

**VAX-11 RSX  
Installation Guide and  
Release Notes**

Order Number: AA-C.139A-TF

digital  
software

# **VAX-11 RSX Installation Guide and Release Notes**

Order Number: AA-CJ39A-TE

**September 1984**

<b>Revision/Update Information:</b>	This is a new manual.
<b>Operating System and Version:</b>	VAX/VMS Version 4.0
<b>Software Version:</b>	VAX-11 RSX Version 1.0

**digital equipment corporation  
maynard, massachusetts**

---

**September 1984**

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

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

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

---

Copyright ©1984 by Digital Equipment Corporation

All Rights Reserved.  
Printed in U.S.A.

---

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

The following are trademarks of Digital Equipment Corporation:

DEC	DIBOL	UNIBUS
DEC/CMS	EduSystem	VAX
DEC/MMS	IAS	VAXcluster
DECnet	MASSBUS	VMS
DECsystem-10	PDP	VT
DECSYSTEM-20	PDT	
DECUS	RSTS	
DECwriter	RSX	

**digital**

ZK-2418

This document was prepared using an in-house documentation production system. All page composition and make-up was performed by T<sub>E</sub>X, the typesetting system developed by Donald E. Knuth at Stanford University. T<sub>E</sub>X is a registered trademark of the American Mathematical Society.

---

# Contents

---

PREFACE	vii
---------	-----

---

<b>SECTION 1 VAX-11 RSX INSTALLATION GUIDE</b>	<b>1-1</b>
--	------------

---

1.1 INTRODUCTION	1-1
1.2 IDENTIFYING THE VAX-11 RSX KIT	1-1
1.3 INSTALLING VAX-11 RSX	1-1
1.3.1 Procedures to Execute Prior to Installation	1-2
1.3.2 Procedures to Execute for Installation	1-6
1.3.3 Procedures to Execute After Installation	1-8
1.3.4 Installation Example	1-8

---

<b>SECTION 2 ERROR MESSAGES</b>	<b>2-1</b>
---------------------------------	------------

---

2.1 ERROR MESSAGE FORMAT	2-1
2.2 ERROR MESSAGES	2-2

---

<b>SECTION 3 NOTES TO VAX-11 RSX VERSION 1.0</b>	<b>3-1</b>
--	------------

---

3.1 SUMMARY OF TECHNICAL CHANGES REFLECTED IN THE DOCUMENTATION	3-1
3.1.1 Changes to MCR Documentation	3-1
3.1.1.1 New MCR Commands	• 3-1
3.1.1.2 New or Revised Keywords and Parameters to MCR Commands	• 3-3
3.1.1.3 New Restrictions for MCR Commands	• 3-8
3.1.1.4 MCR Commands and/or Keywords No Longer Supported	• 3-9

## Contents

<b>3.1.2</b>	<b>Changes to Indirect Command Processor (Indirect) Documentation</b>	<b>3-9</b>
3.1.2.1	New or Revised Directives • 3-9	
3.1.2.2	New or Revised Symbols • 3-11	
3.1.2.3	New Switches • 3-15	
3.1.2.4	New Functionality • 3-15	
<b>3.1.3</b>	<b>Modifications to the Documentation for Executive Directives</b>	<b>3-15</b>
<b>3.1.4</b>	<b>Modifications to the User Environment</b>	<b>3-18</b>
<b>3.1.5</b>	<b>Modifications to the I/O and General Programming Documentation</b>	<b>3-19</b>
<hr/>		
<b>3.2</b>	<b>VAX-11 RSX RELEASE NOTES</b>	<b>3-20</b>
<b>3.2.1</b>	<b>Restriction for Common Cluster System Disks</b>	<b>3-20</b>
<b>3.2.2</b>	<b>Accessing Cluster Devices From Compatibility Mode Tasks</b>	<b>3-21</b>
<b>3.2.3</b>	<b>Enhancements to VAX-11 RSX Version 1.0</b>	<b>3-22</b>
3.2.3.1	Terminal Driver • 3-22	
3.2.3.2	Device-Independent Device Characteristics • 3-23	
3.2.3.3	Sticky Device and Directory Names on VAX-11 RSX Command Lines • 3-23	
3.2.3.4	SPWN\$ Directive • 3-24	
3.2.3.5	Send/Receive Directives • 3-24	
3.2.3.6	Longword UIC Support • 3-25	
3.2.3.7	Cluster Device Names and Rooted Directory "Devices" Support • 3-25	
3.2.3.8	Reading from TI: in Spawned MCR Command • 3-25	
3.2.3.9	Changes to VAX-11 RSX Task Names (for Multiuser or Unnamed Tasks) • 3-26	
3.2.3.10	Changes Required to User-Created RSXUSR.EXE Images • 3-27	
3.2.3.11	Default File Specifications for Common and Library Task Images • 3-29	
3.2.3.12	File Transfer Program (FLX) • 3-29	
<b>3.2.4</b>	<b>Restrictions to MCR for VAX-11 RSX Version 1.0</b>	<b>3-30</b>
3.2.4.1	Logical Name Handling • 3-30	
3.2.4.2	Device Allocation • 3-30	
3.2.4.3	Process Context • 3-30	
3.2.4.4	Old Indirect Commands • 3-31	
3.2.4.5	Differences Between VAX-11 RSX Indirect (ICM) and RSX-11M-PLUS Indirect (ICM) • 3-31	
3.2.4.6	Invoking the RMS DEF Utility and the MCR DEFINE Command • 3-32	
<b>3.2.5</b>	<b>RMS-11 V2.0 Under VAX-11 RSX</b>	<b>3-32</b>
<b>3.2.6</b>	<b>RSX-11S System Generation For VAX/VMS</b>	<b>3-32</b>

<b>3.2.7</b>	<b>DCL Restrictions</b> _____	<b>3-33</b>
3.2.7.1	Changing UIC • 3-33	
3.2.7.2	Running VAX-11 RSX Tasks from DCL • 3-34	
3.2.7.3	Entering MCR Commands in DCL • 3-35	
3.2.7.4	Restriction to Invoking RSX-11 Utilities from DCL Commands • 3-35	
<b>3.2.8</b>	<b>Restrictions to Installing Compatibility Mode Tasks</b> ____	<b>3-35</b>
<b>3.2.9</b>	<b>TMPMBX Requirement</b> _____	<b>3-36</b>
<b>3.2.10</b>	<b>BYTLM Requirement</b> _____	<b>3-36</b>
<b>3.2.11</b>	<b>Requirement for Tasks that Need to Receive Escape Sequences</b> _____	<b>3-36</b>
<b>3.2.12</b>	<b>Restrictions to the Disk Save and Compress Utility (DSC)</b> _____	<b>3-36</b>



---

# Preface

---

## Manual Objectives

This manual provides the system manager with a step-by-step procedure for installing VAX-11 RSX on a VAX/VMS operating system, and it provides users with information pertaining to the Version 1.0 release of VAX-11 RSX.

---

## Intended Audience

This manual is for system managers and system users familiar with both VAX/VMS and RSX-11 operating systems.

---

## Structure of This Document

This manual is organized into three chapters.

Chapter 1 describes the procedures that the system manager must follow to copy the files from the distribution kit and to install and verify the VAX-11 RSX software on a VAX/VMS system.

Chapter 2 describes the error messages displayed if the installation or verification fails and suggests possible user action.

Chapter 3 contains release notes for VAX-11 RSX. It describes the technical changes made in this release and any information that supplements the *VAX-11 RSX Compatibility Mode Reference Manual*.

---

## Associated Documents

- *Guide to VAX/VMS Software Installation*
- *Guide to VAX/VMS System Management and Daily Operations*
- *VAX-11 RSX Compatibility Mode Reference Manual*



---

### Conventions Used in This Document

The following conventions are observed in this manual.

Convention	Meaning
CTRL/x	A symbol that indicates the CTRL key; it must be held down while another key is pressed. For example, CTRL/C means hold down CTRL while pressing C.
xxx	A one- to three-character key symbol. For example, ESC indicates the ESCAPE key, LF indicates the LINE FEED key, and RET indicates the RETURN key.
n	A lowercase n; indicates a variable for a number.
,	Comma; separates parameters in commands.
.	A period; separates the file name from the file type in a file specification.
;	A semicolon; separates the file type from the file version number in a file specification.
...	Horizontal ellipsis; indicates that you can enter additional parameters, values, or other information.
.	Vertical ellipsis; indicates that not all the statements in an example or figure are shown.
MCR>	The explicit prompt of the Monitor Console Routine (MCR), the command interpreter provided by VAX-11 RSX.
>	The MCR prompting character that appears whenever control is returned to the compatibility mode terminal and you can type input.
\$	The VAX/VMS DIGITAL Command Language (DCL) prompting character. (Whenever control is returned to the native mode terminal and you can type input, this prompt appears.)
[ ]	Square brackets; usually indicate optional syntax. Brackets that are part of directory names, however, do not indicate optional syntax. In addition, certain MCR directives use brackets as part of their required syntax.
red ink	Color of ink used to show all user-entered commands in examples.
black ink	Base color of ink used throughout manual. In examples, however, black ink has a special meaning: All output lines and prompting characters that the system prints or displays are shown in black ink.

---

---

# VAX-11 RSX Installation Guide

---

## .1 Introduction

VAX-11 RSX is a VAX/VMS layered product that allows you to execute RSX-11 tasks on VAX-11 processors. The *VAX-11 RSX Installation Guide* details the procedures necessary to install, verify, and maintain the VAX-11 RSX Version 1.0 software on VAX/VMS Version 4.0.

---

## .2 Identifying the VAX-11 RSX Kit

VAX-11 RSX Version 1.0 is distributed on the following types of media:

- Magnetic tape  
Order number: BB-CJ34A-BE
- RK07 disks  
Order number: AY-CJ35A-BE
- RL02 disks  
Order number: BC-CJ36A-BE
- RC25 removable disks  
Order number: BK-CJ37A-BE
- RA60 disks  
Order number: BD-CJ38A-BE

Each distribution kit contains two files that are BACKUP save sets, RSX010.A and RSX010.B. RSX010.A contains the command files needed to perform both the installation and the Installation Verification Procedure (IVP). RSX010.B contains the task images, libraries, and other files that constitute VAX-11 RSX Version 1.0.

---

## .3 Installing VAX-11 RSX

The following sections describe how to install the VAX-11 RSX Version 1.0 kit.

# VAX-11 RSX Installation Guide

## 1.3.1 Procedures to Execute Prior to Installation

This section details the steps you should follow before you install VAX-11 RSX Version 1.0.

### 1 Invoke an editor to modify SYSTARTUP.COM and perform the following

- Insert the following command line:

```
$ @SYS$MANAGER:VAX11RSX [spool_device]
```

This indirect command invokes VAX11RSX.COM. The command file, VAX11RSX.COM, automatically installs VAX-11 RSX each time your system is bootstrapped and establishes the environment for VAX-11 RSX to work in.

During the installation of VAX-11 RSX, VAX11RSX.COM is placed in the system manager's account [SYSMGR]. The spool\_device parameter if specified, defines the logical device names SP and SP0. If this parameter is not specified, these logical device names are defined to be LPA0:.

- Remove any old RSX-11 start-up commands. For example:

```
$ INSTALL SYS$SYSTEM:MCR  
$ INSTALL SYS$SYSTEM:MCRTABLES  
$ INSTALL SYS$SYSTEM:RSX  
$ INSTALL SYS$SYSTEM:RSXSHR  
$ INSTALL SYS$SYSTEM:RSXUSR  
$ INSTALL SYS$SYSTEM:BACKTRANS
```

You can include the systemwide page-length value that RSX-11 utilities (for example, the Task Builder (TKB) and the PDP-11 MACRO-11 Assembler (MACRO-11)) use to determine the number of lines per page for a printed listing. If you want to specify this value, include the following line in the definition table of SYSTARTUP.COM:

```
$ DEFINE/SYSTEM SYS$LP_LINES [page-length-value]
```

SYS\$LP\_LINES is used by the run-time library routine, \$PGLN. The \$PGLN routine computes the default number of lines on a printer page. This routine can be used by any task that produces listing files and does pagination.

The new page-length value range for SYS\$LP\_LINES is 30 to 99 lines per page. If you do not specify a value, the page-length value defaults to 66 lines per page.

- Remove the symbol definitions CL, CO, and LB from the file. Also, remove any start-up commands that install compatibility mode components. VAX11RSX.COM specifies all the necessary symbol definitions and compatibility mode start-up commands.

### 2 Remove all unwanted or redundant files from the appropriate media to ensure there are enough free blocks to install VAX-11 RSX.

The following summary indicates the approximate disk block utilization requirements for VAX-11 RSX Version 1.0:

Peak disk block utilization: 4600 blocks

Net disk block utilization: 4400 blocks

**Note:** To preserve space on the system disk, DIGITAL recommends that you not log in under the SYSTEM account. The SYSTEM account, which has all privileges (including BYPASS), is intended only for software installation, bootstrapping, and system problem diagnosis. You can avoid problems by creating another account and assigning to it the minimum privileges you require.

A significant number of system disk blocks are used to contain the work files required for the installation and the files read from the distribution volume. The number of blocks (4600 blocks) used during the installation is called the **peak disk block utilization**. DIGITAL recommends that you make enough blocks available on the system disk to cover peak utilization. If you choose not to do so, VMSINSTAL will operate in an alternate mode that reduces the peak utilization.

If a system failure occurs in the alternate mode, the installation will fail. To recover from this system failure, you must provide more free blocks and begin the installation again.

When the installation is complete, the system disk returns to its normal state. However, many disk blocks are permanently used up, over and above the number in use before you began the installation. This number (4400 blocks) is called the **net disk block utilization**.

After you rebootstrap the system, you can recover additional disk blocks by purging system files that VMSINSTAL cannot purge during the installation.

Confirm the free block count by issuing the following DCL command line:

```
$ SHOW DEVICE system-device-name
```

### 3 Back up your system disk, using the Backup Utility:

```
$ BACKUP
From:  indev:
To:    outdev:
$
```

A system failure at a critical point in the installation can result in unusable files. Therefore, you should always back up your system disk before you attempt any software installation. You should use the backup copy for the installation. For more information on the VAX/VMS Backup Utility, see the *VAX/VMS Utilities Reference Volume*.

- 4 Log in at the console terminal under the system manager's account (or any other account that has SETPRV privilege).
- 5 Ensure that all users are logged out and that all batch jobs are completed by issuing the following commands:

```
$ SHOW USERS
.
.
.
$ SHOW QUEUE /BATCH
.
.
.
```

- 6 Prevent users from gaining access to the system by issuing the following command:

```
$ SET LOGINS/INTERACTIVE=0
```

## VAX-11 RSX Installation Guide

- 7 Make sure DECnet-VAX is turned off. If you are not sure whether DECnet-VAX is running on your system, enter the following command:

```
$ SHOW NETWORK
```

If the message "Network unavailable" appears, proceed to step 8. If DECnet-VAX is running, turn it off by issuing the following command:

```
$ RUN SYS$SYSTEM:NCP
```

The system responds with the NCP prompt (NCP> ). At the prompt, enter the following commands:

```
NCP>SET EXECUTOR STATE SHUT  
NCP>EXIT  
$
```

- 8 Ensure that the limits in the SYSTEM account authorization record are equal to or greater than the default limits designated in the following list:

Buffered byte count (BYTLM)	=	20480
Enqueued quota (ENQLM)	=	20
Direct I/O limit (DIOLM)	=	12
Buffered I/O limit (BIOLM)	=	12
Open file quota (FILLM)	=	20
AST limit (ASTLM)	=	20

Also ensure that the account does not have the DEFCLI flag set. This flag prevents the account from logging in under a CLI other than DCL and will cause the VAX-11 RSX IVP to fail.

To check these limits, run the VAX/VMS Authorize Utility by entering the following commands:

```
$ SET DEFAULT SYS$SYSTEM  
$ RUN AUTHORIZE
```

The system responds with the User Authorization File (UAF) prompt (UAF> ). At the prompt, enter the following command:

```
UAF>SHOW SYSTEM
```

The Authorize Utility displays the current values of the SYSTEM account's authorization record.

If necessary, you can change these values by issuing the MODIFY command in response to the UAF prompt:

```
UAF>MODIFY SYSTEM/limit=new-value
```

For example:

```
UAF>MODIFY SYSTEM/DIOLM=n
```

To modify the DEFCLI flag, issue the following command in response to the UAF prompt:

```
UAF>MODIFY SYSTEM/FLAGS=NODEFCLI
```

Reissue the SHOW command to verify that you set the new values:

```
UAF>SHOW SYSTEM
```

To exit from the Authorize Utility, type EXIT at the UAF prompt. You must log out and then log back in for the changes to take effect.

- 9** Ensure that your system has at least 9 unused global sections and at least 225 unused global pages. To determine how many global sections and global pages are currently available, enter the following commands:

```
$ RUN SYS$SYSTEM:INSTALL
INSTALL> /GLOBAL
```

If you have VAX/VMS Version 4.0 INSTALL defined as a foreign command in your SYSTARTUP.COM file, use the following command:

```
$ INSTALL LIST/GLOBAL
```

Read the last line of the display and note the following

- The number of global sections **USED**
- The number of global pages **USED**
- The number of global pages **UNUSED**

To determine the **unused** global section count, invoke the VAX/VMS System Generation Utility (SYSGEN) as follows:

```
$ RUN SYS$SYSTEM:SYSGEN
SYSGEN>USE CURRENT
SYSGEN>SHOW GBLSECTIONS
```

The number in the first column of the SYSGEN display is the current number of global sections. Subtract the number of used global sections (which was shown by INSTALL) from the current number of global sections (which is shown by the SYSGEN display). The difference is the unused global section count.

Since you know how many unused global sections and global pages are available, take the following steps, depending on your counts:

- If the unused global section count is less than 9, you must increase the GBLSECTIONS parameter with the following command:

```
SYSGEN>SET GBLSECTIONS new-value
```

To compute the new value, add 9 to the number of used global sections shown by INSTALL.

- If the number of unused global pages shown by the INSTALL /GLOBAL display is less than 225, you must increase the GBLPAGES parameter with the following command:

```
SYSGEN>SET GBLPAGES new-value
```

To compute the new value, add 225 to the number of used global pages.

- If you have modified either parameter (GBLPAGES or GBLSECTIONS), issue the following commands to write the new values into SYSGEN and exit:

```
SYSGEN>WRITE CURRENT
SYSGEN>EXIT
```

**Note:** Neither GBLPAGES nor GBLSECTIONS is a dynamic parameter. You must reboot your system before installing VAX-11 RSX Version 1.0 for the new value to take effect.

# VAX-11 RSX Installation Guide

## 1.3.2 Procedures to Execute for Installation

If you successfully completed all the steps in Section 1.3.1, you are ready to install VAX-11 RSX Version 1.0. The following installation takes approximately 30 minutes to complete.

- 1 Log in at the console terminal using a privileged account (an account that has SETPRV privilege).

- 2 Enter the following command line to initiate the installation:

```
$ @SYS$UPDATE:VMSINSTAL RSX010 ddcu:
```

ddcu: is the drive in which you have placed the VAX-11 RSX distribution kit.

At any time during the execution of VMSINSTAL, you can type a question mark (?) for help.

The installation procedure is performed in two phases: the first phase copies the files and the second phase verifies that the VAX-11 RSX utilities are working (see step 4).

At the start of the first phase, you must answer the following question:

\* Are you satisfied with the backup of your system disk [YES]?

Assuming that you have performed the backup as recommended in Section 1.3.1, step 3, press RETURN (YES is the default response).

Next, VMSINSTAL prompts you as follows:

Please mount the first volume of the set on ddcu:

\* Are you ready?

Physically mount the distribution kit in the appropriate drive (if you have not done so already) and then answer YES.

A prompt will appear to request each additional volume. Enter the response YES after you insert each volume. When all the volumes have been copied, remove the last volume from the drive.

- 3 If your system disk is configured with a common system root, the VAX-11 RSX installation will display a warning message and information about using VAX-11 RSX on common system disks. Refer to Chapter 3, Section 3.2.1 for complete information and suggestions.

- 4 VMSINSTAL prompts you as follows:

Please select one of the following sections:

- 1) Perform the installation
- 2) Create a file with a description of the installation
- 3) Both of the above

\* What would you like to do [3]:

Choose option 1 if you want to perform the installation without creating a file containing a description of the installation.

Choose option 2 if you do not want to do the installation, but want the description file.

Choose option 3 (the default) if you want to both perform the installation and create a description file.

If you choose either option 1 or 3, VMSINSTAL prompts you with the following questions:

\* Do you want to install the MCR help library [YES]?

This library supplies help for VAX-11 RSX MCR commands and for the RSX utilities supplied with VAX-11 RSX. The library requires an additional 2000 disk blocks. If you want the library, answer YES.

\* Do you want to purge files replaced by this installation [YES]?

If you do not have sufficient free disk space or you do not want the old copies of the files, answer YES.

- 5 After you decide whether to install the library, the second phase of the installation begins with a VMSINSTAL question concerning the Installation Verification Procedure (IVP):

\* Do you want to run the IVP after the installation [YES]?

IVP tests the various VAX-11 RSX components to ensure that each component was installed correctly and is working properly. IVP is not meant to be a comprehensive test of the components, but it does test some of the features of each component. For example, IVP tests LBR by having it create an object library, extract a module from that library, delete the module in the library, and then reinsert the extracted module into the library. Note that you can invoke IVP only while running VMSINSTAL.

When IVP is running and, for example, testing ICM, the following messages appear:

```
%RSX010-I-IVPSTART, Installation Verification Procedure starting
%RSX010-I-TESTSTART, ICM testing started at 17:01:48
%RSX010-S-TESTSUCCESS, ICM testing completed successfully at 17:02:26
%RSX010-S-IVPSUCCESS, the installation verification of VAX-11 RSX Version 1.0 succeeded
```

Various compatibility mode images tested during the IVP may produce status messages that are not in standard VAX/VMS format. If an error occurs, IVP will detect it and produce the message in the standard VAX/VMS format.

If an error message is displayed during the verification procedure, find the message in Section 1.3.2.1 and take corrective action.

At this point, you have answered all the questions. The VAX/VMS installation procedure will complete your VAX-11 RSX Version 1.0 installation.

- 6 For a complete description of the installation, please read SYS\$UPDATE:RSX010.TXT. This file lists each of the new images and files that VAX-11 RSX uses (providing you chose either option 2 or option 3 in step 3).
- 7 When the installation is complete, VMSINSTAL invokes the start-up procedure SYS\$MANAGER:VAX11RSX.COM (which VMSINSTAL copied from the distribution kit into the system manager's account (SYSTEM)), and then exits. You can now use VAX-11 RSX Version 1.0.



# VAX-11 RSX Installation Guide

## 1.3.3 Procedures to Execute After Installation

This section lists the steps that you should follow after the installation of VAX-11 RSX.

- 1 Read the console listing carefully. Look for warning or error messages that indicate tasks you must perform manually. Many informational messages will be displayed; you can ignore them.
- 2 Perform an incremental backup of the system disk.
- 3 Establish minimum user account quotas for individual users (for example, set quotas for open file, paging file, working set size, and subprocess creation).

## 1.3.4 Installation Example

The following example shows a typical VAX-11 RSX Version 1.0 installation on VAX/VMS Version 4.0 using the RK07 distribution kit.

**Note: If you have a common system disk, the installation procedure will vary. However, the only variance is additional system output. You will not be prompted for any additional output.**

```
Username: SYSTEM
Password: [RET]

Welcome to BABBLE, VAX/VMS Version 4.0
$ SET DEF SYS$UPDATE
$ @VMSINSTAL RSX010 DMA1:

      VAX/VMS Software Product Installation Procedure Version 4.0

It is 23-DEC-1984 at 12:00
Enter a question mark (?) at anytime for help.
* Are you satisfied with the backup of your system disk [YES]? [RET]
The following products will be processed:
  RSX   Version 1.0

      Beginning installation of RSX Version 1.0 at 12:01
(%)VMSINSTAL-I-RESTORE, Restoring product saveset A...

Please select one of the following sections:
  1) Perform the installation
  2) Create a file with a description of the installation
  3) Both of the above
* What would you like to do [3]? [RET]

An optional help library is provided for use with the MCR CLI.
This library supplies help for commands supported by MCR and for
the RSX utilities supplied with VAX-11 RSX. This library will
require approximately 2000 additional disk blocks. If you wish to
install this library on your system, answer yes to the following
question:
* Do you want to install the MCR help library [YES]? [RET]
* Do you want to purge files replaced by this installation [YES]? [RET]
* Do you want to run the IVP after the installation [YES]? [RET]

VAX11RSX.COM, the startup command procedure, is used to set up
the environment for VAX-11 RSX. During the installation it will
be placed in the [SYSMGR] directory of the system root on which
this installation is being performed. Your system startup
procedure should be modified to invoke this procedure when the
system boots. However, it will not be necessary to reboot the
```

system after the installation, since this procedure is invoked as part of the installation. See the VAX-11 RSX Installation Guide and Notes, or the procedure itself for information.

The installation procedure has no further questions to ask and can continue from this point without user assistance. The rest of the installation will take approximately 10 minutes. If you chose to run the IVP, it will take an additional 10 minutes.

(%)VMSINSTAL-I-RESTORE, Restoring product saveset B...  
(%)RSX-I-DESCRIBE, The installation is described in SYS\$UPDATE:RSX010.TXT

- 1) BACKTRANS (new image)
- 2) BCK (new image)
- 3) BRU (new image)
- 4) CNV (new image)
- 5) CRF (new image)
- 6) DEF (new image)
- 7) DES (new image)
- 8) DMP (new image)
- 9) DSC1 (new image)
- 10) DSP (new image)
- 11) EDI (new image)
- 12) FLX (new image)
- 13) ICM (new image)
- 14) IFL (new image)
- 15) INDSYS (new file)
- 16) LBR (new image)
- 17) MAC (new image)
- 18) MCR (new image)
- 19) MCRHELPLIB (new file)
- 20) MCRTABLES (miscellaneous)
- 21) ODT (new file)
- 22) PAT (new image)
- 23) PIP (new image)
- 24) QIOSYM (new file)
- 25) RMS11ODL (new file)
- 26) RMSDES (new file)
- 27) RMSLIB (new file)
- 28) RMSMAC (new file)
- 29) RST (new image)
- 30) RSX (new image)
- 31) RSXMAC (new file)
- 32) RSXSHR (new image)
- 33) RSXUSR (new image)
- 34) SLP (new image)
- 35) SYSLIB (new file)
- 36) TKB (new image)
- 37) VAX11RSX (new file)
- 38) VMLIB (new file)
- 39) ZAP (new image)

# VAX-11 RSX Installation Guide

%VMSINSTAL-I-MOVEFILES, Files will now be moved to their target directories...  
%RSX010-I-IVPSTART, Installation Verification Procedure starting

The Installation Verification Procedure (IVP) tests that VAX-11 RSX installed correctly by testing various compatibility mode utilities. This serves as a test of the utilities themselves and of the RSX AME. Also, when DCL commands exist for compatibility mode utilities (i.e. LIBRARY/RSX for LBR), these will be used to test the translation of DCL commands to MCR syntax for these utilities

Several of the compatibility mode utilities will log their normal actions to SYS\$OUTPUT. These informational messages as well as any error messages that may be produced by these utilities, will not be in the standard VAX/VMS message format. If an error occurs, the IVP will detect it and produce an error message in the standard VAX/VMS format. The IVP will also produce informational and success messages in the standard format.

%RSX010-I-TESTSTART, DMP testing started at 12:12:24  
DMP---DMP Version M08.01 (ANSI)

%RSX010-I-TESTSUCCESS, DMP testing completed successfully at 12:12:48  
%RSX010-I-TESTSTART, ICM testing started at 12:12:48  
%RSX010-I-TESTSUCCESS, ICM testing completed successfully at 12:13:26  
%RSX010-I-TESTSTART, LBR testing started at 12:13:26

Modules deleted:

OBJ3

Module "OBJ1 " replaced

Module :OBJ2 " replaced

Entry points deleted:

ENT\$A1

ENT\$B2

ENT\$A3

Modules deleted:

MCLBR3

Module "MCLBR1" replaced

Module "MCLBR2" replaced

%RSX010-I-TESTSUCCESS, LBR testing successfully completed at 12:14:47

%RSX010-I-TESTSTART, PAT testing started at 12:14:48

%RSX010-I-TESTSUCCESS, PAT testing successfully completed at 12:14:54

%RSX010-I-TESTSTART, PIP testing started at 12:14:55

PIP---PIP Version M1600 (ANSI)

%RSX010-I-TESTSUCCESS, PIP testing successfully completed at 12:16:27

%RSX010-I-TESTSTART, SLP testing started at 12:16:27

%RSX010-I-TESTSUCCESS, SLP testing successfully completed at 12:16:36

%RSX010-I-IVPSUCCESS, Installation verification of VAX-11 RSX Version 1.0 succeeded

%VMSINSTAL-I-IVPSUCCESS

VMSINSTAL procedure done at 12:16

---

## 2 Error Messages

This chapter lists the messages that are displayed if the installation or verification fails and suggests possible user action for each. Messages are displayed or printed in both uppercase and lowercase letters (when the output device is capable of displaying or printing lowercase letters).

In this section, the fixed and variable portions of the text of a message are distinguished by enclosing the variable portions in apostrophes (single quotation marks); the distinction is obvious when a message is displayed.

---

### 2.1 Error Message Format

Messages displayed by VAX-11 RSX have the following format:

```
%FACILITY-L-IDENT, TEXT  
[-FACILITY-L-IDENT,TEXT]
```

where

#### **FACILITY**

is a VAX-11 RSX facility or component name. A percent sign (%) prefixes the first message issued, and a hyphen (-) prefixes each subsequent message. For the VAX-11 RSX kit installation procedure, the facility name is RSX010.

#### **L**

is a severity level indicator with one of the following values:

---

<b>Code</b>	<b>Meaning</b>
S	Success
I	Information
W	Warning
E	Error
F	Fatal, or severe error

---

#### **IDENT**

is an abbreviation of the message text; the message descriptions are alphabetized by this abbreviation.

#### **TEXT**

is the explanation of the message.

## Error Messages

---

### 2.2 Error Messages

**BASELEVEL,** RSX-11 "version" must be installed on VAX/VMS field test baselevel(s) "bl"

**Explanation:** This version of VAX-11 RSX can be installed only on the VAX/VMS field test base level specified.

**User Action:** You have either the wrong version of VAX-11 RSX or the wrong field test base level of VAX/VMS. You must obtain the correct version of either VAX/VMS or VAX-11 RSX.

**DESCRIBE,** the installation is described in SYS\$UPDATE:RSXvvu.TXT

**Explanation:** Indicates where the installation description can be found after the installation is complete.

**User Action:** None.

**ERRORS,** of the "nn" action listed above, the following "nn" should be reviewed:

**Explanation:** If errors occur during the kit installation, this message will summarize the actions on which the errors occurred.

**User Action:** Review the specified actions.

**IVPFAILURE,** the installation verification of VAX-11 RSX Version 1.0 failed

**Explanation:** The installation did not pass the kit Installation Verification Procedure (IVP). Previous error messages should have indicated the specific reason(s) for the failure.

**User Action:** Submit an SPR describing the problem and include the console listing of the kit installation.

**IVPSUCCESS,** the installation verification of VAX-11 RSX Version 1.0 succeeded

**Explanation:** IVP did not detect any errors in the kit installation.

**User Action:** None.

**NOBLOCKS,** RSX-11 "version" requires "n" additional blocks after installation

**Explanation:** After installation, the specified version of VAX-11 RSX will use the specified number of blocks on the system disk. If the specified number of blocks is not available, the installation will fail.

**User Action:** Purge or delete unnecessary files from the system disk.

**TESTERROR,** "utility" fail test of "test description"

**Explanation:** An error occurred in performing the specified utility test during the IVP.

**User Action:** Submit an SPR describing the problem and include the console listing of the kit installation.

**TESTSTART,** "utility" testing started

**Explanation:** IVP has begun testing the specified utility.

**User Action:** None.

TESTSUCCESS, "utility" testing completed successfully

**Explanation:** IVP did not detect any errors.

**User Action:** None.

IVPSTART, starting installation verification procedure

**Explanation:** IVP is beginning.

**User Action:** None.

VERSION, RSX-11 "version" must be installed on existing VAX/VMS "version" system

**Explanation:** This version of VAX-11 RSX can be installed only on the specified version of VAX/VMS.

**User Action:** You have the wrong version of either VAX-11 RSX or VAX/VMS. You must obtain the correct version.



# 3 **Notes to VAX-11 RSX Version 1.0**

---

## 3.1 **Summary of Technical Changes Reflected in the Documentation**

---

This section contains all the new and modified features for VAX-11 RSX. The new and modified features are documented in the *VAX-11 RSX Compatibility Mode Reference Manual*.

### 3.1.1 **Changes to MCR Documentation**

---

The following sections describe the changes to MCR for VAX-11 RSX.

#### 3.1.1.1 **New MCR Commands**

---

##### **ATTACH**

Enables you to switch control of your terminal from your current process to another process in your job.

##### **BAD**

Tests disks and tapes for the location and number of bad blocks.

##### **CREATE**

Creates one or more sequential disk files from records that follow the command in the input stream.

##### **CREATE/DIRECTORY**

Creates a new directory or subdirectory for cataloging files.

##### **CREATE/FDL**

Invokes the Create/FDL Utility (CREATE/FDL) to use the specifications in an FDL file to create a new, empty data file.

##### **CREATE/NAME\_TABLE**

Creates a logical name table in which to define logical names. The /NAME\_TABLE keyword is required.

##### **DEASSIGN**

Cancels one or more logical name assignments made with the ALLOCATE, ASN, DEFINE, or MOUNT commands.

##### **DEFINE**

Creates a logical name table entry and assigns one or more equivalence name strings to the specified logical name.



### **DIFFERENCES**

Compares the contents of two disk files and creates a listing of those records that do not match.

### **DUMP**

Displays or prints the contents of files or volumes in ASCII, hexadecimal, or octal representation.

### **HELP**

Provides information available in the system HELP files or in any HELP library you specify.

### **MAIL**

Invokes the VAX/VMS Personal Mail Utility (MAIL), which is used to send messages to other users of the system.

### **MERGE**

Invokes the VAX-11 Merge Utility to combine similarly sorted input files and create a single output file.

### **RESUME**

Resumes execution of an indirect command file that was interrupted when a .PAUSE directive was issued by the file.

### **RUNOFF**

Invokes the DIGITAL Standard Runoff (DSR) text formatter to format one or more ASCII files.

### **RUNOFF/CONTENTS**

Invokes the DIGITAL Standard Runoff (DSR) table of contents utility to create a file that can be processed by DSR to make a table of contents.

### **RUNOFF/INDEX**

Invokes the DIGITAL Standard Runoff (DSR) indexing utility to create a file that can be processed by DSR to make an index.

### **SEARCH**

Searches one or more files for a specified string or strings and lists all the lines containing occurrences of the string(s).

### **SET COMMAND**

Adds commands that are defined in a command description file to your process command set or a command tables file.

### **SHOW CLUSTER**

Displays cluster activity and performance.

### **SORT**

Invokes the VAX-11 SORT utility to sort the records in a file based on one or more key fields within each record.

## SPAWN

Creates a subprocess of the current process.

---

### 3.1.1.2 New or Revised Keywords and Parameters to MCR Commands

#### ALLOCATE

/[NO]LOG—Controls the message displayed at your terminal when the allocate operation is completed.

/GENERIC—Specifies the type of device to be allocated.

#### APPEND

/BACKUP—Selects files according to the date of their most recent backup.

/BEFORE[=time]—Selects only those files that are dated before the specified time.

/BY\_OWNER[=uic]—Selects one or more files only if their UIC matches the specified owner UIC.

/[NO]CONFIRM—Controls whether a request is issued before each individual operation to confirm that the operation should be performed on that file.

/CREATED—Selects files based on their dates of creation.

/EXCLUDE=(file-spec[,...])—Excludes any files that match the listed file specification from the APPEND operation.

/EXPIRED—Selects files according to the dates on which they will expire.

/MODIFIED—Selects files according to the dates on which they were last modified.

/[NO]NEW\_VERSION—Replaces the keyword /[NO]NEW in functionality.

/SINCE[=time]—Selects only those files that are dated after the specified time.

#### ASN

/GLOBAL—Replaces the keyword /GBL in functionality.

#### COPY

/BACKUP—Selects files according to the date of their most recent backup.

/BEFORE[=time]—Selects only those files that are dated before the specified time.

/BY\_OWNER[=uic]—Selects one or more files only if their UIC matches the specified owner UIC.

/[NO]CONFIRM—Controls whether a request is issued before each individual operation to confirm that the operation should be performed on that file.

/CREATED—Selects files based on their dates of creation.

/EXCLUDE=(file-spec[,...])—Excludes any files that match the listed file specification from the copy operation.

## Notes to VAX-11 RSX Version 1.0

/EXPIRED—Selects files according to the dates on which they will expire.

/MODIFIED—Selects files according to the dates on which they were last modified.

/SINCE[=time]—Selects only those files that are dated after the specified time.

### DELETE

/BACKUP—Selects files according to the date of their most recent backup.

/BY\_OWNER[=uic]—Selects one or more files only if their UIC matches the specified owner UIC.

/[NO]ERASE—Writes a block of zeroes for each block in each file that is deleted.

/EXCLUDE=(file-spec[,...])—Excludes any files that match the listed file specification from the DELETE operation.

### DIRECTORY

/ACL—Controls whether the Access Control List (ACL) for each directory is displayed.

/BACKUP—Selects files according to the date of their most recent backup.

/BEFORE[=time]—Selects only those files that are dated before the specified time.

/BY\_OWNER[=uic]—Selects one or more files only if their UIC matches the specified owner UIC.

/[NO]CONFIRM—Controls whether a request is issued before each individual operation to confirm that the operation should be performed on that file.

/[NO]FILE\_ID—Includes the file identification number for each specified file.

/[NO]GRAND\_TOTAL—Inhibits the listing of all individual file information and all trailing line information except for a one-line summary of the number of directories and files specified.

/SECURITY—Controls whether information about file security is displayed.

/SELECT—Allows you to control the files selected based on /SELECT options.

/WIDTH—Allows you to change the format of the directory display for four elements: file name field width, entire display width, owner field width, and full size field width.

### EXECUTE PROCEDURE (@)

p1,p2,...,p8,p9—Specifies from one to nine optional parameters to pass to the indirect command file.

/[NO]CLI, /LB, /[NO]LO, /[NO]MC, /[NO]TR—These keywords are described fully in Section 5 of the *VAX-11 RSX Compatibility Mode Reference Manual*.

## INITIALIZE

/[NO]ERASE—Controls the Data Security Erase (DSE) operation on the volume before initializing it.

/[NO]HIGHWATER—Controls highwater-marking on the specified volume.

/OWNER\_UIC=[g,m]—The group (g) field can have a value in the range 0 through 37777(octal) and the member field can have a value in the range 0 through 177777(octal).

## LOGIN

/TABLES—Specifies the name of an alternate CLI table to override the default listed in the User Authorization File (UAF).

## LOGOUT

/HANGUP—For dial-up terminals, determines whether the phone will hang up whenever you log out.

## MOUNT

/BLOCK=n—Replaces the /BLOCKSIZE=n keyword in functionality.

/[NO]UNLOAD—Controls whether the disk or tape volume(s) specified in the MOUNT command is unloaded when it is dismounted.

## PRINT

/[NO]FLAG[=option]—Replaces the /[NO]FLAG\_PAGE keyword in functionality.

/NOTE=string—Allows you to specify a message that will appear on the flag page for the job.

/OPERATOR=string—Allows you to specify a message that will be sent to the operator.

/PAGES=(n[,m,...])—Replaces the /PAGE\_COUNT=n keyword in functionality.

/PARAMETERS=(parameters[,...])—Specifies from zero to eight optional parameters to be passed to the job.

/[NO]PASSALL—Specifies that the output symbiont print the file in PASSALL mode.

/[NO]RESTART—Specifies whether the job can restart after a crash or a STOP/REQUEUE command.

/SETUP=module[,...]—Calls for the specified module(s) to be extracted from the device control library and copied to the printer before a file is printed.

/[NO]TRAILER[=option]—Controls whether a trailer page is needed at the end of a file.

/USER=username—Allows you to submit a print job for another user.

### PURGE

/BACKUP—Selects files according to the date of their most recent backup.

/BEFORE[=time]—Selects only those files that are dated before the specified time.

/BY\_OWNER[=uic]—Selects one or more files only if their UIC matches the specified owner UIC.

/[NO]CONFIRM—Controls whether a request is issued before each individual operation to confirm that the operation should be performed on that file.

/CREATED—Selects files based on their dates of creation.

/[NO]ERASE—Writes a block of zeroes for each block in each file that is purged.

/EXCLUDE=(file-spec[,...])—Excludes any files that match the listed file specification from the PURGE operation.

/EXPIRED—Selects files according to the dates on which they will expire.

/MODIFIED—Selects files according to the dates on which they were last modified.

/SINCE[=time]—Selects only those files that are dated after the specified time.

### RENAME

/BACKUP—Selects files according to the date of their most recent backup.

/BEFORE[=time]—Selects only those files that are dated before the specified time.

/BY\_OWNER[=uic]—Selects one or more files only if their UIC matches the specified owner UIC.

/CREATED—Selects files based on their dates of creation.

/EXCLUDE=(file-spec[,...])—Excludes any files that match the listed file specification from the RENAME operation.

/EXPIRED—Selects files according to the dates on which they will expire.

/MODIFIED—Selects files according to the dates on which they were last modified.

/SINCE[=time]—Selects only those files that are dated after the specified time.

### RUN (PROCESS)

/[NO]DETACHED—Controls whether a detached process or subprocess is created to execute the task image.

/[NO]DUMP—Writes the contents of the address space to a file SYS\$LOGIN:IMAGEDUMP.DMP if a task image terminates due to an unhandled condition.

/ENQUEUE\_LIMIT=quota—Specifies the maximum number of lock requests that a process can queue at any one time.

/EXTENT=quota—Limits the maximum working set size of a task image.

/PROCESS=*process-name*—Replaces, in functionality, the keyword  
/PROCESS\_NAME=*process-name*.

### SET

ACL—Allows you to set up an Access Control List (ACL) for a file or device.

AUDIT—Enables or disables forms of security auditing.

FILE/ACL—Modifies the Access Control List (ACL) of one or more files.

QUEUE—Changes the current status or attributes of the specified queue.

RESTART\_VALUE—Establishes a test value for restarting portions of batch jobs.

### SHOW

AUDIT—Shows the security features that are enabled.

CLUSTER—Shows cluster activity and performance.

KEY—Shows key definitions.

QUEUE/CHARACTERISTICS—Shows characteristic names and numbers that are enabled on queues.

QUEUE/FORM—Shows form names and numbers that are available on queues.

### STOP/REQUEUE

*alt-queue-name*—Specifies a new queue to hold the job that is being requeued.

/PRIORITY—Specifies a priority for the requeued job.

### SUBMIT

/CHARACTERISTIC=(*characteristic[,...]*)—Specifies one or more characteristics desired for the job.

/CLI=*file-spec*—Enables you to specify a different command interpreter to use in processing the job.

/PARAMETERS—Specifies from one to nine optional parameters to be passed to the indirect command file.

/USER=*username*—Allows you to submit a job for another user.

### TYPE

/BACKUP—Selects files according to the date of their most recent backup.

/BEFORE[=*time*]—Selects only those files that are dated before the specified time.

/BY\_OWNER[=*uic*]—Selects one or more files only if their UIC matches the specified owner UIC.

/[NO]CONFIRM—Controls whether a request is issued before each individual operation to confirm that the operation should be performed on that file.

/CREATED—Selects files based on their dates of creation.

`/EXCLUDE=(file-spec[,...])`—Excludes any files that match the listed file specification from the TYPE operation.

`/EXPIRED`—Selects files according to the dates on which they will expire.

`/MODIFIED`—Selects files according to the dates on which they were last modified.

`/[NO]PAGE`—Controls whether the output is displayed one page at a time.

`/SINCE[=time]`—Selects only those files that are dated after the specified time.

### UFD

`/PROTECTION=code`—Protection must be specified in the following format:

`SYSTEM:RWED,OWNER:RWED,GROUP:RWED,WORLD:RWED`

---

### 3.1.1.3 New Restrictions for MCR Commands

#### ALLOCATE

You can allocate and mount devices in an indirect command file. The devices remain mounted after the command file executes. However, if you allocate the devices in the indirect command file, but do not mount them, the devices will not remain allocated after Indirect exits.

#### EXAMINE

You must have user mode read and write access to the virtual memory location contents you want to examine. Also, the EXAMINE command analyzes expressions arithmetically. Therefore, keywords are interpreted correctly only when they appear immediately after the command name.

#### MOUNT

You cannot allocate a device interactively and then mount the device from an indirect command file.

#### SET /UIC

If you use SET /UIC to change to a UIC greater than [377,377], your UIC will be changed if you have CMKRNL privilege. However, your default directory will not change because UIC-formatted directories are not allowed to have a group or member code that exceeds 377(octal).

#### SUBMIT

The batch file is not processed automatically by Indirect. Your batch file must contain an @file command to invoke an indirect command file. Also, the current directory for the batch job is initially the default login directory. However, the batch command file can change the current directory for the batch job. The current directory for the batch job is the directory that is searched.

---

**3.1.1.4 MCR Commands and/or Keywords No Longer Supported**

EXECUTE PROCEDURE (@) /OUTPUT=file-spec

EXIT

GOTO

ON Severity Level

SET [NO]ON

SET [NO]VERIFY

UFD /ALLOCATION=number-entries

---

**3.1.2 Changes to Indirect Command Processor (Indirect) Documentation**

The following sections describe the changes for the RSX-11M-PLUS Indirect Command Processor (Indirect) that is supported under VAX-11 RSX.

Note that Indirect now accepts directives and command lines in lowercase as well as uppercase.

---

**3.1.2.1 New or Revised Directives**

**.ASK**

Allows a default value and a timeout parameter to be specified in the command line.

**.ASKN**

Allows a timeout parameter to be specified in the command line.

**.ASKS**

Allows a default value and a timeout parameter to be specified in the command line.

**.DATA**

Allows a tab after the command.



## Notes to VAX-11 RSX Version 1.0

### **.ENABLE/.DISABLE**

New options for enabling and disabling are (default setting and default action):

ATTACH	Enabled: Indirect attaches to TI: for comment lines.
CONTROL-Z	Disabled: Indirect exits when CTRL/Z is used in response to a .ASKx directive.
DECIMAL	Disabled: Indirect creates or redefines numeric symbols as octal.
DETACH	Enabled: Indirect detaches from the terminal when processing command lines.
DISPLAY	Enabled: Indirect displays the current fields for the .ASKx directives and "@ <EOF>".
GLOBAL	Enabled: Indirect defines as global, to all levels of indirect files, symbol names that begin with a dollar sign (\$).
MCR	Enabled: Indirect passes to MCR commands that it does not process.
OVERFLOW	Disabled: Indirect does not allow signed arithmetic in numeric expressions.
TIMEOUT	Enabled: Indirect uses the timeout parameters specified for the .ASKx directives.
TRACE	Disabled: Indirect does not display processed commands on the terminal.
TRUNCATE	Disabled: Indirect does not ignore truncate errors on a .READ directive.

### **.ERASE SYMBOL**

Deletes a symbol from the global symbol table.

### **.IF ENABLED/DISABLED**

Tests the .ENABLED/.DISABLED options.

### **.IFACT/.IFNACT**

Tests whether a task is active. For VAX-11 RSX, a task is always inactive.

### **.IFINS/.IFNINS**

Tests whether a task is installed. For VAX-11 RSX, a task is always installed.

### **.IFLOA/.IFNLOA**

Tests whether a device driver is loaded. For VAX-11 RSX, a device driver is always loaded.

### **.OPENR**

Opens a file for reading with the .READ command.

### **.PARSE**

Parses strings into substrings.

**.READ**

Reads a record from a file into a specified string variable.

**.SETD/.SETO**

Redefines the radix of a numeric symbol.

**.SETL**

Defines or redefines a logical symbol and assigns the symbol a true or false value.

**.TEST**

Can also locate a substring.

**.TESTDEVICE**

Returns information about a device in the system.

**.TESTFILE**

Determines whether a specified file exists and the physical device associated with a logical name.

**.TESTPARTITION**

Returns the following default information (because there are no partitions for VAX-11 RSX):

partition-name,400,size,SYS

---

**3.1.2.2 New or Revised Symbols**

**<ACCOUN>**

For VAX-11 RSX, the <ACCOUN> fields are null.

**<ALTMOD>**

Tests whether the last question was answered with an altmode or an escape. Synonymous with <ESCAPE>.

**<BASLIN>**

Always set to false for VAX-11 RSX.

**<CLI>**

For VAX-11 RSX, assigned the acronym MCR.

**<CONFIG>**

Contains the parameter defaults specified when the current Indirect task was built.

**<DIRECT>**

Contains the current default directory for the process.

## Notes to VAX-11 RSX Version 1.0

### <EOF>

Tests whether the last .ASKx or .READ directive resulted in reading past the end of the file.

### <ERRCTL>

Controls Indirect processing when an error is encountered.

### <ERRNUM>

Contains the class number of an error that Indirect has finished processing.

### <ERROR>

One of five symbols that define the exit status of an executed MCR-specified command line. The other symbols are <SEVERE>, <SUCCES>, <WARNIN>, and <NOSTAT>. These symbols are associated with the numeric codes for <EXSTAT>.

### <ERRSEV>

Contains the error severity mask associated with the error that Indirect has finished processing.

### <ERSEEN>

Tests the error code of a file, the exit status of an executed CLI-specified command line, whether <EOF> is true, the value for <ERRNUM>, or whether the .SETT directive was used.

### <EXSTRI>

Contains build-time information about the Indirect task or the string results from a more deeply nested indirect command file. The results are sent to the calling indirect command file.

### <FALSE>

Logical constant used for comparisons with the .IF directive or as a default for the .ASK directive.

### <FILATR>

Contains file-attribute and device information obtained from offsets for the File Descriptor Block (FDB).

### <FILERR>

Contains the File Control Services (FCS-11) or I/O driver status code resulting from a .TESTFILE, .OPENx, or .READ directive information.

### <FILR2>

Assigned the error code 0 for VAX-11 RSX.

### <FILSPC>

Contains the fully qualified specification for a file referred to with the last .OPENx or .TESTFILE directive operation or in the last specification for a nested command file.

### <FMASK>

Contains octal values that best reflect the kind of RSX-11 environment VAX-11 RSX can provide.

### <FORATT>

Contains the octal value of the file attributes used to open the data files.

### <IAS>

Always set to false for VAX-11 RSX.

### <LIBUIC>

Assigned the system logical name RSX\$LIBUIC for VAX-11 RSX. If the logical name does not exist, the symbol is assigned the value [1,1].

### <LOCAL>

Always set to true for VAX-11 RSX.

### <LOGDEV>

Assigned to SY0: for VAX-11 RSX.

### <LOGUIC>

Assigned the current default UIC for the process for VAX-11 RSX.

### <MAPPED>

Always set to true for VAX-11 RSX.

### <MEMSIZ>

Assigned a default value of 2044 (10) words, which represents the largest memory size available under RSX-11.

### <NETUIC>

Assigned the system logical name RSX\$NETUIC and set to [1,54] for VAX-11 RSX.

### <NETNOD>

Contains the DECnet node name for the system or VAXAME.

### <NUMBER>

Tests whether the last string entered in response to a .ASKS directive or tested with a .TEST directive contains only numeric characters.

### <OCTAL>

Tests whether the value used in a .ASKN or .TEST directive operation is octal.

### <PRIVIL>

Always set to false for VAX-11 RSX.

## Notes to VAX-11 RSX Version 1.0

### <RSX11D>

Always set to false for VAX-11 RSX.

### <SYMTYP>

Contains the numeric code for the type of symbol tested with the .TEST directive.

### <SYSDEV>

For VAX-11 RSX, assigned to SY0:.

### <SYSID>

Contains the base-level number for the system.

### <SYSTEM>

Assigned the octal value 5 for VAX-11 RSX.

### <SYSUIC>

For VAX-11 RSX, assigned the system logical name RSX\$SYSUIC if it exists; otherwise, assigned the value [1,54].

### <SYUNIT>

Assigned the number 0 for VAX-11 RSX.

### <TICLPP>

Contains the current page length setting for the terminal.

### <TICWID>

Contains the current page width setting for the terminal.

### <TIMOUT>

Tests whether the last .ASK directive timed out while waiting for a user response.

### <TISPED>

Assigned the baud rate 0 for VAX-11 RSX system.

### <TITYPE>

Contains the terminal type of the terminal from which Indirect is running.

### <TRUE>

Logical constant used for comparisons with the .IF directive or as a default for the .ASK directive.

---

### 3.1.2.3 New Switches

**/[NO]CLI**

(/CLI default) Passes commands not processed by Indirect to MCR. Synonymous with /[NO]MC.

**/LB**

Indicates that the specified file is a universal library of command procedures and that the specified module is the procedure to be executed.

**/LO**

(/NOLO default) Indicates that when a new file is executing, it can have access to the local symbols created by its calling command file. Also indicates that any local symbols created by the new command file will be defined as local symbols for the calling command file.

**/[NO]MC**

Passes commands not processed by Indirect to MCR. Synonymous with /[NO]CLI.

---

### 3.1.2.4 New Functionality

**Universal Library Support**

When command procedures, which are indirect command files, are inserted into a universal library by the Librarian Utility Program (LBR), they can then be referenced with the new /LB:module switch.

Command libraries are built by creating a universal library and inserting command files into it. The procedures in the library can then be referenced in the following form:

```
@command-library/LB:module
```

**Substitution Format Control**

Format control is available for numeric values being converted to string values and for string and logical values being placed in a substitution operation.

---

## 3.1.3 Modifications to the Documentation for Executive Directives

**CLEF\$**

A task image must be associated with group global event flags to access flags in the group global cluster.

### **CRGF\$**

A task image must be associated with group global event flags to access flags in the group global cluster.

A task image is associated with a group global event flag cluster only if it issues this directive.

### **EXIF\$**

A task image must be associated with group global event flags to access flags in the group global cluster.

### **GDVI\$**

Documentation for the Get Device Information directive (GDVI\$) has been added. The GDVI\$ directive is a new directive for VAX-11 RSX. GDVI\$ instructs the system to fill a buffer with information about a physical device unit to which a LUN is assigned.

### **GLUN\$**

Mailboxes may have unit numbers greater than eight bits. To refer to mailboxes from a compatibility mode task, a logical name must be used. The mailbox can be the task's SYS\$INPUT file or can have a \$\$n logical name. See Section 2 of the *VAX-11 RSX Compatibility Mode Reference Manual* for a discussion on device naming.

### **GMCR\$**

The system processes all command lines to translate all symbols known to the CLI (DCL only).

### **MRKT\$**

A task image must be associated with group global event flags to access flags in the group global cluster.

### **QIOS\$**

A task image must be associated with group global event flags to access flags in the group global cluster.

### **QIOW\$**

A task image must be associated with group global event flags to access flags in the group global cluster.

### **RADF\$**

A task image must be associated with group global event flags to access flags in the group global cluster.

### **RCST\$**

VAX-11 RSX dequeues the oldest message received for the task. If the sender task name is specified, the oldest message received from that task is dequeued. If there is no such message (either no messages or none from the specified task), VAX-11 RSX issues a Hibernate system service on behalf of the process issuing this directive.

Because the logical name of a temporary mailbox is implicitly qualified by VAX/VMS with the group number of the owner, this directive is useful only for passing data between processes within the same group.

### **RCVDS\$**

VAX-11 RSX dequeues the oldest message received for the task. If the sender task name is specified, the oldest message received from that task is dequeued. If there is no such message (either no messages or none from the specified task), VAX-11 RSX returns status IE.ITS (no data currently available).

Because the logical name of a temporary mailbox is implicitly qualified by VAX/VMS with the group number of the owner, this directive is useful only for passing data among processes within the same group.

### **RCVX\$**

VAX-11 RSX dequeues the oldest message received for the task. If the sender task name is specified, the oldest message received from that task is dequeued. If there is no such message (either no messages or none from the specified task), VAX-11 RSX causes the task to exit by issuing a \$EXIT system service to VAX/VMS for the process.

Because the logical name of a temporary mailbox is implicitly qualified by VAX/VMS with the group number of the owner, this directive is useful only for passing data among processes within the same group.

### **RDXF\$**

A task image must be associated with group global event flags to access flags in the group global cluster.

### **SDAT\$**

A task image must be associated with group global event flags to access flags in the group global cluster.

Because the logical name of a temporary mailbox is implicitly qualified by VAX/VMS with the group number of the owner, this directive is useful only for passing data among processes within the same group.

### **SETF\$**

A task image must be associated with group global event flags to access flags in the group global cluster.

### **SPWN\$**

A task image must be associated with group global event flags to access flags in the group global cluster.

If a CLI task is specified, a mailbox is created and the command line is written to the mailbox. Because the input part of TI (\$I) is set equivalent to the mailbox, the CLI reads the command from the mailbox. The output part of TI (\$O) will be the parent process' SYS\$OUTPUT, or the terminal that the unum and dnam parameters specify.

The task names "MCR," "MCR...," "CLI," and "CLI..." request the MCR CLI. The task names "DCL" and "DCL..." request the DCL CLI.



## Notes to VAX-11 RSX Version 1.0

When a command line that affects the process context is spawned to MCR it affects the parent process as well as the subprocess. When an equivalent command line is spawned to DCL, only the subprocess is affected.

The subprocess this directive creates will have a process name that is the same as the specified task name. However, if another process running in the same UIC group has the same process name, VAX-11 RSX modifies the subprocess name to make it unique.

### **STLOS**

A task image must be associated with group global event flags to access flags in the group global cluster.

### **STSES**

A task image must be associated with group global event flags to access flags in the group global cluster.

### **WTLOS**

A task image must be associated with group global event flags to access flags in the group global cluster.

### **WTSES**

A task image must be associated with group global event flags to access flags in the group global cluster.

---

## 3.1.4 Modifications to the User Environment

### **Command Line Editing**

VAX-11 RSX supports command line editing using terminal control key sequences and function keys.

### **Job Logical Name Table**

Job logical names are shared by an interactive or batch process and all its subprocesses. The job logical name table is deleted when an interactive or batch process terminates. No privilege is required for any process in the job to insert, translate, or delete logical names in this table.

### **\$\$n Logical Names**

\$\$n logical names are logical names that consist of two ASCII characters (\$\$) followed by an octal unit number ranging from 0 to 377. You can place these logical names in any desired logical name table (shared or private), provided the table is included in the logical name search list LNM\$FILE\_DEV.

### **Mounting Devices**

If you mount a device in a subprocess, the device will be owned by the main process. This allows the main process and all subprocesses in your job to access a device. When the subprocess exits, the device remains mounted by the main process.

## Search Lists

The use of multiple equivalence names is referred to as the use of search lists. RSX-11 tasks using FCS-11 or ACP QIOs cannot use search lists. An attempt to use one results in a "bad device name" error. RSX-11 tasks using RMS-11 can use search lists in the same manner as native mode images.

## 1.5 Modifications to the I/O and General Programming Documentation

### F2.FDX

F2.FDX (full-duplex terminal driver) in word 1, bit 7, is now returned by the IO.GTS I/O function.

### IO.CRE

A nonstandard form of the IO.CRE function is documented and supported for VAX-11 RSX only. The user can migrate programs that are linked to FCS to RSX-11 because FCS uses the nonstandard version of IO.CRE only when running under VAX-11 RSX.

### TC.TPP

TC.TPP terminal characteristic code now supports the following terminals: LA12, LA100, LQP02, and the VT200 series.

### Unsupported Devices

Documentation is provided on unsupported devices. The VAX-11 RSX AME recognizes the unsupported device category for an RSX-11 task image that may need to perform I/O to a different type of device (for example, a mailbox).

### Floating Point Instruction Emulation

The following notes apply to floating point instruction emulation:

- If a floating point overflow or division by zero occurs during the emulation of a floating point instruction, the emulator generates a reserved operand in the destination of that instruction.
- If a floating point underflow occurs during the emulation of a floating point instruction, the emulator generates a zero value in the destination of that instruction.
- A floating point OP code exception interrupt cannot be disabled with the LDFPS instruction as it can be on the PDP-11/70. If a task specifies a floating point exception AST with the SFPA\$ directive and a floating point error code is detected by the emulator, the task will always get the AST with the appropriate exception code placed on the stack.
- Floating point maintenance mode is not supported by the emulator.

### I/O Operations to Mailboxes

The following notes apply to I/O operations to mailboxes:

- When a native mode image creates a mailbox that is intended for access by an RSX-11 task image, the native mode image must create a logical name resembling the format of an RSX-11 device name. If the native mode image does not do so, the RSX-11 task image will be unable to access the mailbox.

The logical name created by the image should be a two-character device name (\$\$) and should have an octal unit number within the range 0 to 377. The name should not have superfluous leading zeros. If a logical name of some other form is defined, then a logical name of the form \$\$ must still be defined to the same mailbox. Note that certain \$\$n logical names are reserved for dynamic definition by images. \$\$1 logical names are described in detail in the *VAX-11 RSX Compatibility Mode Reference Manual*.

- Mailboxes created by VAX-11 RSX for the purpose of emulating send/receive directives do not have \$\$ logical names. The use of these mailboxes by VAX-11 RSX is transparent to RSX-11 task images.

---

## 3.2 VAX-11 RSX Release Notes

Sections 3.2.1 through 3.2.10 describe new or modified functionality for and restrictions to VAX-11 RSX Version 1.0.

---

### 3.2.1 Restriction for Common Cluster System Disks

When a VAX/VMS Version 4.0 system is configured with a common system disk (regardless of whether it is in a VAXcluster), the standard system root logical name SYS\$SYSROOT becomes a search list. The root logical name consists of SYS\$SPECIFIC (the node-specific part of the system), and SYS\$COMMON (the common part of the system shared by all nodes).

Due to the way in which RSX-11 FCS works with devices and file names FCS cannot handle search lists in a compatible and transparent manner. VAX-11 RSX code has been implemented to reject (as a "bad filename") any compatibility mode task reference to a search-listed logical name.

**Note:** Tasks that use RMS-11 calls (which VAX-11 RSX translates directly to VAX RMS calls) can properly use search lists.

As a result, compatibility mode tasks that use FCS do not function correctly on a system configured with a common system disk. This causes problems for the VAX-11 RSX installation and IVP procedure. However, the following steps offer a solution:

- 1 You should redefine SYS\$SCRATCH (WK:) so that it refers to a search-listed logical name (for example, redefine it to SYS\$COMMON:[SYSMGR]). This allows the VAX-11 RSX installation and IVP procedure to complete successfully.
- 2 You should redefine SYS\$DISK (SY:) so that it refers to a search-listed logical name. Note that VAX-11 RSX treats the RSX-11 pseudodevice SY as a synonym for SYS\$DISK. You must use the SET DEFAULT command (or an equivalent command) to redefine SYS\$DISK.

## **.2.2 Accessing Cluster Devices From Compatibility Mode Tasks**

The device names recognized by RSX-11 and all RSX-11 compatibility mode tasks running under VAX-11 RSX, have two components:

- 1 A two-character ASCII device name
- 2 A unit number that may range from 0 to 377 (8)

However, these components inadequately describe VAX/VMS Version 4.0 devices, particularly devices in a VAXcluster environment. A VAX/VMS device name consists of the following components:

- 1 A device name that includes as many as 15 characters
- 2 A unit number that ranges from 0 to 65535

In previous versions, the AME provided an internal device name translation from VAX/VMS format to the equivalent RSX-11 format. This translation assumed that the VAX/VMS device name consisted of the following:

- 1 A two-character ASCII device name that was the same as an RSX-11 name
- 2 A single alphabetic character controller letter that followed the device name
- 3 A unit number that ranged from 0 to 15 (10) and followed the controller letter

This assumption was usually valid on VAX/VMS versions prior to Version 4.0. It was, however, incorrect for devices such as mailboxes and remote terminals that had unit numbers greater than 15 (10). However, in Versions 3.0 and 4.0 of VAX/VMS, many changes were made that invalidate these assumptions.

To support the VAX/VMS concept of a rooted directory for FCS tasks properly, VAX-11 RSX must treat the device and rooted directory as a virtual device. The results are as follows:

- The device and rooted directory cannot be separate.
- The device name cannot be represented by a two-character ASCII name and byte-bounded unit number.
- Disk devices on UDA-50 and HSC-50 controllers (that may have unit numbers greater than 15 (10)) cannot be handled.
- Shared VAXcluster devices with device names that exceed two ASCII characters (because the ASCII portion of the name includes the VAXcluster node name or the allocation class) cannot be handled.

The previous compatibility mode algorithm for device name mapping still exists and, for VAX/VMS systems that are not VAXcluster members, is usually sufficient. However, in any case where the device cannot be represented in an RSX-11 compatible format, VAX-11 RSX Version 1.0 utilizes new logical names. Some of these logical names are provided automatically by the VAX-11 RSX AME or by the start-up command file (VAX11RSX.COM).

An RSX-11 task that uses FCS cannot access files (without making use of the new logical names) if any of the following conditions are true:

## Notes to VAX-11 RSX Version 1.0

- If a logical name that includes a root directory name (for example, SYS\$SYSROOT) is used to specify the file
- If the device specified is a shareable device on a VAXcluster and contains the VAXcluster node name (for example, FOO\$DRA1)
- If the device has a unit number greater than 15 (10) or if the RSX-11 equivalent unit number exceeds 377 (8)

Refer to 2.7 in the VAX-11 RSX Compatibility Mode Reference Manual for a complete discussion on device name mapping. Specific information on \$\$n logical names is provided in 2.7.2.3. The documentation provides information for devices that VAX-11 RSX does not automatically support

### 3.2.3 Enhancements to VAX-11 RSX Version 1.0

The following sections describe the enhancements and modifications for VAX-11 RSX Version 1.0.

#### 3.2.3.1 Terminal Driver

New features of the terminal driver under VAX-11 RSX are as follows:

- The TF.GTC function indicates that the terminal driver is full-duplex.
- VAX-11 RSX has added terminal driver support for more RSX-11M Version 4.1 terminal types and characteristics, including characteristics corresponding to the VAX/VMS characteristics DECCRT, ANSICRT, EDITMODE, and REGIS.
- When a compatibility mode task performs an IO.ATA function to a terminal, VAX-11 RSX uses a VAX/VMS "out-of-band character" rather than the normal CTRL/C handler mechanism. The CTRL/C character (^C) is not echoed now when the terminal has been attached for CTRL/C and a CTRL/C has been typed.

To coordinate the differences in terminal timeout operations between RSX-11 and VAX/VMS, VAX-11 RSX multiplies the specified RSX-11 timeout parameter by 10 (10) to equal the VAX/VMS timeout parameter. Also, VAX-11 RSX does not specify a timed function to VAX/VMS unless the TF.TMO function modifier is present. The result is that the use of the timeout parameter is now identical for VAX-11 RSX and RSX-11.

To execute terminal input functions terminated either by the escape character or by an escape sequence in a manner compatible with RSX-11, the VAX-11 RSX terminal function translator executes escape terminators, as follows:

- If the terminal generates escape sequences, the terminator character will be the ASCII escape character with the high bit of the byte set to 233 (8). The escape sequence will be included in the input length for use by the task.
- If the terminal does not generate escape sequences (VAX/VMS terminal characteristic TT\$V\_ESCAPE and RSX-11 terminal characteristic TC.ESQ are cleared), the terminator character returned in the IOSB will be the ASCII character escape 33 (8). The input length returned in the IOSB will not include the escape character.

---

### 3.2.3.2 Device-Independent Device Characteristics

Previously, the GLUN\$ directive returned device-independent characteristic bits that were VAX/VMS characteristics instead of RSX-11 characteristics. VAX-11 RSX now returns device-independent characteristic bits that are compatible with RSX-11. Because some of the RSX-11 device-independent characteristic bits have no meaning under VAX/VMS, some of these bits return constant values. For example, DV.EXT (22-bit extended UNIBUS addressing) is always 0. The new form of the device-independent characteristics word is documented in the *VAX-11 RSX Compatibility Mode Reference Manual* under the description of the GLUN\$ directive.

Instead of returning a device characteristic that was valid when a LUN was assigned to the device, the GLUN\$ directive dynamically determines the device's characteristics. This ensures that the characteristics changed since the LUN was assigned are returned to the calling program.

---

### 3.2.3.3 Sticky Device and Directory Names on VAX-11 RSX Command Lines

VAX-11 RSX utilities parse command lines using the RSX SYSLIB routines CSI\$1 and CSI\$2. This provides for the stickiness of file specification fields. However, this code does not understand the concept of logical names. Because of this, undesired behavior can result from including both device and directory within a logical name.

The Command String Interpreter (CSI) treats a logical name as a device name and is not aware that a directory name may be contained in the logical name. Therefore, if the next file name in the list contains only a directory name, the logical name applies to the second file specification only. This results in a file specification with two directory names and makes the utility unable to open the file. For example:

```
$ DEFINE ABC DRA1:[TEST]
$ MCR MAC X,X/-SP=ABC:X,[DAVEB]Y
```

CSI treats the occurrence of "abc" as a device name and applies it as a "sticky default" to the file specification [DAVEB]Y. Once VAX RMS has expanded the logical name, the final file specification becomes DRA1:[TEST][DAVEB]Y.MAC. This file specification is illegal and results in an error when MAC attempts to open it.

A similar problem results if the ABC:X and [DAVEB]Y are reversed: CSI applies the sticky default [DAVEB] to the file specification ABC:X, creating the illegal file specification DRA1:[TEST][DAVEB]X.MAC. The solution is to specify the command as follows:

```
$ MCR MAC X,X/-SP=ABC:X,DRA1:[DAVEB]Y
```

This prevents CSI from applying incorrect sticky defaults to the second file specification.

---

### 3.2.3.4 SPWN\$ Directive

The SPWN\$ directive has been modified as follows:

- Previously, the AME SPWN\$ directive spawned a CLI task when the directive Device Parameter Block (DPB) specified a command line. The spawned subprocess ran the user's default CLI.
- Now VAX-11 RSX accepts a CLI name as a task to be spawned. The CLI task name forms normally accepted by RSX-11 are accepted by VAX-11 RSX. For more information, see the *VAX-11 RSX Compatibility Mode Reference Manual*.

Now, when a task being spawned does not have a VAX/VMS process name that is unique within the current UIC group, VAX-11 RSX now modifies the process name by appending the string ".n" (where "n" represents a decimal number). The decimal number is incremented until the resulting process name is unique. For example, if two offspring tasks are both spawned running the MCR CLI, their names will be "MCR" and "MCR.1."

- Previously, the SPWN\$ directive propagated the parent task's TI to the offspring if the parameter UNUM was 0. This resulted in a task unable to specify explicitly that a TI of TT0: was to be used by the offspring task. Also, when TI was propagated, the parent task's SYS\$INPUT was propagated to both SYS\$INPUT and SYS\$OUTPUT of the offspring, causing errors unless SYS\$INPUT was a terminal.
- Now, to propagate the parent task's TI to the offspring, both UNUM and DNAM must be either null (0) or specified as TI0:. The parent task's SYS\$INPUT and SYS\$OUTPUT are both propagated to the offspring task.

---

### 3.2.3.5 Send/Receive Directives

Previously, only tasks that had an RSX-11 Task Builder (TKB) assigned task name could use the send and receive data directives. Now, all VAX-11 RSX tasks have access to these directives, although any task that was not given a task name by TKB or that was given a multiuser task name will have the VAX/VMS process name for a "task name".

VAX-11 RSX's handling of the receive data mailbox has changed:

- Instead of waiting for messages to be buffered in VAX-11 RSX dynamic memory after a Receive Data directive is executed, VAX-11 RSX now reads messages immediately upon receipt. Receive Data AST routines no longer need to read the data immediately and that the task-name argument that receives data directives can be specified. (All these changes improve the send and receive compatibility with RSX-11.)
- The format of the message sent to the Receive Data mailbox of a compatibility mode task has changed:
  - 1 The format includes a type-code field that designates the message as a Send Data rather than some other AME control message.
  - 2 The format includes the Radix-50 task name of the sender task, as well as the actual data supplied by the sender.

The new format of the Send Data message is described in the *VAX-11 RSX Compatibility Mode Reference Manual*.

- Any native mode image that communicates to a compatibility mode task using the Send Data mailbox interface should use the new message format to eliminate problems caused when the sending task image exits before the message is actually read. However, for compatibility, VAX-11 RSX continues to support the old message format.

---

**3.2.3.6 Longword UIC Support**

When VAX-11 RSX receives from native mode (the process default UIC) a UIC that has a group or member number (or both) greater than 377 (8), it treats the UIC as if it had a value of [377,377].

VAX/VMS UIC codes now have a maximum value of [37777,177777]. This is incompatible with VAX-11 RSX tasks. Therefore, the UIC stored in the task header and returned by the directive GTSK\$ may not actually be the correct process UIC (if the process UIC is not legal for RSX-11). The handling of longword UIC values is described in Section 2 of the *VAX-11 RSX Compatibility Mode Reference Manual*.

---

**3.2.3.7 Cluster Device Names and Rooted Directory "Devices" Support**

VAX-11 RSX can accept cluster device names and rooted directories by using a new logical name format (\$\$n) and a modified device mapping algorithm. VAX-11 RSX returns RSX-11-type logical names instead of VAX/VMS physical device names if the device is SY, LB, SP, TI, CO, CL, or OV; or if the device corresponds to one of the \$\$n (\$\$0 through \$\$377) logical names defined by you or by the system. This allows you to shield VAX-11 RSX tasks from cluster device names and rooted directories. For more information, see Section 2 of the *VAX-11 RSX Compatibility Mode Reference Manual*.

---

**3.2.3.8 Reading from TI: in Spawned MCR Command**

When the Indirect Command Processor encounters an MCR command in an indirect command file, it executes the command by spawning the command (using the SPWN\$ directive) as a command line to the task "CLI..." To execute this, VAX-11 RSX forces SYS\$INPUT for the subprocess to a mailbox. This means that any input from the spawned task originates from the mailbox and not from the parent process's subprocess with a CLITI: device.

VAX-11 RSX has a special mechanism that allows the parent process to read from its TI: and then pass the data to the subprocess by using the subprocess' SYS\$INPUT mailbox. However, this read operation does not allow terminal device-dependent I/O functions except for IO.RVB or IO.RPR (with no function modifiers attached). Therefore, a task such as PIP will function under RSX-11 as if invoked by the command "PIP" in an indirect command file. This is also true for MAC, TKB, and other tasks that do simple line-oriented read operations.

Any task that attempts to use other I/O functions or subfunctions will not operate properly from an indirect command file. Also, native mode images cannot be used in prompting mode from an indirect command procedure.



### 3.2.3.9 Changes to VAX-11 RSX Task Names (for Multiuser or Unnamed Tasks)

When the VAX-11 RSX AME activated a VAX-11 RSX task, the AME read the task's header label. The header label contained the value specified for the TKB "TASK" option. If this field contained a task name that did not begin with the characters "...", the AME translated the Radix-50 string to ASCII and set the VAX/VMS process name to this ASCII value. The Receive Data mailbox was then created with the GROUP logical name "RCVDtask," where "task" was the first six characters of the VAX/VMS process name. The Radix-50 task name (to which task communication directives could refer) consisted of whatever initial characters in the ASCII process name were converted to Radix-50.

The conversion algorithm resulted in a meaningless correlation between the Radix-50 task name (used by other tasks), the actual VAX/VMS process name (used by other copies of the VAX-11 RSX AME to emulate intertask communication directives), and the Receive Data mailbox name (which other copies of the VAX-11 RSX AME use to emulate the Send Data directives).

VAX-11 RSX now processes the VAX/VMS process name to form an ASCII string that is not longer than six characters and that can be converted intact to a Radix-50 encoded task name. The conversion is accomplished by the following steps:

- 1 Converting lowercase alphabetic characters to uppercase
- 2 Removing all characters not in the Radix-50 character set (including removal of the spaced character, which is a Radix-50 character)
- 3 Truncating the result to six characters if it is longer

In addition, for batch jobs with the default process name assigned by the VAX/VMS Job Controller (of the form BATCH\_n, where "n" is the job sequence number), the AME performs different processing to ensure a unique task name. The initial task name consists of "B" followed by the last (low order) five digits of the job sequence number (for example, BATCH\_123 becomes B123, while BATCH\_1234567 becomes B34567).

If the string resulting from this algorithm is not a unique VAX/VMS process name the following will occur:

- 1 VAX-11 RSX will replace the final character of the taskname with the ASCII character 0 (zero).
- 2 If the resulting name is not unique, VAX-11 RSX will increment the final character until the name is unique or the value 9 (nine) is reached.
- 3 If the value 9 (nine) is reached and no unique name has been found, VAX-11 RSX will abort the task with an error.

**Note:** VAX-11 RSX does not support the use of a null (empty) process name, or one composed of blanks (for example, " ").

As with previous versions of compatibility mode, if the task was built with a nonmultiuser task name, the TKB-assigned task name will supercede the results of the previous algorithm. The final task name is used consistently as follows:

- The VAX/VMS process name is set to the task name while the task is executing.
- The name of the Receive Data mailbox is "RCVDtask".

- The task name is accurately represented by a six-character Radix-50-encoded string, thus allowing other tasks within the same UIC group to use the Receive Data and task control directives.

**WARNING: Digital intends to improve the RSX-11 compatibility of VAX-11 RSX task names in a future release. The algorithms used for the Send Data and Receive Data directives and other intertask communication directives may change radically as a result. Digital will attempt to retain as much compatibility with previous releases of the AME as possible. However, since the primary objective is to improve RSX-11 compatibility, compatibility with previous releases of the AME may be affected in future releases of VAX-11 RSX.**

### 3.2.3.10 Changes Required to User-Created RSXUSR.EXE Images

The VAX-11 RSX AME consists of four images: RSXSHR.EXE, RSXUSR.EXE, RSX.EXE, and BACKTRANS.EXE. The shareable images RSXSHR.EXE and RSXUSR.EXE constitute the actual AME, and the executable images RSX.EXE and BACKTRANS.EXE provide shell functions that are invoked by the current CLI when VAX-11 RSX tasks are run.

For VAX-11 RSX Version 1.0, the change in the structure of the AME requires changes to the RSXUSR.EXE shareable image. If you created your own RSXUSR.EXE image, you must modify it before attempting to use it with VAX-11 RSX Version 1.0. Changes affecting the RSXUSR.EXE are as follows:

- RSXUSR.EXE supports user-defined RSX-11 directives; when the AME encounters an RSX DIC code that it does not support, it dispatches control to RSXUSR through the transfer vector RSX\$UNKNOWN, to allow emulation of the directive user-supplied code. The RSXUSR.EXE supplied with VAX-11 RSX supplies the directive error status IE.SDP (DIC or DPB size invalid). You can substitute a user-written RSXUSR.EXE for the supplied image by defining the logical name RSXUSR to the full file name of the desired replacement or by replacing the image SYS\$SHARE:RSXUSR.EXE.
- Prior to VAX-11 RSX Version 1.0, all AME images were linked with base addresses. The base address for the RSXUSR.EXE shareable image was 21000 (16). This placed size constraints on the RSXSHR.EXE shareable image since it fit beneath RSXUSR.EXE. With the increased functionality, the location of the RSXUSR.EXE image had to be modified. For Version 1.0 of VAX-11 RSX, RSXSHR.EXE and RSXUSR.EXE were converted to unbased shareable images. This means that they will be placed in the process virtual address space at image activation time (by VAX/VMS), regardless of how large either image may grow in the future.
- To use your own RSXUSR.EXE with the VAX-11 RSX AME, you must modify the source code and the data to be Position Independent Code (PIC). You must then link the object files as an unbased shareable image (by removing the "BASE=n" statement in the option file for the DCL command LINK).
- When converting your source code and data to PIC, ensure that your RSXUSR.EXE adheres to the following rules:
  - 1 Any data references to addresses must use the MACRO-32 .ADDRESS directive rather than .LONG. Address references using .LONG cannot be modified at run time by the image activator and, therefore, can prevent the shareable image from being relocated

## Notes to VAX-11 RSX Version 1.0

properly. For example, your RSXUSR.EXE may contain code like the following:

```
XYZZY:
.
.
.
FOO: .LONG    XYZZY
```

Change the code to:

```
XYZZY:
.
.
.
FOO: .ADDRESS XYZZY
```

- 2 Any code references to addresses outside of the RSXUSR.EXE shareable image must use the G<sup>^</sup> addressing mode. This allows the image activator to modify the target address at run time. References to addresses within the RSXUSR.EXE image must use either G<sup>^</sup> (general) or relative (L<sup>^</sup>, W<sup>^</sup>, or B<sup>^</sup>) addressing mode. Relative addressing mode within a shareable image is PIC and works correctly wherever the image is located. General addressing mode allows the image activator to modify the target addresses at runtime.

In particular, be careful to ensure that the RSXUSR.EXE transfer vector uses relative addressing for the Jump To The DIC Dispatch routine.

Assuming that CBERRY was an AME routine defined as a universal symbol by RSXSHR.EXE, an RSXUSR.EXE that called CBERRY might look like this:

```
.TRANSFER    RSX$UNKNOWN
JMP          L^RSX$UNKNOWN
.
.
.
RSX$UNKNOWN::                ; Dispatch user DIC codes
.
.
.
JSB          G^CBERRY        ; Call CBERRY routine
.
.
.
```

- 3 Do not use labels in RSXUSR.EXE as displacements because the image activator will not be able to relocate the references properly. Use of labels is common when an offset into a static data structure must be calculated. However, the offset could be calculated in a register, so the data can be referenced using the data structure label as a displacement to the register. Change such code to use index mode rather than displacement mode. For example:

```
MULL2 #4, R3                ; R3 contains structure offset
MOVAB TABLE(R3), R4        ; Data element is 4 bytes
                                ; Get address of element
```

Change the code to:

```
MOVAL L^TABLE[R3], R4 ; Get address of data element
```

Note that in this case, using index mode also allowed the computation to be compressed into a single instruction. If the data element size had been 3 bytes, the replacement code would have been as follows:

```
MULL2 #3, R3 ; Calculate structure offset
MOVAB L^TABLE[R3], R4 ; Get address of data element
```

- 4 Do not use immediate mode in referencing labels. For example,

```
MOVL #TABLE, R0 ; Get address of table
```

should be replaced by

```
MOVAB L^TABLE, R0 ; Get address of table
```

For reference, a simplified link command for the VAX-11 RSX RSXUSR image is:

```
$ LINK/SHARE=RSXUSR/MAP RSX/INCLUDE=RSXUNKNOWN,SYS$INPUT/OPTIONS
GSMATCH=LEQUAL,3,0 ! MATCH CONTROL, VERSION 3
```

**Note:** For more detailed information, refer to the *VAX/VMS Linker Reference Manual*.

### 3.2.3.11 Default File Specifications for Common and Library Task Images

If you are using an RSX-11 task image that was linked to a common or library using the COMMON, LIBR, RESCOM, or RESLIB options to the Task Builder, that common or library image is assumed by default to reside in SYS\$LIBRARY: and to have a default file extension of .EXE. Previously, you could not override that default.

Under VAX-11 RSX Version 1.0, you can override the default by defining a logical name for the library name that translates to the file specification of the common or library image. For example, if you have a program that links to

```
DISK$WORK: [MYDIR.LIBRARY]LIBRARY.EXE
```

and the task-built name of the library is MYLIB, you may define the logical name MYLIB to translate to

```
DISK$WORK: [MYDIR.LIBRARY]LIBRARY.EXE
```

This will allow your program to work correctly.

This feature is more useful for the RESCOM and RESLIB options in TKB since these options allow a full file specification of the common or library image. The COMMON and LIBR options already assume that the library is in LB:[1,1], which is equivalent to SYS\$LIBRARY:. There is generally no reason to override this default with the logical name feature previously described. However, if you define such a logical name, it will always take precedence.

### 3.2.3.12 File Transfer Program (FLX)

When using the File Transfer Program (FLX) with a TU78 tape drive, the default density is set to 6250 bpi. To create a tape at 1600 bpi, you must specify the /DNS:1600 switch when you create the tape with the /ZE switch. The documentation describing FLX in the *RSX-11M/M-PLUS Utilities Manual* erroneously indicates that the default density is 800 bpi.

### 3.2.4 Restrictions to MCR for VAX-11 RSX Version 1.0

In VAX-11 RSX Version 1.0, the Indirect Command Processor is a slightly modified version of the RSX-11M-PLUS Version 2.1 Indirect Command Processor.

For Version 1.0, Indirect spawns a command line to MCR, that command line is executed by a subprocess with the CLI MCR. Command exit status and process-context information changed by the command are propagated to the main process when the command completes in the subprocess. The following sections describe the restrictions imposed by this implementation.

#### 3.2.4.1 Logical Name Handling

- If you redefine SYS\$OUTPUT in a command procedure, the new definition remains in effect only through the execution of the command procedure. When the command procedure exits, SYS\$OUTPUT reverts to the original definition specified for your process.
- If you create process-local logical name tables in a command procedure, the tables are created in the MCR subprocess but do not exist in your main process when the command procedure exits.
- If, during a command procedure, you create logical names in a process-local table that created, for example, a process-local table other than LNM\$PROCESS, the logical names are not available after the command procedure completes. However, if you create logical names in the LNM\$PROCESS table or any table that is not process-local, the logical names are available after the command procedure exits.

#### 3.2.4.2 Device Allocation

If you allocate a device during a command procedure and do not mount or deallocate the device during the course of the procedure, the device is automatically deallocated when the command procedure completes.

#### 3.2.4.3 Process Context

- Because MCR commands in command procedures are actually executed in a subprocess, the output of a SHOW PROCESS command in a command procedure gives information on the subprocess. The process name, process identification number (pid), and the deductible quotas appear different from those of the main process in which Indirect is running. The remaining information looks the same.
- If you use the SET TERMINAL command in a command procedure, you must explicitly specify the terminal (even for your own terminal).
- In a command procedure, the SET PROCESS/NAME command changes only the name of the subprocess being used to execute the MCR commands within the command procedure. When the command procedure exits, your main process will have the same process name it had when you invoked the command procedure.

- If you issue the following VAX/VMS-specific MCR commands from an indirect command procedure, changes in context are lost when the command procedure exits:

```
SET COMMAND          SET MESSAGE
SET RMS_DEFAULT     SET WORKING_SET
```

All MCR commands issued in an indirect command procedure are executed within a subprocess. Most of the commands that alter process-context information propagate that information to the main process to make it available after the subprocess exits. The four commands shown do not do this.

---

#### 3.2.4.4 Old Indirect Commands

- Because VAX-11 RSX now supports indirect command processing through the RSX-11 Indirect Command Processor, Indirect commands are not imbedded in MCR. To issue Indirect commands interactively, you must type "@TI:" to the MCR prompt. Indirect will return its prompt (IND> ) and allow you to issue both Indirect and MCR commands.
- Because DCL lexical functions cannot interact with the Indirect Command Processor, the following functions were removed from MCR for VAX-11 RSX Version 1.0:

```
F$DIRECTORY    F$LOGICAL      F$TIME
F$EXTRACT      F$MESSAGE    F$USER
F$LENGTH      F$MODE       F$VERIFY
F$LOCATE      F$PROCESS
```

---

#### 3.2.4.5 Differences Between VAX-11 RSX Indirect (ICM) and RSX-11M-PLUS Indirect (ICM)

The VAX-11 RSX Indirect Command Processor (Indirect) does not support the .FORM, .XQT, and .WAIT directives included in the RSX-11M-PLUS Version 2.1 Indirect Command Processor (Indirect). The .FORM directive is not included because the FMS/RSX-11 driver is not supported under VAX-11 RSX. Complexities in correctly handling the asynchronous .XQT and .WAIT directives prevent their implementation in VAX-11 RSX Indirect for Version 1.0.

A few Indirect directives have little meaning in the VAX/VMS environment and, as a result, they have been set to standard values which allow most command files to operate as expected. These directives include: .IFACT/.IFNACT, .IFINS/.IFNINS, .IFLOA/.IFNLQA, and .TESTPARTITION. The .TESTDEVICE directive returns the device-independent characteristics in the RSX-11 format, but the device-dependent characteristics it returns in words 2 and 3 of the device characteristics words are a VAX/VMS device-dependent longword.

A few of the special symbols are also slightly different due to the constraints and differences in the VAX/VMS environment. The symbols <BASLIN>, <LOCAL>, <PRIVIL>, <FILER2>, <MEMSIZ>, <SYSTEM>, <SYUNIT>, <ACCOUN>, <CLI>, <FMASK>, <SYSDEV>, and <SYDISK>, which can have varying values for RSX-11M-PLUS, have constant values for VAX-11 RSX.

Refer to Section 5 of the *VAX-11 RSX Compatibility Mode Reference Manual* for more details on indirect command file processing.

### **3.2.4.6 Invoking the RMS DEF Utility and the MCR DEFINE Command**

Each MCR (and DCL) command is unique to the first four characters of the command. However, many commands can be uniquely recognized in less than the first four characters. The DEFINE command (new to MCR for VAX-11 RSX Version 1.0) can be uniquely recognized under DCL by the first three characters (DEF). However, under MCR, you must specify the first four characters (DEFI) to distinguish the DEFINE command from the RMS-11 DEF utility. Under MCR, typing the following:

```
>DEF command-line
```

invokes the RMS-11 file definition utility, but typing any of the following:

```
>DEFI command-line
```

```
>DEFIN command-line
```

```
>DEFINE command-line
```

invokes the DEFINE command to define a logical name.

## **3.2.5 RMS-11 V2.0 Under VAX-11 RSX**

Previously, RMS-11 Version 1.8 issued RSX-11 ACP QIOs that VAX-11 RSX translated into the equivalent VAX/VMS QIOs. Now, RMS-11 Version 2.0 (used under VAX-11 RSX) issues a special directive that performs the following:

- 1 Translates the RMS-11 user structures into equivalent VAX RMS structures
- 2 Issues the corresponding VAX RMS QIO function
- 3 Translates the appropriate fields in the VAX RMS user structures back to the RMS-11 structures

## **3.2.6 RSX-11S System Generation For VAX/VMS**

Following are modifications to the preparation steps for an RSX-11S system generation under VAX/VMS:

- You must mount the target disk prior to running the RSX-11S system generation procedure.
- Your kit disk, labeled 11SKIT1, must not be allocated or mounted when you invoke the RSX-11S system generation procedure.

## 3.2.7 DCL Restrictions

The following sections describe restrictions to DCL for VAX-11 RSX.

### 3.2.7.1 Changing UIC

Under VAX/VMS Version 3.0, when you changed your UIC group with either the DCL command SET UIC or the MCR command SET /UIC, you automatically gained access to the group logical name table of your new UIC group. This allowed you to create a group logical name in the new group logical name table, provided you had the GRPNAM privilege. For VAX/VMS Version 4.0, logical name handling has been altered. Now, the process logical name LNM\$GROUP points to the group logical name table that will be accessed by your process.

The behavior of the SET UIC command in DCL and the SET /UIC command in MCR is different as follows:

- In DCL, the command changes the process UIC but leaves the LNM\$GROUP logical name unchanged. This means that a user who executes the SET UIC command to change his or her group number does not automatically receive the new group's logical names. In addition, logical name protection prevents a nonprivileged user from accessing his or her old logical name table. This is different behavior from Version 3.0.
- In MCR, the command changes the process UIC and also redefines LNM\$GROUP to reflect the new group number. A user who executes the SET /UIC command to change his or her group number gains access to the group logical name table of the new group. MCR will also create the new group logical name table for you if it does not already exist. MCR does this by creating a temporary detached process with the new UIC. The detached process will not run an image or a CLI and will be run down immediately after its creation. The redefinition of LNM\$GROUP preserves Version 3.0 behavior.
- If you have the GRPNAM privilege, you will be able to define group logical names for your new UIC group.

In general, a user operating under the control of either CLI can obtain whichever behavior he or she desires by choosing an appropriate definition for LNM\$GROUP. The area likely to be affected by this change is compatibility mode under DCL.

The VAX-11 RSX AME must create group logical names on start-up. If you change your UIC under DCL, you will not be able to run any VAX-11 RSX tasks without changing your group logical name table. If you need to change your UIC group in DCL prior to running a VAX-11 RSX task, use the VAX-11 RSX Indirect Command Processor (Indirect) to change the UIC. Indirect will spawn the SET /UIC command to MCR and the LNM\$GROUP group logical name table pointer will be changed. To do this, type the following:

```
$ MCR ICM @TI:
IND>SET /UIC=[group,member]
IND>[CTRL/Z]
```



### 3.2.7.2 Running VAX-11 RSX Tasks from DCL

When VAX-11 RSX tasks issue the GMCR\$ directive to the AME to obtain the command line issued for that task, the AME requests the command line from the CLI.

Under MCR, the AME receives what was entered (after Indirect resolves symbols, if the command is part of an indirect command file) with multiple blanks or tabs stripped to a single blank or tab. This is exactly compatible with RSX-11.

However, under DCL, blanks that occur before delimiters that are significant to DCL are also stripped. Significant DCL delimiters include the following:

- Slash (/)
- Comma (,)
- Colon (:)
- Plus sign (+)
- Dash (-)
- Left and right square brackets ([ ])
- Left and right angle brackets (< >)
- Question mark (?)
- Equal sign (=)
- Circumflex (^)
- Exclamation point (!)
- At sign (@)
- Asterisk (\*)
- Single quote (')
- Semi-colon (;)
- Period (.)

The AME receives from DCL (and must pass on to the compatibility mode task) a command line that may have fewer blank delimiters than the VAX-11 RSX task expects. For instance, the following DCL command line issued to the Macro-11 Assembler,

```
$ MCR MAC ,A.LIS=A.MAC
```

is passed by DCL to the AME and then to Macro-11 as

```
MAC,A.LIS=A.MAC
```

Macro-11 cannot correctly parse this input.

To work around this restriction when running VAX-11 RSX tasks from DCL, specify null arguments as NL:. For example:

```
$ MCR MAC NL: ,A.LIS=A.MAC
```

Under the MCR CLI, the example command

```
> MAC ,A.LIS=A.MAC
```

(and all other task commands) work as expected.

---

### 3.2.7.3 Entering MCR Commands in DCL

When the MCR command is entered in DCL, the AME runs in a special prompting mode. Each line entered to the prompt "MCR>" is taken as the name of a VAX-11 RSX task on SYS\$SYSTEM:, followed by a command line to be passed to that task (for example, PIP /LI).

Previously, when the MCR command was used in this mode from a command procedure or in a batch job (as opposed to entering it from a terminal), the AME would type the input line to the output file, preceded by the string "MCR>" (for example, MCR> PIP /LI). Now that VAX/VMS includes a built-in image input verification (SET VERIFY=IMAGE), this code has been removed from the AME. This means that the "MCR>" string will no longer be typed at the beginning of the echoed input line.

The DCL command MCR does not provide access to the MCR CLI; it only allows you to enter command lines for VAX-11 RSX tasks.

---

### 3.2.7.4 Restriction to Invoking RSX-11 Utilities from DCL Commands

Section 6 of the *VAX-11 RSX Compatibility Mode Reference Manual* describes the DCL commands that you can use to invoke various RSX-11 utilities. If, when using these commands, you use concatenated input file specifications, place all of the general command keywords at the beginning of the command after the command verb (rather than after the list of concatenated input files). For example, use the command:

```
$MACRO/RSX11/LIST/OBJECT A+B
```

to produce A.OBJ and A.LST, which contain the binary object and listing files, for the combined assembly of A.MAC and B.MAC.

---

## 3.2.8 Restrictions to Installing Compatibility Mode Tasks

When a VAX-11 RSX task contains a read-only shareable region (for example, Indirect or a resident library), you can install the image (using the VAX/VMS Install Utility) as shared. Install creates shared global sections for the shareable regions, and these sections can be mapped by any number of processes running the image, minimizing the need for physical memory (all the processes will share the same physical memory).

Install creates the shared global section with a name that consists of the image file name (for example, ICM for SYS\$SYSTEM:ICM.EXE) and is concatenated with a sequence number.

However, when VAX-11 RSX attempts to map a global section for a compatibility mode task image, the name that it uses is formed from the task name (from the Task Builder option TASK) the task was built with (the leading "..." sequence is removed, if present), rather than from the image name. If you want to install a compatibility mode task image as shareable, you must build it with a task name that is the same as the image file name.

---

### 3.2.9 TMPMBX Requirement

The privilege, TMPMBX, is needed for many different operations performed by VAX-11 RSX. Specifically, you need the TMPMBX privilege to:

- Run any VAX-11 RSX task
- Run any indirect command file under MCR
- Use the MCR command SPAWN.

---

### 3.2.10 BYTLM Requirement

Each process that runs VAX-11 RSX components should have a BYTLM quota greater than  $1804 + (\text{DEFMBXBUFQUO} * 2) + \text{DEFMBXMXMSG}$ , which is approximately 5000 bytes. This amount is needed for VAX-11 RSX components only. The BYTLM quota should be set to a value higher than this to accommodate other system requirements. DIGITAL recommends a BYTLM quota of no less than 8192 for VAX-11 RSX users.

---

### 3.2.11 Requirement for Tasks that Need to Receive Escape Sequences

Before running any compatibility mode task image or native-mode image that needs to receive escape sequences as input, turn off terminal line editing as shown in the following MCR and DCL examples:

MCR:

```
>SET TERMINAL/NO_LINE_EDITING
```

DCL:

```
$$SET TERMINAL/NO_LINE_EDITING
```

The VAX-11 RSX editor, EDI, is an example of a task that needs to receive escape sequences as input.

---

### 3.2.12 Restrictions to the Disk Save and Compress Utility (DSC)

Under RSX-11, DSC requires that you specify a colon (:) at the end of each device name. This is true under VAX-11 RSX, even when you are using logical names that contain colons in the translation string. The following command example shows how to specify devices to DSC on a VAXcluster system:

```
DSC>$$1:=$$3:
```

where

```
$$1 translates to "MYVAX$DMAO:"
```

```
$$3 translates to "MYVAX$DMA1:"
```

On a system that is not a VAXcluster member, the following command

```
DSC>DMO:=DM1:
```

shows a second way to specify devices to DSC.

## Notes to VAX-11 RSX Version 1.0

You must also mount as foreign the devices that you plan to use with DSC (both input and output devices). Under RSX-11M-PLUS, if you do not mount the devices foreign, you receive a privilege violation. For VAX-11 RSX, if you do not mount the devices foreign, DSC returns a syntax error (due to slightly different parsing and device handling by DSC).



---

## READER'S COMMENTS

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

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

---

---

---

---

---

---

---

---

---

---

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

---

---

---

---

---

---

---

---

---

---

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

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

Name \_\_\_\_\_ Date \_\_\_\_\_

Organization \_\_\_\_\_

Street \_\_\_\_\_

City \_\_\_\_\_ State \_\_\_\_\_ Zip Code \_\_\_\_\_  
or Country

Do Not Tear - Fold Here and Tape

**digital**

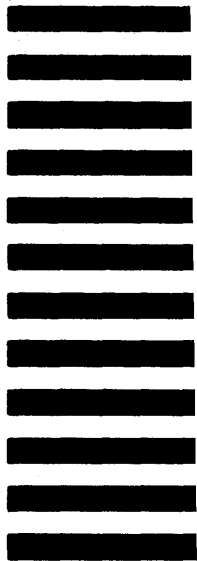


No Postage  
Necessary  
if Mailed in the  
United States

**BUSINESS REPLY MAIL**  
FIRST CLASS PERMIT NO.33 MAYNARD MASS.

POSTAGE WILL BE PAID BY ADDRESSEE

SSG PUBLICATIONS ZK1-3/J35  
DIGITAL EQUIPMENT CORPORATION  
110 SPIT BROOK ROAD  
NASHUA, NEW HAMPSHIRE 03062-2698



Do Not Tear - Fold Here