;     DISK HANDLERS FOR INSTALLATION IS LEFT TO THE USER.
;
;     ANY OF THE ABOVE MAY BE INSTALLED SIMPLY BY EDITING THIS FILE TO
;     DELETING THE ";" AND TAB PRECEDING THE FILE SPECIFICATION,
;
;
;     THE DISMOUNT VOLUME MCR FUNCTION
;
;     INS [11,1]DMO
;           THE FILE DUMP MCR UTILITY
;
;     INS [11,1]DMP
;           THE FLOPPY DISK HANDLER
;
;           INS [11,1]DX
;
;           THE DECTAPE HANDLER
;
;           ALTERNATE:
;           ----
;
;           THE DIAGNOSTIC HANDLER:
;
;           INS [11,27]DTD
;
;     INS [11,1]DT
;           LINE TEXT EDITOR (LETTER) "...EDI"
;
;           ADDITIONAL:
;           ----
;
;           FOR THE SLIPR EDITOR:
;           INS [11,1]SLP
;
;     INS [11,1]EDI/PRI=60
;
;           THE HARDWARE ERROR LOGGING TASK USED IN CONJUNCTION
;           WITH ERROR LOGGING HANDLERS.

```

EXAMPLE OF SYSTEM GENERATION FILES

```

)
INS [11,1]ERRLOG
)
)   THE FILE INTERCHANGE UTILITY (TO AND FROM DOS, RT-11, RSTS SYSTEMS)
)
INS [11,1]FLX/PRI=55
)
)   FILES-11 MESSAGE OUTPUTTING TASK
)
INS [11,1]F11MSG
)
)   THE LOG ON MCR FUNCTION
)
INS [11,1]HEL
)
)   FILES-11 VOLUME INITIALIZATION MCR FUNCTION
)
INS [11,1]INI
)
)   THE LIBRARIAN (RELOCATABLE OBJECTS AND RANDOM ACCESS MACROS)
)
INS [11,1]LBR
)
)   LINE PRINTER HANDLER
)
INS [11,1]LP
)
)   THE LOGICAL UNIT INFORMATION MCR FUNCTION
)
INS [11,1]LUN
)
)   THE MESSAGE OUTPUT (PSEUDO) HANDLER MO....
)
INS [11,1]MO
)
)   THE MEMORY EXAMINATION AND PATCH UTILITY
)
INS [11,1]OPE
)
)   THE OPERATOR'S SPOOLER CONTROL MCR FUNCTION
)
INS [11,1]OPR
)
)   THE FILES-11 PERIPHERAL AND FILE INTERCHANGE FUNCTION
)
INS [11,1]PIP/PRI=55
)
)   THE POOL USAGE DISPLAYING UTILITY
)
INS [11,1]POOL
)
)   THE ONLINE PRESERVE UTILITY "...PRE"
)
INS [11,1]PRE
)
)   THE VERSIONS OF THE MACRO-11 ASSEMBLER ARE AS FOLLOWS:
)
)   FOR SMALL, HEAVILY OVERLAYED TASKS:
)   INS [11,1]MAC
)   FOR THE LARGE, UNOVLAYED ASSEMBLER TASKS,
)   TASK BUILD COMMAND FILES ARE PROVIDED ON THE AUXILIARY MEDIA
)   UNDER UFD [11,10] TO BUILD THEM.
)
)   [11,10]MACBLD.COM BUILDS [11,1]MAC.TSK, THE OVERLAYED VERSION.
)
)   [11,10]PURMACBLD.COM BUILDS [11,1]PURMAC.TSK, THE UNOVLAYED

```

EXAMPLE OF SYSTEM GENERATION FILES

```

;
;   VERSION,
;
;       ALL HAVE THE TASK NAME "...,MAC"
;
;           THE EXECUTION TIME OF THESE ASSEMBLERS MAY
;   BE DECREASED CONSIDERABLY WHEN THE NUMBER OF SYMBOLS IS LARGE
;   BY ALLOCATING THE ASSEMBLER MORE CORE RESIDENT WORK SPACE.
;   THIS MAY BE ACHIEVED BY INSTALLING THE TASK USING THE
;   "/INC=NNNNN" SWITCH ON THE COMMAND TO INSTALL. THIS IS
;   RECOMMENDED IF IT IS NECESSARY TO RE-BUILD THE TERMINAL HANDLER
;   TASK (TT,...). THE VALUE FOR 'NNNNN' MAY BE
;   0 TO 15000 (APPROX) FOR [11,1]MAC      (DEFAULT=8192)
;   0 TO 12000 (APPROX) FOR [11,1]PURMAC  (DEFAULT=12000)
;
;           FOR THE SMALL, OVERLAYED TASK:
INS [11,1]MAC
;           FOR A LARGER, FASTER, OVERLAYED TASK:
;INS [11,1]MAC/INC=15000
;           FOR THE FAST, LARGE, UNOVERLAYED TASK:
;INS [11,1]PURMAC/INC=4000
;           FOR THE FASTEST, LARGE, UNOVERLAYED TASK:
;INS [11,1]PURMAC
;
;           THE MCR FUNCTION FOR INSERTING PASSWORDS INTO SPECIFIED UFD'S
;           ON THE SYSTEM DISK.
INS [11,1]PWD
;
;           THE MCR FUNCTION TO QUEUE A FILE FOR OUTPUT
;INS [11,1]QUE
;
;           THE LOGICAL UNIT REASSIGNMENT FUNCTION
;INS [11,1]REA
;
;           THE DEVICE REDIRECTION MCR FUNCTION
;INS [11,1]RED
;
;           THE MCR FUNCTION FOR REMOVING TASKS AND SGA'S
;INS [11,1]REM
;
;           THE TASK RUNNING MCR FUNCTION
;INS [11,1]RUN
;
;           THE SYSTEM SAVING AND RESTARTING MCR FUNCTION
;INS [11,1]SAV
;
;           THE SYSTEM PARAMETER SETTING MCR FUNCTION
;INS [11,1]SET
;
;           THE SPOOLER TASK "SPR,..."
;INS [11,1]SPR
;
;           THE DESPOOLER TASK "SPR2,.." SET FOR 1 DEVICE. INCREMENTS OF
;           460 WORDS SUPPORT ONE MORE DEVICE
;INS [11,1]SPR2/INC=460
;

```

EXAMPLE OF SYSTEM GENERATION FILES

```

)      THE SYSTEM INFORMATION MCR FUNCTION
)
INS [11,1]SYS
)
)      THE ON-LINE TERMINAL CHARACTERISTIC CHANGING FUNCTION
)
INS [11,1]TER
)
)      THE TIME SETTING AND ENQUIRING FUNCTION
)
INS [11,1]TIM
)
)      THE LINKER (TASK BUILDER)
)
)      THERE ARE TWO VERSIONS OF TASK BUILDER --
)      1) OVERLAYED   INSTALLED BY   INS [11,1]TKB
)      2) UNOVERLAYED THIS HAS TO BE BUILT USING COMMAND
)      FILES SUPPLIED ON THE AUXILIARY MEDIA
)      UNDER UFD [11,1]. WHEN BUILT, IT IS
)      INSTALLED BY   INS [11,1]UNOVRTKB
)
)      BOTH VERSIONS HAVE TASK NAME "...TKB" .
)
)      THE EXECUTION TIME OF THE OVERLAYED VERSION MAY BE
)      DECREASED BY ALLOCATING IT MORE CORE SPACE AT INSTALL
)      TIME USING THE /INC=NNNNN SWITCH. NNNNN MAY BE ANY
)      VALUE FROM 0 TO 12000 APPROX. HIGHER VALUES GIVE
)      FASTER EXECUTION.
)
INS [11,1]TKB
)
)      THE MAGTAPE HANDLER.
)
)      ALTERNATES:
)      -----
)
)      "MM...." FOR TJU16 MASSBUS CONTROLLER:
INS [11,1]TU16
)      AND THE DIAGNOSTIC VERSION:
)      INS [11,27]TU16D
)
)      "MT...." FOR TM-11 TU10 DEVICES:
INS [11,1]TU10
)
)      AND ITS DIAGNOSTIC VERSION:
)      INS [11,27]TU10D
)
)      THE USER FILE DIRECTORY CREATION MCR FUNCTION
)
INS [11,1]UFD
)
)      THE DEVICE HANDLER UNLOADING FUNCTION
)
INS [11,1]UNL
)
)      THE FILES-11 SYSTEM VERIFICATION MCR FUNCTION
)
INS [11,1]VFY
)
)      THE TERMINALS IN USE DISPLAY UTILITY "...WHO"
)
INS [11,1]WHO
)
)      THE FILE PATCH UTILITY "...ZAP"
)
INS [11,1]ZAP
)

```

EXAMPLE OF SYSTEM GENERATION FILES

```
)
)
) THE SPOOLER USES A PSEUDO DEVICE SP: FOR QUEUING AND
) BUFFERING. THIS DEVICE MUST BE REDIRECTED TO SOME RANDOM
) ACCESS FILES=11 DEVICE WITH A UFD (1,4)/PRO*(RWED,RWED,RWED,RWED)
)
) ASSUME HERE THAT IT IS THE SYSTEM DEVICE.
)
RED SY=SP
)
) AFTER SGN2 HAS PROCESSED ALL THE COMMANDS IN THIS FILE
) IT WILL THEN PROCEED TO ALLOCATE THE CHECKPOINT FILE (IF
) AND AS) SPECIFIED DURING PHASE 1.
)
) NOTE 1.:
) -----
) THE DEVICE TO WHICH CHECKPOINTED TASKS ARE TO BE WRITTEN
) MUST HAVE ITS HANDLER RESIDENT; MUST BE INITIALIZED AS A
) FILES=11 VOLUME AND MUST BE MOUNTED FOR SGN2 TO CREATE THE
) CHECKPOINT FILE.
) THEREFORE IF IT IS A VOLUME OTHER THAN THE SYSTEM DEVICE
) INSERT THE APPROPRIATE INSTALL, LOAD AND MOUNT COMMANDS IN
) THIS FILE.
) DIRECTIVE.
)
) NOTE 2.:
) -----
) AFTER SGN2 COMPLETES, THE USER SHOULD LOG ON TO THE SYSTEM,
) PERFORM THOSE FUNCTIONS DEEMED DESIRABLE AND THEN SAVE THE SYSTEM.
)
) TYPICAL POST SGN2 FUNCTIONS ARE:
) LOA MO ; LOAD MO HANDLER
) LOA LP ; LOAD THE LINE PRINTER HANDLER
) RED LP=CL ; LINE PRINTER IS CONSOLE LISTING DEVICE
) REA ...TKB 8 DS ; REASSIGN THE WORKFILE LUNS OF
) REA CRF... 7 DS ; TKB, MAC AND CRF TO
) REA ...MAC 8 DS ; A FIXED HEAD DISK,
) TIM HH:MM:SS DD=MMH=YY ; ENTER TIME AND DATE
) RUN ACCLOG ; RUN THE ACCOUNT LOGGER
)
) ALTERNATIVELY, THIS FILE MAY BE EDITED TO
) INCORPORATE THESE COMMANDS.
)
```


APPENDIX B

DEVICE CHARACTERISTICS WORDS

```

;
;
; THIS MODULE CONTAINS:
;
; 1 -- CODE FOR DVSCH.
;
.PAGE
.SBTTL DVSCH -- THE DEVICE CHARACTERISTICS MACRO
;
; MACRO TO GENERATE AN ENTRY IN THE DEVICE CHARACTERISTICS
; TABLE.
;
; MACRO FORMAT:
;
; DEVCH NAME,AC1,AC2,AC3,AC4,ACP,BNAM,VSIZH,VSIZL,VECT,PRI,CSR
;
; WHERE: NAME = DEVICE NAME STORED AS TWO RAD50
;          WORDS.
;          AC1 = PUD CHARACTERISTICS WORD 1 (DEV. INDEP.)
;          AC2 = " " " 2 (DEV. DEPEND.)
;          AC3 = " " " 3 (DEV. DEPEND.)
;          AC4 = " " " 4 (TXFR. SIZE)
;
; (FOR DEFINITION OF THE ABOVE, SEE EXECUTIVE AND APPROPRIATE
; DEVICE HANDLER)
;
; ACP = DEFAULT ACP FOR THIS DEVICE
; BNAM = FOUR ASCII CHARACTERS WHICH
; IDENTIFY THE BOOTSTRAP
; PROGRAM.
; (2 ZERO WORDS FOR NON-BOOTABLE
; DEVICES)
; VSIZH = HIGH ORDER PART OF VOLUME SIZE
; FOR DIRECTORY DEVICES.
; (ZERO FOR OTHERS)
; VSIZL = LOW ORDER VOL. SIZE
; VECT = DEFAULT VECTOR ADDRESS
; PRI = DEFAULT PRIORITY
; CSR = DEFAULT CONTROL/STATUS REGISTER ADDRESS
;
; MACRO DEVCH ANAME,AC1,AC2,AC3,AC4,ACP,BNAM,VSIZH,VSIZL,VECT,PRI,CSR,?Z
; ZI /ANAME/
; #Z+4
;
; .WORD AC1
; .WORD AC2
; .WORD AC3
; .WORD AC4
; .IIF NB <ACP> .RAD50 /ACP/
; .IIF B <ACP> .WORD 0
; .IIF NB <BNAM> .ASCII /BNAM/
; .IIF B <BNAM> .BLKW 2
; .IIF NB <VSIZH> .WORD VSIZH
; .IIF B <VSIZH> .WORD 0
; .IIF NB <VSIZL> .WORD VSIZL
; .IIF B <VSIZL> .WORD 0

```

DEVICE CHARACTERISTICS WORDS

```

.IIF NB <VECT> .WORD VECT
.IIF B <VECT> .WORD 0
.IIF NB <PRI> .WORD PRI
.IIF B <PRI> .WORD 0
.IIF NB <CSR> .WORD CSR
.IIF B <CSR> .WORD 0
;
LNG      * .Z          ;LENGTH OF CHARACTERISTICS TABLE
.ENDM
.PAGE
.SBTTL   DVSCH -- STANDARD SUPPORTED DEVICE CHARACTERISTICS
;
;        DEFINE THE DEVICE CHARACTERISTICS TABLE
DCHST:
DEVCH   PTR,40,0,0,1000,,,,,70,4,177550
DEVCH   PTP,40,0,0,1000,,,,,74,4,177554
DEVCH   RK03,140010,21060,1014,512,,F11,RK05,,4800,,220,5,177400
DEVCH   LP11A,3,0,0,80,,,,,200,4,177514
DEVCH   LP11B,3,0,0,132,,,,,200,4,177514
DEVCH   LP11WU,3,0,0,132,,,,,200,4,177514
DEVCH   LP11WL,3,1,0,132,,,,,200,4,177514
DEVCH   LP11NU,3,0,0,80,,,,,200,4,177514
DEVCH   LP11NL,3,1,0,80,,,,,200,4,177514
DEVCH   LS11WU,3,2,0,132,,,,,200,4,177514
DEVCH   LS11WL,3,3,0,132,,,,,200,4,177514
DEVCH   LS11NU,3,2,0,80,,,,,200,4,177514
DEVCH   LS11NL,3,3,0,80,,,,,200,4,177514
DEVCH   DT11,140010,4060,401,512,,F11,,,578,,214,6,177340
DEVCH   AF11,0,1023,,0,0,,,,,134,4,172570
DEVCH   AD01,0,100,0,0,,,,,130,6,176770
DEVCH   LPS11,0,16,,0,0
DEVCH   UDC11,0,0,0,0,,,,,234,6,171000
DEVCH   RK05,140010,1060,1014,512,,F11,RK05,,4800,,220,5,177400
DEVCH   RK05F,140015,41060,1014,512,,F11,RK05,,4800,,220,5,177400
DEVCH   RP02,140010,41460,12012,512,,F11,RP03,,40000,,254,5,176710
DEVCH   RP03A,140010,101460,12012,512,,F11,RP03,1,34200,254,5,176710
DEVCH   RP03B,140010,141460,12012,512,,F11,RP03,1,34200,254,5,176710
DEVCH   RF1,140010,400,10,512,,F11,RF11,,1024,,204,5,177460
DEVCH   RF2,140010,20400,10,512,,F11,RF11,,2048,,204,5,177460
DEVCH   RF3,140010,40400,10,512,,F11,RF11,,3072,,204,5,177460
DEVCH   RF4,140010,60400,10,512,,F11,RF11,,4096,,204,5,177460
DEVCH   RF5,140010,100400,10,512,,F11,RF11,,5120,,204,5,177460
DEVCH   RF6,140010,120400,10,512,,F11,RF11,,6144,,204,5,177460
DEVCH   RF7,140010,140400,10,512,,F11,RF11,,7168,,204,5,177460
DEVCH   RF8,140010,160400,10,512,,F11,RF11,,8192,,204,5,177460
DEVCH   CR11,1,0,0,80,,,,,230,6,177160
DEVCH   CD11,1,0,0,80,,,,,230,4,172460
DEVCH   RP04,140010,2060,11426,512,,F11,RP04,2,117426,254,5,176700
DEVCH   RP05,140010,22060,11426,512,,F11,RP04,2,117426,254,5,176700
DEVCH   RP06,140010,42060,11426,512,,F11,RP04,5,030434,254,5,176700
DEVCH   RS03,140010,2400,100,512,,F11,RS04,,1024,,204,5,172040
DEVCH   RS04,140010,22400,100,512,,F11,RS04,,2048,,204,5,172040
DEVCH   TU10,140070,0,0,512,,MTA,,,,,224,5,172520
DEVCH   TU16,140070,100000,0,512,,MTA,,,,,224,5,172440
DEVCH   TA11,100070,0,0,128,,,,,260,6,177500
.IF DF IAS
DEVCH   BATCH,3,0,0,132,
DEVCH   PSEUDO,10000,0,0,0
.ENDC
DEVCH   RK06,140010,3160,1426,512,,F11,RK06,,27126,,210,5,177440
DEVCH   TERM,7,0,0,512,
DEVCH   CONSOL,7,0,0,512,,,,,60,4,177560
DEVCH   RX01,140010,2430,415,512,,F11,,,494,,264,5,177170
DEVCH   ,0,0,0,0
DCHEN:

```

INDEX

- ACP, 4-8, 4-9

- Baĉ blocks, 2-2
- Bootstrap, hardware, 2-8, 3-6
- Bootstrap, software, 3-6
- Bootstrapping, 2-1, 2-4, 5-7, 5-12
- BP device, 4-11

- CI device, 4-11
- CKPNT directive, 4-15, 5-4
- CL device, 3-7, 4-11
- Clock, 4-3, 5-12
- CO device, 4-11
- Communication interfaces, 2-13
- CONFIG.MAC, 2-13, A-1
- CPU specification, 4-3, 5-3
- Crash dump module, 6-1

- *DELAY directive, 4-18
- DEV directive, 4-8, 5-3
- Device addresses,
 - BM792-YB, 2-9
 - BM873-YA, 2-10
 - BM873-YB, 2-11
 - M9301-YC, 2-12
 - MR11-DB, 2-8
- Device information, 4-10, B-1
- Device priority, 4-8
- Directives, system generation, 4-1, 4-5
- Directories, 2-4
- Distribution disks, 2-1, 2-4
- Distribution tapes, 2-1

- Editing system generation files, 5-1
- Error messages,
 - Phase 1, Messages-1
 - Phase 2, Messages-7
 - terminal handler, 2-21
- EXEC directive, 4-4, 5-3

- Fl1ACP, 3-7, 4-17
- Floating point option, 4-3

- Hardware bootstrap, 2-8
- Host system, 3-1, 3-6

- /INC, INSTALL switch, 6-1
- INS directive, 4-17
- Installation of required tasks, 5-4

- Memory allocation, 4-3, 5-12
- MO device, 4-11

- Node pool requirement for SGN1, 1-2, 4-5, 4-6

- PAR directive, 4-5, 4-7, 5-3
- Parameters, directive, 4-1
- PARAMS.MAC, 2-13
- Partitions, 4-7
- PDP11 directive, 4-3, 5-3
- Phases 1 and 2, 1-1, 3-1, 4-1
- Procedures,
 - Phase 1, 3-2, 5-2
 - Phase 2, 3-5
- Pseudo-devices, 4-11
- PSWD directive, 4-16, 5-4

- RED, MCR command, 3-7, 4-11
- Required task installation, 5-4
- RSXSYS, 2-3
- RUN \$, 4-14

- Sample system generation files, A-1, A-2
- Saving a generated system, 3-7, 5-10
- SCOM directive, 4-5, 5-3
- SCOM size, 4-5
- SGAs, 4-17
- SGN1 task, 5-8
- SGN2 task, 4-4, 5-9
- SP device, 4-11
- SY device, 3-7, 4-11
- SY directive, 4-12, 5-4

INDEX (CONT.)

SYSBLD.COMD file, 5-5; A-4
SYSBLD directive, 4-18
System defaults specification,
5-3

TARGET directive, 4-2, 5-2
Target specification, 5-2
Target system, 1-1, 3-1

Terminal handler, 2-1, 2-13,
3-7, A-1
error messages at system
generation, 2-21, 3-7
Terminology, 1-2
TI device, 4-11

xxxxBOOT.TSK, 5-8

ERROR MESSAGES

During system generation, errors detected cause a message to be printed on the console. Phase 1 messages are preceded by a SGN1; phase 2 messages are preceded by SGN2. System generation produces three types of error messages.

1. Diagnostic messages -- These messages are informative and do not result in termination of system generation. The user should not force termination if a diagnostic message is printed. Diagnostic messages are prefixed as follows.

SGn -- *DIAG*

2. Diagnostic message dependent on console input -- These message are nonfatal if console intervention is possible. If input is from an indirect file, the error is declared fatal.
3. Fatal error messages -- These errors are not recoverable. System generation terminates the current attempt to build the system. Fatal error messages are prefixed as follows.

SGn -- *FATAL*

Error messages in this appendix are divided into one section for phase 1 messages and another for phase 2 messages. Within each section, messages are presented in alphabetic order.

For terminal handler configuration error messages see Section 2.4.5.

1. PHASE 1 ERROR MESSAGES

In the following section, the number associated with each message is the internal message number within SGN1.

CANNOT REQUEST ...INV 40

Failure of the REQUEST directive or corruption of the communication data sent by SGN1 to ...INV (the special version of INSTALL) has occurred. If the latter is true, remove and reinstall ...INV and then try rerunning phase 1.

DUPLICATE NAME 3

A device name or partition name has appeared more than once.

ERROR IN READING TASK IMAGE 27

An attempt to read a task has failed. The failure is probably due to hardware errors on the device.

ERROR IN STD TABLE CONSTRUCTION	20
SGN1 has failed internally.	
ERROR ON COMMAND INPUT	9
A hardware failure has occurred on attempting to read command input	
EXECUTIVE CROSSES 124K BOUNDARY	41
The base address assigned to the Executive is such that it does not reside completely within the save file on PDP-11/70. The Executive must be below the 124K boundary to fit in the save file. Reassign the base addresses.	
FAILED TO PICK VIRTUAL SCOM NODE	52
No more nodes available from node pool of system being generated. Specify a larger SCOM size so that more nodes are available.	
FAILED TO REMAP GCD NODE ADDRESS	53
Indicates an internal error within SGN1 while converting the GCD nodes of the present system to the addresses of the new system.	
FP OPTION ILLEGAL FOR 11/35 AND 11/40	47
The FP field of the PDP11 directive specifies the presence of the FP11 floating point processor. This option is not available on a PDP-11/35 OR 11/40.	
To perform floating point calculations on a PDP-11/35 or PDP-11/40, the floating point instruction set (FIS) must be present. The presence of the floating point instruction set is not specified during system generation.	
I-O ERROR ON FILE filename	37
I/O error has occurred on reading or writing the named file. For example, the volume may be write-locked during a write operation.	
ILLEGAL CHECKPOINT DEVICE	28
The checkpoint device was not defined by a DEV=directive.	
ILLEGAL ERROR -- SEVERITY number	5
SGN1 has failed internally.	
ILLEGAL MACHINE TYPE	45
The only CPU types supported by RSX-11D are PDP-11/34,35,40,45,50,55 and 70.	
ILLEGAL MEMORY SIZE OR BASE ADDRESS	46
A system or partition size specification is larger than that allowed on the specified CPU; that is, greater	

than 124K for a PDP-11/34,35,40,45,50,55 or greater than 1920K on a PDP-11/70. Alternatively, a base address specified is beyond the usable memory of a system with the specified CPU type.

ILLEGAL MULTI-PARAMETER SETS	32
Multiple parameter sets have occurred in a directive that permits only one parameter set.	
ILLEGAL 'SY' INPUT	6
This message can be caused by any one of the following conditions.	
Undefined device (no DEV command associated with it)	
Invalid mnemonic	
Device for which no bootstrap program exists (see Section 5.4)	
IMPROPER "@" FILE SPECIFICATION	8
File syntax specification incorrect.	
INSUFFICIENT PARAMETERS	33
All the parameters required by the directive have not been submitted.	
INVALID ACP TASKNAME	39
The specified name of the file primitive tasks for this device was not nnnACP; i.e., the last three characters are erroneous.	
INVALID DEVICE TYPE <type>	16
Device type not recognizable. The device type is not defined internally in system generation.	
INVALID KEYWORD IDENTIFIER	29
Keyword in a directive cannot be recognized.	
INVALID PARTITION NAME	35
An attempt has been made to install a task in a nonexistent partition.	
INVALID PARTITION PRIORITY	51
The priority of a T partition must be between 2 and 200 (decimal).	
INVALID PARTITION TYPE	50
Only user-controller (U), system-controlled (S), and time scheduled (T) partitions are acceptable.	
INVALID SYSBLD DIRECTIVE	55
SYSBLD directive file specification contained device name or was more than 31 characters in length.	

INVALID TARGET DEVICE SPECIFICATION	34
<p>A syntax error was made in specifying the target device and filename.</p>	
INVALID TASK HEADER	22
<p>Bad data was found in the header of a task INV has installed. Possibly a task has become corrupted. Start phase 1 again.</p>	
MARK TIME OR WAIT FOR DIRECTIVE FAILURE	19
<p>SGN1 failed to issue one of these directives successfully.</p>	
MORE THAN 15 INSTALLS TO A PARTITION	11
<p>No more than 15 tasks and/or Shareable Global Areas can be installed in a partition during phase 1. This message shows this limit has been exceeded.</p>	
MULTIPLE PARTITIONS WITH * SIZE SPECIFICATION NOT PERMITTED	49
<p>Only one partition can use the wild card memory size specification. Specify a partition size for all partitions except one and retry.</p>	
MULTIPLE SYSBLD DIRECTIVES NOT PERMITTED	54
<p>Only one SYSELD directive is permitted. Delete any additional SYSBLD directives and retry.</p>	
MULTIPLE TARGET DIRECTIVES NOT PERMITTED	48
<p>Only one target device and filename specification can be used. Delete any extra target specifications and retry.</p>	
NO DYNAMIC STORAGE AVAILABLE	4
<p>No more nodes are available. Indicates nested generations may be required. Thus, the user must generate successively larger systems until he reaches his target system. Alternatively, some running on or installed tasks should be removed; then try again.</p>	
NON-EXISTENT "@" FILE SPECIFIED	7
<p>The indirect file specified cannot be found.</p>	
NO SPACE FOR POOL	12
<p>SCOM size insufficient to provide for any pool.</p>	
NO TELETYPE HANDLER (TT....) INSTALLED	18
<p>After all the installs for phase 1 were processed, SGN1 found that no task with the name TT.... was installed. Insert the directive to install TT.... in the phase 1 command file. This message is a warning; the system is operable.</p>	
NO TELETYPE ZERO DEFINED CI, CO, AND CL NOT REDIRECTED	17

TT0 was not assign. CI, CO, and CL have no assignments and should be redirected when the target system is initiated.	
OPEN FAILURE ON FILE filename	38
File cannot be opened. The failure has occurred in attempting to open EXEC.TSK, one of the bootstrap .TSK or .STB files, or the system image output file. Usually either the file does not exist, or no room exists on the disk to create the output file.	
OPTION LINE SYNTAX ERROR	30
A syntax error exists in a directive line.	
OUTPUT I-O ERROR ON FILE filename	25
An I/O error has occurred when writing the save file.	
OVERLAP IN REAL ADDRESS SPACE	21
The base specifications and size specifications result in a space allocation conflict.	
POSITIONING ERROR ON FILE filename	0
This message indicates an internal system failure. Contact local software support.	
REAL MEMORY SIZE EXCEEDED	2
After summing memory requests, those requests are found to exceed the amount specified by the PDP11 directive.	
SCOM CROSSES 124K BOUNDARY	42
The base addresses and sizes specified for the system are such that SCOM cannot be contained completely within the save file on a PDP-11/70. SCOM must be below the 124K boundary to fit in the save file. Reassign the base addresses and/or adjust the sizes of the components.	
SGN2 CROSSES 124K BOUNDARY	44
Because SG1 writes the system generation phase 2 task out to the save file, which has a maximum length of 124K, SGN2 must reside below the 124K boundary in a PDP-11/70. Adjust the partitions or install SGN2 in another partition.	
STD/GCD ENTRY NOT FOUND FOR A TASK/SGA AFTER INSTALL	36
INSTALL has failed to create an STD or GCD entry. INSTALL may issue a message in addition to the one issued by SGN1. Alternatively, INV may not have run to completion.	
SY DISK HANDLER CROSSES 124K BOUNDARY	43
Since SG1 writes the system disk handler into the save file, which has a maximum length of 124K, the system disk handler must reside below that boundary on a PDP-11/70. Adjust the partitions or install the disk	

handler in another partition.	
SYMBOL NOT FOUND symbol	1
A symbol needed by SGN1 cannot be found in the appropriate bootstrap .STB file. The file either has not been built or has been built incorrectly.	
SYSGEN PHASE 2 NOT INSTALLED	26
After all phase 1 installs were processed, SGN1 determined that SGN2 was not installed.	
SYSTEM DISK HANDLER NOT INSTALLED	24
No task with the name xy.... was installed. xy is the device specified in the SY directive, which was valid. See Section 4.2.5.	
SYSTEM DISK NOT DEFINED, 'SY' NOT REDIRECTED	23
SY has not been defined explicitly. Therefore, SGN1 is attempting, by default, to redirect SY to DK0. However, no such device has been defined by a DEV directive; SY cannot be redirected.	
TASK ...INV NOT IN SYSTEM	15
SGN1 requires the task ...INV to operate but the task cannot be found.	
TOO MANY PARAMETERS	31
More than the allowable number of parameters have appeared in the directive.	
VIRTUAL ADDRESS SPACE OF SCOM OUT OF RANGE	13
The size specified for SCOM makes the virtual starting address less than 1000000; Administrator must reduce the size of SCOM. The maximum size of SCOM is 12K.	

2. PHASE 2 ERROR MESSAGES

ALL MEMORY SPECIFIED DOES NOT RESPOND

SGN2 attempts to initialize hole pointers in partitions whose base addresses are beyond 124K on a PDP-11/70. Memory access at one of these base addresses produced a memory system timeout.

ASSIGN LUN ERROR

SGN2 cannot assign a LUN to SY.

CHECKPOINT FILE ALLOCATION ERROR

There is not sufficient contiguous space on the disk to allocate the checkpoint file, the device has not been mounted, or its handler is not loaded.

ERROR ON LOGGING DEVICE

Could not successfully log.

ILLEGAL FUNCTION FOR NON PRIVILEGED TERMINAL

The user must be privileged to run phase 2 of system generation. In fact, phase 2 should only be run automatically by performing the phase 1 procedures.

IO ERROR

An I/O error has occurred probably due to hardware failure during creation of the checkpoint file.

MARK TIME ERROR

SGN2 has attempted a MARK TIME and the request failed.

OPEN ERROR ON SYSBLD.CMD

[11,17]SYSBLD.CMD could not be opened.

OPTION LINE SYNTAX ERROR

A syntax error exists in an SGN2 directive.

READ ERROR ON COMMAND FILE

Command file cannot be read. This is probably due to hardware errors.

REQUEST ERROR

SGN2 has attempted to REQUEST a task and the request failed for a reason other than not being installed.

TASK NOT INSTALLED

The task ...MFT, which performs some of the MCR functions, is not installed. It may be possible to perform the installation after SGN2 completes.

UNABLE TO FIND ATL FOR SYSTEM DISK HANDLER

SGN2 searches the active task list expecting to find an entry for the system disk. No entry is found which

suggests that the system disk handler has terminated abnormally. Progress beyond this point is impossible.

UNABLE TO FIND ATL FOR TERMINAL HANDLER (TT....)

SGN2 searches the active task list for an entry for the task TT...., but did not find one. It is possible for phase 2 to complete, but no errors can be reported and no communication can be established with the system until a terminal handler is resident.

WAITFOR ERROR

A WAITFOR request has failed.

READER'S COMMENTS

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

Did you find errors in this manual? If so, specify by page.

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

Is there sufficient documentation on associated system programs required for use of the software described in this manual? If not, what material is missing and where should it be placed?

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

- Assembly language programmer
- Higher-level language programmer
- Occasional programmer (experienced)
- User with little programming experience
- Student programmer
- Non-programmer interested in computer concepts and capabilities

Name _____ Date _____

Organization _____

Street _____

City _____ State _____ Zip Code _____

or

Please cut along this line.

Fold Here

Do Not Tear - Fold Here and Staple

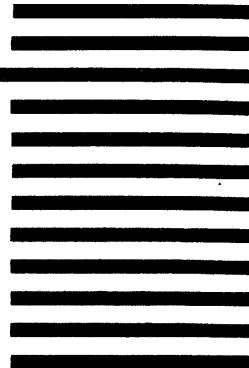
**FIRST CLASS
PERMIT NO. 33
MAYNARD, MASS.**

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

Postage will be paid by:

digital

Software Documentation
146 Main Street ML5-5/E39
Maynard, Massachusetts 01754



digital

digital equipment corporation