

RSTS/E Release Notes

Order Number: AA-NB17A-TC

These release notes describe new features of the RSTS/E system and explain the differences between this and previous versions of RSTS/E. System managers and system maintainers should read this document prior to system installation.

Operating System and Version: RSTS/E V9.7

Software Version: RSTS/E V9.7

July 1989

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.

© Digital Equipment Corporation 1989.

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

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

DATATRIEVE	FMS-11	RSX
DEC	LA	RT
DECmail	MASSBUS	UNIBUS
DECmate	PDP	VAX
DECnet	P/OS	VMS
DECsystem-10	Professional	VT
DECsystem-20	Q-Bus	Work Processor
DECtape	Rainbow	WPS-PLUS
DECUS	ReGIS	
DECwriter	RSTS	
DIBOL		

IBM is a registered trademark of International Business Machines Corporation.

Contents

Preface	vii
----------------------	-----

Chapter 1 **New Features**

1.1	New Hardware Support	1-1
1.2	New Software Features	1-1
1.2.1	Electronic Message Reception	1-1
1.2.2	Supervisor Mode RMS Libraries	1-2
1.2.3	DCL	1-2
1.2.3.1	Virtual Disk Commands	1-2
1.2.3.2	Apostrophe Substitution	1-2
1.2.3.3	Work File	1-3
1.2.3.4	Performance Improvements	1-3
1.2.3.5	LINK Command	1-4
1.2.3.6	SET SYSTEM Command	1-4
1.2.4	TKB	1-5
1.2.4.1	Fast Map Overlay Handler Switch	1-5
1.2.4.2	D Space APR Protection When Using Libraries	1-5
1.2.5	New RSX-11/M-PLUS Directives	1-5
1.2.6	Layered Product Features	1-6
1.2.6.1	DECnet/E	1-6
1.2.6.2	PDP-11 Fortran-77 Support	1-6
1.2.6.2.1	Virtual Arrays	1-6
1.2.6.2.2	Changes to Resident Library Task-Build Files	1-6

Chapter 2 **Differences Between RSTS/E V9.7 and V9.6**

2.1	Unsupported Devices Removed	2-1
2.2	Software Changes	2-1
2.2.1	Installation	2-1
2.2.2	EDT Performance Improvement	2-2
2.2.3	Virtual Disk as Part of the Public Structure	2-2
2.2.4	System Command Files	2-3
2.3	Monitor	2-3
2.3.1	Corrections	2-3
2.3.2	Resident Library Handling	2-4
2.3.2.1	Mapping Changes	2-4
2.3.2.2	Running Tasks That Use Libraries	2-4

2.3.3	SYS-Call Changes	2-5
2.3.3.1	UU.MNT Directive	2-5
2.3.3.2	UU.DIE Directive	2-5
2.3.3.3	UU.CFG Directive	2-5
2.3.3.4	UU.RTS Directive	2-5
2.3.3.5	RSX-11/M-Plus Directive	2-5
2.4	INIT.SYS	2-6
2.4.1	Corrections	2-6
2.4.2	Default Option	2-6
2.4.3	INIT Now Recognizes DR11/DRV11	2-6
2.4.4	INIT Checks For Rev 0.0 Disks	2-6
2.4.5	Virtual Disk No Longer Shows as Disabled Device	2-6
2.5	Commonly Used System Programs (CUSPs)	2-7
2.5.1	DCL	2-7
2.5.1.1	Command Processing	2-7
2.5.1.2	LAT Commands	2-7
2.5.1.3	SHOW LOGICAL Command	2-7
2.5.1.4	ASSIGN Command	2-7
2.5.1.5	SHOW DEVICE Command	2-7
2.5.1.6	SHOW SYSTEM Command	2-7
2.5.1.7	SHOW TERMINAL/PERM Command	2-8
2.5.1.8	SHOW QUOTA Command	2-8
2.5.1.9	SET SYSTEM/HOLD Command	2-8
2.5.1.10	SET JOB/PRIVILEGE Command	2-8
2.5.1.11	SET TERMINAL Command	2-8
2.5.1.12	BACKUP/DIRECTORY Command	2-8
2.5.1.13	HELP Command	2-10
2.5.1.14	DISMOUNT Command	2-10
2.5.1.15	INITIALIZE Command	2-10
2.5.1.16	COBOL Command	2-10
2.5.1.17	DELETE/SERVICE/LAT Error Message	2-10
2.5.2	PBS Package	2-10
2.5.3	LOGIN	2-11
2.5.4	SHUTUP	2-11
2.5.5	OPSER Package	2-11
2.5.6	PIP	2-11
2.5.7	DIRECT	2-12
2.5.8	DISPLY	2-12
2.5.9	STATUS	2-12
2.5.10	SYSTAT	2-12
2.5.11	DSKINT	2-12
2.5.12	Error Package	2-13
2.5.13	TKB and STK	2-13
2.5.14	EMTCPY	2-14
2.5.15	ANALYS Package	2-14
2.5.16	Update Package	2-14
2.5.17	Removal of Unsupported Software	2-14
2.6	Basic-Plus Run-Time System	2-15
2.7	RMS-11	2-15
2.8	Layered Products Changes	2-15
2.8.1	DECnet/E	2-15
2.8.1.1	Monitor Corrections	2-15
2.8.1.2	Installation	2-15
2.8.1.3	NCP Help	2-16
2.8.1.4	NCP Circuit Maximum Routers Parameter	2-16

2.8.2	DECmail-11	2-16
2.8.3	WPS-PLUS	2-16
2.8.4	DIBOL	2-17
2.8.5	COBOL-81	2-17

Chapter 3 Documentation Changes

3.1	Changes to the System Directives Manual	3-1
3.1.1	General RSTS/E Environment	3-1
3.1.2	.PLAS Directive	3-1
3.1.3	UU.CFG Directive	3-2
3.1.4	UU.DIE Directive	3-4
3.1.5	UU.MNT Directive	3-4
3.1.6	UU.RTS Directive	3-5
3.1.7	UU.TRM Directive	3-6
3.1.8	CRAW\$ Directive	3-7
3.1.9	FEAT\$ RSX-11 Directive	3-7
3.1.10	FEADF\$ RSX-11 Directive	3-8
3.1.11	GMCR\$ RSX-11 Directive	3-8
3.1.12	SCCA\$ RSX-11 Directive	3-9
3.1.13	TFEA\$ RSX-11 Directive	3-9
3.1.14	SSTX\$ RSX-11 Directive	3-10
3.2	Changes to the RMS-11 User's Guide	3-11
3.3	Changes to the Task Builder Reference Manual	3-12
3.4	Layered Product Documentation Changes	3-34
3.4.1	DECnet/E V4.1 Network Installation and Update Guide	3-34
3.4.2	DECnet/E V4.1 Release Notes	3-38

Chapter 4 Known Problems and Restrictions

4.1	Installation	4-1
4.1.1	Booting Magnetic Tape	4-1
4.1.2	Flag Files	4-1
4.2	Initialization Code (INIT.SYS)	4-1
4.3	Monitor	4-2
4.3.1	DB and DR Disk Drivers	4-2
4.3.2	MSCP Disk Driver	4-2
4.3.3	TMSCP Tape Driver	4-2
4.3.4	Monitor Overlay Handling	4-2
4.4	Commonly Used System Programs (CUSPs)	4-3
4.4.1	DCL	4-3
4.4.1.1	DIRECTORY Command	4-3
4.4.1.2	TIME Command	4-3
4.4.1.3	F\$MID Function	4-3
4.4.1.4	ON ERROR Handling	4-3
4.4.1.5	MOUNT Command	4-3
4.4.1.6	GOSUB Command	4-3
4.4.1.7	BROADCAST Command	4-3
4.4.2	ANALYS Package	4-4

4.4.3	LOGIN and LOGOUT	4-4
4.4.4	ODT	4-4
4.4.5	PIP	4-4
4.4.6	RT-11 Utilities on DU Disks Greater Than Unit 9	4-5
4.4.7	Log File Corruption	4-5
4.5	RSX-11 and RMS Tasks	4-5
4.5.1	RMS-11	4-6
4.6	Layered Products	4-6
4.6.1	DECnet/E	4-6
	4.6.1.1 NFT and FAL	4-7
4.6.2	FORTTRAN-77	4-7

Index

Tables

4-1	RSX-11 and RMS Tasks	4-5
-----	----------------------------	-----

Preface

The *RSTS/E V9.7 Release Notes* describe new features for this release and the differences between this release and RSTS/E V9.6.

The procedure to install a new RSTS/E V9.7 system or to upgrade an existing RSTS/E system to V9.7 follows the same general process developed for V9.0.

Because of the V9.0 changes to the installation process, users of earlier versions of RSTS/E should read the *RSTS/E System Installation and Update Guide* prior to attempting an installation of RSTS/E V9.6.

If you are installing a new RSTS/E V9.7 system, refer to Part 1 of the *RSTS/E System Installation and Update Guide*.

If you are upgrading a pre-V9.0 system to V9.7, refer to Part 2 of the *RSTS/E System Installation and Update Guide*.

If you are updating a RSTS/E V9.0 or later system to V9.7, refer to Part 4 of the *RSTS/E System Installation and Update Guide*.

In addition, users and programmers familiar with RSTS/E V9.6 should read these notes to learn about the changes and new features in RSTS/E V9.7.

The V9.7 Release Notes are divided into four chapters:

- Chapter 1 Describes new features
- Chapter 2 Describes the differences between RSTS/E V9.7 and V9.6
- Chapter 3 Describes documentation changes
- Chapter 4 Describes known problems and restrictions

NOTE

If you last used a version of RSTS/E older than V9.6, you should read the release notes applicable to those versions.

1.1 New Hardware Support

RSTS/E V9.7 has no new hardware support.

1.2 New Software Features

The following sections describe the new software features for RSTS/E V9.7.

1.2.1 Electronic Message Reception

Electronic messaging systems are becoming more common in the telecommunications world. Many of these systems have been available for several years. These systems allow subscribers to send messages to any other subscriber in the network (sometimes even to nonsubscribers). In the past, subscribers generally had dedicated terminals that could send or receive messages. Addressing of messages was handled by a terminal feature called *answerback*. An answerback message was defined for a terminal or group of terminals to receive messages for that address.

RSTS/E V9.7 has the capability to receive these electronic messages. The SET SYSTEM command now has the qualifier /ANSWERBACK="text" for defining a systemwide answerback message. Using the SET TERMINAL KBnn:/ANSWERBACK/PERM command enables reception of electronic messages on a given terminal line. You can also use the terminal for normal logins as well.

When you enable this characteristic for a terminal line and a messaging system calls into the RSTS/E system, the system responds with its answerback message to confirm its address to the caller. RSTS/E then waits for data from the messaging system, and stores the data in a unique file in the EMS\$: account. The file names are made unique based on the current time of day. The files can then be handled as any other file on the system: they can be printed for manual distribution, or an application that scans the file to determine which user on the system should receive this message can be written. The UNSUPP\$: account supplies a demonstration command file (EMS.COM) that forwards received messages to DECmail-11 users.

In this way, you can use a RSTS/E system to replace all of the terminals that previously used a single answerback message to receive messages.

1.2.2 Supervisor Mode RMS Libraries

You can now access RMS-11 Resident libraries using supervisor mode. This feature is available on all CPUs that have the supervisor mode hardware available (11/44/45/50/53/55/70/73/83/84). By mapping the RMS-11 libraries into supervisor mode rather than user mode, the application may be able to gain an additional 8K words of address space. The address space previously used by the RMS-11 libraries may now be available for use by the application itself.

This feature results in the addition of files to the RMS\$: account and the LB: account. The new file LB:RMSLX.ODL is the correct ODL file to use when building against RMS as a supervisor mode library. The file LB:DAPSLX.ODL is provided for building against the combination of supervisor mode RMS library and the remote access (DAPRES) RMS library. The files in RMS\$: (RMSRES.TSK and RMSRES.STB) have the same names as files already in LB:, but their contents are different. Do **not** move these files into LB: because they would replace the existing files that have the same names. You must have the files in LB: to properly build to RMSRES as a user mode library, and you must have the files in RMS\$: to properly build to RMSRES as a supervisor mode library.

1.2.3 DCL

The following sections describe new DCL features.

1.2.3.1 Virtual Disk Commands

You can now create the virtual disk (DV0:) directly from DCL during timesharing operation. Two commands, CREATE/VIRTUAL_DISK and DELETE/VIRTUAL_DISK, have been added to support new features that manage the virtual disk. Digital recommends that you place the commands for creating and initializing the virtual disk in the system startup command file (SY0:[0,1]START.COM) because the virtual disk memory must be contiguous. You should place these commands in the startup command file prior to the installation of run-time systems or resident libraries. The formats for these commands are:

```
$ CREATE/VIRTUAL_DISK size          !Where size is the size in blocks
      [/ADDRESS=n]                  !Where n is the address in Kwords
      /[NO]LOG                       !Requests a confirmation message

$ DELETE/VIRTUAL_DISK                !No parameters
      /[NO]LOG                       !Requests a confirmation message
```

Creation of the virtual disk simply reserves a section of memory for use by the virtual disk. You must still use the INITIALIZE and MOUNT commands, as with any other RSTS/E disk. If an attempt is made to use the device before it is created, a ?Not a valid device error message is issued. You can use the /ADDRESS qualifier to specify the physical location of the memory. If you do not specify this qualifier, RSTS/E selects an appropriate location. You cannot delete the virtual disk if it is mounted, assigned, or has open files. See the respective help topics for more information on these commands.

1.2.3.2 Apostrophe Substitution

Prior to RSTS/E V9.7, apostrophe substitution allowed only a single symbol (contained within two apostrophes) to be substituted in a command line. In RSTS/E V9.7, you can place any expression within apostrophes. This allows for greater flexibility and clarity when creating DCL command procedures.

For example, consider the following command procedure:

```
$ run PROGRAM
$ time = f$right(f$time,12)
$ broadcast [] "PROGRAM completed at 'time'"
$ exit
```

In the following example, notice that the introduction of a temporary symbol is not necessary. This makes the command procedure run faster and makes it easier to read.

```
$ run PROGRAM
$ broadcast [] "PROGRAM completed at 'f$right(f$time,12)'"
$ exit
```

1.2.3.3 Work File

A new system logical, DCLWRK\$, has been added to RSTS/E V9.7. When a user logs into an account, DCL attempts to open its work file in the DCLWRK\$ account. If this open fails for any reason, DCL opens the work file in the user's account, as before. Users whose DCL work files are on DCLWRK\$ can now reorder their directories while they are logged in without having RSTS/E allocate the work file blocks to their account's quota.

By default, RSTS/E assigns DCLWRK\$ to SY:[0,19]. However, you can reassign it to any disk or disk account. If you assign it to a disk without specifying an account, RSTS/E places the work files in the login accounts of users who have accounts defined on that disk. For users not having accounts on the specified disk, RSTS/E uses their account on the system disk.

Because the speed of the disk directly affects command file performance, Digital recommends that you assign DCLWRK\$ to a specific account on a disk that has fast access. Additionally, do not set logged in and logged out quotas for the DCLWRK\$ account. Quotas could cause users without EXQTA privilege to receive the error message ?Unable to write DCL work file.

You can assign DCLWRK\$ to DV0: (the virtual disk). However, on a system where CPU time is in short supply, this may reduce overall system performance. If there is ample CPU time available, then using DV0: for DCLWRK\$ may improve system performance.

1.2.3.4 Performance Improvements

Several enhancements to improve DCL command procedure processing performance have been made. The areas of change responsible for this improvement are:

- Apostrophe and command substitutions take less time to perform. Even command procedures that do not take advantage of the new apostrophe substitution feature will run faster as a result of these changes.
- DCL no longer needs to perform a disk access for certain commands. DCL drops privileges when it is entered and regains them before executing a system command.
- The monitor now executes 17 fewer instructions for every character read from a DCL command procedure.

1.2.3.5 LINK Command

The DCL LINK command now allows for the specification of supervisor mode libraries. The syntax for the LINK qualifier is:

```
LINK/RMS [ =[NO]RESIDENT ]  
          [ =SUPERVISOR   ]  
          [ =USER         ]
```

1.2.3.6 SET SYSTEM Command

The following qualifiers have been added to the SET SYSTEM command:

- The /DYNAMIC_REGION_LIMIT=*n* qualifier sets the limit for the total size of all dynamic regions and FORTRAN virtual arrays that users can create without the INSTAL privilege. Any user having the INSTAL privilege can override this new systemwide limit. The SHOW SYSTEM command displays this qualifier value. See the online HELP and the *RSTS/E System Manager's Guide* for more information.
- The /EMT_LOGGING=*emt* qualifier enables or disables EMT logging for the specified EMT(s), where "emt" is a valid EMT mnemonic, such as CRE. Refer to online HELP for a list of EMT mnemonics. This qualifier eliminates the need for Feature Patch 3.5.1. The SHOW SYSTEM command now includes the /EMT_LOGGING qualifier to display the status of each EMT specified. The /[NO]LOG qualifier has been added to the SET SYSTEM command for use in conjunction with the /EMT_LOGGING qualifier. This means that the minimum abbreviation for the /[NO]LOGIN qualifier is now four characters. See the online HELP and the *RSTS/E System Manager's Guide* for more information.
- The /ANSWERBACK="*text*" qualifier sets the system response for electronic messaging systems. The SHOW SYSTEM command displays the response text. See the online HELP and the *RSTS/E System Manager's Guide* for more information.
- The /MONITOR_NAME=*name* qualifier sets the monitor file name that is started the next time the system is rebooted. The SHOW SYSTEM command displays the name of the monitor currently installed and, if different, the name of the monitor started at reboot. The SHOW SYSTEM command displays this information only if you have the WREAD privilege. See the online HELP and the *RSTS/E System Manager's Guide* for more information.
- The /SWAP_MAXIMUM=*n* qualifier sets the value that will be in effect the next time the system is rebooted. The SHOW SYSTEM command displays the current swap maximum and, if different, the value that will be in effect upon reboot. The SHOW SYSTEM command displays this information only if you have the TUNE privilege. See the online HELP and the *RSTS/E System Manager's Guide* for more information.

1.2.4 TKB

The following sections describe the new software features for TKB.

1.2.4.1 Fast Map Overlay Handler Switch

The Task Builder now supports the `/FO` and `/-FO` switches. The `/FO` switch indicates that the overlay handler for memory resident overlays should use the Fast Map version. This version is faster but does take more space. Use the `/FO` switch in conjunction with the `/FM` switch to tell RSTS/E to use the IOT instruction for Fast Map rather than for application use for this task.

The default for these switches is `/FO/FM`. You must specify `/-FO` if an application does not use Fast Map overlay handling. Likewise, you must specify `/-FM` if an application uses the IOT instruction for its own use but not for the Fast Map operation.

1.2.4.2 D Space APR Protection When Using Libraries

The `LIBR` and `RESLIB` options had an additional parameter added in V9.6 to specify the D space APR protection mask. You need to specify this mask only if the library requires D space to overmap the I space of the library and if the library was built on V9.6 or later and was not built using the `/LI:nn` switch (described in section 1.2.7 of the *RSTS/E V9.6 Release Notes*).

The parameter is appended to the end of the option. The mask follows the same pattern as described for the `/LI:nn` mask. For example:

```
LIBR=FOOLIB:RO:6:300
```

reserves D space APRs 6 and 7 for FOOLIB and those APRs are not available for the task.

If the library meets one of the following conditions, the D mask (D APR protection mask) is not required in the `RESLIB` or `LIBR` options:

- User library built with the `/LI:mask` switch
 - Library was coded I space only (as RMS is)
 - System library built with the D APRs protected (as CSPLIB is). This fact would be declared in the library documentation.
-

1.2.5 New RSX-11/M-PLUS Directives

RSTS/E V9.7 has several new RSX-11/M-PLUS directives. The new directives are:

- `FEAT$` - for determining available system-level features
- `FEADF$` - a macro to define `FEAT$` symbols
- `TFEA$` - for determining available task-level features
- `SSTX$` - a macro for exiting SST trap routines

Chapter 3 describes these directives.

1.2.6 Layered Product Features

The following sections describe the new layered product features.

NOTE

The use of RMS in supervisor mode is not supported unless specifically stated by the specific version of the layered product in question. Refer to the SPD for the specific version of the layered product to determine if supervisor mode is supported.

1.2.6.1 DECnet/E

You must use DECnet/E V4.1 with RSTS/E V9.7; you **cannot** use DECnet/E V4.0.

1.2.6.2 PDP-11 Fortran-77 Support

The following sections describe the changes for PDP-11 FORTRAN-77 support.

1.2.6.2.1 Virtual Arrays

A virtual array is an array whose storage is allocated in physical main memory outside of the program's directly addressable main memory. The use of virtual arrays in a program frees directly addressable memory for executable code and data storage.

The *PDP-11 FORTRAN-77 Language Reference Manual*, order number AA-V193B-TK, sections 2.5.1 and 5.5 describe the syntax of the VIRTUAL statement and the restrictions on the use of virtual arrays.

The *PDP-11 FORTRAN-77 User's Guide*, order number AA-V194B-TK, section 3.7 describes virtual array options. Section 3.7.2 describes the process of converting DIMENSION statements in a program to VIRTUAL statements.

RSTS/E V9.7 includes support for Fortran-77 virtual arrays up to a size of 255K words.

1.2.6.2.2 Changes to Resident Library Task-Build Files

The Task Builder library switch with bit mask (/LI:[nnn]), a feature since RSTS/E V9.6, allows the user to specify which APRs in a resident library contain data.

In order to ensure that the D space of the PDP-11 FORTRAN-77 resident libraries is protected, the task build files for the PDP-11 FORTRAN-77 resident libraries, F7RCLS.CMD and F7FRES.CMD, have been modified to indicate that APRs 6 and 7 contain data. See the Task Builder manual and the RSTS/E V9.6 Release Notes for additional information.

It should be noted that since the PDP-11 FORTRAN-77 clustered RMS resident library, F7RCLS, and the PDP-11 FORTRAN-77 FCS resident library, F7FRES, are vectored, it is only necessary to rebuild the resident library. It is not required to rebuild the tasks linked against the resident library.

Differences Between RSTS/E V9.7 and V9.6

2.1 Unsupported Devices Removed

RSTS/E has not supported the following devices since December 1988 and no longer includes installation questions for these devices:

- RP02 and RP03 disk drives
- TU56 DEctape
- PC11 and PR11 papertape devices
- DJ11 Terminal multiplexer

2.2 Software Changes

The following sections describe the software changes for RSTS/E V9.7.

2.2.1 Installation

The RSTS/E V9.7 Installation Procedure is different from that of previous versions. For complete information, refer to the updated *RSTS/E Installation and Update Guide*. Highlights of these differences are:

- You no longer need the V9 Library kit because all the modules are now on the RSTS/E V9.7 installation kit. The installation procedure does not ask you to mount the V9 Library kit.
- The installation of RSTS/E on any system now requires only one procedure; we have eliminated the need of a separate Micro/RSTS installation procedure. This allows you to custom build a RSTS/E monitor for a Micro PDP-11 machine. For convenience, the RSTS/E installation media now includes the prebuilt monitor formerly used for Micro/RSTS installations. You can select this instead of building a monitor.
- If you are installing RSTS/E from a video terminal that recognizes ANSI ESCAPE sequences, the installation procedure uses the terminal video capabilities to produce output formatted especially for the screen. For example, the top 6 lines of the screen display header information; the next 13 lines are the scrolling region for general information; and the bottom 5 lines are exclusively for input and error information. Reverse video and bold screen attributes highlight different types of information.

- The concepts of installation and update are no longer different, but now produce the same results for the RSTS/E Monitor and System Programs. If you boot the installation media, a full installation takes place and the new modules on the installation media replace the specified Monitor and System Programs. If you invoke the installation procedure by typing @[0,1]INSTALL or @[0,1]UPDATE, the new modules on the installation media replace the specified Monitor and System Programs. Layered Products still work as they did in previous versions.
- In general, the new installation procedure is more automated and requires less effort on the part of the user. In addition, it provides more information about what choices are available and about what is going on at any particular time. For example, a percentage-completed display allows you to keep track of progress during a monitor build.

Prior to RSTS/E V9.7, the TKB.TSK and MAC.TSK programs were deleted prematurely when installing SORT. This problem has been fixed.

When installing a layered product that uses the new Auto-Install procedure, you no longer have to manually assign the logicals AUTOIN\$: and AUTWRK\$:. The system startup procedure SYSINI.COM now assigns the logicals.

The following obsolete and unsupported layered products have been removed from the installation procedure: ADE, COBOL, DECGRAPH, DECWORD, DIBS, DMS500, F77DBG, INDENT, MSGROUTER, PRO IV, and RPG. If you are still using any of these products, you will have to assign the product logicals (ADE\$: or COBOL\$: for example), in your START.COM file. These logicals are no longer defined in the SYSINI.COM file.

The files DIBOL.LNK and DMS.LNK are no longer included in the RSTS/E kit. However, these files are included in the DIBOL kit and are restored when you install DIBOL.

The following obsolete and unsupported programs are no longer included in the RSTS/E kit: MONEY.TSK, MONEY.BAS, PPEXER.TSK, PREXER.TSK, DTEXER.TSK, and UPDATE.DOC.

2.2.2 EDT Performance Improvement

When using the shared version of EDT, the performance of some functions has been improved significantly for RSTS/E V9.7. On a standalone PDP 11/73 system, a top-to-bottom search of a 600 block file has been reduced from 66 seconds to 35 seconds. This performance improvement applies to the shared version only. There have been no changes to the overlaid versions of EDT.

2.2.3 Virtual Disk as Part of the Public Structure

Because the File Processor selects the public structure disk having the most free space, data files and programs could have been lost if they were placed on the Virtual Disk and the system was shut down or crashed. In RSTS/E V9.6, requiring that DV0: be mounted read-only when it was part of the public structure prevented this problem. For RSTS/E V9.7, the file processor has been modified so that DV0: can be mounted as a read-write public disk. However, files will never be placed on DV0: unless explicitly specified (even when DV0: is a part of the public structure).

2.2.4 System Command Files

The following system command files have been changed:

- In START.COM, we have added for your convenience, some sample commands to create and initialize a virtual disk and to install some files on it. By default, the code is not executed, but you can active it by editing START.COM and changing the appropriate comment lines. In your editor, search for "virtual disk" to locate the correct code.

See the [0,1]START.097 file for these changes if you are installing on an existing system.

- Previously, RECOVER.COM generated a simple SYSINI.COM file and left it in [0,1]. This could have caused confusion when running the installation procedure. RECOVER.COM now causes its own SYSINI.COM to delete itself after performing its intended task.

2.3 Monitor

The following sections describe changes to the monitor.

2.3.1 Corrections

We have corrected the following problems in the RSTS/E V9.7 Monitor:

- Previously, when you attempted to deallocate a device that was open in multiple terminal service and binary mode, it was possible for the system to crash because of a race condition in the monitor. In *The Software Dispatch* for December 1988, sequence number 3.3.18 corrected this problem; it is no longer required.
- Sequence number 3.3.19 described an echo control mode problem with the bell character. You no longer need this patch.
- Opening a keyboard in Echo Control, Binary or Enable XON/XOFF mode previously caused clearing of the typeahead buffer. For March 1989, sequence number 3.3.20 corrected this problem; this patch is no longer required.
- The problem of spawning a job on dynamic pseudo-keyboards, described in sequence number 3.5.14, has been corrected. This patch is no longer required.
- The changing priority problem described in sequence number 3.5.15 has been corrected. This patch is no longer required.
- The spawning detached jobs problem described in sequence number 3.5.16 has been corrected. This patch is no longer required.
- Overlaid I and D space programs without I space overlays did not load properly. This problem has been corrected. The programs now load correctly for both I and D space overlays. The size of the overlay still must be larger than 32 words. This removes the restriction noted in sequence number 3.1.7.
- In previous versions, asynchronous I/O requests on TMSCP tape drives (TK50 and TU81) could have failed in unusual ways if the system ran out of small buffers while processing the request. This problem has been corrected.

- In V9.6, opening a terminal in mode 33% caused flow control characters to be stored as input and passed characters to the users program if a read occurred or exhausted the terminal's input buffer quota. This problem was corrected in V9.6 as described in *The Software Dispatch*, sequence number 3.3.22 M. This problem has been corrected in source for V9.7.
- Connections from DECnet/DOS LAT failed for no apparent reason. The connection would be dropped and no reason or error message issued. The correction described in April 1989 in *The Software Dispatch*, sequence number 3.3.21 is no longer required.
- Under certain conditions, the system could crash during a UU.STL stall system SYScall. This problem has been corrected.
- For systems using system disks having a disk cluster size greater than 16, the monitor would create an invalid CRASH.SYS file during a system crash. In addition, the monitor would write data beyond the file, possibly corrupting other data on the disk.

2.3.2 Resident Library Handling

The following sections describe the changes for resident library handling.

2.3.2.1 Mapping Changes

On machines that support I and D space, RSTS/E V9.7 permits the independent assignment of I space and D space APRs within libraries and dynamic regions. For example, a library containing code only can be assigned to the I space APRs, and at the same time, a second library containing data only can be assigned to the same D space APRs concurrently. The CRAW\$ directive has been extended so that you can now map resident libraries into supervisor mode instead of user mode. Note that only the create address window (CRAW) monitor call determines into which space a region is mapped. The ATTACH and MAP functions do **not** determine or change the space a region is mapped into. See Section 3.1.8, CRAW\$ Directive of these Release Notes for details about the directive changes.

2.3.2.2 Running Tasks That Use Libraries

On RSTS/E versions prior to V9.7, a task D space could be extended only up to the lowest APR used by the library. This has been changed for V9.7. Now, tasks are loaded the same as on RSX-11/M+. That is, the task D space can be extended up to the lowest **protected** D space APR.

NOTE

This change does **not** affect existing .TSK images. All existing tasks will continue to load and run as they have in the past. This change affects only tasks built on V9.7.

Beginning with RSTS/E V9.6, a new task builder that allowed for the protection of D space APRs was provided. Refer to section 1.2.7 in the V9.6 *RSTS/E Release Notes*.

If a library was built on V9.6 or later using the /LI switch instead of the /LI:nn switch, tasks built on V9.7 and using the library may need to protect the D space APRs used by those libraries. You do this by using the RO:apr:Dmask option in either the LIBR or RESLIB options. Refer to section 1.3.4.2 for additional information.

NOTE

The RMS and CSPLIB libraries provided with V9.7 have been built with the necessary APRs already protected. No additional action is required to use these libraries.

2.3.3 SYS-Call Changes

The following sections describe the changes for SYS-Call.

2.3.3.1 UU.MNT Directive

The UU.MNT directive has been modified to add a new Disk Status subfunction. You can use this subfunction to return the CSR value and additional information about the status of a disk. See Section 3.1.5, UU.MNT Directive for additional details on the directive changes.

2.3.3.2 UU.DIE Directive

The UU.DIE directive has been modified to allow automatic restarts while a DCL LOG file is open. This corrects a problem in SHUTUP where the system would not automatically restart while a log file was open. See Section 3.1.4, UU.DIE Directive for additional details on the directive changes.

2.3.3.3 UU.CFG Directive

The monitor configuration sys-call has been updated to allow setting and returning new system configuration parameters. These new parameters are:

- Non-privileged Dynamic Region Limit
- System Answerback Message

See Section 3.1.3, UU.CFG Directive for the details on the directive changes.

2.3.3.4 UU.RTS Directive

Dynamic regions can now be up to 255K words in size. The run-time system resident library sys-call has also been updated to allow for the creation and removal of the virtual disk. See Section 3.1.6, UU.RTS Directive, for the details on the directive changes.

2.3.3.5 RSX-11/M-Plus Directive

- The RSX-11/M directive Get Command Line has been modified to allow command lines up to 127 characters. Previously, this limit was 80 characters. See Section 3.1.11, GMCR\$ RSX-11 Directive, for the details on the directive changes.
- Additional information on exiting from Ctrl/C traps (SCCA\$ directive) using the new SSTX\$ macro is supplied in Section 3.1.12, SCCA\$ RSX-11 Directive, and Section 3.1.14, SSTX\$ RSX-11 Directive.

- The SVTK\$ directive has been updated to allow specification of user mode or supervisor mode service routines. See section 12.15.1.2 in the new chapter 12 of the *Task Builder Reference Manual* for additional details.
- The SVDB\$ directive has been updated to allow specification of user mode or supervisor mode service routines. See section 12.15.1.2 in the new chapter 12 of the *Task Builder Reference Manual* for additional details.

2.4 INIT.SYS

The following sections describe the changes for INIT.SYS.

2.4.1 Corrections

The following problems have been corrected in INIT.SYS:

- The missing memory problem described in *The Software Dispatch*, sequence number 1.1.12 has been corrected. This patch is no longer required.
- The large number of terminals problem described in *The Software Dispatch*, sequence number 1.1.13 has been corrected. This patch is no longer required.
- In V9.6 during system startup, INIT failed to check for the carrier detect signal on interfaces that supported modem control. This change broke Leased Lines support. This problem has been corrected in V9.7.

2.4.2 Default Option

The VIRTUAL suboption of the Memory Table section in DEFAULT has been removed. This function is now available for DCL commands during timesharing. See Section 1.2.3.1, Virtual Disk Commands, for a description of the DCL syntax.

2.4.3 INIT Now Recognizes DR11/DRV11

INIT.SYS now recognizes the presence of DR11 or DRV11 parallel DMA interfaces on the system. The device name is XA. No drivers or other operating system support is provided.

2.4.4 INIT Checks For Rev 0.0 Disks

If you perform an update on a pre-V8.0 system, the error message ?Warning-system disk will be software write-protected DCL commands may give unexpected errors is issued. This is an informational message and indicates that errors may occur.

2.4.5 Virtual Disk No Longer Shows as Disabled Device

In previous versions, the virtual disk was counted in the devices disabled message that appeared during system startup. Because the virtual disk is now dynamic, it is no longer counted as a disabled device.

2.5 Commonly Used System Programs (CUSPs)

The following sections describe the changes for CUSPS.

2.5.1 DCL

The following sections describe changes to DCL.

2.5.1.1 Command Processing

The following changes apply to DCL command processing:

- The DCL READ command would sometimes incorrectly handle RMS fixed-length files having the SPAN attribute if the record size was not a power of two and was less than 512. This problem has been corrected.
 - Prior to RSTS/E V9.7, the IF command returned the ?If statements nested too deeply error message when the number of IF-THEN commands was repeated more than 14 times. This no longer results in an error. The only limit to the number of IF-THEN commands allowed on one line is the length of the command line.
 - The error message ?Invalid function name has been changed to ?Invalid function.
-

2.5.1.2 LAT Commands

We have corrected several messages for the SET SERVICE/LAT and SHOW SERVICE/LAT commands to change the word *server* to *service*. Messages now also contain the name of the service where possible.

The following DCL messages have been changed from severe errors to errors:

?LAT is not installed
?DECnet is not started

2.5.1.3 SHOW LOGICAL Command

The SHOW LOGICAL command no longer displays SY: for a logical that has not been assigned a device.

2.5.1.4 ASSIGN Command

The ASSIGN/SYSTEM command no longer defaults the device of a system logical to SY:.

2.5.1.5 SHOW DEVICE Command

When using the SHOW DEVICE command on systems having multiple MSCP type controllers, the CSR address for DU type disks now displays correctly.

The SHOW DEVICE command now operates correctly for disks having a device cluster size greater than 16.

2.5.1.6 SHOW SYSTEM Command

The SHOW SYSTEM display has been consolidated to ensure that all information fits on one screen. The online HELP for SHOW SYSTEM now lists the privileges you need to view certain parts of the display.

2.5.1.7 SHOW TERMINAL/PERM Command

The SHOW TERMINAL/PERM command has been changed to allow users not having the HWCFG privilege to see the permanent characteristics of any terminal that they have allocated to their job.

2.5.1.8 SHOW QUOTA Command

The SHOW QUOTA command normally shows disk quota information for the users account on each mounted disk. Now the user can supply an account specification, such as [249,1], [249,*], [* ,1], [* ,*]. The command displays the quota information for each account that matches the specification on each mounted disk where that account exists.

2.5.1.9 SET SYSTEM/HOLD Command

Error trapping for the SET SYSTEM/HOLD command has been changed to display more meaningful error messages. If you execute the command from a pseudo keyboard (batch or set host job), the error message ?System cannot be placed on hold from a pseudo keyboard. is displayed. If you execute the command while the system is already on hold, the error message ?System already on hold is displayed.

2.5.1.10 SET JOB/PRIVILEGE Command

The SET JOB/PRIVILEGE command no longer accepts a privilege in the form =NONOpriv. For example, the command SET JOB/PRIVILEGE=NONOWACNT returns an error message.

2.5.1.11 SET TERMINAL Command

The SET TERMINAL/WIDTH command has been changed to always send the appropriate escape sequences to ANSI scope terminals and change the terminal width accordingly. Previously, the command sent the escape sequences only if the specified width was different from the terminal's current width.

Prior to RSTS/E V9.7, executing a SET TERMINAL/INQUIRE command on a VT125 terminal did not set the Advanced Video and 132 Columns attributes. These attributes are now set correctly.

The /[NO]ANSWERBACK qualifier has been added to the SET TERMINAL command. Specifying /ANSWERBACK allows the terminal to receive messages from electronic messaging systems. Because this is a permanent attribute of the terminal, you must specify the /PERMANENT qualifier with /[NO]ANSWERBACK. In addition, the terminal must not be set to Autobaud. The SHOW TERMINAL/FULL command displays this attribute. See the online HELP and the *RSTS/E System Manager's Guide* for more information.

2.5.1.12 BACKUP/DIRECTORY Command

The following changes have been made to the BACKUP/DIRECTORY command.

- BACKUP no longer requires the INSTAL privilege to run. In order to use BACKUP without the INSTAL privilege, the system manager must specify a value of 3 or greater in the \$SET SYSTEM/DYNAMIC_REGION command. See the RSTS/E help files for more information on the SET SYSTEM/DYNAMIC_REGION command.

If BACKUP is running without INSTAL privilege and cannot allocate space from the system dynamic region pool, the ?Not enough free memory to create dynamic region error message is issued and BACKUP stops.

- BACKUP no longer attempts to recreate its dynamic region when doing a BACKUP verify or RESTORE verify operation. Previously, there was a problem where BACKUP was unable to recreate the dynamic region for the verify operation and the program would issue a fatal error. BACKUP now creates the dynamic region before the BACKUP or RESTORE operation and does not deallocate it until just before prompting for the next command.
- BACKUP now uses dynamic pseudo keyboards. This was also noted in the V9.6 Release Notes but a problem in BACKUP prevented the feature from working correctly in V9.6.
- The RESTORE command has been corrected so that it does not override the /BUFFER qualifier. In RSTS/E V9.6, RESTORE would always attempt to create a 55K dynamic region when using MS or MV tape drives regardless of any value specified with the /BUFFER qualifier.
- The /DIRECTORY qualifier no longer conflicts with the /LIST qualifier.
- The BACKUP/DIRECTORY command now includes header information. The header information, stored at the beginning of every RSTS/E and VMS backup set, contains information about the backup set. The command now displays the header information regardless of the /BRIEF and /FULL qualifiers that may be specified with the command. An example of the new feature follows:

```

$.BACKUP/DIRECTORY NEWDCL.BCK
Please mount volume 1 of Backup set NEWDCL.BCK
Where can this volume be located? <SY:>

Directory of save set:
Save set:          NEWDCL.BCK
Written by:       DCL Account
PPN:              [195,20]
Date:             26-Oct-88 08:57 AM
Command:          backup s:[170,20]*.*/exclude=[170,]**.E?? NEWDCL
Operating system: RSTS/E V9.0
BACKUP version:   1.0
Node name:        _NODE::
Written on:        _SY0:
Block size:       2048
Group size:       10

Disk pack attributes
Label:            97S
Clustersize:     16
Pack size:       891072

Account Name .Typ   Size   Prot  Accessed   Date       Time      Clu  RTS
[170,20 ]STATUS.CMS  1 P   < 60> 23-Oct-88 02-Nov-87 03:52 PM  16  ...RSX
[170,20 ]AUXMSG.MAC  4 P   < 60> 02-Nov-87 02-Nov-87 03:52 PM  16  ...RSX
[170,20 ]AUXTRN.MAC  3 P   < 60> 02-Nov-87 02-Nov-87 03:52 PM  16  ...RSX
[170,20 ]DCL .CTL    2 P   < 60> 02-Nov-87 02-Nov-87 03:52 PM  16  ...RSX
[170,20 ]DCLDBG.MAC  3 P   < 60> 02-Nov-87 02-Nov-87 03:52 PM  16  ...RSX
Please dismount volume 1 of Backup set NEWDCL.BCK

```

2.5.1.13 HELP Command

In the HELP program, the CCL HELP command now works the same as the DCL HELP command. The HELP program now prompts for the next topic instead of immediately exiting.

In addition, the system HELP files now contain a help file for the \$WAIT command. For help on the \$WAIT command, type HELP WAIT at the DCL prompt.

We have also updated several other help files.

All known problems with the HELP files have been corrected.

2.5.1.14 DISMOUNT Command

In some instances, when you issued the DISMOUNT command for a disk that had already been spun down, you received an error message and the disk did not dismount. This happened when certain blocks from the disk were in cache and other specific blocks were not. This problem has been corrected and the disk can now be dismounted no matter what error occurs.

2.5.1.15 INITIALIZE Command

The INITIALIZE command now correctly initializes a disk having a device cluster size of 64.

Previously, when you used the INITIALIZE command for a tape and then typed Ctrl/Z at the prompt, DCL exited with the ?End of file on device error message. The command now exits without an error.

2.5.1.16 COBOL Command

The /STANDARD=option qualifier has been added to the COBOL command. Option can be either V2 or 85. See the COBOL V3.0 User's Guide for details.

2.5.1.17 DELETE/SERVICE/LAT Error Message

The error message has been changed from ?Invalid characters in service name or identification string to ?Invalid characters in service name.

2.5.2 PBS Package

Prior to version RSTS/E 9.7, Print Batch Services (PBS) always issued a form feed after printing each file, even if this occurred at the top of a page. This was a problem when printing payroll checks or some other form where blank pages were not desired.

PBS now keeps track of whether or not a page has any characters printed on it. If the end of a file is reached and the printer is on the first line of a page containing no printable characters, PBS does not issue a form feed. Instead, PBS immediately starts printing the next file or copy.

PBS now correctly determines if the owner specified in the SUBMIT/OWNER command is granted access to the command and log files.

2.5.3 LOGIN

The LOGIN program has been modified to accept the Ctrl/E character as input when the /ANSWERBACK characteristic is enabled on a terminal line.

The LOGIN program no longer looks up new mail messages in the user's MAIL.MAI file. The system wide LOGIN.COM (located in [0,1]) now runs the program MAIL\$:NEWMAI.TSK to display the number of new mail messages. This allows systems with DECmail-11 to continue using this feature, and improves LOGIN performance on those systems that do not have DECmail-11.

2.5.4 SHUTUP

The automatic restart feature of the SHUTUP program can now be used while a DCL log file is open. A new phase to close a log file has been added to the SHUTUP program. See the *RSTS/E System Manager's Guide* for details.

Instead of simply exiting with an error number when it finds open files on the disk, SHUTUP now exits with the ?Can not proceed because of open file(s) on the disk error message.

2.5.5 OPSEER Package

The following changes have been made to the OPSEER package:

- When OPSEER attempted to broadcast to an OSC that no longer existed, such as a dynamic KB, OPSEER crashed with the following error message:

```
ERR 8 AT LINE 15120
Stop at line 19200 in SUB:OPSRUN
```

This problem has been corrected. OPSEER now changes the OSC to KB0: when this error occurs and then continues.

- If \$LOGIN.COM<104> has had its protection code changed to not include the executable bit (bit value = 64), BATRUN generates the ??Unable to Login Batch Job error message and requeues itself. BATRUN.TSK has been changed to run \$LOGIN.TSK instead of \$LOGIN.COM when creating a batch job at a pseudo keyboard.

2.5.6 PIP

PIP did not print an account number with a file name. This resulted from an unsuccessful attempt to delete and erase a file. Additionally, if the original filespec contained wildcards, the delete procedure would terminate with an error instead of continuing to completion.

The PIP /REname command no longer requires a device name in the output specification if there is a device specified in the input specification.

The PIP /RMS:FA copy routine would fail to copy all of a file if a record in the input file happened to end at the end of the PIP internal buffer. PIP now copies these files correctly.

PIP copies of RMS files to ANSI magnetic tape now use the correct block size. This means that a VMS system now correctly reads RSTS/E backup sets transferred to ANSI magnetic tape.

2.5.7 DIRECT

The following errors have been corrected in TKB and STK for RSTS/E V9.7:

- Specifying a HISEG option caused the error message:
TKB—*FATAL*-Illegal Get Command Line error code
- Specifying an option line such as RESLIB=LIBRARY/RO:base:bitmap caused a syntax error.
- Building an I and D task using a PIC resident library caused a D-space APR conflict if a base APR was not specified.

NOTE

STK features have been available in TKB using the /SB switch since RSTS/E V9.5. STK will be removed from RSTS/E in a future release. Digital recommends that you immediately update command files to use this TKB switch.

2.5.14 EMTCPY

In RSTS/E V9.7, the unsupported EMTCPY.TSK program now includes the EMTs that were added since RSTS/E V9.0. This program now traps Ctrl/C and properly closes its files before exiting.

The EMTCPY program now displays wildcard filetypes correctly.

2.5.15 ANALYS Package

ANALYS now dumps the supervisor mode APRs when appropriate. The annotations have been improved significantly. The .SIL module physical addresses are now correct, including KISAR5 and KISAR6. The job numbers are annotated in the job control region.

2.5.16 Update Package

Because of the new /EMT_LOGGING qualifier for the SET SYSTEM command, feature patch file PA0305.001 has been removed. Use this new qualifier to selectively enable and disable logging of EMTs.

In V9.6, patch files PA0303.001 and PA0303.002 would not install correctly because of changes to the module name. These feature patches were superseded as described in May 1989 in *The Software Dispatch*. The patch files in the UPDATE\$: account have been updated with the new module names.

2.5.17 Removal of Unsupported Software

As announced in June 1987, the following CUSPS or software capabilities were retired as of December 1988:

- The SAVE option of SAVRES has been removed. This is the first step in the retirement process for the SAVRES program. For RSTS/E V9.7, as in previous versions, the RESTORE and IMAGE options are still available.

To create bootable recovery medium, use the RECOVER.COM procedure documented in the *RSTS/E V9.0 Release Notes*, section 2.13, and the updated *RSTS/E System Managers Guide*.

- FLINT.TSK
- 5- and 6-bit character lengths

2.6 Basic-Plus Run-Time System

When you are operating in the Basic_Plus environment and execute a command file, you no longer see the "Ready" prompt between the DCL commands in the file.

2.7 RMS-11

Several problems that occurred when using the /EO switch with RMSCNV have been corrected. RMSCNV would fail to remove a Ctrl/Z character located in the first byte of a block. It would also fail, with an odd address trap error, if a Ctrl/Z character was located in the last byte of the block. These problems have been corrected.

RMSIFL would, at times, incorrectly return an exit status of success to DCL after encountering a read error in the input file. RMSIFL has been corrected to return an error exit status under those conditions.

2.8 Layered Products Changes

The following sections describe changes to layered product support.

2.8.1 DECnet/E

You must use DECnet/E V4.1 with RSTS/E V9.7. You cannot use DECnet/E V4.0 with RSTS/E V9.7.

2.8.1.1 Monitor Corrections

The Ethernet XEDVR (for DELUA and DEUNA devices) and XHDVR (for DEQNA and DELQA devices) have been modified so that Ethernet Maintenance Counter Requests are answered with a correct Counters Packet. In the past, this request could be answered incorrectly or not at all because of a problem in the request code. If Counter information is polled and compiled for your network, you may notice an increase in complete responses.

2.8.1.2 Installation

Several modifications have been made to the DECnet/E installation and update procedure. Major enhancements include:

- A new Node Definition Phase during a DECnet/E Installation
- A new question allowing the installer to optionally install or update the new NCP Help information
- A new question allowing the installer to decide if the node should be defined as a router or as an end node
- Removal of a prior restriction not allowing the volatile parameter file to be located on a nonsystem disk

In addition, the node definition procedure and the installation test procedures are now available for use outside of the Installation and Update Procedure. For more information see the V4.1 *DECnet/E Network Installation and Update Procedure*.

2.8.1.3 NCP Help

With RSTS/E V9.7, NCP now includes online NCP HELP. You can select the NCP HELP facility during a DECnet/E V4.1 installation or update. Once installed, NCP HELP is available right from the NCP> prompt, or can be accessed under PROGRAMS in the standard RSTS/E HELP facility.

The use of NCP HELP does have the following limits:

- You cannot use NCP HELP on any system that does not have the standard RSTS/E HELP package installed.
- With DECnet/E V4.1, NCP HELP is available only on the local node; that is, no help is available when NCP is operating as a REMOTE EXECUTOR, regardless of the help that may be available locally to the remote system or to your local node.

2.8.1.4 NCP Circuit Maximum Routers Parameter

Since V4.0, DECnet/E has had an EXECUTOR MAXIMUM ROUTERS parameter and a CIRCUIT MAXIMUM ROUTERS parameter for Ethernet Circuits. These two parameters are very similar and are closely related. However, disrupting this delicate relationship has sometimes caused network problems that were very difficult to diagnose.

In RSTS/E V9.7, the NCP parameter CIRCUIT MAXIMUM ROUTERS no longer has any meaning. You can still set the parameter within the limits documented in the *DECnet/E System Manager's Guide*, but the result has no impact on the operation of your DECnet/E implementation and is no longer displayed with your Ethernet circuit characteristics.

This means that the EXECUTOR MAXIMUM ROUTERS parameter is the only maximum routers parameter. Successful network operation is no longer disrupted by the interaction of it with the CIRCUIT MAXIMUM ROUTER parameter.

2.8.2 DECmail-11

DECmail-11 now supplies the program NEWMAI.TSK located in the MAIL\$: account. When running this program, it prints the number of new mail messages in the MAIL.MAI folder for the current account.

2.8.3 WPS-PLUS

WPS-PLUS/RSTS V1.0 has been updated to V1.1.

WPS-PLUS/RSTS V1.0 required certain privileges to use the Spell Checker. This update corrects that problem and allows the WPS-PLUS user to run Spell Checker from a nonprivileged account. It also supersedes the patch described in the WPS-PLUS/RSTS *The Software Dispatch* sequence number 26.0.2M.

2.8.4 DIBOL

RSTS/E V9.7 contains changes for both RSTS DIBOL V5.2 and V6.0. The installation procedure detects which version of RSTS DIBOL is being installed and uses the appropriate updated components. RSTS/E V9.7 also provides online release notes documenting the changes to RSTS/E DIBOL.

2.8.5 COBOL-81

RSTS/E V9.7 includes a change to the ANSI 85 file status values. Previously, when an attempt was made to rewrite a record that was not the same size as the record being replaced, an erroneous file status of 14 was given. The corrected file status is 44.

V9.7 also includes changes to modules to reflect the new COBOL-81 V3.0 header at the C81> prompt, as well as in the .LST file. The updated modules are:

Library	Module	Problem Description
Compiler	CMND	Display C81 V3.0-01 at the C81> prompt
Compiler	LSTUTS	Print COBOL-81 V3.0-01 in .LST file
OTS	STA85	Fix problem with incorrect file status 14

Updating of the COBOL-81 compiler and OTS libraries is controlled by the Auto-Install procedure that was shipped with COBOL-81 V3.0. You must first copy the update files to the PATCH\$ account. Then install COBOL-81 using the COBOL-81 V3.0 installation kit, and installation guide. The Auto-Install procedure first copies the update files to the correct accounts, and then continues with the installation.

NOTE

Before you begin the COBOL-81 update, be sure to delete C81UPD.DON from PATCH\$:. This file is left over from any previous COBOL-81 updates, and indicates that the update was done.

Documentation Changes

The documentation for RSTS/E V9.7 consists of these release notes, a revised *RSTS/E System Installation and Update Guide*, and a revised *RSTS/E System Manager's Guide*.

3.1 Changes to the System Directives Manual

The following sections describe the changes made to the *RSTS/E System Directives Manual*.

3.1.1 General RSTS/E Environment

On page 2-4, replace the note text with the following:

The PDP-11/44, 45, 50, 55, 70, 73, and 84 have three sets of APRs; the additional set is for supervisor mode mapping. RSTS/E supports supervisor mode only under the RSX run-time system. Using supervisor mode outside of the RSX run-time system may cause unpredictable results, including fatal abort.

3.1.2 .PLAS Directive

On page 3-163, under FIRQB+FQMODE, replace the definition for *bit 1* with the following:

bit 1 = 1	Write access to the window is desired if bit 7=1.
= 0	No write access to the window is desired if bit 7=1.
bit 4 = 1	CRAFQ will allocate this window in the user's data space. Any mapping will occur in D space APRs only.
= 0	This window is not a data only window.
bit 5 = 1	CRAFQ will allocate this window in the supervisor I Space. Any mapping will occur in the supervisor I space APRs only.
= 0	This window is not a supervisor mode window.

At the bottom of page 3-163, add the following text:

If neither bit 4 nor bit 5 is set, then user instruction space is used. There is a functional correspondence between bits 4 and 5 in the CRAFQ mode byte and the region options available in TKB. This correspondence is as follows:

TKB option	CRAW mode bit setting
RESCOM & COMMON	bit 4 set to 1
RESSUP & SUPLIB	bit 5 set to 1
RESLIB & LIBR	neither, bit 4 & 5 set to 0

If bit 4 = 1, the window is created in the D APR space only; no mapping in the I space is done. If another window is already created in both the I and D space at this APR, the I space window remains as is, but the D window is moved to this request. If a CRAFQ is done with bit 4 set and a second CRAFQ is done to the same APR with neither bit 4 nor bit 5 set, the resulting mapping is such that the D space APR is given to the CRAW with bit 4 set and the I space APR is assigned to the region of the second CRAFQ.

It is illegal to CRAFQ a region with both bits 4 and 5 set. In RSTS/E, supervisor data space is always mapped the same as user data space.

3.1.3 UU.CFG Directive

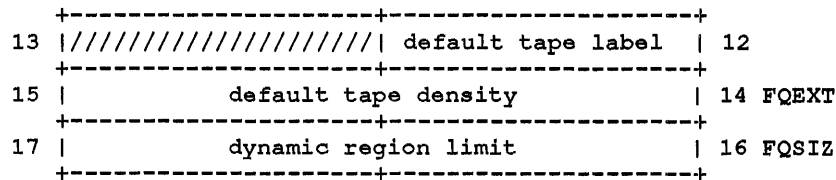
On page 3-312, under the section SWCFG is required to:, add the following:

- o Set Dynamic Region Limit

SWCTL is required to:

- o Set or return system Answerback

On page 3-320, in the diagram of the FIRQB, update the following sections:



On page 3-321, replace the section FIRQB+FQNAM1+3, with the following:

FIRQB+FQNAM1+3 Reserved

FIRQB+FQEXT Default magnetic tape density in bits per inch (bpi).

FIRQB+FQSIZ Non-privileged user dynamic region limit. Zero indicates no change, minus one (-1) sets the limit to 0K words. A value between 1 and 2048 sets the limit to that value in K words. The sum of the dynamic region sizes owned by all nonprivileged users cannot exceed the limit value given here. If a nonprivileged user (without INSTAL privilege) attempts to create a region whose size, when added to all other nonprivileged region sizes, exceeds this limit, a NOROOM error message is issued and the region is not created.

On page 3-321, in the diagram of the FIRQB, update the following sections:

13		default tape label	12
15		default tape density	14 FQEXT
17		dynamic region limit	16 FQSIZ

On page 3-323, replace the section FIRQB+FQNAM1+3 with the following:

FIRQB+FQNAM1+3 Reserved
 FIRQB+FQEXT Default magnetic tape density in bits per inch (bpi).
 FIRQB+FQSIZ Dynamic region limit in K words.

Add page 3-324a with the following information:

Data Passed (Set and Return System Answerback Message)

Mne- monic	Octal Offset	FIRQB		Octal Offset	Mne- monic
	1			0	
FQFUN	3	UU.CFG (=42)		2	
FQSIZM	5	Flag	subfunction code =7	4	FQFIL
	7	\ Answerback Message Text		6	FQPPN
	35			34	
	37			36	

FIRQB+FQSIZM Flag = -1 to return the Answerback text
 > 0 to set the Answerback text

FIRQB+FQPPN The ASCII string for the Answerback text

Data Returned

Mne- monic	Octal Offset	FIRQB		Octal Offset	Mne- monic
	1			0	
	3			2	
	5		subfunction code =7	4	FQFIL
	7	\ Answerback Message Text		6	FQPPN
	35			34	
	37			36	

FIRQB+FQPPN Answerback message text returned if Flag = -1.

Errors

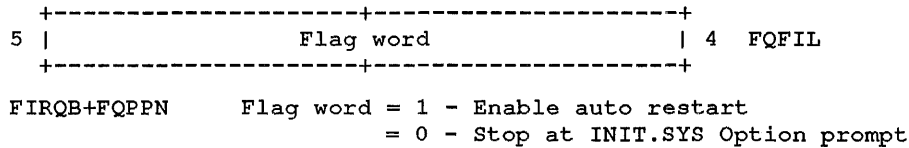
PRVIOL Current user does not have SWCTL privilege.

NOTAVL There is no Answerback text to return.

NOBUFS There are no buffers available to store the text.

3.1.4 UU.DIE Directive

On page 3-353, in the diagram of the FIRQB, update the following sections:



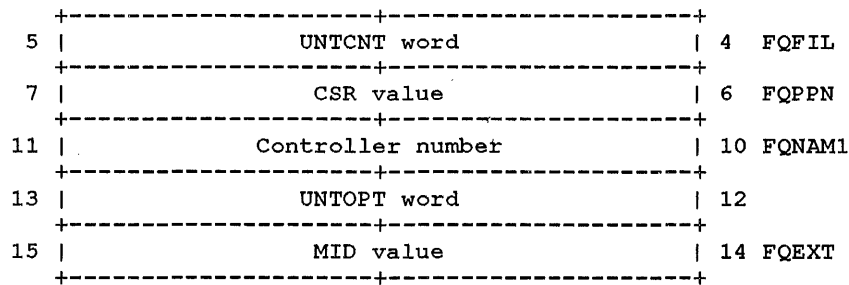
This change was made in V9.6.

3.1.5 UU.MNT Directive

On page 3-400, under the section: FIRQB+FQFIL The subfunction codes are:, add the following subfunctions:

Code	Meaning
10	Load the SATT from a disk pack or cartridge
12	Unload the SATT to a disk pack or cartridge
14	Reserved for future use
16	Return disk drive status information

On page 3-401, in the diagram of the FIRQB, update the following sections:



On page 3-401, between the FIRQB diagram and the returned errors, add the following:

When disk drive status information is requested, the following information is returned in the FIRQB:

FIRQB+FQFIL	The internal UNTCNT value for the specified disk
FIRQB+FQPPN	The CSR value for the specified disk. The disk does not exist if this value is 0. If a CSR is returned, the specified unit may or may not exist.
FIRQB+FQNAM1	The controller number if the specified disk is a "DU" type disk. If the disk is not a "DU" type, then this value is random.
FIRQB+FQNAM1+2	The internal UNTOPT value for the specified disk.
FIRQB+FQEXT	The internal drive type identifier for the specified disk.

3.1.6 UU.RTS Directive

On page 3-440, replace the section FIRQB+FQSIZ with the following:

FIRQB+FQSIZ	bits 0-7	Size region requested (maximum of 255.) in K words of memory.
	bit 14	Set to 1 if the request can accept a smaller size region than requested if not enough memory exists. A zero value will cause an error if there is not enough memory to satisfy the request.
	bit 15	Set to 1 to use bits 0-7 for size. A zero value indicates only bits 0-6 contain size.

Add page 3-442a with the following information:

Data Passed (Create/Delete Virtual Disk)

Mne- monic	Octal Offset	FIRQB	Octal Offset	Mne- monic
	1		0	
FQFUN	3	UU.RTS (= -22)	2	
	5	subfunction code=32	4	FQFIL
	7		6	
	11		10	
	13		12	
	15	Start Virtual disk Address	14	FQEXT
	17	Virtual disk size in K words	16	FQSIZ
	21		20	
	23		22	
	25		24	
	27		26	
	31		30	
	33		32	
	35		34	
	37		36	

FIRQB+FQFUN	-22	(UU.RTS)
FIRQB+FQFIL	32	(Create/delete virtual disk)
FIRQB+FQEXT		K word address at which to place the virtual disk (from 0 to 2043), or -1 to let RSTS/E select the "best-fit" address.
FIRQB+FQSIZ		Size of virtual disk to create, in K words (from 1 to 2044), or 0 to delete the current virtual disk. If you specify 0, the address parameter in FQEXT is ignored.

Discussion

This call allocates memory for the virtual disk. Any memory so allocated is not available for other use, nor can it be windowed or mapped using the .PLAS directives. When the virtual disk is deleted, the memory is then made available for other uses.

When a request is made to create a virtual disk, the memory manager must sometimes "shuffle" things in memory (such as jobs, run-time systems, etc.) to make enough contiguous memory space for it. This task may take some time, particularly on very busy systems. Therefore, the virtual disk driver has been modified so that, before the first access to a newly-created virtual disk is allowed, it ensures that all memory that is supposed to be available for the virtual disk has really been allocated. If not, then a '?Device hung or write locked' error will be returned to the job requesting the access. If this should occur, it is recommended that the requesting job should sleep for a short period of time (a few seconds) and then retry the operation, thus giving the memory manager time to do its job.

Note that the virtual disk created by this system call is identical in structure and action to the one that could be created offline (using INIT.SYS) in previous versions. When a virtual disk is created, it is not automatically initialized as a RSTS/E volume. As in the past, it must still be initialized (using the INITIALIZE command) and mounted (using the MOUNT command) if required.

HWCFG and INSTAL privileges are required to create or delete the virtual disk.

Data Returned

Except for a possible error in byte 0 of the FIRQB, this subfunction does not return any meaningful data.

Errors

When creating:

PRVIOL User or program did not have HWCFG privilege
NOROOM There was not enough contiguous memory available to create the virtual disk at the requested size.
BADCNT The specified size or address was invalid, or, if an address was given, the virtual disk would not fit at that address.
NOBUFS Not enough small buffers were available to build the memory descriptor blocks.
FIEXST A virtual disk already exists.

When deleting:

PRVIOL User or program did not have HWCFG privilege
INUSE The virtual disk is mounted, has open files, or is open in non-file-structured mode.
NODEVC There is no virtual disk to delete.

3.1.7 UU.TRM Directive

On page 3-481, under the section FIRQB+FQDEVN, add the following bit definitions:

11 Set Non-Interactive
12 Set Answerback mode

On page 3-481, under the section FIRQB+FQCLUS, add the following bit definitions:

- 11 Clear Non-Interactive
- 12 Clear Answerback mode

3.1.8 CRAW\$ Directive

On page 5-15, under the section adr+W.NSTS, update the following text:

is the access-flags word. Two bits of this word define whether the window is to be mapped and whether read-only access or read/write access is desired. Two additional bits define if the window is to be mapped into supervisor I space or User D space. If neither WS.SIS or WS.UDS are set then User I space is used.

Mnemonic	Bit	Meaning
WS.MAP	7 = 1	The window is to be mapped.
	= 0	The window is not to be mapped.
WS.SIS	5 = 1	CRAW will allocate this window in the supervisor I Space. Any mapping will occur in the supervisor I space APRs only.
	= 0	This window is not a supervisor mode window.
WS.UDS	4 = 1	CRAW will allocate this window in the user's data space. Any mapping will occur in D space APRs only.
	= 0	This window is not a data only window.
WS.WRT	1 = 1	Read/write access is desired if WS.MAP=1.
	= 0	Read-only access is desired if WS.MAP=1.

3.1.9 FEAT\$ RSX-11 Directive

Add page 5-30a with the following information:

Test for specified system feature (FEAT\$)

The test for specified system feature directive tests for the presence of a specific system software or hardware option, such as floating point hardware or the presence of the commercial instruction set.

Macro Call:

FEAT\$ sym

Where:

sym = Symbol for the specified system feature

Supported System Feature Symbols Table

Symbol	Value	Meaning
FE\$EXT	1	22-Bit extended memory support
FE\$PLA	5	PLAS calls supported
FE\$LIB	18.	Supervisor mode library support
FE\$UDS	37.	User data space supported
FE\$FMP	59.	Fast mapping directive supported
HF\$UBM	-1.	Processor has UNIBUS mapping registers
HF\$EIS	-2.	Processor has Extended Instruction Set
HF\$QB	-3.	Processor has a Q-BUS backplane
HF\$DSP	-4.	Processor supports separate I & D spaces
HF\$CIS	-8.	Processor supports Commercial Instruction Set
HF\$PPP	-16.	Processor has NO floating point unit

Macro Expansion:

```
FEAT$      FE$FMP
.BYTE      177.,2      ;FEAT$ MACRO DIC, DPB size = 2 words
.WORD      FE$FMP      ;feature identifier
```

Local symbol definitions:

F.EAF Feature identifier

DSW return Codes:

IS.CLR - Successful completion; feature not present(=0)

IS.SET - Successful completion; feature present(=2)

IE.ADP - Illegal DPB address

IE.SDP - Invalid DIC or feature identifier code

Note:

The full list of RSX-11/M-PLUS feature identifiers is not supported on RSTS/E.

3.1.10 FEADF\$ RSX-11 Directive

Add page 5-30b with the following information:

Feature Test Definition Macro (FEADF\$)

The FEADF\$ macro is provided in the system macro library for ease of proper definition of the system feature labels.

Macro Call:

FEADF\$

This is purely a definition macro and therefore there are no results or errors.

3.1.11 GMCR\$ RSX-11 Directive

On page 5-34 after the first paragraph, add the following:

If the CCL command line is longer than 80 bytes, the first 79 bytes followed by a hyphen in position 80 will be returned on the first call to GMCR\$. A subsequent call will return the remaining portion of the command line. The maximum CCL command line length that can be retrieved via the GMCR\$ call is 127. characters.

After the line IE.SDP DIC or DPB size is invalid, add the following:

Notes:

- 1) The command line interpreter processes all lines to:
 - Convert tabs to a single space
 - Convert multiple spaces to a single space
 - Convert lowercase characters to uppercase
 - Remove all comment text after an exclamation point
 - Remove all trailing blanks
- 2) The hyphen indicating that additional characters remain is generated by RSTS/E and is not part of the command line.
- 3) Differences with RSX-11/M+:
 - RSTS/E does not transfer the terminating character.
 - Comments between exclamation points are not supported.
 - A maximum of 127. characters can be transferred.

3.1.12 SCCA\$ RSX-11 Directive

On page 5-55 in the section SCCA\$\$, replace the third paragraph with the following:

Your routine can handle the Ctrl/C in any way it sees fit. The Ctrl/C routine should be exited with the use of the SSTX\$ macro. This will return control back to the point where it left off at the time of the exception. If the task is going to use a library in Supervisor mode, the SSTX\$ exit is required.

3.1.13 TFEA\$ RSX-11 Directive

Add page 5-65a with the following information:

Test for Specified Task Feature (TFEA\$)

The test for specific task feature directive tests for the presence of a task specific software option, such as fast-mapping turned on by the /FM switch in the TKB when the task was built.

Macro Call:

TFEA\$ sym

Where:

sym = Symbol for the specified task feature

Supported Task Feature Symbols Table

Symbol	Value	Meaning
T2\$DST	15.	AST recognition disabled (IS.SET=disabled)
T4\$DSP	34.	Task was built /ID or has turned I & D on
T4\$FMP	40.	Task has fast mapping turned on

Macro Expansion:

TFEA\$ T4\$FMP
.BYTE 177.,2 ;TFEA\$ MACRO DIC, DPB size = 2 words
.WORD T4\$FMP ;feature identifier

Local symbol definitions:

F.TEAF Feature identifier

DSW return Codes:

IS.CLR - Successful completion; feature not present(=0)

IS.SET - Successful completion; feature present(=2)

IE.ADP - Illegal DPB address

IE.SDP - Invalid DIC or feature identifier code

Note:

The full list of RSX-11/M-PLUS feature identifiers is not supported on RSTS/E.

3.1.14 SSTX\$ RSX-11 Directive

Add page 5-59a with the following information:

System Synchronous Trap Exit (SSTX\$)

In the RSTS/E environment, you can use the SSTX\$ directive to terminate a routine that handles a synchronous system trap such as BPT and Ctrl/C exceptions among others, only if the program has indicated its intention to handle such traps with a SVTK\$ or SVDB\$ directive. When the SSTX\$ is issued, control is returned to the point where the program left off at the time of the trap.

If the task also uses supervisor mode libraries, then the SSTX\$ is the required exit from any synchronous trap service routine. There is no equivalent directive in RSX-11/M+.

If the trap was a memory protection violation, the service routine must remove the additional memory register information from the stack before issuing the SSTX\$ directive.

Privileges Required:

None

Macro Call

SSTX\$

Macro Expansion:

```
TST  2(SP)      ;Is this a user or super return?
BPL  10$        ;Super!
RTI                      ; User
```

10\$: CSM #0 ;So, return to Supervision mode via Digital Standard.

Local Symbol Definitions

None

DSW Return Codes

None (SSTX\$ does not execute an EMT; control does not pass to the RSX emulator when SSTX\$ is executed.)

NOTE

The SSTX\$ call, when used with supervisor mode, is supported only when the Digital-supplied Supervision Linkage modules are used.

3.2 Changes to the RMS-11 User's Guide

Add the following to section 8.1.2.1 on page 8-6:

8.1.2.1.1 Building A Supervisor Mode RMS Library.

On the RSTS/E systems that support supervisor mode, you may choose to use RMSRES as a supervisor mode library instead of user mode. Because this configuration uses two otherwise idle supervisor mode APRs to map most of the RMS-11 code, the impact of the RMS-11 code on your user-mode virtual address space is reduced to the absolute minimum; there also may be slight performance advantages over the clustered RMS-11 configuration.

To use RMSRES as a supervisor mode library, use the following sequence of commands:

```
TKB>command string
TKB>/
ENTER OPTIONS:
TKB>RESSUP=RMS$:RMSRES/SV:0
TKB>//
```

See the RSTS/E Task Builder Reference Manual for more information on the command string.

The root of the task must include the following modules:

```
LB:RMSLIB/LB:ROEXSY:ROAUTS:ROIMPA
```

This can be done either with the task builder command string or an ODL file.

If the task requires global definitions of the user-visible RMS-11 symbols, the following should also be included:

```
LB:RMSLIB/LB:RMSSYM
```

To include remote access (DAP) support while also using RMSRES as a supervisor mode library, several options are available. The module ROAUTS is used for task resident DAP support. The module ROAULS is needed for resident library DAP support, which also requires DAPRES to be specified as a LIBR or CLSTR option in the task builder command sequence. The module ROAUOS is needed for overlaid DAP support. The following example includes DAP using the resident library support:

```
LB:RMSDAP/LB:ROAULS
```

If inconsistencies are found in the modules at execution time, a BPT trap will be generated and the value 175744 (the error code ER\$LIB) will be in R0. This can happen if not all segments of the library are installed or if the version numbers of one or more segments do not match the root segment, the RMSDAP code, or the task itself.

3.3 Changes to the Task Builder Reference Manual

Add the following as page 10-9a following section 10.5:

10.5.1 CMPRT Completion Routine

You use the CMPRT option to identify a global location within a supervisor mode library that is to be used to execute a return to user mode when each call is complete. The CMPRT option requires an argument that specifies the entry point of the completion routine in the library. The completion routine switches the processor back from supervisor mode to user mode and returns program control to the user task after the supervisor mode library subroutine that was called from the user task has executed. The completion routine is invoked by the RTS PC at the end of the supervisor mode subroutine.

The following completion routines are available in the system library:

- \$CMPCS restores only the carry bit in the user mode PSW
- \$CMPAL restores all the condition code bits in the user mode PSW

These routines perform all the necessary overhead to switch the processor from supervisor mode to user mode and return program control to the user task at the instruction following the call to supervisor mode library subroutine.

Although you can write your own completion routines, it is best to use either \$CMPCS or \$CMPAL whenever possible. Chapter 12 discusses these completion routines in detail.

Syntax:

CMPRT=name

where:

name A one- to six- character name identifying the completion routine global entry point.

Default

none

Add the following as page 10-27a following section 10.21:

10.21.1 RESSUP Resident Supervisor Mode Library

You use the RESSUP option to declare that your task intends to access a user owned supervisor mode library. The term "user owned" means that the library and the symbol definition file associated with it can reside in any directory that you choose. You can specify the directory along with the other portions of the file specification. You must not place comments on the line with RESSUP.

Syntax:

```
RESSUP=filespec/[-]access-code[:apr]
```

where:

filespec	is the specification identifying the supervisor mode library. The Task Builder expects to find a symbol table file and a task image file with the same file name (filename.STB and filename.TSK) on the device and account specified. You must omit the file type from the specification.
[-]	indicates whether TKB includes mode switching vectors within the user task. If you specify /SV or /SW, TKB includes a 4-word mode switching vector within the address space of the user task for each call to a supervisor mode library subroutine. If you specify /-SV or /-SW, you must provide your own mode switching vectors. This is useful if your library contains threaded code. We recommend however, to use system supplied vectors whenever possible.
access-code	is the code SV (for read-only) or SW (for read/write), indicating the type of access required by your program.
apr	An integer in the range 0 to 7 that specifies the first supervisor Active Page Register (APR) that you want TKB to reserve for this supervisor mode library. You can specify an APR only for position independent supervisor mode libraries. The default is the lowest available APR. One supervisor mode library is required to be at virtual 0 (ie /SV:0) and must have the CSM (change supervisor mode) dispatcher present together with the completion routines as described in chapter 12. Most uses would be /SV:0.

Add the following as page 10-28a following section 10.22:

10.22.1 SUPLIB Resident Supervisor Mode Library

Use the SUPLIB option to declare that your task intends to access a system wide supervisor mode library. The term "system wide" means that the Task Builder expects to find the supervisor mode library and the symbol definition file associated with it in the system library account (LB:).

Syntax:

```
SUPLIB=name:[-]access-code[:apr]
```

where:

name is the 1- to 6-character name specifying the supervisor mode library. The Task Builder expects to find a symbol table file and a task image file of the same name (name.STB and name.TSK) on the device and under the account specified by the system logical LB:.

If the files do not exist on LB:, you must use the RESSUP option.

[-] indicates whether TKB includes mode switching vectors within the user task. If you specify :SV or :SW, TKB includes a 4-word mode switching vector within the address space of the user task for each call to a supervisor mode library subroutine. If you specify :-SV or :-SW, you must provide your own mode switching vectors. Providing your own mode switching vectors is useful if your library contains threaded code. We recommend however, to use system supplied vectors whenever possible.

access-code is the code SV (for read-only) or SW (for read/write), indicating the type of access required by your program.

apr An integer in the range 0 to 7 that specifies the first supervisor Active Page Register (APR) that you want TKB to reserve for this supervisor mode library. You can specify an APR only for position independent supervisor mode libraries. The default is the lowest available APR.

One supervisor mode library is required to be at virtual 0 (ie :SV:0) and must have the CSM (change supervisor mode) dispatcher present together with the completion routines as described in chapter 12.

Most uses would be :SV:0.

Add the following information as a new chapter 12:

Chapter 12

Supervisor Mode Libraries

A supervisor mode library is a resident library that doubles a user task's virtual address space by mapping the instruction space of the processor's supervisor mode.

A call from within a user task to a subroutine within a supervisor mode library causes the processor to switch from user mode to supervisor mode. The user task transfers control to a mode-switching vector that the Task Builder includes within the task. The mode-switching vector performs the mode switch and then transfers control to the called subroutine within the supervisor mode library. The library routine executes with the processor in supervisor mode. When the library routine finishes executing, it transfers control to a completion routine within the library. The completion routine mode switches the processor back to user mode. The user task continues executing with the processor in user mode at the return address on the stack. This process recurs whenever the user task calls a subroutine in the supervisor mode library.

12.1 Mode-Switching Vectors

In a task that links to a supervisor mode library, TKB includes a 4-word, mode-switching vector in the user task's address space for each entry point referred to in a subroutine in the library.

The following example shows the contents of a mode-switching vector:

```
MOV #COMPLETION-ROUTINE,-(SP)
CSM #SUPERVISOR-MODE-ROUTINE ADDRESS
```

Note

When switching from user mode to supervisor mode, all registers of the referencing task are preserved. All condition codes in the PSW are saved on the stack before they are cleared and must be restored by the completion routine.

12.2 Completion Routines

After the subroutine finishes executing, its RETURN statement transfers control to a completion routine that switches from supervisor mode to user mode. The completion routine returns program control back to the referencing task at the instruction after the call to the subroutine. The system library (SYSLIB) contains two completion routines, as follows:

```
$CMPCS restores only the carry bit in the user-mode PSW.
$CMPAL restores all condition code bits in the user-mode PSW.
```

12.3 Restrictions on the Contents of Supervisor Mode Libraries

The following restrictions are placed on the contents of a supervisor mode library:

- Only subroutines using the form JSR PC, x should be used within the library.
- The library must not contain subroutines that use the stack pointer (R6) to pass parameters.
- Data maybe placed on the stack prior to calling a supervisor routine as long as the data is pointed to by a register other than R6 and that the calling parameters are removed from the stack only after return to user mode.
- No asynchronous I/O calls (.WRITA/.READA) can be made from supervisor mode.

- If both the library and the referencing task link to a subroutine from the system library, then the entry point name of the subroutine must be excluded from the STB file for the library.
- The library must not contain data of any kind (even read-only) because the user supervisor D-space APRs map the user task. This includes user data, buffers, I/O status blocks, and Directive Parameter Blocks (only the \$\$ directive form can be used, because the DPB for this form is pushed onto the user stack at run time).

12.4 Supervisor Mode Library Mapping

Supervisor mode libraries are mapped with the supervisor I-space APRs. Supervisor D space APRs map either the user I space in non-I & D tasks or the user D space in tasks with separate I & D spaces. An example of non-I & D task mapping is shown in figure 12-1.

12.4.1 Supervisor Mode Library Data

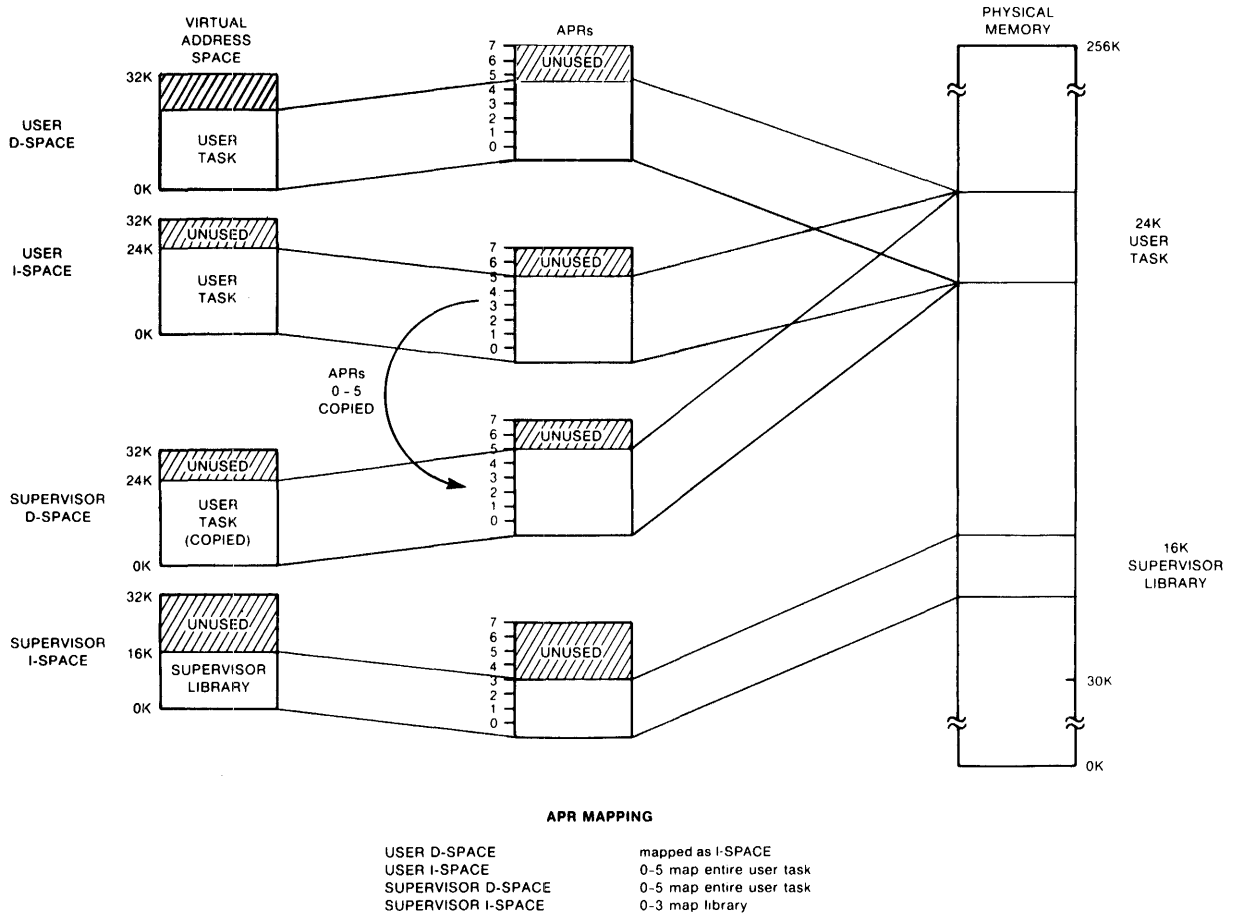
The supervisor D-space APRs always over map the user D- space APRs (for programs built using /ID). The supervisor D- space APRs over map the user I-space APRs (for programs that are built with /-ID).

12.4.2 Supervisor Mode Libraries with I- and D-Space Tasks

I- and D-space tasks may link to supervisor mode libraries. Instead of mapping to the entire user task, the supervisor mode library's D-space APRs map the task's data space. Because the I- and D-space task maps its data with the D-space APRs, the task's D-space APRs are copied into the supervisor mode library's D-space APRs. Therefore, the supervisor mode library maps its own instructions with supervisor mode I-space APRs and maps the task's data with supervisor mode D-space APRs.

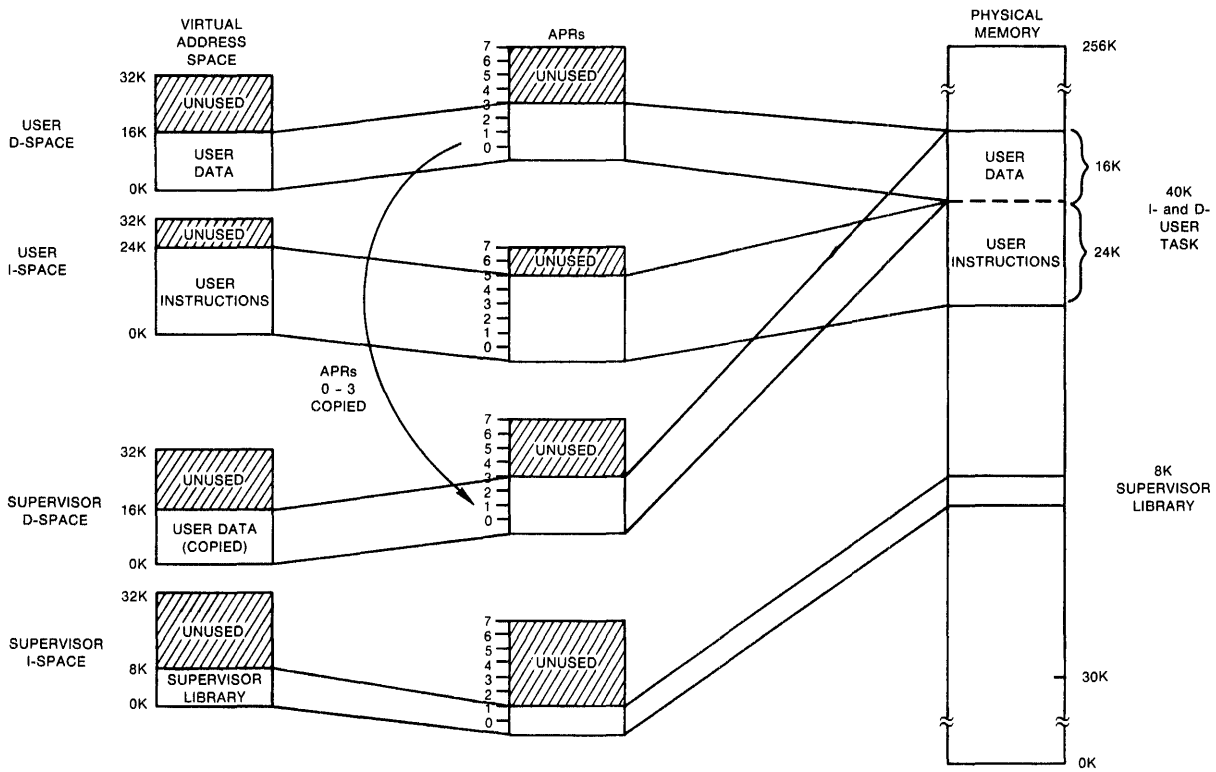
Figure 12-2 illustrates the mapping of an I- and D-space task linked to a supervisor mode library.

FIGURE 12-1. Mapping of a 24K Word Conventional User Task That Links to a 16K Word Supervisor Mode Library



ZK-439-81

FIGURE 12-2. Mapping of a 40K Word I and D Space Task That Links to an 8K Word Supervisor Mode Library



APR MAPPING

- USER D-SPACE 0-3 map user data
- USER I-SPACE 0-5 map user instructions
- SUPERVISOR D-SPACE 0-3 map user data
- SUPERVISOR I-SPACE 0-1 map library

Figure 8-3: Mapping of a 40K I- and D-Space Task that Links to an 8K Supervisor-Mode Library

ZK-1105-82

12.5 Building and Linking to Supervisor Mode Libraries

Building and linking to a supervisor mode library is essentially the same as building and linking to a conventional resident library (discussed in Chapter 7). When you build a supervisor mode library using the TKB command line, you suppress the header by attaching `/-HD` to the task image file. During option input, you suppress the stack area by specifying `STACK=0`. You specify the partition in which the library is to reside and, optionally, the base address and length of the library with the `PAR` option.

12.5.1 Relevant TKB Options

Use the following options to build and reference supervisor mode libraries:

<code>CMVRT</code>	Indicates that you are building a supervisor mode library and specifies the name of the completion routine
<code>RESSUP (SUPLIB)</code>	Indicates that your task references a supervisor mode library
<code>GBLXCL</code>	Excludes a global symbol from the STB file of the supervisor mode library

These options are discussed briefly in the next sections and are fully documented in Chapter 10.

12.5.2 Building the Library

You indicate to TKB that you are building a supervisor mode library with the `CMVRT` option. The argument for this option identifies the entry symbol of the completion routine. When TKB processes this option, it places the completion routine entry point in the library's STB file. To exclude a global symbol from the library's STB file, you specify the name of the global symbol as the argument of the `GBLXCL` option. You must exclude from the STB file of a supervisor mode library any symbol defined in the library that represents the following:

- An entry point to a subroutine that uses the stack pointer (R6) to pass parameters
- An entry point to a subroutine mapped in user mode that the referencing user task calls and also exists in the supervisor library

12.5.3 Building the Referencing Task

When you build a task that references a supervisor mode library, use the `RESSUP` option if you are referencing a user-owned, supervisor mode library, and `SUPLIB` if you are referencing a system-owned, supervisor mode library. (As with the `RESLIB` and `LIBR` options for linking to conventional libraries, `RESSUP` and `SUPLIB` are functionally the same.) The arguments for these options are as follows:

- The file specification (`RESSUP` option) or name (`SUPLIB` option) of the library to be referenced
- A switch that tells TKB whether to use system-supplied vectors to switch from user mode to supervisor mode.
- For position-independent libraries, the first available supervisor mode I-space APR that you want to map the library.

12.5.4 Mode-Switching Instruction

Mode switching occurs with a hardware instruction available with all processors that support the CSM instruction. Processors that do not support the hardware CSM instruction but do support separate I- & D-spaces (ie the PDP-11/70) have the CSM instruction emulated by RSTS/E. Throughout the remainder of the chapter, supervisor mode libraries are referred to as CSM (change supervisor mode) libraries.

12.6 CSM Libraries

This section discusses how you build and link to CSM libraries. It also shows an extended example of building and linking to a CSM library and explains the context-switching vectors and completion routines for CSM libraries.

12.6.1 Building a CSM Library

You can indicate to the Task Builder that you are building a CSM library by specifying the name of the completion routine as the argument for the CMPRT option. This option places the name of the completion routine into the library's STB file. Link the completion routine, either \$CMPAL or \$CMPCS, located in LB:SYSLIB.OLB, as the first input file. Although the completion routines are located in the system library (which is ordinarily referenced by default), you must explicitly indicate it and link it as the first input file. You must also specify in the PAR option a base of 0 for the partition in which the library will reside. These two steps locate the completion routine at virtual 0 of the library's virtual address space. This placement is a requirement.

Specify the name of any global symbols that you would like to exclude from the library's STB file as the argument to the GBLXCL option. You must exclude from the STB file of a supervisor mode library any symbol defined in the library that represents the following:

- An entry point to a subroutine that uses the stack pointer to address parameters
- An entry point to a subroutine mapped in user mode that the referencing user task calls and also exists in the CSM library.

The following sample TKB command sequence builds a CSM library called SUPER in directory [30,55] on device SY:

```
TKB>SUPER/-HD/LI/PI,SUPER/MA,SUPER=  
TKB>LB:SYSLIB/LB:CMPCAL,SY:[30,55]SUPER  
TKB>/  
Enter Options:  
TKB>STACK=0  
TKB>PAR=GEN:0:2000  
TKB>CMPRT=$CMPCS  
TKB>GBLXCL=$SAVAL  
TKB>//  
>
```

RSTS/E requires the output of the Task Builder to be converted to a loadable library by the MAKSIL program (Please refer to the RSTS/E Programmer's Utilities Manual for additional information on MAKSIL). An example of MAKSIL is shown here:

```
$ run $maksil  
MAKSIL V9.5-04 RSTS V9.7-08  
Resident Library name? SUPER  
Task-built Resident Library input file <SUPER.TSK>?  
Include symbol table (Yes/No) <Yes>?  
Symbol table input file <SUPER.STB>?  
Resident Library output file <SUPER.LIB>?  
SUPER built in 1 K-words, 21 symbols in the directory  
SUPER.TSK renamed to SUPER.TSK<40>
```

The library is built without a header or stack, like all shared regions. It is position-independent and has only one program section named .ABS. The /LI switch in TKB accomplishes this, eliminating program section name conflicts between the library and the referencing task.

The completion routine module CMPAL is specified first in the input line. The library will run in partition GEN at 0 and is not more than 1K words. These are two aspects of building supervisor mode libraries specific to CSM libraries: the completion routine must be linked first and must reside at virtual 0. (Why the CSM library must reside at virtual 0 is discussed in Section 12.6.2.)

The CMPRT option specifies the global symbol \$CMPCS, which is the entry point of the completion routine. Note that the name for the system library module is CMPCS and its corresponding global symbol is \$CMPCS.

The GBLXCL option excludes \$SAVAL from the library's STB file because the user task must reference a copy of \$SAVAL that is mapped with user mode APRs.

12.6.2 Linking to a CSM Library

If your task links to a user-owned CSM library, use the RESSUP option. If your task links to a system-owned CSM library, use the SUPLIB option. These options tell TKB that the task will link to a supervisor mode library. The option takes up to three arguments, as follows:

- The file specification (RESSUP option) or name (SUPLIB option) of the library
- A switch that tells TKB whether to use system-supplied, mode-switching vectors
- A switch that determines whether the library is to be attached read/write or read-only
- For position-independent libraries, an APR that must be APR 0 so that the library's completion routine is mapped at virtual 0

This information enables TKB to find the STB file for the CSM library, include a 4-word mode-switching vector within the user task for each call to a subroutine within the library, and correctly map the library at virtual 0 in the library image.

The following examples of TKB command sequences build a task named REF, which references the library SUPER that you built in the previous section:

```
TKB>REF,REF=REF
TKB>/
Enter Options:
TKB>RESSUP=SUPER/SV:0
TKB>//
>
```

This sequence tells TKB to include in the logical address space of REF a user-owned, supervisor mode library named SUPER. TKB includes a 4-word mode-switching vector within the user task for each call to a subroutine within the library. The CSM library is position-independent and is mapped with APR 0.

```
TKB>REF/ID/DA,REF=REF
TKB>/
Enter Options:
TKB>RESSUP=SUPER/SW:0
TKB>//
>
```

This sequence does the same function except that it also includes ODT for use in debugging the task together with the CSM library. Note the /ID is required to select the proper ODT version that contains supervisor mode commands. The /SW switch attaches the CSM library read/write so that breakpoints may be placed in the library. The CSM library must also be INSTALLED with read/write access for ODT to be able to make changes in it.

12.6.3 Example of a CSM Library and Building a Task

This example shows you the code and maps and the TKB command sequence for building with a CSM library. Example 12-1 shows the code for the library SUPER and Example 12-2 shows its accompanying map. Example 12-3 shows the code for the completion routine \$CMPCS that is linked into SUPER from the system library. Example 12-4 shows the code for referencing task TSUP and Example 12-5 shows its accompanying map.

Example 12-1: Code for SUPER.MAC

```

        .TITLE  SUPER
        .IDENT  /01/

SORT::
        CALL   $$SAVAL      ; SAVE ALL REGISTERS
        TST    (R5)+        ; SKIP OVER NUMBER OF ARGUMENTS
        MOV    (R5)+,R0     ; GET ADDRESS OF LIST
        MOV    (R5)+,R4     ; GET ADDRESS OF LENGTH OF LIST
        MOV    (R4),R4      ; GET LENGTH OF LIST
        BEQ    40$          ; IF NO ARGUMENTS
        MOV    R0,R5        ;
        DEC    R4           ;
10$:
        MOV    R5,R0        ; COPY
        MOV    R4,R3        ; COPY LENGTH OF LIST
20$:
        TST    (R0)+        ; MOVE POINTER TO NEXT ITEM
        CMP    (R5), (R0)   ; COMPARE ITEMS
        BLE    30$          ; IF LE IN CORRECT ORDER
        MOV    (R5),R2      ; SWAP ITEMS
        MOV    (R0), (R5)   ;
        MOV    R2, (R0)     ;
30$:
        DEC    R3           ; DECREMENT LOOP COUNT
        BGE    20$         ; IF NE LOOP
        DEC    R4           ; DECREMENT
        BLE    40$         ; IF EQ SORT COMPLETED
        TST    (R5)+        ; GET POINTER TO NEXT ITEM
        BR     10$         ; TO BE COMPARED
40$:
        RETURN

SEARCH::
        CALL   $$SAVAL      ; SAVE ALL THE REGISTERS
        CMP    #4, (R5)+    ; FOUR ARGUMENTS?
        BNE    20$         ; IF NE NO
        MOV    (R5)+,R0     ; GET ADDRESS OF NUMBER TO LOCATE
        MOV    (R5)+,R1     ; ADDRESS OF LIST SEARCHING
        MOV    (R5)+,R2     ; GET ADDRESS OF LENGTH OF LIST
        MOV    (R2),R2      ; GET LENGTH OF LIST
        BEQ    20$         ; IF NO ARGUMENTS
        MOV    (R5),R5      ; ADDRESS OF RETURNED VALUE
        MOV    R2,R3        ; COPY LENGTH

```

```

10$:
    CMP      (R0), (R1)+      ; IS THIS THE NUMBER?
    BEQ      30$              ; IF EQ YES
    BMI      20$              ; IF MI NUMBER NOT THERE
    DEC      R2                ; DECREMENT LOOP COUNT
    BNE      10$              ; IF NE NOT AT END OF LIST
20$:
    MOV      #-1, (R5)        ; END OF LIST PASS BACK ERROR
    RETURN
30$:
    SUB      R2, R3            ; NUMBER FOUND - GET INDEX INTO LIST
    INC      R3                ;
    MOV      R3, (R5)         ; RETURN INDEX
    RETURN
.END

```

Note the use of the routine \$SAVAL in both the Library SUPER.MAC and the main program TSUP.MAC in example 12-4, this double usage is the reason the global is excluded from the library's STB with the GBLXCL option.

Example 12-2: Memory Allocation Map for SUPER

SUPER.TSK Memory allocation map TKB M43.00 Page 1
11-AUG-88 15:41

Partition name : GEN
Identification : 03.01
Task UIC : [30,55]
Task attributes: -HD,PI
Total address windows: 1.
Task image size : 128. words
Task address limits: 000000 000343
R-W disk blk limits: 000002 000002 000001 00001.

Root segment: CMPAL

R/W mem limits: 000000 000341 000342 00226.
Disk blk limits: 000002 000002 000001 00001.

Memory allocation synopsis:

Section	Title	Ident	File
.BLK.: (RW, I, LCL, REL, CON)	000000 000342 00226.		
	000000 000140 00096. CMPAL	03.01	SYSLIB.OLB
	000140 000140 00096. SUPER	01	SUPER.OBJ
	000300 000042 00034. SAVAL	00	SYSLIB.OLB

Global symbols:

SEARCH 000220-R SORT 000140-R \$CMPAL 000022-R \$CMPCS 000110-R
\$SAVAL 000300-R \$SRTI 000002-R

Task builder statistics:

Total work file references: 300.
Work file reads: 0.
Work file writes: 0.
Size of core pool: 6466. words (25. pages)
Size of work file: 1024. words (4. pages)
Elapsed time:00:00:08

Example 12-3: Completion Routine \$CMPCS from SYSLIB.OLB

```
.TITLE  CMPAL
.IDENT  /0204/

;
;          COPYRIGHT (c) 1987,1989 BY
;          DIGITAL EQUIPMENT CORPORATION, MAYNARD
;          MASSACHUSETTS.  ALL RIGHTS RESERVED.
;
; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED
; AND COPIED ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE
; AND WITH THE INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS
; SOFTWARE, OR ANY OTHER COPIES THEREOF, MAY NOT BE PROVIDED OR
; OTHERWISE MADE AVAILABLE TO ANY OTHER PERSON.  NO TITLE TO AND
; OWNERSHIP OF THE SOFTWARE IS HEREBY TRANSFERRED.
;
; THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE WITHOUT
; NOTICE AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL
; EQUIPMENT CORPORATION.
;
; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF
; ITS SOFTWARE ON EQUIPMENT THAT IS NOT SUPPLIED BY DIGITAL.
;
;          .ENABL  LC
;
;
; This module supports the "new" transfer vector format generated
; by the Task Builder for entering super mode libraries.  This
; format is optimized for speed and size and supports user data
; space tasks.
;
; The CSM dispatcher routine and the standard completion routines
; $CMPAL and $CMPCS are included in this module due to the close
; interaction between them.
;
;
; **--CSM Dispatcher-Dispatch CSM entry
;
; This module must be linked at virtual zero in the supervisor mode
; library.  It is entered via a four word transfer vector of the
; form:
;
;          MOV      #completion-routine,-(SP)
;          CSM      #routine
;
; Note: Immediate mode emulation of the CSM instruction is required
;       in the Executive for 11/70s.
;
; The CSM instruction transfers control to the address contained in
; supervisor mode virtual 10.  At this point the stack is the
; following:
;
;          (SP)  Routine address
;          2(SP) PC (past end of transfer vector)
;          4(SP) PSW with condition codes cleared
;          6(SP) Completion-routine address
;          10(SP) Return address
;
; A routine address of 0 is special-cased to support return to
; supervisor mode from a user mode debugging aid (ODT).  In this
; case stack is the following:
;
;          (SP)  zero
;          2(SP) PC from CSM to be discarded
;          4(SP) PSW from CSM to be discarded
;          6(SP) Super mode PC supplied by debugger
;          10(SP) Super mode PSW supplied by debugger
;
;
```

```

; To allow positioning at virtual zero, this code must be in the
; blank PSECT which is first in the TKB's PSECT ordering.

.PSECT
.ENABL LSB

; Debugger return to super mode entry. Must start at virtual zero
CMP      (SP)+, (SP)+      ; Clean off PSW and PC from CSM

;
; **-$SRTI-SUPER mode RTI
;
; This entry point performs the necessary stack management to
; allow an RTI from super mode to either super mode or user mode.
; In this case, the stack is the following:
;
;      (SP)   Super mode PC
;      2(SP)  Super mode PSW
$SRTI:: TST      2(SP)      ; Returning to user mode?
BR       70$           ; Join common code
; CSM transfer address, this word must be at virtual 10 in super
; mode

.WORD    CSMSVR        ; CSM dispatcher entry

; Dispatch CSM entry
CSMSVR: MOV      6(SP), 2(SP) ; Set completion routine address for RETURN
JMP      @(SP)+        ; Transfer to super mode library routine

;
; **-$CMPAL-Completion routine which sets up NZVC in the PSW
;
; Copy all condition codes to stacked PSW. Current stack:
;
;      (SP)   PSW with condition codes cleared
;      2(SP)  Completion routine address (to be discarded)
;      4(SP)  Return address
;
$CMPAL::BPL      40$           ;
BNE      20$           ;
BVC      10$           ;
BIS      #16, (SP)      ; Set NZV
BR       $CMPCS        ;
10$:  BIS      #14, (SP)      ; Set NZ
BR       $CMPCS        ;
20$:  BVC      30$           ;
BIS      #12, (SP)      ; Set NV
BR       $CMPCS        ;
30$:  BIS      #10, (SP)     ; Set N
BR       $CMPCS        ;
40$:  BNE      60$           ;
BVC      50$           ;
BIS      #6, (SP)       ; Set ZV
BR       $CMPCS        ;
50$:  BIS      #4, (SP)      ; Set Z
BR       $CMPCS        ;
60$:  BVC      $CMPCS        ;
BIS      #2, (SP)       ; Set V

;
; **-$CMPCS-Completion routine which sets up only C in the PSW
;
; Copy only carry to stacked PSW. Current stack:
;
;      (SP)   PSW with condition codes cleared
;      2(SP)  Completion routine address (to be discarded)
;      4(SP)  Return address
;

```

```

$CMPCS::ADC      (SP)                ; Set up carry
                MOV      4(SP),2(SP)  ; Set up return address for RTT
                MOV      (SP)+,2(SP)  ; And PSW. Returning to super mode?
70$:            BPL      80$           ; If PL yes
                MOV      #6,-(SP)     ; Number of bytes for (SP), PSW, and PC
                ADD      SP,(SP)      ; Compute clean stack value
                MTPI     SP           ; Set up previous stack pointer
80$:            RTT      RTT          ; Return to previous mode and caller
                .DSABL   LSB
                .END

```

Example 12-4: Code for TSUP.MAC

```

                .TITLE   TSUP
                .IDENT   /O1/
                .MCALL   QIOW$,DIR$,QIOW$$
WRITE:          QIOW$    IO.WVB,5,1,,,,<OUT,,40>
READIN:         QIOW$    IO.RVB,5,1,,,,<OUT,5>
IARRAY:         .BLKW   12.
LEN:            .BLKW   1
IART:           .BLKW   1
INDEX:          .WORD   0
OUT:            .BLKW   100.
ARGBLK:         .BLKW   10.
FMT1:           .ASCIZ   /%2SARRAY(%D)=/
FMT2:           .ASCIZ   /%N%2SNUMBER TO SEARCH FOR?/
FMT3:           .ASCIZ   /%N%2S%D WAS FOUND IN ARRAY(%D)/
FMT4:           .ASCIZ   /%N%2S%D WAS NOT IN ARRAY/
FMT5:           .ASCIZ   /%2SARRAY(%D)=%D/
                .EVEN

START:          MOV      #IARRAY,R0    ; GET ADDRESS OF ARRAY
                MOV      #10,R1        ; SET LENGTH OF ARRAY
5$:             CLR      (R0)+         ; INITIALIZE ARRAY
                DEC      R1            ; LOOP
                BNE     5$
                MOV      #IARRAY,R0   ;
                MOV      #INDEX,R2

10$:            MOV      #FMT1,R1      ; FORMAT SPECIFICATION (ADDRESS
                ; OF INPUT STRING)
                MOV      (R2),EDBUF    ; GET INDEX
                INC      EDBUF         ;
                CALL     PRINT         ; PRINT MESSAGE
                CALL     READ         ; READ INPUT
                MOV      IART,(R0)+    ; PUT BINARY KEYBOARD INPUT INTO ARRAY
                BEQ     20$           ; ZERO MARKS END OF INPUT
                INC      (R2)         ;
                CMP      (R2),#10.    ;
                BNE     10$           ; IF NE YES
20$:            MOV      (R2),LEN      ; CALCULATE LENGTH OF ARRAY
                MOV      #ARGBLK,R5    ; GET ADDRESS OF ARGUMENT BLOCK
                MOV      #2,(R5)+     ; NUMBER OF ARGUMENTS
                MOV      #IARRAY,(R5)+ ; PUT ADDRESS OF ARRAY
                MOV      #LEN,(R5)    ;
                MOV      #ARGBLK,R5   ;
                CALL     SORT         ; SORT ARRAY
;+
;Task Builder replaced call to SORT subroutine in SUPLIB with
;4-word context switching vector. Flow of control switches to SUPLIB
;via the vector and back via the completion routine $CMPCS. TSUP
;continues executing at the next instruction.
;-
                CLR      R2           ;
                MOV      #IARRAY,R0   ; GET ARRAY ADDRESS

```



```

30$:      INC      R2          ; INCREMENT INDEX
          MOV      R2,EDBUF   ; GET INDEX FOR PRINT
          MOV      (R0)+,EDBUF+2 ; GET CONTENTS OF ARRAY
          MOV      #FMT5,R1   ; GET ADDRESS OF FORMAT SPECIFICATION
          CALL     PRINT      ;
          CMP      R2,LEN     ; MORE TO PRINT?
          BLT     30$        ; IF LE YES
          MOV      #FMT2,R1   ; GET ADDRESS OF FORMAT SPECIFICATION
          CALL     PRINT      ; OUTPUT MESSAGE
          CALL     READ       ; READ RESPONSE
          MOV      #ARGBLK,R5 ;
          MOV      #4,(R5)+   ; SET NUMBER OF ARGUMENTS
          MOV      #IART,(R5)+ ; SET ADDRESS OF NUMBER LOOKING FOR
          MOV      #IARRAY,(R5)+ ; SET ADDRESS OF ARRAY
          MOV      #LEN,(R5)+ ; SET ADDRESS OF LEN OF ARRAY
          MOV      #INDEX,(R5) ; ADDRESS OF RESULT
          MOV      #ARGBLK,R5 ;
          CALL     SEARCH     ; SEARCH FOR NUMBER IN IART
;
;Call to SUPLIB for SEARCH subroutine.
;
          TST     INDEX       ; WAS NUMBER FOUND?
          BLT     40$        ; IF LT NO
          MOV     IART,EDBUF   ; GET NUMBER LOOKING FOR
          MOV     INDEX,EDBUF+2 ; GET ARRAY NUMBER
          MOV     #FMT3,R1    ; GET FORMAT ADDRESS
          CALL    PRINT       ;
          BR     100$        ; DONE
40$:      MOV     #FMT4,R1    ; GET FORMAT ADDRESS
          MOV     IART,EDBUF   ; GET NUMBER
          CALL    PRINT
100$:     CALL    $EXST       ; EXIT WITH STATUS
PRINT:    CALL    $$SAVAL     ; SAVE ALL REGISTERS
          MOV     #OUT,R0     ; ADDRESS OF OUTPUT BLOCK
          MOV     #EDBUF,R2   ; START ADDRESS OF ARGUMENT BLOCK
          CALL    $EDMSG      ; FORMAT MESSAGE
          MOV     R1,WRITE+Q.IOPL+2 ; PUT LENGTH OF OUTPUT
          ; BLOCK INTO PARAMETER BLOCK
          DIR$   #WRITE      ; WRITE OUTPUT BLOCK
          RETURN
READ:     CALL    $$SAVAL     ; SAVE ALL REGISTERS
          DIR$   #READIN     ; READ REQUEST
          MOV     #OUT,R0     ; GET KEYBOARD INPUT
          CALL    $CDTB       ; CONVERT KEYBOARD INPUT TO BINARY
          MOV     R1,IART     ; PUT INPUT INTO BUFFER
          RETURN
          .END    START

```

TSUP prompts you to enter numbers at your terminal. It calls a subroutine in SUPER to sort the numbers. Then it displays the numbers you entered as array entries and prompts you to request a number to search for. TSUP calls a subroutine in SUPER.LIB to search for the number. Finally, TSUP indicates at your terminal either that the number was not found or the array location in which the number is stored.

Example 12-5: Memory Allocation Map for TSUP

TSUP.TSK Memory allocation map TKB M43.00 Page 1
11-AUG-88 15:41

Partition name : GEN
Identification : 01
Task UIC : [30,55]
Stack limits: 000274 001273 001000 00512.
PRG xfr address: 002130
Total address windows: 2.
Task image size : 1344. words
Task address limits: 000000 005133
R-W disk blk limits: 000002 000007 000006 00006.
*** Root segment: TSUP
R/W mem limits: 000000 005133 005134 02652.
Disk blk limits: 000002 000007 000006 00006.

Memory allocation synopsis:

Section	Title	Ident	File
-----	-----	-----	-----
. BLK.: (RW, I, LCL, REL, CON) 001274 002334 01244.			
	001274 001234 00668.	TSUP 01	TSUP.OBJ
CMPAL : (RW, I, LCL, REL, CON) 000000 000474 00316.			
PUR\$D : (RO, I, LCL, REL, CON) 003630 000076 00062.			
PUR\$I : (RO, I, LCL, REL, CON) 003726 000752 00490.			
\$\$RESL: (RO, I, LCL, REL, CON) 004700 000212 00138.			
\$\$SLVC: (RO, I, LCL, REL, CON) 005112 000020 00016.			

TSUP.TSK;1 Memory allocation map TKB M43.00 Page 2
11-AUG-88 15:41

*** Task builder statistics:

Total work file references: 2477.
Work file reads: 0.
Work file writes: 0.
Size of core pool: 6988. words (27. pages)
Size of work file: 1024. words (4. pages)
Elapsed time:00:00:05

12.6.3.1 Building the Library SUPER

To build SUPER in directory [30,55] on device SY:, use the following TKB command sequence:

```
TKB>SUPER/-HD/LI/PI, SUPER/MA, SUPER=  
TKB>LB:SYSLIB/LB:CMPAL, SY:[30,55]SUPER  
TKB>/  
Enter Options:  
TKB>STACK=0  
TKB>PAR=GEN:0:2000  
TKB>CMPRT=$CMPCS  
TKB>GBLXCL=$SAVAL  
TKB>//  
>  
  
$ run $maksil  
MAKSIL V9.5-04 RSTS V9.7-03  
Resident Library name? SUPER  
Task-built Resident Library input file <SUPER.TSK?>  
Include symbol table (Yes/No) <Yes?>  
Symbol table input file <SUPER.STB?>  
Resident Library output file <SUPER.LIB?>  
SUPER built in 1 K-words, 21 symbols in the directory  
SUPER.TSK renamed to SUPER.TSK<40>
```

SUPER is built without a header or stack. It is position-independent and has only one program section, named .BLK. The /LI switch eliminates program section name conflicts between the library and the referencing task.

The completion routine module CMPAL is specified first in the input line. The library will run in partition GEN at 0 and is not more than 1K words.

The GBLXCL option excludes \$SAVAL from the library's STB file. Exclude \$SAVAL from the STB file because the referencing task, TSUP, also calls \$SAVAL. If TSUP finds \$SAVAL in the STB file of SUPER, it will not link a separate copy of \$SAVAL into its task image from the system library. If TSUP could not link to a copy of \$SAVAL that is mapped through user APRs, TSUP would call \$SAVAL as a subroutine residing within the supervisor mode library but without the necessary mode-switching vector and completion routine support. This option forces TKB to link \$SAVAL from the system library into the task image for TSUP.

The memory allocation map in Example 12-2 shows the following information:

- SUPER begins at virtual 0.
- The completion routine, \$CMPAL, is linked into the library from the system library at virtual 0.
- The entry point \$CMPAL is located at virtual 22, SEARCH is located at 220, and SORT is located at 140. All of these entry points are relocatable.

12.6.3.2 Building TSUP

Use the following TKB command sequence to build a task, TSUP, that links to SUPER:

```
TKB>TSUP,TSUP=TSUP
TKB>/
Enter Options:
TKB>RESSUP=SUPER/SV:0
TKB>//
>
```

This command sequence tells TKB to include in the logical address space of TSUP a user-owned supervisor mode library named SUPER. TKB includes a 4-word mode-switching vector within the task image for each call to a subroutine within the library. The library is position-independent and is mapped with supervisor I-space APR 0. This is a requirement for CSM libraries because the CSM library expects to find the entry point of the completion routine at location 10.

The memory allocation map for TSUP (Example 12-5) shows the following information:

- \$CMPAL is linked from the STB file of the library and begins at location 0.
- The mode-switching vectors begin at 5112 and are 16 bytes in length. This means that TSUP calls subroutines within the library two times (four words for each vector).
- The initiation routine \$SUPL is located at 4700.
- The SEARCH and SORT subroutines that were located at virtual 220 and 140, respectively, in the virtual address space of SUPER have been relocated to the mode-switching vectors residing at 5112 and 5122, respectively, in TSUP.
- The system library module SAVAL, containing \$SAVAL, has been linked into the task image instead of including \$SAVAL from the library's STB file.

12.6.3.3 Running TSUP

After building SUPER and TSUP as indicated in the task-build command sequence discussed previously, install the library SUPER and run TSUP. TSUP prompts you for the position in which to store the number in the array:

```
ARRAY (x)
x
```

Enter a number. TSUP stores the number in the array and prompts you again for a number. This continues until you have entered a 0, an invalid number, or 10 numbers. Then TSUP calls the SORT routine in SUPER.

When you enter a number, TSUP calls the SEARCH routine in SUPER. TSUP then outputs a message indicating whether the number was in the array.

12.6.4 The CSM Library Dispatching Process

When you build the referencing task and specify the SV or SW argument to the RESSUP or SUPLIB option, TKB includes a 4-word context-switching vector for each call to a subroutine in the library. This has been very generally discussed in Section 12.2. This section discusses the CSM library vector in more detail.

CSM mode switching occurs as follows:

- The vector is entered with the return address on top of the stack (TOS).
- The vector pushes the completion-routine address on the stack.
- A CSM instruction is executed with the supervisor mode entry point as the immediate addressing mode parameter.

The CSM instruction executes the following steps either through hardware action or software emulation:

- Evaluating the source parameter and storing the entry point address in a temporary register
- Copying the user stack pointer to the supervisor stack pointer
- Placing the current PSW and PC on the supervisor stack, clearing the condition codes in the PSW
- Pushing the entry point address on the supervisor stack
- Placing the contents of location 10 in supervisor I-space into the PC and transferring execution to that address in supervisor mode

The stack looks like this when the processor begins to execute at the contents of virtual 10 in supervisor mode:

```
user sp ----> return address
               completion routine address
               PSW
               PC
super sp ---> entry point address
```

Because the CSM library mode-switching vector processor begins executing at the contents of virtual 10 in supervisor mode, the completion routine must be located at virtual 0. In this way, virtual location 10 is within properly mapped memory.

12.7 Passing Parameters Using Stack Space

Note also the existence of the two independent stack pointers: (NOT stacks) one for user mode and one for supervisor mode. The fact that a variable amount of additional data is placed on one stack and not the other by the system is why the SP cannot be used as a pointer to parameters. The following example shows a method of properly using the stack to pass parameters to and from a CSM library routine:

In the user mode program:

```
.
.
MOV   R0,-(SP)    ;make a register available
MOV   DATA1,-(SP) ;parameters needed by CSM library routine
MOV   DATA2,-(SP) ;placed on stack as a packet
MOV   SP,R0      ;put address of packet in another register
JSR   PC,SUPER_ROUTINE ;transfer to CSM library routine
                                ;on return to user mode
ADD   #4,SP      ;remove data packet from user stack
MOV   (SP)+,R0   ;restore register
.
.
```

In the CSM library code:

```
SUPER_ROUTINE:
      MOV   (R0),R2      ;put DATA2 in R2
      MOV   2(R0),R3     ;put DATA1 in R3
.
.
```

Data may be returned from a CSM library in the same manner as long as the data packet was reserved on the stack and addressed in the user mode prior to the call to the CSM library routine.

12.8 Using Supervisor Mode Libraries as Resident Libraries

Supervisor mode libraries can double as conventional resident libraries. For position-independent supervisor mode libraries, rebuild the referencing task using the RESLIB option instead of the RESSUP option. Indicate the first available user-mode APR that you want to map the library. For CSM libraries, this will always change because you cannot map a shared region with APR 0. You do not have to rebuild the library.

For absolute supervisor mode libraries, rebuild the referencing task using the RESLIB option instead of the RESSUP option. Rebuild the library only if the beginning partition address in the PAR option is incompatible with the address limits of your referencing task.

Once installed, the library can be used in either mode at any time. All APR relocation is done when the task that references the library is built. There is no change to the library itself.

12.9 Multiple Supervisor Mode Libraries

A user task can reference multiple supervisor mode CSM libraries. However, all the CSM libraries must use the completion routine that begins at virtual 0 in supervisor mode instruction space.

12.10 Linking Supervisor Mode Libraries

You cannot link supervisor mode libraries together, and you cannot link a supervisor mode library to a user-mode library. Calling a user-mode library is not possible because its code is not mapped through the I-space APRs while in the supervisor mode library. However, you can link user-mode libraries to a supervisor mode library.

12.11 Writing Your Own Vectors and Completion Routines

You can write your own mode-switching vectors and completion routines. This may be necessary for threaded code. If you use your own vectors, build them into the task and use the /-SV or the /-SW switch on the RESSUP or RESLIB option when you build the referencing task. If you create your own completion routines, write your completion routine to resemble the system-supplied completion routines (see Example 12-3) as much as possible. If you do not retain the last three lines of code as indicated in Example 12-3, the task may crash if the Executive processes an interrupt before the switch back to user mode has completed.

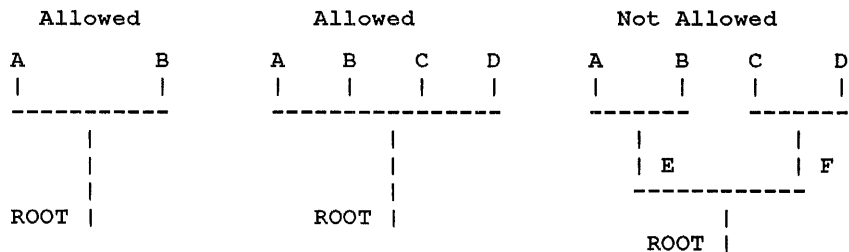
The debugger ODT expects the presence of the system-supplied completion routines to handle breakpoints and all other traps while in supervisor mode. Lack of this code may cause the task to crash should a trap occur.

12.12 Overlaid Supervisor Mode Libraries

It is possible to use overlaid supervisor mode libraries. However, the following restrictions must be noted when building these libraries:

- The completion routine for the library must be in the root.
- Only one level of overlaying is allowed (Figure 12-4).

Figure 12-4: Overlay Configuration Allowed for Supervisor Mode Libraries



12.13 USING ODT TO DEBUG CSM LIBRARY

The use of ODT with a CSM library is the same as with other types of tasks except for following:

- The CSM library must have been installed as RW and 1 user. `INSTAL/LIB/NOREAD_ONLY/NOSHAREABLE file[ppn]`
- When the task that uses the CSM library is Task built, the parameters change:
 - Indicate the inclusion of the proper ODT.OBJ file by using the /DA and /ID switches.
 - Indicate that the library should be loaded Read/Write by replacing the supervisor mode vector switch /SV:0 with the /SW:0 switch.
- Once in ODT, the Z command switches ODT to the supervisor mode and the U command switches it back to user mode.
- Only the UI, UD and ZI spaces are defined in ODT. Any attempt to use the ZD (supervisor data) space will result in the non-existent memory indicator from ODT. However, the data in supervisor D space is mapped the same as the user data space and therefore is available via the UD command.
- Breakpoints and single step work the same as in user space. There is, however, no indicator as to which space, Z or U, the breakpoint was in. It is the task of the user to determine or keep track of which space the program is in when in ODT.
- ODT operating with supervisor mode requires the use of the Digital system-supplied supervisor mode completion routines which handle the special mode switching needs of ODT.

12.14 Trap Handling With Supervisor Libraries

Although Asynchronous I/O calls (READA & WRITA) are illegal from a task that is also using supervisor mode, other forms of traps may require handling by a task that is using supervisor mode.

The two system asynchronous traps, FPP exception and Control-C interception, as well as all forms of Synchronous Service Traps (SSTs) are permitted. The trap service routines for these may be located in either user or supervisor mode.

If the service routines reside in supervisor mode, they must adhere to additional requirements that their user mode counterparts do not need.

- The routine must exit via either the SSTX\$ or ASTX\$ calls. Whereas in user mode, only service routines are allowed to clean the PC & PSW off the stack and continue without returning, tasks using supervisor mode must use an exit call. Note that the hardware prohibits an RTI instruction from returning a task to supervisor mode from a service routine in user mode.
- Routines located in supervisor space must obey all the rules of supervisor library routines such as not calling user mode routines and having no data within the code.

12.14.1 Locating Service Routines

The task informs RSTS/E in which space the service routine is at the time the respective service vectors are set up. The following calls set up the trap service vectors for the task:

FPPA\$	Floating point exception
SCCA\$	Control-C interception
SVTK\$	SST trap vector table
SVDB\$	SST debugging trap vector table

12.14.1.1 FPPA\$ and SCCA\$

In the cases of FPPA\$ and SCCA\$, the task issues a single address vector. Which space the service routine resides in is indicated by bit 0 of the vector. If it is zero, the service routine is in user space. If the bit is 1 (an odd address vector), the routine is in supervisor space.

12.14.1.2 SVTK\$ and SVDB\$

In the SVTK\$ and SVDB\$ calls, the task issues a list of vectors that will be associated with the different events (BPT etc.). The bit 0 in each of the individual service routine vector addresses indicates which mode that routine is in, but in a different way from FPPA\$ and SCCA\$.

An even vector entry causes the SST routine to be executed in the same mode (either user or supervisor) that the processor was in when the SVTK\$ or SVDB\$ call was issued. An odd vector entry causes the SST routine to be executed in the other mode. For example, if the processor was in supervisor mode when an SVTK\$ was issued and the vector was odd (bit 0 set), the SST routine is executed in the user mode. This method of designation of service routine location is the same as RSX-11/M-Plus. This method allows the individual SSTs to be different modes at the same time (BPT in user and address trap in supervisor). Bit 0, which is the flag(s), is a member of the vector list, **not** the address pointer to the vector list.

12.15 Building To a Supervisor Mode RMS library.

On the RSTS/E systems that support supervisor mode, you may choose to use RMSRES as a supervisor mode library instead of user mode. Because this configuration uses two otherwise idle supervisor mode APRs to map most of the RMS-11 code, the impact of the RMS-11 code on your user-mode virtual address space is reduced to the absolute minimum. There also may be slight performance advantages over the clustered RMS-11 configuration.

To use RMSRES as a supervisor mode library, use the following sequence of commands:

```
TKB>command string
TKB>/
ENTER OPTIONS:
TKB>RESSUP=RMS$:RMSRES/SV:0
TKB>//
```

See the RSTS/E Task Builder Reference Manual for more information on the command string.

The following modules must be included in the root of the task:

```
LB:RMSLIB/LB:ROEXSY:ROAUTS:ROIMPA
```

This can be done by either adding it to the task builder command string or by adding @LB:RMSSLX in the users's ODL file.

If the task requires global definitions of the user-visible RMS-11 symbols, the following should also be included:

```
LB:RMSLIB/LB:RMSSYM
```

To include remote access (DAP) support while also using RMSRES as a supervisor mode library, several options are available. Use the module ROAUTS for task resident DAP support. For resident library DAP support, use the module ROAULS and specify DAPRES as a LIBR or CLSTR option in the task builder command sequence. For overlaid DAP support, use the module ROAUOS. The following example includes DAP using the resident library support:

```
LB:RMSDAP/LB:ROAULS
```

This can be done either by adding it to the task builder command string or by adding @LB:DAPSLX in the users's ODL file.

If inconsistencies are found in the modules at execution time, a BPT trap will be generated and the value 175744 (the error code ER\$LIB) will be in R0. This can happen if not all segments of the library are installed or if the version numbers of one or more segments do not match the root segment, the RMSDAP code, or the task itself.

3.4 Layered Product Documentation Changes

The following sections discuss the documentation changes to the DECnet/E layered product.

3.4.1 DECnet/E V4.1 Network Installation and Update Guide

The RSTS/E V9.7 installation kit includes updates to DECnet/E V4.1. These enhancements appear mainly in the form of changes to the dialogue for the DECnet/E installation and update procedures.

To update your DECnet/E V4.1 software, first install or update the RSTS/E software to RSTS/E V9.7. Then perform the update procedure in the *DECnet/E Network Installation and Update Guide*.

Make the following changes to the *DECnet/E Network Installation and Update Guide*:

1. In Chapter 2, on page 2-5, immediately after the note in step 10, replace the last sentence with the following:

If you answer **NO** to this question, the procedure displays the following message and you go to step 12 in Chapter 3.

```
Installation canceled - proceeding with DECnet/E update
```

2. In both Chapter 2 on page 2-5 and Chapter 3 on page 3-5 after step 12, add the following:

```
Do you want to install (update) the HELP files for NCP? (YES/NO) <YES>
```

If you want to enable online **HELP** for NCP, press **[RETURN]** to accept the default of **YES**. If you don't want to install (update) the files, enter **NO**.

3. In Chapter 2 on page 2-8, step 24, add the following to the explanatory text:

That is, the **MAXIMUM CIRCUITS** parameter is set to 1.

4. In Chapter 2 on page 2-10, between steps 31 and 32, add the following steps:

- a. You see the following question only if your node has just one circuit defined and more than 256K words of memory.

```
Only one circuit is defined on this node. The node will be defined as an end node unless you enter YES in response to the following prompt
```

```
Do you want to define this node as a router? (YES/NO) <NO>
```

Only 1 circuit is defined for your node and the node would normally be defined as an end node. If you still want your node defined as a router, enter **YES**. Otherwise, press **[RETURN]** to accept the default of **NO**.

If you answer **YES**, the **MAXIMUM CIRCUITS** parameter is set to 2; otherwise the parameter is set to 1.

- b. You see the following question only if your node has more than 1 circuit defined and has more than 256K words of memory.

```
circuits-found circuits are defined on this node. The node will be defined as a router unless you enter NO in response to the following prompt
```

```
Do you want to define this node as a router? (YES/NO) <YES>
```

Circuits-found represents the actual number of circuits defined for your node. Normally, you would define your node as a router. If you still want to define your node as an end node, enter **NO**. Otherwise, press **RETURN** to accept the default of **YES**.

If you answer **NO** to this question, the **MAXIMUM CIRCUITS** parameter is set to 1; otherwise, the parameter is set to the value of *circuits-found*.

5. In Chapter 2 on page 2-16, following step 59 and in Chapter 3 on page 3-13, following step 52, place the following text at the end of the testing phase:

You can perform the standard DECnet/E verification tests outside of the installation or update procedure by following these steps:

- a. Turn off all of your communications circuits, but leave your network running. Do this using the NCP command:

```
$ NCP SET KNOWN CIRCUITS STATE OFF
```

- b. Create the test file using the command:

```
$ @DECNET$:CRTEST
```

- c. Test your network using the command:

```
$ @DECNET$:DNTEST
```

Proceed by answering the questions for the installation or update testing phase.

- d. Shut down DECnet/E using the command:

```
$ RUN DECNET$:NETOFF
```

- e. Start DECnet/E and turn on your normal circuits using the command:

```
$ @DECNET$:DSTART
```

6. In Chapter 2 on page 2-16, the current Phase 7 becomes Phase 8 and the new Phase 7 is as follows:

Phase 7: Defining Remote Nodes

During this phase, you can define the remote nodes with which your node can communicate.

- a.

```
Do you want to define any remote nodes? (YES/NO) <YES>
```

Before you can communicate with the other nodes in your network, define them in the node database. Answer **YES** to define one or more nodes in your node database, or **NO** if you do not want to define any nodes. Press **[RETURN]** to accept the default of **YES**. If you enter **NO**, go to phase 8.

- b.

```
Name of node? (Or press RETURN to quit)
```

You see this question only if you answered **YES** in step **a**. Enter a 1- to 6-character name of the remote node, or press **[RETURN]** if you don't want to define any more nodes.

- c.

```
Address of node node-name? (area.node, e.g. 1.25)
```

Node-name represents the node name you specified in step **b**. Enter the address of the remote node in the form *area.node* where *area* is a number from 1 to 63 and *node* is a number from 1 to 1023.

- d.

```
Node node-name defined at address node-address
```

node-name represents the node name you specified in step **b** and ***node-address*** represents the node address you specified in step **c**. You see this message if the procedure successfully defines the specified remote node. If the definition fails, you see the following error message and go back to step **b**.

```
%Error encountered while attempting to define node node-name  
Please try again
```

The procedure repeats steps **b** through **d** as long as you wish to continue defining remote nodes.

e.

Are you finished defining nodes for your network? (YES or NO) <YES>

You see this question if you press **[RETURN]** without entering a node name in step **b**. Enter **NO** if you want to define more remote nodes and go to step **b**, or press **[RETURN]** to accept the default of **YES** and go to phase 8.

At any time after performing the DECnet/E installation procedure, you can define additional remote nodes using the command:

```
$ @DECNET$:NODDEF
```

7. In Chapter 2 on page 2-16, make the following changes:
 - a. At step **61**, delete the note text at the beginning of the display message.
 - b. In the note at the bottom of the page, delete the first bulleted list item.
8. In Chapter 3 on page 3-7, step **21**, delete the display message and replace it with the following message:

```
%This system has less memory than the recommended minimum amount
for a routing node (minimum = 256K words of memory)
```
9. In Chapter 3 on page 3-8, between steps **25** and **26**, add the following steps:
 - a. You see the following question only if your system has less than 256K words of memory and more that one circuit is found on your system.

circuits-found circuits are defined on this node. However, due to system memory limitations, the node will be defined as an end node unless you enter YES in response to the following prompt

Do you want to define this node as a router? (YES/NO) <NO>

Circuits-found represents the actual number of circuits defined for your node. Your system has more than one circuit but has less than the recommended minimum memory to be defined as a router. If you still want to define your node as a router, enter **YES**. Otherwise, press **[RETURN]** to accept the default of **NO**.

If you answer **NO** to this question, the **MAXIMUM CIRCUITS** parameter is set to 1; otherwise, the parameter is set to the value of *circuits-found*.

- b. You see the following question only if there is one circuit defined for your node.

Only 1 circuit is defined on this node. The node will be defined as an end node unless you enter YES in response to the following prompt

Do you want to define this node as a router? (YES/NO) <NO>

There is just one circuit defined for your node and the procedure would normally define your node as an end node. If you want to define your node as a router, enter **YES**. Otherwise, press **[RETURN]** to accept the default of **NO**.

If you answer **YES** to this question, the **MAXIMUM CIRCUITS** parameter is set to 2; otherwise, the parameter is set to 1.

- c. You see the following question only if your node has more than one circuit defined and your system has more than 256K words of memory.

circuits-found circuits are defined on this node. The node will be defined as a router unless you enter NO in response to the following prompt

Do you want to define this node as a router? (YES/NO) <YES>

Circuits-found represents the actual number of circuits defined for your node. Your node has more than 1 circuit defined and has more than 256K words of memory. Normally, the procedure would define your node as a router. If you want to define your node as an end node, enter **NO**. If you want to define your node as a router, press **[RETURN]** to accept the default of **YES**.

If you answer **NO** to this question, the **MAXIMUM CIRCUITS** parameter is set to 1; otherwise, the parameter is set to the value of *circuits-found*.

10. In the Chapter 3 testing phase on page 3-11, add the following step ahead of step 44:

The volatile parameter file must be created on disk *volatile-device*

Please mount *volatile-device* and press **RETURN** when ready

Volatile-device represents the device designation of the disk on which you want to locate the volatile parameter file. You see this message only if you locate your volatile parameter file on a nonsystem disk. To proceed with the update, mount this device and press **[RETURN]**.

3.4.2 DECnet/E V4.1 Release Notes

Restriction 4.1.3 has been removed. You can now locate the volatile parameter file on a nonsystem disk during a DECnet/E update.

Known Problems and Restrictions

4.1 Installation

The following sections describe RSTS/E installation problems and restrictions.

4.1.1 Booting Magnetic Tape

You must boot magnetic tape distributions of RSTS/E or any bootable recovery tapes of RSTS/E from unit zero. This problem affects MM and MT drives. It will be corrected in a future release of RSTS/E.

4.1.2 Flag Files

The installation or update of RSTS/E leaves several zero length flag files in various system accounts after installation. (The installation procedure uses flag files to determine which parts of the installation are complete.)

Deleting any of these files may cause unpredictable results. In particular, deleting the flag file [1,2]VER097.SYS may cause certain layered product installations and updates to fail.

4.2 Initialization Code (INIT.SYS)

An unnecessary error message prints when you boot an 800 BPI magnetic tape on a TE16 subsystem with a TM02 formatter from the Option: prompt of INIT.SYS. For example:

```
MMO Error   MTCS1   MTWC    MFC     MTCS2   MTDS    MTER    MTC
            144270  000000  001007  000100  154640  102100  102300
```

The error message you receive may be different. This message prints as a result of the density autosizing process that occurs on TM02 formatters. It does not represent a true hardware error.

4.3 Monitor

The following sections describe RSTS/E monitor problems and restrictions.

4.3.1 DB and DR Disk Drivers

The DB and DR disk drivers may count or log errors for the wrong disk unit if there is heavy DB/DR disk activity when you attempt to mount a unit that is spun down or off-line.

This problem will be corrected in a future release of RSTS/E.

4.3.2 MSCP Disk Driver

On systems having multiple UDA50-A MSCP controllers, a system crash may result if disk unit numbers are swapped between two drives on different UDA50-A controllers during timesharing.

The preferred workaround is to swap drive numbers only when timesharing is not up. The best method is to shut the system down and swap the unit numbers when the Start timesharing prompt appears. After swapping the unit numbers, reboot the system disk and restart timesharing.

4.3.3 TMSCP Tape Driver

If your system contains only one TMSCP tape drive (TU81 or TK50), you must designate it unit 0. If you have two drives and unit 0 does not work, you must make the working drive unit 0 and disconnect the non-working drive. This restriction will be fixed in a future release of RSTS/E.

NOTE

The controller number must also match the drive number. Valid examples are:

Name	Address	Vector	Comments
MU0:	174500	P350	TU81 Units: 0(TU81)
MU1:	160504	P354	TK50 Units: 1(TK50)

Invalid examples are:

Name	Address	Vector	Comments
MU0:	174500	P350	TU81 Units: 1(TU81)
MU1:	160504	P354	TK50 Units: 0(TK50)

4.3.4 Monitor Overlay Handling

If the system is to have both an installed overlay file and memory loaded overlays, the overlay file must be installed before any monitor overlays are loaded. Under normal conditions, both of these steps should occur in the startup command file. This restriction will be fixed in a future release of RSTS/E.

4.4 Commonly Used System Programs (CUSPs)

The following sections describe problems and restrictions for CUSPS.

4.4.1 DCL

The following sections describe problems and restrictions with DCL commands.

4.4.1.1 DIRECTORY Command

The **DIRECTORY** command shows incorrect file sizes for magnetic tape files containing more than 65535 records. This is a restriction and will be fixed in a future release of RSTS/E.

4.4.1.2 TIME Command

DCL interprets 12:00 PM as midnight instead of noon.

4.4.1.3 F\$MID Function

The **F\$MID** function returns the ?Invalid character error message if the second parameter and closing parenthesis are missing. The correct error message is ?Additional argument required. This will be corrected in a future release of RSTS/E.

4.4.1.4 ON ERROR Handling

If an error occurs in a DCL command procedure and the cursor is not at the start of the line, the first line of the error trap routine is displayed, even if **\$SET NOVERIFY** is in effect.

4.4.1.5 MOUNT Command

When issuing a **MOUNT** command with the **/OVER** qualifier and without a pack label, the logical name, if specified, must not have a colon (:).

Use a command of the following format:

```
MOUNT DU1:/OVER/PRIVATE S
```

rather than:

```
MOUNT DU1:/OVER/PRIVATE S:
```

4.4.1.6 GOSUB Command

You cannot use continuation lines with the **GOSUB** command.

4.4.1.7 BROADCAST Command

You cannot use exclamation marks in continuation lines with the **BROADCAST** command.

4.4.2 ANALYS Package

In the monitor dump section of the ANALYS report, the virtual disk area of the Memory Layout is incorrectly shown as locked out. This will be fixed in a future release of RSTS/E.

4.4.3 LOGIN and LOGOUT

The LOGIN and LOGOUT programs control dropping of the carrier on a dialup line. If you access the system from a captive account over a dialup line and the command procedure you are running aborts abnormally, the carrier will not be dropped. To work around this problem, trap all possible errors within the command procedure and exit by using the LOGOUT command.

4.4.4 ODT

ODT.TSK does not correctly display addresses greater than 177777(8). However, ODT does correctly access and display the contents of the address location. It is only the display of the address that is incorrect.

4.4.5 PIP

The following PIP problems have been reported and will remain as restrictions while they are being investigated for resolution in a future release of RSTS/E:

- PIP cannot access files on a tape created on an RT-11 system.
- A restriction exists when copying files into themselves using the Open Next directive with PIP. If you are copying files into themselves on the system disk that was initialized as New Files Last (NFL), and you specify the disk device name differently on the input and output sides of the command line, PIP copies the files in an endless loop. Type Ctrl/C to terminate the operation. For example, if your system disk is _DU1: and you issue the following command, PIP endlessly copies files:

```
$ PIP _SY:[1,2]=_DU1:[1,2]*.*/*LOG
```

To prevent this, do not specify a device name for the system disk or specify the same device name on both the input and output sides of the PIP command line (_SY: = _SY:, _SY0: = _SY0:, _DU1: = _DU1:, and so on.).

- You can copy only RMS files to magnetic tape using the ANSI format. PIP allows you to copy RMS files to a DOS magnetic tape, but the file attributes will be lost and the files will be unusable. We will implement safeguards in future releases of RSTS/E to prevent inadvertent copying of untranslatable RMS files to DOS magnetic tape.

4.4.6 RT-11 Utilities on DU Disks Greater Than Unit 9

The following utilities will not correctly print the disk name and unit number in error messages for disks having unit numbers greater than 9:

- LINK.SAV
- LIBR.SAV
- MACRO.SAV

For example, you see this message if the source file TEST01 does not exist on DU15 and you issue the following command:

```
$ RUN $MACRO
*DU15:TEST01=DU15:TEST01
?MACRO-F-File not found DVE:TEST01.MAC
DU15:TEST01=DU15:TEST01
```

This problem will be fixed in a future release of RSTS/E.

4.4.7 Log File Corruption

If the disk to which a DCL or BATCH log file is being written becomes full, the resultant log file may contain one or two blocks of random data as space becomes available.

4.5 RSX-11 and RMS Tasks

Usually you can omit one or both of the Project-Programmer Number (PPN) fields when specifying a file or an account. You can also use logical names longer than six characters and logical names containing dollar signs. You cannot include a trailing asterisk (*) in file specifications. Table 4-1 lists the RSX-11 and RMS tasks that do **not** allow you to specify one or more of these features.

Table 4-1: RSX-11 and RMS Tasks

Task Name	More than 6 Characters	Use \$	Omit PPN Field(s)
TKB	YES	YES	NO
MAC	YES	YES	NO
PAT	NO	NO	YES
LBR	NO	YES	NO
RMSDEF	YES	NO	NO
RMSDSP	YES	NO	NO

4.5.1 RMS-11

- RMS-11 does not correctly handle a file specification that contains an underscored device. RMS-11 ignores the underscore.
- When you use a key value of zero for a nonstring datatype to access a remote indexed file, RMS-11 incorrectly returns a Record Not Found error message. This is documented as a correct way to pass a nonstring key but currently works correctly only for local operations. This problem will be fixed in a future release of RSTS/E.

4.6 Layered Products

The following sections describe problems and restrictions for layered products.

4.6.1 DECnet/E

The problems and restrictions that apply to DECnet/E are:

- The NCP Command SET EXECUTOR SEGMENT BUFFER SIZE does not work when the EXECUTOR STATE is ON. This command should fail with an appropriate error message, but instead appears to succeed without taking any action. This will be corrected in a future release of RSTS/E.
- When attempting to issue a command such as SHOW NODE COUNTERS and the node number is not in your volatile database, NCP reports that the node does not exist, even though you may have an active link to that node. To work around this, ensure that all the nodes you normally access are defined in your volatile parameter file. This will be corrected in a future release of RSTS/E.
- DECnet/E does not successfully create a network parameter file for networks containing more than 26,000 nodes. To successfully create a node database (NETPRM.SYS), you must specify a MAXIMUM NODES value smaller than 26,000. This is a restriction to the current release of DECnet/E and may be lifted in any future release or update.
- When you use the SET HOST command from a RSTS/E node to some versions of VMS, logging off the VMS system may display the following error message:

```
%Logical link failure to remote node xxxxxx  
NSP reason code = 0  
Control returned to node xxxxxx
```

The receipt of an unexpected message from the VMS node during disconnect will trigger this error message. However, no actual error has occurred and the user session is not otherwise affected.

Using the SET HOST command from some versions of VMS to RSTS/E may result in the same error message. In addition, the RSTS/E system detaches the job.

4.6.1.1 NFT and FAL

- NFT and FAL do not preserve the correct protection codes when creating files. Protection codes for the output files are computed by adding together the default system protection code and the codes for executable (64), protected (128), or executable and privileged (64+128) as set on the input file. Note that these codes are decimal values.
- NFT and FAL do not correctly report the size of large files in directory listings. When the value exceeds 65535, the size information cannot be contained in 16 bits. It ignores extra bits and misrepresents the value. This will be fixed in a future release of RSTS/E.
- During a wildcard copy operation, NFT ignores the IBL block mode switch after the first copy is performed. This will be fixed in a future release of RSTS/E.
- The NFT and FAL programs add RMS attributes to files when they are copied. Since RSTS/E V9.5, an NFT copy of a nonattributed stream file created an Implied Carriage Control output file. When RT-11 listings and maps are copied in this way, the PBS program is unable to perform correct output of these files.

Use the /NOA switch to strip the attributes of the output file (see Chapter 4, Network File Transfer Utility, of the *DECNET/E Utilities Guide* for further information on the use of this switch).

This problem will be corrected in a future release of RSTS/E.

4.6.2 FORTRAN-77

If you are using FORTRAN-77 V5.2, you may get undefined symbols when using the LINK/F77 command. This problem will be fixed in a future release of FORTRAN-77. Make the following additions to LB:RMS11M.ODL to correct this problem:

```
OTSLIO: .FCTR  LB:[1,1]F77RMS/LB:$LSTI:$LSTO-$$FIOC-$$FIOD
```

becomes

```
OTSLIO: .FCTR  LB:[1,1]F77RMS/LB:LICSB$:$LSTI:$LSTO-$$FIOC-$$FIOD
```

```
OTSMS2: .FCTR  OTSVRT-LB:[1,1]F77RMS/LB:$VIRT:$MADBV
```

becomes

```
OTSMS2: .FCTR  OTSVRT-LB:[1,1]F77RMS/LB:$VIRT:$MADBV:$MKADB
```


Index

A

/ADDRESS qualifier
 CREATE/VIRTUAL_DISK command, 1-2
ANALYS.TSK, 4-4
ANALYS Package
 changes, 2-14
Answerback, 1-1
/ANSWERBACK qualifier
 SET SYSTEM command, 1-1
 SET TERMINAL command, 1-1, 2-8
 SHOW SYSTEM command, 1-4
Apostrophe substitution
 DCL, 1-2
ASSIGN command, 2-7

B

BACKUP/DIRECTORY command, 2-8
Basic-Plus RTS
 changes, 2-15
BATCH
 log files, 4-5
BROADCAST command, 4-3

C

COBOL command, 2-10
Continuation lines, 4-3
CRAW\$ directive, 2-4, 3-7
CREATE/VIRTUAL_DISK command, 1-2
 /ADDRESS qualifier, 1-2

D

DB disk errors, 4-2
DCL changes, 2-7
 apostrophe substitution, 1-2
 ASSIGN Command, 2-7
 BACKUP/DIRECTORY Command, 2-8
 COBOL command, 2-10
 command processing, 2-7
 continuation lines, 4-3
 DISMOUNT command, 2-10
 F\$MID function, 4-3
 INITIALIZE command, 2-10
 LINK command, 1-4
 log files, 2-5, 2-11, 4-5
 ON ERROR handling, 4-3
 Performance improvements, 1-3
 SET JOB/PRIVILEGE command, 2-8
 SET SYSTEM command, 2-8

DCL changes (cont'd.)

 SET TERMINAL command, 2-8
 SHOW DEVICE Command, 2-7
 SHOW LOGICAL Command, 2-7
 SHOW QUOTA command, 2-8
 SHOW SYSTEM Command, 2-7
 SHOW TERMINAL/PERM command, 2-8
 work file, 1-3
DCLWRK\$: logical, 1-3
 Performance issues, 1-3
DECmail-11, 1-1
 changes, 2-16
DECnet, 4-6
DECnet/E, 4-6
 changes, 2-15
 installation changes, 2-15, 3-34
 NCP.TSK, 2-16
 NCP circuit maximum routers parameter, 2-16
 new features, 1-6
DELETE/VIRTUAL_DISK command, 1-2
DIBOL
 changes, 2-17
DIRECTORY Command, 4-3
DIRECT Program
 changes, 2-12
DISMOUNT Command, 2-10
DISPLY Program
 changes, 2-12
DR disk errors, 4-2
DSKINT.TSK
 changes, 2-12
D space APR protection, 1-5
DU disks, 2-7, 4-2, 4-5
DVO:, 4-4
 virtual disk, 1-2, 2-2, 2-6
/DYNAMIC_REGION_LIMIT qualifier
 SET SYSTEM command, 1-4

E

EDT
 performance improvements, 2-2
Electronic messaging reception, 1-1
EMS\$: Logical, 1-1
EMTCPY
 changes, 2-14
/EMT_LOGGING qualifier
 SET SYSTEM command, 1-4
 SHOW SYSTEM command, 1-4
ERRDIS.TSK
 changes, 2-13

F

FAL.TSK, 4-7
Fast map switch
 default, 1-5
 performance issues, 1-5
FEADF\$ macro, 1-5, 3-8
FEAT\$ directive, 1-5, 3-7
/FM switch
 TKB, 1-5
FORTRAN-77, 1-6, 4-7
 new features, 1-6
/FO switch
 TKB, 1-5

G

GMCR\$ directive, 2-5, 3-8
GOSUB command, 4-3

H

HELP program
 \$WAIT command, 2-10
/HOLD qualifier
 SET SYSTEM command, 2-8

I

INIT.SYS, 4-1
 Default option, 2-6
INIT.SYS changes, 2-6
INITIALIZE Command, 2-10
Installation, 2-1
Installation of DECnet/E, 2-15, 3-34 to 3-38
Installation procedure changes, 2-2

L

LAT commands, 2-7
LIBR.SAV, 4-5
LINK.SAV, 4-5
LINK command
 Supervisor mode, 1-4
LOGIN.COM
 changes, 2-11
LOGIN.TSK, 4-4
 changes, 2-11
LOGOUT.TSK, 4-4
/LOG qualifier
 SET SYSTEM command, 1-4

M

MACRO.SAV, 4-5
Monitor corrections, 2-3
/MONITOR_NAME qualifier
 SET SYSTEM command, 1-4
MOUNT command, 4-3
MSCP disks, 2-7, 4-2, 4-5

N

NCP.TSK, 3-35, 4-6
NCP HELP
 changes, 2-16
New features

New features (cont'd.)
 DECnet/E, 1-6
 FORTRAN-77, 1-6
NFT.TSK, 4-7

O

ODT.TSK, 4-4
OPSER Package
 changes, 2-11
OPSRUN.TSK
 changes, 2-11

P

PBS Package
 changes, 2-10
Performance improvements
 DCL, 1-3
 EDT, 2-2
Performance issues
 fast map switch, 1-5
PIP.SAV, 4-4
 changes, 2-11
.PLAS directives revisions, 3-1
Protection
 D space APR, 1-5

R

Resident library
 mapping changes, 2-4
 supervisor mode, 2-4
RMS-11, 3-11, 4-4, 4-5, 4-6, 4-7
 changes, 2-15
 supervisor mode, 1-4
RMS libraries
 supervisor mode, 1-2
RSX-11/M
 FEADF\$ macro, 1-5
 FEAT\$ directive, 1-5
 new directives, 1-5
 SSTX\$ macro, 1-5
 TFEA\$ directive, 1-5

S

SCCA\$ directive, 2-5, 3-9
SET HOST command, 4-6
SET JOB/PRIVILEGE command, 2-8
SET SERVICE/LAT Command, 2-7
SET SYSTEM command
 /ANSWERBACK qualifier, 1-1
 /DYNAMIC_REGION_LIMIT qualifier, 1-4
 /EMT_LOGGING qualifier, 1-4
 /HOLD qualifier, 2-8
 /LOG qualifier, 1-4
 /MONITOR_NAME qualifier, 1-4
 /SWAP_MAXIMUM qualifier, 1-4
SET TERMINAL command
 /ANSWERBACK qualifier, 1-1, 2-8
 /WIDTH qualifier, 2-8
SHOW DEVICE command, 2-7
SHOW LOGICAL command, 2-7
SHOW QUOTA Command, 2-8
SHOW SERVICE/LAT Command, 2-7
SHOW SYSTEM command

SHOW SYSTEM command (cont'd.)
 /ANSWERBACK qualifier, 1-4
 /EMT_LOGGING qualifier, 1-4
 SHOW SYSTEM Command, 1-4
 SHOW TERMINAL/PERM command, 2-8
 SHUTUP.TSK, 2-5
 changes, 2-11
 Software changes
 installation procedure, 2-1
 system command files, 2-3
 SSTX\$ macro, 1-5, 2-5, 3-10
 STATUS Program
 changes, 2-12
 STK.TSK
 changes, 2-13, 2-14
 Supervisor mode, 3-15 to 3-34
 LINK command, 1-4
 resident library, 2-4
 RMS libraries, 1-2
 SVDB\$ directive, 2-6
 SVTK\$ directive, 2-6
 /SWAP_MAXIMUM qualifier
 SET SYSTEM command, 1-4
 SYS-call changes, 2-5
 SYSTAT.TSK
 changes, 2-12
 System command file changes, 2-3

T

TFEA\$ directive, 1-5, 3-9
 Time, 4-3
 TK50, 4-2
 TKB, 1-5
 changes, 2-13, 2-14
 CMPRT option, 3-12
 COMMON option, 2-13
 /FM switch, 1-5
 /FO switch, 1-5
 LIBR option, 2-13
 RESCOM option, 2-13
 RESLIB option, 2-13
 RESSUP option, 3-13
 SUPLIB option, 3-14
 TKB.TSK, 3-2
 TMSCP tapes, 4-2
 TU81, 4-2

U

Unsupported devices
 DEctape, 2-1
 DJ11, 2-1
 Papertape, 2-1
 PC11, 2-1
 PR11, 2-1
 RP02, 2-1
 RP03, 2-1
 TU56, 2-1
 Unsupported software
 5- and 6-bit characters, 2-15
 CSPCOM.OLB, 2-14
 CSPCOM.TSK, 2-14
 FLINT.TSK, 2-15
 SAVRES, 2-14
 Update Package
 changes, 2-14

UU.CFG directive, 2-5, 3-2 to 3-3
 UU.DIE directive, 2-5, 3-4
 UU.MNT directive, 2-5, 3-4
 UU.RTS directive, 2-5, 3-5 to 3-6
 UU.TRM directive, 3-6 to 3-7

V

Virtual disk, 2-2, 4-4
 creating DV0:, 1-2
 creating with DCL, 1-2
 VT125
 SET TERMINAL command, 2-8

W

\$WAIT command
 HELP program, 2-10
 /WIDTH qualifier
 SET TERMINAL command, 2-8
 Work file
 DCL, 1-3
 WPS-PLUS
 changes, 2-16

Reader's Comments

RSTS/E
Release Notes
AA-NB17A-TC

Please use this postage-paid form to comment on this manual. If you require a written reply to a software problem and are eligible to receive one under Software Performance Report (SPR) service, submit your comments on an SPR form.

Thank you for your assistance.

I rate this manual's:	Excellent	Good	Fair	Poor
Accuracy (software works as manual says)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Completeness (enough information)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Clarity (easy to understand)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Organization (structure of subject matter)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Figures (useful)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Examples (useful)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Index (ability to find topic)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Page layout (easy to find information)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

I would like to see more/less _____

What I like best about this manual is _____

What I like least about this manual is _____

I found the following errors in this manual:

Page	Description
_____	_____
_____	_____
_____	_____
_____	_____

Additional comments or suggestions to improve this manual:

I am using **Version** _____ of the software this manual describes.

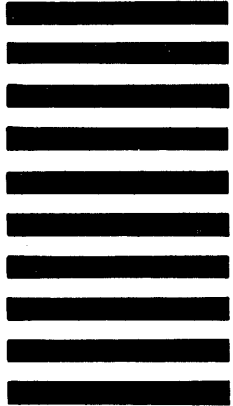
Name/Title _____ Dept. _____
Company _____ Date _____
Mailing Address _____ Phone _____

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

digital™



No Postage
Necessary
if Mailed
in the
United States



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

POSTAGE WILL BE PAID BY ADDRESSEE

Digital Equipment Corporation
Corporate User Publications
Continental Boulevard
MK01-2/E12
Merrimack, New Hampshire 03054



--- Do Not Tear - Fold Here ---

Cut Along Dotted Line