RSTS/E V8.0 Release Notes

Order No. AA–5246F–TC

March 1983

The Release Notes describe new features on the RSTS/E system and explain differences between this version and previous versions. System managers and system maintainers should read this document prior to system installation.

OPERATING SYSTEM AND VERSION:	RSTS/E	V8.0
SOFTWARE VERSION:	RSTS/E	V8.0

digital equipment corporation, maynard, massachusetts

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.

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

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

The following are trademarks of Digital Equipment Corporation:

digital ™
MASSBUS
PDP
P/OS
Professional
DEC

DECwriter
DIBOL
UNIBUS
VAX
VMS
VT
Rainbow

RSTS
RSX
DECmate
DECsystem–10
DECSYSTEM–20
DECUS
Work Processor

Commercial Engineering Publications typeset this manual using DIGITAL's TMS–11 Text Management System.

TABLE OF CONTENTS

Preface

The RSTS/E V8.Ø Release Notes describe new features in RSTS/E for this release as well as differences between this and the previous release, RSTS/E V7.2.

Anyone who is about to generate a RSTS/E V8.Ø system should be thoroughly familiar with the information presented in these Release Notes. In addition, users and programmers familiar with earlier versions of RSTS/E should read this manual to learn about changes and new features.

NOTE

> RC25 Device -- There are a number of references in the RSTS/E manuals to the RC25 device. This device is, as of the first publication date of the manuals, not available for sale or distribution; nor is its mention in the documents a commitment by DIGITAL to sell or distribute this device.

1.Ø NEW FEATURES OF RSTS/E V8.Ø

1.1 New Device Support

For RSTS/E V8.Ø, new device support is included for the RD/RX disks for micro RSTS and the Professional 35Ø with VT1Ø2 emulation with a few restrictions. Support for the TU8Ø was added for RSTS/E V7.2 in October 1982, and for the TSVØ5, RA81, RA6Ø, LNØ1, DZS11, LA5Ø, LA1ØØ/LA12, MS11-P, LPV11 and VT131 in December 1982. They are also supported for RSTS/E V8.Ø.

1.1.1 Restrictions

The following hardware is supported on RSTS/E V8.Ø only for systems upgrading from RSTS/E V7.2. RSTS/E support for these devices will cease in October, 1983.

Device	Suggested Replacement
RF11	RKØ7, RMØ2, or RA8Ø
RSØ3	RKØ7, RMØ2, or RA8Ø
RSØ4	RMØ3 on 11/7Ø; RMØ2 or RA8Ø
RKØ5J (as a system disk)	RLØ2
RKØ5F (as a system disk)	RLØ2
LT33	LA34 or LA38
VTØ5	VT1ØØ series
VT5Ø	VT1ØØ series
IBM 2741-compatible terminals	LA38 or LA12Ø
DP11 synchronous line interface	DUP11

RF11, RSØ3, and RSØ4 disks cannot be used as system disks in RSTS/E V8.Ø.

The TU58 will be supported on RSTS/E V8.Ø only for systems upgrading from RSTS/E V7.2. RSTS/E support for the TU58 will be stopped in January, 1984.

After RSTS/E V8.Ø, the LOAD and FILL options of INIT.SYS will no longer be supported.

RSTS V8.Ø and all of its Update Kits will continue to be distributed on 8ØØ BPI 9-track magnetic tape, RKØ5, RLØ1, and RKØ6. The next release after V8.Ø will not be distributed on 8ØØ BPI 9-track magnetic tape, RKØ5, RLØ1, and RKØ6.

1.2 Disk Structure

RSTS/E V8.Ø supports a new disk directory structure. This new structure is more efficient than the old one and allows for extensibility so that new features can be implemented in the future. It also allows more accounts; instead of approximately 17ØØ accounts, it is possible to have thousands. The DSKINT program (both online and offline) will create disks with this new structure. However, RSTS/E V8.Ø still supports the older disk structure, and disk packs of either kind can be mounted and accessed under V8.Ø. Note, however, that new structure disk packs are not acceptable to RSTS/E V7.2 or earlier releases -- an attempt to mount a new format pack under V7.2 will result in a "?Protection violation" error.

The support of "old" structure packs is temporary; RSTS/E releases subsequent to V8.Ø will not support the old structure. Furthermore, the performance and capacity improvements resulting from the new structure are not available on old format packs. DIGITAL recommends that you convert your old format packs as soon as convenient. However, do not convert them until you are satisfied that you will no longer need to access the packs using RSTS/E V7.2 or older systems. Some installations have programs, either written in-house or supplied by third parties, that directly access the RSTS/E directory structure (although this is not supported) -- such programs may need to be modified in order to handle the new format disks. Be sure to do this prior to conversion.

Another difference is that more space is now available on RPØ4, RPØ5, and RPØ6 disks: on RPØ4 and RPØ5 disks, 171798 blocks are now available rather than 1672ØØ, and on RPØ6 disks 34Ø67Ø blocks rather than 3344ØØ blocks.

1.2.1 Summary of New Disk Structure

DISCLAIMER

> The following information is included as a service to users who have software that directly accesses the RSTS file structure, to aid in converting such programs. Note that this is not a supported capability of RSTS/E; therefore, the information provided here is subject to change in future releases of RSTS/E. DIGITAL assumes no responsibility for the use of this information, and will not support software that uses the data structures discussed.

All numbers in the following discussion are in octal unless stated otherwise.

To minimize the impact of the directory structure change, all UFD data has been left unchanged. The only changes are in the structure of the MFD. The changes are such that most programs that reference the MFD directly (either non-file-structured or by its old name "[1,1]") will fail in a "reasonably sane" fashion.

Formerly, the UFDs were listed in the MFD, which was also the UFD for account [1,1] and could be accessed either by opening [1,1] as a file or by looking at Fip block number 1, which was the starting cluster of the MFD. The entries in the MFD were linked together in the same fashion as the file entries in a UFD; account entries were distinguished from file entries by a flag bit, but both could occur in the [1,1] directory.

In the new structure, block 1 contains the pack label. Block 1 contains the same information as before in the first 2Ø bytes; the rest of the block is reserved. New packs can be distinguished from old ones by the fact that the 2ØØØØ bit is set in the pack flags (offset 12 of the pack label). If set, offset 6 contains the structure revision number (4Ø1 for this version), and offset 4 contains the starting device cluster number of the MFD.

The MFD no longer points directly to the UFDs, but instead points to GFDs (Group File Directories), which in turn point to the UFDs. There is a GFD for each group that has accounts defined, and it contains pointers to each of the UFDs for that group. Both the MFD and the GFDs have the same basic format: they look like a UFD, except for the second and third blocks. These blocks are not formatted as UFD blocks (i.e. no clustermaps) but instead are interpreted as tables. The tables consist of word entries indexed by group number times 2 (for the MFD) or user number times 2 (for a GFD). The second block contains the starting device cluster number of the GFD for the group (in case of the MFD) or UFD for the user number (in the GFD). The third block is reserved in the MFD, and contains a directory link word for the name entry of the account in the GFD.

The UFD name entries and associated accounting entries (contained in the GFD) are very similar to the old form with the following differences: The name entries are not linked together, because they are accessed through the table in the third block of the GFD. Instead, the link word of the name entry contains a link to a chain of account attribute blocks. Each attribute block contains a link to the next one (Ø to end the chain) and a type field in the third byte. A type code of 3 indicates a password block; other type values are reserved for future use. Bytes 4 and up contain data (password in ASCII padded with nulls, in the case of the password block).

Unlike the old [1,1], the MFD and GFDs contain account entries only, not file entries. Note that in the new structure [1,1] is no longer the MFD and, therefore, need not be present on a disk. In addition, files can now be stored in [1,1] without affecting the number of accounts allowed on a pack. The limit of 1734 accounts per disk no longer applies; the disk structure allows every legal account number to be defined, although disk space restrictions make it impossible to allocate a UFD for that many accounts. However, the number of legal accounts is now limited by the amount of disk space available rather than by some inherent limitation in the file structure.

Neither the MFD nor the GFDs have file names. Consequently they can be accessed only by non-file-structured I/O. Note that the description just given is a summary only; we strongly recommend you do not attempt to directly access the RSTS directory structure since the structure is subject to future change and extension. System directives are available that allow most if not all programs to be implemented without access to the RSTS directory structure. For example, the recommended way to find out what accounts exist on a disk is to use the wildcard PPN lookup SYS call.

1.2.2 Disk Stucture Conversion Program

To facilitate conversion of old format packs, a conversion utility, DSKCVT, is included as part of RSTS/E V8.Ø. DSKCVT is installed if you install the "BIGPRG" programs. This utility performs an in-place conversion of the directory structure to the new format. To do so, it needs some amount of work space on the pack being converted. You should ensure that there is enough space available; the amount required is approximately 32 blocks times the number of different group numbers used on the pack. However, in most cases the amount used is significantly less than this value.

The conversion program guarantees no loss of data even if the program aborts because of lack of work space, a system crash, or disk hardware problem. This means that should any problem be encountered during the conversion, you need only correct the problem and rerun the conversion. In such a case, it is not necessary to restore the pack from a backup. Nevertheless, we strongly recommend you make a backup of any pack before conversion.

The conversion program uses two workfiles in account [1,1] on the disk being converted: these are named NEW11.DAT and MFDGFD.DAT. If files by that name already exist in [1,1], they will be deleted without warning. Please verify that you are not using these file names in [1,1] before you convert a pack.

To convert a pack, proceed as follows:

```
        run $dskcvt
        DSKCVT V8.Ø-Ø6 RSTS V8.Ø-Ø6

        Disk to convert? DLØ:           [Specify the pack to convert]
        Ready to convert DLØ:NEWODS     [DSKCVT displays the packid]
        Proceed (Y or N)? y             [Enter Y to proceed, N to quit]
                                        [DSKCVT performs the conversion]
        Sorting PPNs
        Processing [Ø,1]                [Each PPN is printed when processed]
        Processing [1,1]
        Processing [1,2]
        Processing [1,3]
        Processing [1,9]
        Processing [1,211]
        Processing [1,234]
        Processing [2,2]
        Processing [7,2]
        Processing [7,72]
           etc.

        Conversion of DLØ: complete
```

DSKCVT accepts either logically mounted or logically dismounted packs. However, the pack should not be in use by anyone when DSKCVT is run. To verify this, DSKCVT dismounts and remounts the pack at the start of processing.

DSKCVT can be run only by privileged users. It will ask you for the name of the disk drive containing the pack to be converted. If the pack appears to be an already converted pack, a message is printed to that effect and the question is asked again. Otherwise, DSKCVT prints the device name and pack label to allow you to verify you are converting the proper pack, and asks for confirmation. If you answer YES, it proceeds without further intervention. First, the account numbers are sorted in ascending order. Next, each account is processed (its data is inserted into the work file which will become the new directory structure). After all accounts have been processed, DSKCVT dismounts the pack and updates the pack label. At this point the pack is a new structure pack; however, the storage allocation table (SAT) does not yet exactly reflect the free space on the pack. Therefore, the pack remains marked as "dirty" while DSKCVT updates the SAT. After this is complete, the pack label is updated once more to indicate the pack is "clean." Finally, if the pack was mounted when DSKCVT started, it will be mounted once again (with its pack ID as a logical name).

If DSKCVT aborts for any reason during processing (for example, due to a hardware error or system crash), any data on the pack being converted is unharmed. After correcting the problem, simply re-run DSKCVT and specify the same pack again. If conversion had proceeded far enough, DSKCVT will report that the pack has already been converted, in which case you can simply mount it using the DCL MOUNT command, (the SAT will be rebuilt at mount time if necessary), or use the ONLCLN utility. Otherwise, DSKCVT will simply restart the conversion from the beginning after deleting the work files (NEW11.DAT and MFDGFD.DAT) that were left behind by the abort. As pointed out before, we still recommend you make a backup of each pack before conversion.

1.3 System Generation (SYSGEN)

Several new questions have been added to the SYSGEN dialogue which reflect new device support and new or expanded features of RSTS/E V8.Ø. These questions and their ramifications are described in detail in the RSTS/E System Generation Manual and are outlined in Section 2.1 of these notes.

1.4 Pre-built CUSPs

RSTS/E V8.Ø offers users the choice of 'BUILD'ing their system by compiling the CUSPs themselves or by selecting to use the pre-built CUSPs supplied in .TSK format on the distribution kit. See Section 2.5 (BUILD) of these notes for information on how to select pre-built CUSPs during the BUILD. Using this option will make the installation faster.

1.5 Initialization Code (INIT.SYS)

Several changes to the initialization code have been made for RSTS/E V8.Ø. Please refer to Section 2.2 of these notes for detailed information.

1.6 Monitor

Several changes have been made to the Monitor for RSTS/E V8.Ø. The RSTS monitor now allows a job's maximum size to be 32kw when an RTS or LIBRARY is not used (and RSX emulation is included in the monitor). Please refer to Section 2.3 of these notes for detailed information.

1.7 File Processor

The following directive is new for RSTS/E V8.Ø.

1.7.1 STALL/UNSTALL Directive
(UU.STL; SYS call subfunction 29)

This SYS call halts all activity on the system by all users, except for the issuing user. This is accomplished by not scheduling any jobs to run except the issuing job. The intent is to enable users whose system disk is on a single spindle/multi-unit drive to stall the system, change the removeable unit, and resume operations without taking down or crashing the system.

For more detailed information, please refer to the RSTS/E Programming Manual and the RSTS/E System Directives Manual.

1.7.2 EMT Logging

EMT logging is an optional feature that provides the system manager with a "window" on the process by which timesharing jobs request and receive services from the RSTS/E Monitor. By using it, you can gather information about certain kinds of activity on your system. For instance, you might be interested in the number of LOGIN's on a particular terminal port, in how many (or which) files are accessed on a certain drive, or in which non-resident FIP overlays get the heaviest use. Such information can help a knowledgeable system manager "tune" a system for improved performance, identify bottlenecks, establish charging algorithms, and watch for potential security problems.

This section describes the EMT logging feature, how it works, and how to include it in your system. Because of the wide variability among RSTS/E systems, however, it does not attempt to describe which data will be most useful to your particular environment. Similarly, it is up to you to decide how to collect (or to use) the data provided.

EMT logging includes optional code in the Monitor, uses XBUF for passing information, and requires a program (not supplied by DIGITAL) running as a timesharing job. You should be aware, therefore, that use of EMT logging can have an impact on performance. This impact is variable, and depends upon which EMT's you decide to log, for which jobs you log them, and how much processing your logging program attempts to do for each EMT.

Documentation and Other Materials

The EMT Logging feature is described in general terms in the RSTS/E Programming Manual, the RSTS/E System Directives Manual, and the System Manager's Guide. Certain details, such as description of data fields returned for a logged EMT, are subject to change in future releases of RSTS/E. This information is included in the Release Notes, instead of in the formal documentation.

The RSTS/E distribution kit contains two additional components related to EMT logging. A feature patch (article number 3.5.12F) allows you to specify which EMT's are to be logged. A demonstration program (EMTCPY.BAS) is included in the UNSUPP (unsupported) package, which illustrates sample techniques for retrieving EMT logging data, including how to declare an EMT logger receiver and how decode the packets passed by the Monitor. This program is not supported, and may neither work with nor be supplied with future releases of RSTS/E.

Background

Every system directive contains an "EMT" hardware instruction (hence this feature's name: "EMT Logging"). To use a Monitor service, a timesharing job loads parameters in its FIRQB and/or XRB and executes an "EMT." This causes a hardware interrupt, and the RSTS/E Monitor gains control. The Monitor then inspects the parameters passed in the FIRQB and/or XRB, and (based upon the parameters, as well as the particular EMT) performs the requested service. Upon completion, the Monitor may pass status information and/or other data back to the job, depending upon the call. For example, the first byte in the job's FIRQB is usually loaded with an error code (zero indicating success, and nonzero values communicating specific kinds of failure).

Data Returned to an EMT Logger

EMT Logging provides information on timesharing activity in terms of what the Monitor sees. Therefore, you will want to be familiar with the MACRO form of system directives, as they are described in the RSTS/E System Directives Manual. The data returned to your logging program is in terms of FIRQB and/or XRB contents, regardless of the programming environment of the job that issues the directives.

Contents of an EMT Logger Packet

(The following description assumes you have read the material on EMT Logging contained in the appropriate sections of the RSTS/E System Manager's Guide and RSTS/E Programming Manual.)

In this release, the Monitor's EMT Logging code returns two fields in each EMT packet: the "header", or context, field, and a field of information extracted from the caller's FIRQB. Although the identities (and lengths) of the fields returned in this release are the same for all packets, you should code your EMT Logger program to respect the variable format (that is, honor the packet's count word and the fields' count bytes), in order to minimize possible changes later.

Recall that the packet described below concerns a single EMT. A single receive may return to your EMT Logging program a variable number of packets, depending upon the buffer space you provide for the data portion of the message. Each packet contains a count field (as its first two bytes) which specifies the packet's length (exclusive of the count bytes). Additionally, the "parameters" returned by the receive call contains a word that gives the number of packets returned to your buffer.

"Bytes" are specified in octal, with the high byte's number to the left for word fields.

Bytes (relative to beginning of packet)	Bytes (relative to beginning of field)	Contents	
.			
.		(possible preceding packets)	
.			
1 - Ø		This packet's length (word)	<66>
--		Start of root (context) field	
2	Ø	Length of "options" area	<Ø>
3	1	Length of XRB data	<Ø>
4	2	Length of FIRQB data	<34>
5	3	Length of context (root) field	<32>
7 - 6	5 - 4	Packet sequence number	
11 - 1Ø	7 - 6	System date at EMT time	
13 - 12	11 - 1Ø	System time at EMT time	
14	12	Seconds till next minute at EMT time	
15	13	Ticks till next second at EMT time	
16	14	Calling job number, times two	
17	15	Reserved	
2Ø	16	IOSTS byte (at directive completion)	
21	17	Function code of directive	
22	2Ø	Reserved	
23	21	Calling job's keyboard number	
25 - 24	23 - 22	Reserved	
27 - 26	25 - 24	Calling job's PPN	
31 - 3Ø	27 - 26	Job's virtual PC (EMT instr + 2)	
32	3Ø	UUO code (if call was a .UUO)	
33	31	Reserved	
--		End of root (context) field	
--		Start of FIRQB data field	
35 - 34	1 - Ø	Third word of caller's FIRQB (*)	
37 - 36	3 - 2	Fourth word of caller's FIRQB	
.	.		
.	.		
67 - 66	33 - 32	Last word of caller's FIRQB	
--		End of FIRQB data field	

(*) - First two words of caller's FIRQB are not returned in the FIRQB data field, since the information they contain is conveyed in the root field.

Notes:

The numbers in angle brackets are the values that will be returned in the indicated fields. (Not variable in this release, but remember the note above about honoring count fields in any program you write.)

"IOSTS byte" contains the returned error code (zero indicates successful completion, no error)

Function code is the FIP function code for this directive. These codes can be found in TBL.LST (one of the listings prepared during SYSGEN; these function codes have MACRO mnemonics of the form xxxFQ). They should be considered as subject to change in future releases. As a current example, a function code of Ø14 corresponds to UUOFQ - the FIP function code for a .UUO directive.

Calling job's keyboard number - in case of a detached job, this will be the one's complement of the number of the keyboard the job detached from.

Job's virtual PC - is the virtual address, in the user's job space, of the instruction following the EMT instruction that invoked this call. This may be of interest if the calling program is written in MACRO. For other languages, it will be less useful: the PC will be within the run-time system in the case of BASIC-PLUS, for instance. (PC = Program Counter, the PDP-11's "next instruction" register.)

UUO code - If the call was a .UUO directive (FIP function code = 14), this byte will contain the code of the UUO. These codes can be found in the System Directives Manual, or in TBL.LST; they have MACRO mnemonics of the form UU.xxx. If the call was not a .UUO directive, this byte will be 177. (Numbers are in octal.)

Which EMT's are Logged

You may select the EMT's for which the Monitor will construct packets and pass them to your program by means of a feature patch (sequence number 3.5.12F, included on the distribution kit).

When you generate a Monitor, it will by default log no EMT's. The feature patch includes a segment for each potentially loggable directive. You may edit the feature patch to select the directives you wish to log. (As shipped, the patch has the effect of disabling logging for every one of the candidate directives).

1.7.3 OPEN Mode for Disk Initialization

A new non-file-structured open mode, mode 512, has been created to allow a program to read the factory bad block file of a disk. This new open mode also affects error logging for that unit. Error messages will be re-routed to the program if it is declared a local receiver with an object type of 64 decimal; otherwise, the error message is discarded.

1.7.4 Changes to SYS Calls

Changes to several SYS calls have been made for V8.Ø. These changes are listed below. Refer to chapter 2 for further details.

Function Code		Change
-29	Tables 3	Additional monitor tables returned for SYSTAT.
-28	Spool	Changes for the new Micro-RSTS spooler
-24	Add CCL	A CCL can now consist of a single, non-alphanumeric, character (e.g., "$"). The "@" character is no longer allowed in positions 2-n of a CCL keyword.
-8	Channel stats	This SYS call has been enhanced to return access counts, current file size, and status byte.
Ø	Create account	Creation of account allows placement and pre-extension of the UFD, and creation of [1,1] on new structure disks.
1	Delete account	Account [1,1] can be deleted from the new disk structure.

Function Code		Change
2	Clean	This SYS call is no longer supported.
5	Logout	LOGOUT checks for violations of disk-usage quotas. Disk quota checking is done on all read/write mounted disks (locked disks are not checked if the user is not privileged). If job is non-privileged, a new non-privileged detach quota is checked. If the quota is not violated, the non-privileged job is killed, and no return is made to the job.
6	Attach	Allows swapping consoles with another job
13	Zero	Zero does not release the UFD clusters unless specified
22	Send/ Receive	Non-privileged jobs can declare themselves as receivers. Local receivers can have Object Types, and extensions have been made for EMT Logging. See Sections 2.3.2 and 1.7.2
24	Create Job (Spawn)	Jobs can be created logged into an account. Either a program can be run in that account or a specified keyboard monitor can be entered. With certain restrictions, non-privileged jobs can create new jobs logged into the same non-privileged account.

1.8 Terminal Service/Line Printer Drivers

A new feature, a "no stall" RECORD modifier to prevent the driver from stalling when the output data cannot be buffered immediately, has been added to terminal service and line-printer drivers in RSTS/E V8.Ø. This feature is only available to programs written in MACRO; it is not available to BASIC-PLUS programs.

Normal character-oriented output (the only kind available in previous releases) works as follows:

When a line printer or terminal output request is made (via the .WRITE directive), the driver first transfers the user data into a system buffer. If there is insufficient room to buffer the user data, because the device cannot empty the system buffer as fast as it is being filled, the driver will stall while waiting for the system buffer to empty sufficiently so that the data transfer can be completed. Such a stall will often occur, for example, when a user's terminal has been XOFFed by the user typing CTRL/S. Once the system buffer becomes full any additional .WRITE request will cause terminal service to stall until sufficient buffer space becomes available (by the user typing CTRL/Q) to complete the data transfer.

While the driver is in a stall state, the user program is also stalled. This can cause problems for programs that have time-critical processing requirements, such as message send/receive, where a .WRITE to an XOFFed terminal might prevent the program from processing any incoming messages until the terminal were XONed and the driver returned control to the program.

In V8.Ø, RECORD modifier bit 8192 can be included with a .WRITE directive to a terminal or line-printer to instruct the driver not to stall when the user data cannot be buffered. Instead, control is returned to the calling program with the byte count in the XRB (XRBC) set to indicate the number of bytes that were not buffered. The calling program may then re-issue the .WRITE, possibly after processing any time-critical sequences, to complete writing any data not sent in the original request. If XRBC is returned with the value Ø, then all data was successfully transferred and no additional .WRITE is required. For .WRITE directives issued without the new "no stall" RECORD modifier, processing will be as in previous releases: control will not be returned to the calling program until the data transfer has been completed, returning Ø in XRBC.

NOTE

> An exception to this is the case of a .WRITE to a hung line printer. The initial .WRITE causes all data to be buffered, but subsequent .WRITEs will not buffer any data (XRBC will be returned unchanged) until the printer error is corrected.

When using the "no stall" option on a line printer, you can perform a special test to see whether the line printer is busy without causing your program to stall. To perform the test, write a single null character with modifier bits 2ØØØØ(8) and 4 (both the "NO STALL" and "STALL UNTIL OUTPUT COMPLETE" bits) set. With both bits set, the system will return control to your program instead of stalling it. The value at XRB+XRBC tells you the status of the line printer. If XRB+XRBC contains zero after this write, it indicates that the line printer's buffers are empty (there are no more characters left to print). If XRB+XRBC contains one, then the line printer is still busy printing the characters that remain in its buffer.

Full use of the new "no stall" feature requires access to the XRBC word in the XRB in order to determine the amount of data not transferred by the .WRITE directive. Therefore, only languages that can access the XRB directly (e.g. MACRO) may use this feature. Note that high-level languages with an external calling interface or commons (e.g. BASIC-PLUS-2) can still use the feature by including a MACRO-coded routine to return the contents of XRBC to the calling program or mapping into the XRB. BASIC-PLUS, which has no such calling mechanism, or commoning, cannot take advantage of the new feature.

1.9 Micro-RSTS Spooling Package

The new micro-RSTS spooling package provides an alternative to the standard RSTS/E Spooling package. The new spooling package has both advantages and disadvantages. Weigh these factors to decide which package is best for your installation.

Advantages of the Micro-RSTS spooling package:

- It reduces system load. It requires only one permanent job on the system, versus a minimum of three for comparable services from QUEMAN. The new spooling package also uses less CPU time, disk I/O, and memory.
- Performance is greatly improved. The programs in the package are more efficient.
- You can print files with any RMS file organization. The standard package is restricted to a few of the more common organizations.
- Operator commands are easier to use. New command syntax for starting the package, initializing printers, and so forth, make managing the package easier.
- The package provides a Forms Definition File to provide easy maintenance of form attributes. The file is keyword driven and can be modified using a standard text editor.
- The package interface is "friendlier." For example, if a user tries to delete a nonexistent job in the queue, the interface displays an error message.

Disadvantages of the Micro-RSTS Spooling Package:

- The package does not have BATCH or RJ278Ø capabilities. (RJ278Ø is optional RSTS/E layered software.)
- The package provides only a single queue (named PRINT:) to hold all print requests.
- The package does not provide a means to determine the status of the print devices, spooler, or queue manager. An operator must manually maintain information on which print devices have been initialized, forms assigned, and so on.

Using the new package:

The new spooling package is designed to be used by DCL commands. See the RSTS/E DCL User's Guide for descriptions of user commands for the package. Chapter 12 of the RSTS/E System Manager's Guide has descriptions of the operator commands for the package. Two versions of DCL can be installed; one to support the new spooling package, and one to support the standard package (you cannot support both within the same DCL run-time system). The syntax you use depends on which package is installed on your system. The system manager is responsible for telling users which spooling package is supported in DCL.

See section 12.2 of the RSTS/E System Manager's Guide for further information on maintaining both old and new spooling packages on the same system.

Installation:

The Micro-RSTS spooling package is installed by running BUILD and using the control file SPL.CTL. Two versions of the spooling package are offered to reflect different usages of RMS. One version requires the RMS resident library (RMSRES) to be installed during execution. The other version does not require the resident library, but is heavily overlaid. The version that uses the RMS resident library will perform better than the overlaid version.

The monitor must be configured to include RSX emulation (the "disappearing" RSX run-time system), and a minimum swap-max of 29K. The interface utility QUEUE.TSK requires approximately 22K plus the RMS resident library for the resident library version, or 28K for the overlaid version.

1.1Ø DCL

The following commands have been added or enhanced for RSTS/E V8.Ø.

1.1Ø.1 LINK Command

The DCL LINK command provides the program with more address space than it did before and is substantially more flexible. It provides easy access to program debugging tools, supports the linking of RSX-11-based MACRO programs, and can be used on a wider variety of systems.

For more information, please refer to the RSTS/E DCL User's Guide.

1.1Ø.2 MOUNT and DISMOUNT Commands for Disks

The MOUNT command for mounting disks has been expanded as follows:

```
MOUNT device-name[:] label [logical-name[:]][/qualifier]
```

A privileged user can specify an alternate logical name in the MOUNT command is allowed.

The qualifiers are:

/[NO]WRITE - can be specified by privileged and non-privileged users to indicate the desired read-only or write enabled access. If the disk is initialized as read/write, then the default action is to mount the disk write-enabled, unless the drive is write protected, in which case it is mounted read-only. If the disk is initialized as read-only, then the default action is to mount the disk read-only.

/[NO]REBUILD - a privileged operation that allows a qualified user to either mount a public or a private disk and unconditionally perform the cleaning operation to rebuild any corrupt disk file structures, or to mount the disk and not rebuild it. The default action is to rebuild the disk if it is "dirty" (was not logically dismounted). Mounting a dirty disk and choosing not to rebuild it allows only read-only access.

/PRIVATE - can be specified by privileged and non-privileged users. A non-privileged user can mount only a private disk as private. A privileged user can mount a private or public disk as private.

/PUBLIC - A privileged operation that allows a qualified user to mount as public a disk that was initialized as public.

/[NO]SHARE - can be specified by privileged and non-privileged users to limit disk access to only the job that mounted the disk (which thus becomes non-shared) or to mount it as shared. A disk mounted as shared is the same as a disk mounted as private. A non-privileged user can mount only a private disk as shared or non-shared. A privileged user can mount a private or public disk as shared or non-shared.

If neither /PRIVATE, /PUBLIC, nor /[NO]SHARE are present, then the default action is to mount the disk as private (shared).

The DISMOUNT command for dismounting disks has been expanded as follows:

```
DISMOUNT device-name[:] label[/qualifier]
```

The only qualifier is:

/PUBLIC - A privileged operation that allows a qualified user to dismount a public disk that was initialized and mounted as public. The default action is to dismount the private disk.

A restriction of the DCL DISMOUNT command is that if a job mounts a disk as non-shared and a second job (which is running under a privileged account) attempts to dismount the disk by specifying the correct pack-id label, then the error "?Pack-id labels don't match" occurs. The work-around for this situation is for the second job to not specify the pack-id label on the DISMOUNT command.

Refer to the RSTS/E System Manager's Guide for more detailed information on the MOUNT and DISMOUNT commands.

1.1Ø.3 MOUNT and DISMOUNT Commands for Tapes

If you do not explicitly specify a density using the /DENSITY qualifier, the MOUNT command assumes the system default density. If the system default density is not the density at which the tape drive is currently set, and your tape drive is connected via a TMØ2 formatter, you will get the error "?Data error or incorrect density". If your tape drive is connected via a TMØ3 formatter, the density will be determined automatically. If you specify a density different than that of the tape, you will get the error message "?Incorrect density".

The MOUNT command deals with the format differently. If you do not explicitly specify a format using the /FORMAT qualifier, the MOUNT command looks at the header label on the tape to determine the format. If you specify a format different from the format on the tape, you will get the error message "?Bad directory for device". You will be prompted for a label, if the tape is in ANSI format. This label is not optional. You will not be prompted for a label if the tape is in DOS format or if you mount the tape FOREIGN. The label is ignored if you specify a label on a DOS tape, and a warning message is given.

If you do not specify a density or format, this information will be supplied.

The DISMOUNT command has one new qualifier.

/[NO]UNLOAD Requests that the tape be unloaded from the drive. The default is /UNLOAD.

The DISMOUNT command will automatically rewind the tape.

1.1Ø.4 COBOL Command

The COBOL command has the following new qualifiers:

- /CHECK now accepts the arguments ALL, [NO]PERFORM, and [NO]BOUNDS. The default is /CHECK=ALL. [NO]PERFORM and [NO]BOUNDS control whether or not checking is to be done on indexes and perform ranges. The arguments can be listed, eg. /CHECK=(PERFORM,NOBOUNDS), to check specific cases.

- /CODE=CIS and /CODE=NOCIS control whether or not the compiler uses CIS (Commercial Instruction Set) in the object module it produces.

- /DEBUG and /NODEBUG control whether or not your program uses the COBOL-81 Symbolic Debugger. To use the debugger specify /DEBUG on both the COBOL and LINK commands.

- /DIAGNOSTICS[=filespec] and /NODIAGNOSTICS control whether or not the compiler produces a diagnostic file.

- /SHOW, /SHOW=MAP, and /NOSHOW control whether or not the compiler produces a data division map showing the addresses for data division entries.

- /SUBPROGRAM indicates that the compiler is compiling a subprogram.

- /TEMPORARY:device changes the storage area for temporary work files from SY: to the device specified.

- /TRUNCATE and /NOTRUNCATE control whether or not the compiler preforms decimal truncation on the values of COMP data items.

- /WARNINGS, /WARNINGS=INFORMATION, and /NOWARNINGS control whether or not the compiler issues informational diagnostics during the compilation.

1.10.5 INITIALIZE Command for Disks

You can now initialize disks online by using the DCL INITIALIZE command. The format of the command is:

```
INITIALIZE device-name[:] pack-id [/qualifier]
```

Default values have been set for all options such as number of patterns to run during bad block checking, retaining the previous bad block file on current RSTS file structured disks, specifying the pack cluster size, MFD cluster size, and positioning of the storage allocation table (SATT.SYS). You can specify values other than the default values by using the INITIALIZE command qualifiers. INITIALIZE always creates the disk for new files last and no pre-extending of directories. It creates account [0,1] but not [1,1] or [1,2].

The command qualifiers are as follows:

```
/CLUSTER_SIZE=n
/DATE=argument
/[NO]EXERCISE[=argument]
/INDEX=position
/MFD_CLUSTER_SIZE=n
/PRIVATE
/PUBLIC
/[NO]QUERY
/[NO]RETAIN
/[NO]WRITE
```

See the RSTS/E DCL User's Guide for full detail on how to use these qualifiers.

1.10.6 INITIALIZE Command for Tapes

The INITIALIZE command for tapes now prompts for a label only if the tape to be initialized in ANSI format. The system default format for tapes is used if you do not specify a format. You will be told the settings that will be used for the density and format if you did not specify them.

1.10.7 DCL Commands Via "$" CCL

In V8.0, you can create a "$" CCL to point to DCL. Doing so allows access to DCL for a single command from any keyboard monitor -- for example, you could type "$COPY FOO BAR" or "$PRINT FOO" from BASIC-PLUS.

See Section 2.3.9 for general information on single-character CCL's.

1.11 DSKINT

The online DSKINT.BAS program has been replaced by DSKINT.TSK, which requires the RSX run-time system. The new version of DSKINT initializes a disk with the new RSTS/E file structure and does pattern checking. The online version of DSKINT provides all the functionality of the offline version with the exception of formatting. INIT's version of DSKINT needs to be used to format disks. The dialogue for the online and offline version is the same except for the FORMAT prompt which is found in the offline version only.

For more information on DSKINT refer to the RSTS/E System Users Guide.

1.12 Layered Product Installation

Because RSTS/E V8.Ø contains a new version of RMS, the following layered products must be re-installed:

COBOL-81
DATATRIEVE
FMS

BASIC-PLUS-2 V1.6 and earlier releases are not supported on RSTS/E V8.Ø. You will have to install BASIC-PLUS-2 V2.Ø or V2.1.

Please refer to the chapter 5 RMS Release Notes included with this document for additional information.

1.13 BASIC-PLUS-2

1.13.1 Compatibility with RMS and BASIC-PLUS-2 V1.6

User programs compiled with BASIC-PLUS-2 V1.6 cannot be task built with RMS V2.Ø. Any BASIC-PLUS-2 V1.6 applications that were task built with RMS V1.8 in the task image and with no RMS resident libraries will continue to run. If they were linked against the RMS V1.8 resident library, they should be rebuilt with RMS V1.8 in the task image and with no RMS resident library or recompiled using BASIC-PLUS-2 V2.Ø or V2.1 and task built against RMS V2.Ø.

1.13.2 New Error Messages

Three new error messages have been added:

185	%RECORDSIZE overflows MAP
186	?Improper error handling
187	?Illegal record lock clause

1.14 FORTRAN-77

One of the new RMS V2.Ø rules impacting FORTRAN-77 states:

> "If you are using disk-overlaid RMS, your ODL cannot make reference to the RMS 'top-level' modules."

These 'top-level' modules are the RØ'op' modules that exist in SYSLIB. Because the existing overlay descriptor file for the FORTRAN-77 RMS OTS, LB:RMS11M.ODL, references RMS 'top-level' modules, it is incompatible with RMS V2.Ø. Therefore, a new version of the FORTRAN-77 RMS overlay descriptor file has been supplied on the update kit in this release.

1.15 DATATRIEVE V2.4

V8.Ø Autopatch "A" must be installed to make DATATRIEVE V2.4 work.

1.16 Documentation

The RSTS/E Quick Reference Guide is new for V8.Ø. A replacement for the RSTS/E Pocket Guide, it summarizes system information (such as file specifications and device names), DCL commands, system programs (such as PIP and SYSTAT), and RMS utilities.

Refer to the RSTS/E Documentation Directory for more information about changes to the manuals for version 8.Ø.

1.17 V8.Ø Release Information

1.17.1 DECnet/E V2.Ø Support; Modified Build/Patch Procedure

DECnet/E V2.Ø is the only version of DECnet/E supported on RSTS/E.

Beginning with RSTS/E V8.Ø, certain components of DECnet/E V2.Ø will be maintained by module replacement. For these components, patching and task-building on-site will no longer be necessary. With update kit "A" (shipped with the RSTS/E V8.Ø distribution), three components of DECnet/E V2.Ø are updated with module replacements: NFT, FAL, and NCP. These replacement modules incorporate all previously-published patches, as well as fixes for several problems which were not patchable.

Update kit "A" also includes a new version of the build/patch top-level control file, DECNET.CTL. NOTE that this control file supersedes the DECNET.CTL file supplied on the DECnet/E V2.Ø distribution. The new control file from update kit "A" will be used to install all future module replacements, and to apply future patches, when the PATCH option of BUILD is invoked. You must also use the new DECNET.CTL file to install the DECnet/E utilities as you build DECnet/E for your V8.Ø RSTS/E system.

You will find additional information in the RSTS/E V8.Ø Maintenance Notebook.

1.17.2 Feature Patches

Please be aware that any feature patches from previous releases may have been changed or dropped. Refer to the RSTS/E V8.Ø Maintenance Notebook to see all feature patches, notes, or restrictions which apply to RSTS/E V8.Ø.

Note that there are no BASIC-PLUS-2 or RMS patches included with this kit.

Article Seq 17.1.1 N describes the update kit included with this release.

1.17.3 Mounting and Rebuilding (Cleaning) Disks

The new disk structure of RSTS/E V8.Ø is incompatible with the CLEAN SYS call (FIP code 2). This call was used by a number of CUSPS in RSTS V7.2. Because of this restriction, the CLEAN SYS call is now obsolete (see section 2.3.6). CUSPS are affected as follows:

1. The CLEAN command in UTILTY has been eliminated.
2. The MOUNT commands in UTILTY and INIT and the $MOUNT command in BATCH will no longer clean (rebuild) a dirty pack.

The recommended way to mount disks is the DCL MOUNT command, which will rebuild disks as needed. You should use FORCE commands in INIT to do this during system startup. If you do not want to use DCL, you should run ONLCLN to rebuild each disk prior to using the UTILTY or INIT MOUNT commands.

2.Ø DIFFERENCES BETWEEN RSTS/E V8.Ø AND V7.2

2.1 System Generation (SYSGEN)

- The "RMØ2/Ø3/Ø5's ? " question has been changed to "RMØ2/Ø3/Ø5/8Ø's ? ." Although the RM8Ø disk drive was fully supported for RSTS/E V7.2, the SYSGEN dialogue did not include RM8Ø.

- Previously, SYSGEN asked two questions regarding paper tape devices:

  "P.T. READER?"

  "P.T. PUNCH?"

  Because the PR11 Paper Tape Reader is no longer supported, these are combined into one question:

  "P.T. reader/punch ? "

  SYSGEN continues to support the PC11 which both reads and punches paper tape.

- The LS11 and LV11 line printer names were removed from the line printer informational paragraph.

- In the disk section, all references to million words have been changed to mega bytes. The UDA question has been changed to ask for MSCP controllers.

- BASIC-PLUS-2 for the default RTS has been removed from the dialogue.

- The maximum number of system wide logicals has been increased to 999. The default is 25.

- A question for "EMT logging" (emulator trap logging) has been added after the question for Monitor Statistics. If you answer YES to this question, the monitor being built includes the EMT Logging code. (See section 1.7.2.) In addition, the send/receive code is automatically made resident (and the corresponding question in the SYSGEN dialogue is skipped).

- TSVØ5 was added to the long and short form of the TS11 question.

- RF disks can no longer be system disks.

- The RC25 was added to the long and short form of the MSCP question.

2.2 Initialization Code (INIT.SYS)

In startup, INIT uses a free 8K area of memory to complete the loading of the monitor. This was the case on systems with a UDA as system disk and is now the case for all systems. This 8K area is allocated from available memory but must be between 6ØK and 124K. It cannot be below 6ØK because INIT.SYS is still running while this space is prepared for use and occupies memory up to 6ØK. "Available" is defined as either "user" or "run-time system" memory. If no 8K area can be found, the following error message is returned:

```
Need at least 8KW non-monitor memory below 124 K to start
```

You must allow room for this 8K area. If the system default RTS is 8 KW or larger (for example BASIC-PLUS) there is no problem. Otherwise, you should allocate "user" space above the RTS and put any desired XBUF at the top of memory or ending at 512K if you have more than 512KW of memory. You can use other layouts as long as the required 8 KW area is available between 6ØK and 124K.

The UNISYS option has been removed from INIT.SYS. You can perform the function of the UNISYS option by using the DISABLE suboption in the HARDWARE option to disable the terminal controllers you do not want. This has the advantage of being selective; you can disable terminal controllers one at a time as opposed to disabling all as UNISYS does.

The default for keyboard devices has been changed from UNPRIV to PRIV.

The COPY option no longer allows copying to DF (RF11) or DS (RSØ3/Ø4) disks. These disks are no longer bootable and, therefore, cannot be used as system disks. You can still copy to DK (RKØ5) disks with the COPY option, even though the RKØ5 is not supported as a system disk.

All device bootstraps were modified for this release. The new bootstraps are not compatible with the ones used in RSTS/E V7.2 and previous releases. This change is invisible to the user except in operations that involve setting up a bootstrap for INIT.SYS ("hooking"). This is done in two places:

1. The COPY option in INIT hooks the output disk when it finishes copying. Because the bootstraps are different, you cannot copy a V8.Ø distribution medium using a V7.2 or earlier INIT, nor copy a V7.2 or earlier distribution medium with the V8.Ø INIT. The normal procedure of booting the distribution medium and then invoking the COPY option to set up the new system disk will work properly.

2. SAVRES, both the off-line and the on-line version, sets up bootstraps in two cases: on the first volume of an output saveset (SAVE option), and on output RSTS disks if there is an INIT.SYS in account [Ø,1] on the output disk (RESTORE or IMAGE option). As a result, there are certain restrictions in the use of SAVRES across versions:

   a. The on-line version of V7.2 SAVRES does not run on a V8.Ø system, nor does the V8.Ø version run on a V7.2 system.

   b. You cannot restore a V7.2 system disk (i.e. any disk that has an INIT.SYS in account [Ø,1]) using the V8.Ø SAVRES. The recommended way to restore such a disk is to boot the saveset and run the off-line (V7.2) SAVRES that resides on the saveset. Likewise, you cannot restore a V8.Ø system disk using the V7.2 SAVRES.

   c. You cannot make an image copy of a V7.2 system disk with the V8.Ø SAVRES, or a V8.Ø system disk with the V7.2 SAVRES. The recommended way to perform this operation is by booting the system disk to be copied and using the off-line IMAGE option.

   d. You should not use the SAVE option to save a V7.2 system disk with the V8.Ø SAVRES or a V8.Ø system disk with the V7.2 SAVRES. Although the operation succeeds, you would not be able to restore the resulting saveset because of restriction b and because the SAVRES on the saveset would be the wrong version. Data disks, (disks that do not contain an INIT.SYS in [Ø,1]) can be saved, restored, or image-copied with either version of SAVRES.

   NOTE

   There are no compatibility problems with data disks. Any disk that does not have an INIT.SYS in account [Ø,1] can be copied using either version of SAVRES and restored with either version.

   If you fail to observe the restrictions under a, b or c, you will get either a system disk that cannot be booted, or the following error message:

   ```
   ???Fatal SAV/RES bug: No BOOT in STB of INIT.SYS
   ```

   Since there are no longer bootstraps for the DF and DS disks, V8.Ø SAVRES will treat these strictly as data disks. If you restore or image copy a DF or DS disk and SAVRES finds an INIT.SYS in account [Ø,1], SAVRES will print out a warning message to inform you that the output disk will not be bootable. You could, of course, restore V7.2 DF or DS system disks with V7.2 SAVRES to make them bootable.

3. There is also a restriction on booting 16ØØ bpi magnetic tapes with the new bootstraps. Due to size restrictions on the bootstrap code, it is not possible to boot 16ØØ bpi magnetic tapes on tape formatters other than Formatter Ø.

   Because of the change in bootstraps, the HOOK program was changed. Only version 8.Ø HOOK will work on RSTS V8.Ø, and it will not work on RSTS V7.2.

The following changes to the SET option have been made:

1. An 8BIT/NO8BIT characteristic has been added to the LP option of SET. When enabled, this option allows 8 bit characters to be sent to the printer. This allows the use of printers with special control characters and extended character sets which include 8 bit characters. The 8BIT characteristic replaces patch article seq 3.7.3 F, with the advantage that it only affects line printers for which screening out all 8 bit characters is not desirable.

2. The VTAB/NOVTAB line printer characteristic has been removed. This printer characteristic was added for a line printer that is no longer supported.

3. A RESET macro has been added to the list of line options. This option resets all of the characteristics for a printer to their default values.

   Previously all unspecified characteristics were set to RSTS defaults, regardless of their values before invoking the SET option. Now unspecified characteristics will not be changed.

4. Set-up Macros for the unsupported LS11 and LV11 printers have been removed.

2.3 Monitor

The RSTS monitor now allows jobs to have a maximum size of 32KW instead of 31KW.

In RSTS/E V8.Ø a number of directive changes were made to facilitate the creation of non-privileged application programs that run detached and perform services for other jobs. Formerly such programs almost always required privilege to perform their functions.

To ensure that non-privileged users do not swamp the system with detached jobs, there exists a "detached job quota." This quota is defined using feature patch 3.2.6 F; its default value is zero. The quota specifies the number of detached non-privileged jobs allowed per attached job, i.e. the ratio of detached to attached jobs. The monitor enforces this quota. Privileged jobs are not affected by the detached job quota.

2.3.1 ATTACH/REATTACH/SWAP CONSOLE Directive (UU.ATT; SYS call subfunction 6)

ATTACH Function

If the job being attached by the ATTACH function is running under the same PPN as the job issuing the ATTACH call, the function is allowed to non-privileged callers. The PPN specified in the call (which must match the PPN of the job being attached) now defaults to the caller's PPN if it is passed as zero.

The password field of the ATTACH function is checked only if the job being attached is running under a PPN different from that of the caller. Note that this case is still restricted to privileged callers.

Non-privileged users only need to specify the function code (ATTACH), the job number, and zeros in the rest of the FIRQB (or SYS call argument string). The PPN is defaulted and the password is not required for matching PPNs.

REATTACH Function

The REATTACH function is now available to non-privileged users to allow a non-privileged program to attach to a terminal. An additional restriction is imposed on non-privileged callers - if the terminal being reattached to is free (i.e. not currently assigned or opened by anyone) then the REATTACH will be permitted only if the terminal has been set as a 'non-privileged' device (using the SET option of INIT.SYS). If the terminal is assigned to the caller at the time of the reattach, this check is skipped. A privileged user can 'reassign' a terminal to a non-privileged job and, thereby, allow that job to reattach to the terminal whether or not the terminal is a privileged device.

SWAP CONSOLE Function

The SWAP CONSOLE to another job call is a new subfunction of the ATTACH/REATTACH directive. It allows two jobs, one of which is detached, to exchange ownership of a terminal. In effect, this call combines a detach of the attached job with a reattach of the detached job. Its purpose is to allow detached 'service' type programs to temporarily obtain ownership of a terminal to perform certain functions at the request of the program running at that terminal and then to return ownership of the terminal to the requesting program when the function is complete.

This function is available to non-privileged jobs. It can be executed by either the detached job or the attached job. Both jobs must be running under the same PPN.

Refer to the *RSTS/E Programming Manual* and the *System Directives Manual* for further detail on this directive.

2.3.2 Message Send/Receive Directive (MESAG, SYS call subfunction 22)

2.3.2.1 Non-privileged SEND/RECEIVE

The "Declare Receiver" system call of BASIC-PLUS and the "Declare Receiver" subfunction of the .MESAG monitor directive can now be used by non-privileged jobs. However, the following restrictions are imposed on non-privileged users:

- Non-blank receiver names must have six characters, and the last two characters must be the job number (not the job number *2) and include a leading zero if appropriate. Blank names are allowed.

- There is a new quota on the number of RIBs (Receiver ID Block) per user. The default value of this quota is 3. This quota can be changed by feature patch Seq 3.1.5 F.

- There is a new upper limit on the number of messages pending per RIB. The default value of this limit is 4. This limit can be changed by feature patch Seq 3.1.6 F.

- Small buffers cannot be used for the data portion of messages. XBUF must be used.

- Receivers are not allowed to accept incoming network connects.

- The number of outbound network links per job is limited to 1. This limit can *not* be changed by a feature patch.

These restrictions require that the following conditions be satisfied in the data passed with the BASIC-PLUS system call:

Bytes	Parameter	Value in non-privileged call
5-10	(name)	six characters (last two are job number)
22	N% O%	0% 0%
23-24	B%	0%
25	M%	within new limit
26	L%	0% (number of inbound links)
29	D%	For privileged jobs: 0%-255% where 0% yields the maximum of 255% For non-privileged jobs: 0%-1% where 0% yields the maximum of 1% (number of outbound links)

In addition, the number of RIBs that can be declared must be within the new quota.

Violation of the restriction on the name of the receiver will cause an ?ILLEGAL FILE NAME error to occur. Violation of the other restrictions will cause a ?PROTECTION VIOLATION error to occur.

On systems using DECnet, mandatory patch Seq 7.1.12 M must be installed. This patch causes DECnet to enforce the limit on outgoing links per job.

NOTE

Patch Seq 3.1.1 F, published in earlier releases of RSTS/E, making receiver declaration non-privileged no longer applies.

2.3.2.2 Local Object Types

Prior to Version 8.Ø, the object type field in a Receiver ID Block (RIB), and the corresponding field in the receiver declare call, only had meaning for network receivers. In V8.Ø, local receivers (if they are privileged) may now assign themselves an object type.

Local object types defined in V8.Ø are:

LOT	Use
Ø	Normal receivers
1	Error Logger (ERRCPY)
2	EMT Logger
64	Reserved for DSKINT

Note that user-written programs should not use any values other than Ø and 2 (an EMT Logger is a user-written program; see Section 1.7.2). All values other than those listed above are reserved.

2.3.2.3 Send/Receive Extensions for EMT Logging

Some parameters of Send/Receive calls have been assigned special meaning for EMT Logging; these are described in Section 1.7.2

2.3.3 Job Creation Directive
(UU.JOB, SYS call subfunction 24)

The Create a Job (Spawn) SYS call has been expanded to allow (a) non-privileged users to create jobs, and (b) new jobs to be created logged-in. The following information that describes the new SYS call is presented in three categories: spawn logged-out, spawn logged-in to run a program, and spawn logged-in to enter a specified keyboard monitor.

Spawn Logged Out

When used to spawn logged-out jobs, the SYS call is used exactly as it has been used in the past, i.e., privilege is required and the bytes passed are described in the current documentation for the SYS call.

Spawn Logged-In to run a Program

The SYS call can now be used to spawn jobs that are automatically logged into an account. This feature can be used by both privileged and non-privileged jobs. The new job can be either attached or detached.

For more detail, please refer to the RSTS/E Programming Manual and the System Directives Manual.

NOTE

Feature Patch 3.1.6 F must be installed before non-privileged jobs can spawn new jobs that are detached. Also, Feature Patch 3.5.14 must be installed before non-privileged users can log out leaving detached jobs in that account. For example, DECWORD spawns spooler jobs in the account where the DECWORD programs reside. If this is a non-privileged account, it will not be possible for a user to log out of the account if a spooler is active and the patch (3.5.14) has not been installed.

Spawn Logged-In to enter Keyboard Monitor

The SYS call can now be used to spawn jobs that enter a keyboard monitor instead of running a program. This feature can be used by both privileged and non-privileged jobs. The new job must be attached.

For more detail, please refer to the RSTS/E Programming Manual and the System Directives Manual.

2.3.4 LOGIN Directive (uu.LIN, SYS call subfunction 4)

The Login SYS call has been changed to prohibit passwords containing the "?" character. In V7.2, this restriction was enforced by the LOGIN CUSP; the change enhances security because possible loopholes (via other programs that use the Login call but do not perform the check), are eliminated.

2.3.5 LOGOUT Directive (UU.BYE, SYS call subfunction 5)

The Logout SYS call has been expanded to have three new functions:

1. Disk usage quotas are enforced. All read/write mounted disks are checked, and if the job is privileged, locked disks are also checked. This function can be suppressed by a control bit in the call parameters.

2. If the job is non-privileged, the number of detached jobs in the non-privileged account is compared against a quota of detached jobs allowed per attached job in that account. If this quota check number is passed (and also the disk quota check), the job is logged out and killed. (No return is made to the job.) See also the note in section 2.3.3 above.

3. It is now possible to LOGOUT without closing channels or deassigning devices. This function can be activated by a control bit in the call parameters.

For more detail, please refer to the RSTS/E Programming Manual and the System Directives Manual.

2.3.6 CLEAN Disk Pack Directive (UU.CLN; SYS call subfunction 2)

In RSTS/E V8.Ø, the Clean up a Disk Pack SYS call (subfunction code 2) is no longer supported. The correct way to rebuild (clean) a disk is by using the ONLCLN utility or the REBUILD option of the DCL MOUNT command. For V8.Ø the SYS call will still work on old format (V7.2) disk packs; however, it will fail with a ?Protection violation error if used on new format disks. In future releases the SYS call will be removed. Use of this SYS call on RPØ4, RPØ5, or RPØ6 type disks will result in a disk which is incompatible with RSTS/E V7.2 and earlier systems.

2.3.7 One-shot Spooling Request Directive

(UU.SPL; SYS call subfunction -28)

Changes have been made to the UU.SPL directive to provide support for the new micro-RSTS spooling package (SPL).

If a UU.SPL directive is issued with a device name of "LP" (indicating a print request) or a null device name (which defaults to "LP"), the monitor will first search its message receiver table for the receiver ID of the Micro RSTS spooler (QMAN). If found, then the print request is sent to that receiver; if not found, the request is sent to the standard spooling package's receiver (QUEMAN), as it has in the past. This design provides upward compatability with the new spooling package; the applications that issue the UU.SPL directive need no modifications to support the new spooling package.

If both spooling packages are running on the same system, the monitor will send one-shot print requests to the new spooling package. A feature patch (3.5.13 F) is provided with version 8.Ø to disable routing of one-shot print requests to the new spooling package, for those installations that prefer to have all print requests routed to the standard spooling package.

For more detail, please refer to the RSTS/E Programming Manual and the System Directives Manual.

2.3.8 RSTS/E Error Message

RSTS/E error message 25, ?Disk pack needs 'CLEANing', has been changed to:

```
?Disk pack needs REBUILDing
```

2.3.9 Single-Character CCLs

The V8.Ø Monitor can recognize and process certain special characters as "self-delimiting" CCL commands. (These characters are "@", "#", and "$", ASCII 64, 35, and 36 decimal.) If such a character is defined as a CCL, typing a string beginning with that character at keyboard monitor command level (or passing such a string to the .CCL directive) will invoke the target CCL program in the same manner as any CCL, except that no delimiter need follow the special character. The monitor will automatically insert a blank character between the CCL character and the rest of the command.

For instance, if you define the "$" character as a CCL with

```
UT  CCL $-=[Ø,1]DCL.DCL;PRIV Ø
```

then any command of the form "$xxxx" will work (invoking DCL) from any keyboard monitor (where "xxxx" is any valid DCL command). Should DCL not recognize the command, it will (as usual) pass it on to the Monitor to be tried as a RSTS/E CCL. (A special case: should "$" not be defined as a CCL, "xxxx" will be tried as a CCL, without reference to DCL. This has the effect of allowing a user to type "$xxxx" (where xxxx is an installed CCL keyword) whether or not DCL is available on the system; a command such as "$PIP", for example, will invoke PIP on any usually-configured RSTS/E system.)

Other single-character CCLs operate normally, and may be set up to point to any program capable of responding to them. The string passed in core common will have a <space> character inserted following the single-character keyword if there was no <space> (or <tab>) typed to separate the command character from its parameters.

The syntax of a valid CCL keyword is now a string of one to nine characters. If one character long, the keyword may be either a letter or one of the characters above. If two to nine characters long, the keyword must begin with a letter; each of the remaining characters may be either a letter or a digit. This is different from V7.2, which allowed "@" characters to occur in other than the first position.

2.3.1Ø Terminal Service: CHR$(155%)

As described in section 4.5.3.1 of the RSTS/E Programming Manual, you may use a CHR$(155%) to transmit an <ESC> character to a terminal (that is, to bypass the normal translation of an <ESC> to a "$" character). Previous versions of RSTS/E transmitted the CHR$(155%) without change; all supported terminals ignored the eighth bit, and "recognized" the received CHR$(155%) as a CHR$(27%), an <ESC> character. In V8.Ø, normal terminal output (e.g., not using binary mode) strips the eighth bit before transmitting any character in the range from CHR$(128%) to CHR$(159%). (That is, the "C1 controls" are transmitted as "CØ controls".) Specifically, CHR$(155%) is now output as a true <ESC>, CHR$(27%).

In RSTS V8.Ø, terminals are by default set up as privileged devices. This is a security precaution. If you want to allow non-privileged users to open some or all keyboards as files, you must use the UNPRIV option of the INIT SET option (see the RSTS/E System Generation Manual.)

2.3.11 Small Buffers: JDB2 Structure Note

One of the small buffers required for each job (the "second job data block", or JDB2) is now permanently allocated, being assembled into the Monitor at SYSGEN time.

The amount of lowcore memory required for a particular JOBMAX is not significantly changed. The permanent allocation is made from the same memory which is used for the General small-buffer pool. The creation of each new job, on the other hand, now diminishes the pool by one fewer small buffer.

2.3.12 ECC Error Correction

The ECC hardware error correction algorithm used with the RKØ6/7, RMØ2/3, RMØ5, RM8Ø, and RPØ4/5/6 families of disks does not always correct multibit errors properly. The monitor has been modified to cause multibit errors to be retried instead of being ECC corrected. This change is transparent to the RSTS/E user, except that the error-log will show that errors that appear to be ECC correctable were retried instead of being ECC corrected.

2.4 File Processor

Non-privileged users are no longer allowed to mount or obtain write access to non-file-structured disks. Feature patches seq 3.5.5 F and 3.5.6 F need to be installed to permit mounting and obtaining write access to non-file-structured disks by non-privileged users.

Non-privileged users can reassign a device to another job only if the target job is logged into the same account. In previous releases of RSTS/E, non-privileged users were allowed to reassign devices unless the feature patch to restrict non-privileged use was installed. The File Processor feature patch has been eliminated.

2.5 Commonly Used System Programs (CUSPs)

This section describes the changes made to the RSTS/E CUSPs (Commonly Used System Programs) for RSTS/E V8.Ø.

All supported BASIC-PLUS CUSPs, with the exception of the CPEXER program in the DEVTST package, can be compiled with any of the three variants of BASIC; BASIC-PLUS, BASIC-PLUS-2, or CSPCOM (which is described below). The CPEXER program is compatible with BASIC-PLUS (provided the TRIG and LOG math functions are generated into BASIC-PLUS) and BASIC-PLUS-2, but not with CSPCOM.

CSPCOM (the CUSP Compiler) generates object modules from the BASIC-PLUS system library programs (CUSPs) that can be task built to run under the RSX run-time system. This eliminates the need for the BASIC-PLUS run-time system on those systems which would not otherwise use BASIC-PLUS or BASIC-PLUS-2.

NOTE

> CSPCOM is included in the RSTS/E kit as a maintenance tool only. CSPCOM is <u>not</u> supported for any use other than the compilation of DIGITAL-supplied utility programs that are included with the RSTS/E distribution kit.
>
> The use of CSPCOM is not supported for the compilation of the utility programs supplied with any optional RSTS/E software product unless the installation guide for that product specifically states that the software is compatible with CSPCOM.
>
> The version of CSPCOM included on the V8.Ø distribution kit includes all patches previously published for earlier releases.

ANALYS

Includes changes in DISK, RTS, and LIB sections as noted under SYSTAT.

ATPK

No changes.

Automated Patching Facility Package

This package has been changed to handle module replacements to the CUSPS.

BACKUP

The BACKUP package has been changed to handle both new and old types of disk structures.

When backup pre-extends accounts upon account creation, the account will be pre-extended by 1 cluster.

BATCH

The $MOUNT command cannot mount a "dirty" disk. Use $ONLCLN to clean disks packs before submitting them to a BATCH processor for MOUNTing.

The $BASIC/BP2 switch does not support BASIC-PLUS-2 beyond V1.6. To use later versions of BASIC-PLUS-2 in BATCH, use commands to invoke the BP@ compiler and task builder directly.

BPCREF

No changes.

BUILD

The BUILD program has been changed to allow compilation under the BASIC-PLUS-2 compiler. If you wish to use BASIC-PLUS-2 to compile your programs, you must answer the following questions in the manner given below:

```
Run-Time System?                                            RSX
Use CSPCOM <Yes> ?                                          NO
Name of BASIC-PLUS-2 compiler <$BP2IC2> ?
```

(The third question, which is a new BUILD question, must be answered with the name and location of the BASIC-PLUS-2 compiler when you installed it - e.g. SY:[1,2]BP2IC2.)

If you wish to use the pre-built CUSPs to build your system, answer RSX to the Run-Time-System question and YES to the CSPCOM question. BUILD will generate the commands necessary to copy the pre-compiled tasks from the distribution media onto your system.

If you create your own BUILD control files, the following restriction applies:

> In the BUILD control files, a file name must follow the 'COMPILE' command. (e.g. COMPILE/OBJ/LIN FOOBAR)

COPY

No changes.

DEVTST - Device Exerciser Package

No changes.

DIRECT

Has been changed to handle both old and new disk structures.

Display Programs (VT50PY/VT5DPY)

No changes.

DSKINT

The DSKINT.BAS program has changed to DSKINT.TSK and requires the RSX run-time system. Please refer to Section 1.11 of these notes for detailed information.

Error Package

The ERRDIS Summary Report has had blank lines removed for a more compressed format.

NOTE

> See section 2.13 of this manual for updating procedures that must be followed for the Error Package to function correctly.

FILCOM

No changes.

FIT

No changes.

FLINT

No changes.

GRIPE

No changes.

HELP

HELP has been modified to refer the user to the appropriate manual for more detailed information on a specific topic.

HELP no longer gains temporary privileges before opening HELP files. This will require that all HELP files not meant for non-privileged users must be set to a protection code of <6Ø>. Other HELP files should be set with a protection code of <4Ø> to allow any user to view them. During the RSTS/E BUILD all HELP files are set at the correct protection code; however, other system independent HELP files may require the user to reset the protection codes. If the message "?Access not granted to file" appears, the HELP file was not set at the right protection code.

A /CHAIN qualifier has been added to HELP which directs help to chain to a specified program upon completion. This allows an application to make use of the help facility by chaining to HELP with a command in core-common followed by the /CHAIN qualifier to cause a return to the original (or another) application.

The format is:

```
          /CH[AIN]:"prog-name[;line-number[;core-common-string]]"
```

where

```
          prog-name is the file spec of the program to chain to
          line-number is the line number to chain to
          core-common-string is a string to be placed in
                                    core-common before chaining
```

INIT

Since the CLEAN SYS call does not support the V8.Ø disk structures, INIT will no longer attempt to automatically rebuild (clean) a disk found to be "dirty" when processing a MOUNT command. It is recommended that you use ONLCLN to rebuild any disk suspected of being dirty.

LOGIN

LOGIN now performs quota checks if the user is already logged in and ATTACHes to another job or HELLOs to another account.

LOGIN now enforces quotas on all disks if logged in and changing accounts.

LOGIN will print a message giving old and new accounts when it is used to change from one account to another. Feature patch 1Ø.12.2 F has been eliminated.

When you log into an account, LOGIN will display the date, time, and terminal at which the account last logged in.

LOGOUT

LOGOUT now enforces quotas on all disks mounted read/write at logout time. LOGOUT does not check LOCKED disks if the user is not privileged, since that user would not be able to access a locked disk to reduce his storage.

MAKSIL

Has been changed to process the new autoload vectors generated in the new TKB.

MONEY

The MONEY program now allows specification of a wildcard account. For example, all accounts matching [2,*] may be listed.

The dialog has changed in the following manner: 1) there are fewer questions, 2) some questions have changed, and 3) defaults are displayed. The "Selective?" and "Device?" questions have been removed. The answers to these questions are determined by the answer given to the "Account?" question. The questions and their defaults appear below.

```
Output report to <_KB:.LST>?

Print passwords <NO>?

Reset <NO>?

Account <_SY:[*,*]>?
```

The format of the report has changed slightly. The phrase "with statistics being reset" appears on a line by itself, and the introductory text lines (Accounts dump of ...) appear each time the "Account <_SY:[*,*]>?" question is answered.

An asterisk is printed next to the quota if an account is over quota.

For disks with the new disk structure, accounts are printed in ascending order. For disks with the old disk structure, accounts are sorted by project number but not programmer number.

There is a single-line command format for privileged users. This format may be used through the CCL facility or from within a program by chaining to MONEY at line 3ØØØØ. Defaults are the same as those given above. The following switches are available:

/PASSWORDS - causes passwords to appear in the report.
Default: passwords do not appear.

/RESET - causes statistics to be reset.
Default: statistics are not reset.

/CHAIN - causes MONEY to chain to a specified program when MONEY is ready to exit.

The format is:

```
/CH[AIN]:"prog-name[;line-number[;core-common-string]]"
```

ODT

No changes.

ONLCLN

To invoke the ONLCLN program via 'RUN $ONLCLN' the user must be privileged. This is the program invoked by the DCL MOUNT command.

ONLCLN has been changed to handle both old and new type disk structures.

OPSER

At system startup time when the Spooling Package is started, or whenever OPSER is run:

1. If OPSER is already declared by another job as a message receiver, then a message will be generated to indicate that the new OPSER can not stay online and that it will go offline.

2. If OPSER was already declared as a receiver by the calling job, then an error will be generated to indicate that OPSER will remove itself as a message receiver and then declare itself again.

3. If OPSER can not obtain required small buffers, it generates a message to indicate that buffers were not available, and that it will try again. It will try for about 1 minute to get resources; upon failure it will notify the user that it is going offline due to lack of small buffers.

PIP

The PIP program is no longer invoked on a DCL INITIALIZE command for tapes.

PMDUMP

The following changes were made to PMDUMP in RSTS/E V7.2, but have not yet been added to the documentation:

PMDUMP now prints information about all the libraries used in the program being dumped. PMDUMP can read input from .TSK and MAKSIL type .SIL's as well as .PMD files. Several new switches have also been added:

/TSK - Treat input file as a .TSK (or .SIL) regardless of file type.

/WIDE - During octal portion of the dump, also print the ASCII characters.

/START:nn - Starting address of octal dump (does not print the header portion of the dump).

/END:nn - Ending address of the octal dump (does not print the header portion of the dump).

QUE

No changes.

QUEMAN

At system startup time when the Spooling Package is started, or whenever QUEMAN is run:

1. if QUEMAN is already declared by another job as a message receiver, then a message will be generated to indicate that the new QUEMAN can not stay online and that it will go offline.
2. if QUEMAN was already declared as a receiver by the calling job, then a message will be generated to indicate that QUEMAN will remove itself as a message receiver and then declare itself again.
3. if QUEMAN can not get any small buffers, it generates a message to indicate that none were available, and that it will try again. It will try for about 1 minute to get resources, upon failure it will notify the user that it's going offline due to lack of small buffers.

QUOLST

No changes.

REACT

The 'Enter' mode has been changed to fully implement the disk directory structure changes for RSTS/E V8.Ø. There are two new questions and two new print outs.

The new questions are:

Number of Clusters - asks the system manager how many clusters to extend the UFD (User's file Directory) by. If the response is zero or nothing, there will be no UFD created for the account. The pre-extension is useful for known large accounts (since the monitor will not have to go through cluster by cluster expansion). The maximum amount by which the UFD can be extended is 7 and the minimum is 1.

NOTE

> ONLCLN and the INIT.SYS REFRESH CLEAN option must examine the complete contents of every directory during the clean operation. Pre-extending directories beyond what is needed can therefore substantially slow the cleaning operation.

Position - asks the system manager where to put the UFD on the disk. This is a disk cluster number. The default is to place the directory near the MFD (as specified when the disk was initialized). A value of -1 will attempt to place the UFD at the middle of the disk.

Example:

```
RUN $REACT
REACT   V8.Ø-Ø1 RSTS T8.Ø-Ø1 *  The Computer  *
System Account Manager
Function? E
Proj,Prog? 235,2
Disk:Password? SY:BOILER
Quota <Ø>? 2ØØØ
Cluster Size <Ø>? 16
Number of Clusters (Ø-7) <1> ? 3
Position <Ø> ? -1
Account Name? KIRTUS
Account SYØ:[235,2]BOILER Created at DCN's 31185  31213  31229
```

The new print outs are:

Account....Created - This message prints out confirmation of the creation of the account and the DCN locations of the UFD.

The Delete function of the program now requires that you specify a disk name. No default is accepted. This is because the Delete function will zero the account before deleting the account.

REORDR

REORDR allows account [1,1] on new RSTS disks to be reordered.

RUNOFF

No changes.

However, as in earlier releases, the PRINT INDEX feature, which requires an obsolete SORT package, is not supported.

SAVE/RESTORE

Has been changed to handle both old and new disk structures.

SPOOLING PACKAGE

See Section 1.9.

SHUTUP

If a user has an EMT logging program running, SHUTUP will send a shutdown message to the EMT logger job after the final job killing phase. If the job doesn't disappear in 6Ø seconds, SHUTUP will kill the job.

SWITCH

No changes.

SYSTAT

The Links/Max field has been changed to Links/InMax/OutMax where InMax is the number of incoming links and OutMax is the number of outgoing links.

SYSTAT prints the level (old or new) of the mounted disks and also prints the names of the disk from which libraries and run-time systems were added.

TALK

No changes.

TTYSET

The terminal macros no longer reference other terminal macros in their definitions, resulting in better performance.

The program size has been reduced by about 4K.

UMOUNT

The UMOUNT program is no longer invoked on a DCL MOUNT or DISMOUNT command for disks or tapes.

UTILTY

The "SEND KBnn: (message)" command will not return the message "KBnn: is busy" if KBnn is a pseudo keyboard. UTILTY no longer recognizes the CLEAN command. Run the ONLCLN program to rebuild disks that are dirty.

2.6 DCL

The following changes were made to DCL in RSTS/E V8.0, and are documented in the *RSTS/E DCL User's Guide*:

- The /ALLOCATION qualifier to the COPY and CREATE commands now accepts any integer. DCL formerly would not allow any integer larger than 65535. If you specify an integer larger than 8388607 you will get the error ?Syntax error.

- The /BLOCK_SIZE=n qualifier was added to the COPY command. This qualifier determines the blocksize of an output magnetic tape file.

- The /CREATE qualifier has been added to the EDIT/EDT command. The /CREATE qualifier tells EDT to start your edit session with an empty buffer if the input file you specify does not exist. When you type EXIT EDT will create a new file. /NOCREATE will cause EDT to exit immediately if the primary input file does not exist. The default is /CREATE.

- The /DEVICE_TYPE qualifier has been added to the SET TERMINAL command. The SET TERMINAL command currently allows the following terminal type qualifiers: /VT05, /VT52, /VT55, /VT100, /LA34, /LA36, /LA38, and /LA120. These terminal types are valid as arguments to the /DEVICE_TYPE qualifier. The following is a list of all valid terminal types:

  ```
  ASR33   ASR35   KSR33   KSR35   LA30    LA30S   LA36    LA12
  LA50    LA100   LA120   LA180S  LA34    LA38    RT02    VT05
  VT05B   VT50    VT50H   VT52    VT55    VT100   VT101   VT102
  VT125   VT131   VK100   2741
  ```

  If you specify a terminal type other than one of these you will get the error message '?Not a valid device type - xxxx'.

- The /DMS qualifier was added to the LINK command. It allows you to link a DIBOL program with DMS (Data Management System) rather than RMS (Record Management Services).

- The /ERASE qualifier was added to the DELETE command. This qualifier requests that the file be written over with zeroes prior to deletion.

- The /OPTIMIZE qualifier has been added to the DIBOL command. This qualifier will cause the DIBOL compiler to create a program which takes less space and runs faster.

All of the DCL feature patches have been re-released; the old patches will not work.

The /F4P qualifier is no longer available in the FORTRAN and LINK commands, as FORTRAN-IV-PLUS is no longer supported.

The LINK command is much more powerful than before. See Section 1.1Ø.1 for a description of the new features of this command.

The MOUNT command will automatically rebuild (clean) a disk that needs it, if the user issuing the command is privileged. See Section 1.1Ø.2 for a detailed description of the changes to the MOUNT command.

The DISMOUNT command automatically rewinds a tape. It also will unload the device. See section 1.1Ø.3 for a detailed description of the changes to the DISMOUNT command.

The INITIALIZE command can initialize both disks and tapes. There are a number of new qualifiers for this command. See Section 1.1Ø.5 and 1.1Ø.6 for a detailed description of the changes to this command.

There are two spooling packages for RSTS/E V8.Ø. A choice must be made at the time of installation as to which package you want DCL to support. See section 1.9 for the advantages and disadvantages of the two packages. See section 2.7 for information on how to install the proper DCL.

The spooler commands DELETE/ENTRY, DELETE/JOB, PRINT, SHOW QUEUE, SET QUEUE/ENTRY, SET QUEUE/JOB, and SUBMIT have added capabilities. Refer to the RSTS/E DCL User's Guide for a detailed description of these commands.

The SET TERMINAL command now accepts the /VK1ØØ qualifier. This declares the terminal to be a VK1ØØ, or "GIGI", graphics terminal. /VK1ØØ is equivalent to /VT1ØØ except that the width for the VK1ØØ is 84.

The DIBOL command has the /DEBUG and /NODEBUG qualifiers to allow use of the DIBOL Debugging Technique (DDT). To use DDT specify /DEBUG on both the DIBOL and LINK commands.

The SET and SHOW commands now prompt for "What:" if no option is given.

The BASIC command will switch you into BASIC-PLUS-2 by default. Patch Seq 2Ø.2.2 F may be applied to change the default to BASIC-PLUS.

DCL assumes that the BASIC-PLUS-2 keboard monitor is called BP2. Patch seq 2Ø.2.3 may be applied to have DCL assume a different name.

DCL takes fuller advantage of DECnet/E capabilities:

- The /TOTAL qualifier is now legal with network DIRECTORY operations. Formerly, it was legal only for local DIRECTORY operations.

- The PRINT command allows printing of a remote file on a local printer.

- The SUBMIT command allows a remote file to be submitted to a local batch processor.

There are several new messages, many of which are only applicable to the MICRO RSTS/E spooling package. Refer to the RSTS/E DCL User's Guide to see a complete list.

One error message has been added and one has been changed that is not reflected in the RSTS/E DCL User's Guide. The following was added:

?Invalid node name
The node name you specified, although syntactically correct, contained more than six characters. The Network File Transfer program (NFT) will not accept node names more than six characters in length.

The following was changed:

?Too many elements in list
Replaces ?Too many items in list. It occurs when you indicate more file specifications on the command line than the command allows.

NOTE

For DCL to work properly, all of the standard CUSPs and certain layered products (DECnet/E, COBOL-81, DIBOL, FORTRAN/FOR, and FORTRAN/77) must be installed in account SY:[1,2].

The DCL HELP file has been changed to reflect the new features of DCL.

Refer to article sequence number 2Ø.1.4 N in the RSTS/E V8.Ø Maintenance Notebook for a description of some of the differences between DCL and BASIC-PLUS keyboard monitor environments.

2.7 DCL Installation

There are two versions of DCL available. One version supports the micro-RSTS spooling package and the other supports the standard RSTS/E spooling package. For information on the differences between the two spooling packages see sections 1.9.

During the BUILD phase at installation time, you must choose the version of DCL you want to use. The DCL build phase will type:

```
    Two versions of the DCL keyboard monitor are available:  one
    recognizes commands for the standard RSTS/E spooling package; the
    other recognizes commands for the micro-RSTS spooling package.

    Type MICRO to install DCL which supports the micro-RSTS spooling
    package.  Type STANDARD to install DCL which supports the standard
    spooling package.
    Which version of DCL do you wish to install <STANDARD> ?
```

If you answer "STANDARD" (the default), you will install the DCL that supports the standard RSTS/E spooling package. If you answer "MICRO", you will install the DCL that supports the new spooling package.

There is a different help package for each of the DCL versions. One help package descibes the micro-RSTS spooling commands, and the other describes the standard spooling commands. If you answer "STANDARD" to the above question DCL will reference the help file to SY:[1,2]DCL.HLP. This file will be copied onto your system. If you answer "MICRO" to the above question, then DCL will reference DCLA.HLP in the same account. Thus, it is possible to have both help files on your system concurrently.

It is not possible to install a single DCL that supports both spooling packages concurrently. See section 1.9 for details on using both spoolers on the same system.

2.8 RSX Emulator and Utilities Package

All of the RSX utilities and libraries are compatible with the corresponding utilities in RSX-11M V4.Ø.

The name of SLOTKB.TSK has been changed to STK.TSK.

The new TKB and the new SYSLIB are incompatible with the old TKB and the old SYSLIB. The RSX package must be completely installed. The old RSX and the new RSX can not be mixed. This is because of changes made in the AUTOLOAD code and bug fixes.

The RSX keyboard monitor CLEAN command has been removed.

If you include the RSX Emulator in your Monitor, the ALUN$ directive will accept user logicals. However, the logical must not be more than two characters.

2.9 RT11 Emulator

The RT11 keyboard monitor CLEAN command has been removed.

MACRO no longer supports the .WEAK directive.

2.1Ø BASIC-PLUS

NOEXTEND/EXTEND - BASIC-PLUS has changed its default from NOEXTEND mode to EXTEND mode in this release of RSTS/E. This will make BASIC-PLUS compatible with BASIC-PLUS-2 and allow the use of long variable names in programs, possibly requiring less documentation within the program. Feature patch Seq 4.1.4 F may be used to change the default from EXTEND mode to NOEXTEND mode. Users will still be able to specify these modes within their programs.

IMMEDIATE MODE - Starting with this release of RSTS/E, BASIC-PLUS will not allow, as the default, immediate mode statements to be executed from a program file being read by BASIC-PLUS as a result of execution of the "OLD" or "APPEND" commands. Since this feature is not compatible with other DIGITAL products, "no immediate mode from .BAS files" will become the default. Feature patch Seq 4.1.5 F may be used to allow your users to continue using immediate mode from .BAS files.

SEND/RECEIVE: SYS(CHR$(6%)+CHR$(18%)+ . . .) - Future releases of RSTS/E will not support SYS call 18. SYS call 22 replaces SYS call 18 and has been available on RSTS/E for several releases. Article Seq 4.1.9 F contains a patch that causes BASIC-PLUS to make note of all occurrences of the obsolete SYS call. You may wish to utilize this patch to create a temporary version of BASIC-PLUS in order to track down and alter all instances of SYS call 18 in your programs. (It should not be applied to the version of BASIC-PLUS used to compile and run Digital supplied RSTS/E CUSPs.)

2.10.1 Handling of floating point numbers

In previous releases, many SPRs have been submitted concerning floating point operations. In an effort to minimize customer confusion in the future, we are including here two typical SPRs and their responses.

The first problem involves possible inaccuracies with floating point numbers. Please note that although this example uses BASIC-PLUS as an example, the same is true of BASIC-PLUS-2.

PROBLEM ABSTRACT 1:

The BASIC statement PRINT INT(.16233*10E4), returns the value 16233, for single precision and 16232 for double precision.

The aberration you are observing is due to the inherent inaccuracy of the floating point number system. Consequently let us assure you that the problem has nothing to due with any of your own code, the BASIC-PLUS software you are invoking, or the machine you are running on.

The floating point number system is an attempt to represent a broad range of infinite precision values with only a finite number of bits. This system has been adopted in some form or another by all computer systems performing scientific application and is consequently widely used.

The floating point number system is analogous to performing decimal arithmetic with only a finite number of significant digits. Using only a finite number of significant digits means that some of the values you would like to represent will contain small inaccuracies, and that results of arithmetic operations will have to be rounded to the proper number of digits. Consider the following sequence of operations:

```
        (1)   int(99*(10/99)) = int(10) = 10.
```

This statement is algebraicly correct. However, suppose the calculations were carried out using a finite number of significant digits, say five. Notice that 10/99 = .10101010101010..., and therefore if only five significant bits are used, 10/99 = .10101, which is slightly less than the actual value. Using only five significant digits equation (1) becomes:

(2) int(99.000*.10101) = int(9.99999000)

Since only five significant digits are being used, the result of the product must be rounded to five places. Now 9.99999000 correctly rounded to five digits is 10.000, so that equation (2) becomes:

(3) int(99.000*.10101) = int(10.000) = 10.

Notice that even though the finite representation of 10/99 was inexact, the rounding procedure compensated for this and the correct answer was obtained.

Now, suppose the number of significant digits is doubled (to increase accuracy). In this case 10/99 = .1010101010 and equation (1) now becomes

(4) int(99.00000000*.1010101010) = int(9.9999999990000000000).

Again, the result of the product contains too many digits and must be rounded. Notice this time however that the correctly rounded product is 9.999999999, so that equation (4) becomes

(5) int(99.00000000*.1010101010) = int(9.999999999) = 9.

In this case, the rounding did not compensate for the initial error in representing 10/99. Also notice that the difference in the two results is due only to the number of significant digits used and the rounding procedure, it has nothing to due with the integer function.

The problem that you have observed is essentially the same as the example above with 10E4 playing the role of 99 and .16233 playing the role of 10/99. The only difference is that on the machine the arithmetic is performed with a finite number of binary digits, instead of a finite number of decimal digits. This difference is important to the problem because while .16233 can be represented in a finite number of decimal digits, it cannot be represented in a finite number of binary digits. (To convince you that this is possible, notice that 1/10 = .1 in base 10, but 1/10 = .00011001100110011001100011... in base 2)

In conclusion then, the statement

```
PRINT INT(.16233*1ØE4)
```

is returning the correct value in both the single and double precision cases. The difference you are observing is due to the inherent inaccuracy of the floating point computation .16233*1ØE4, which returns slightly different values for the different data types.

The second area of confusion is with mixed mode arithmetic when using BASIC-PLUS. This topic only pertains to BASIC-PLUS, not BASIC-PLUS-2.

PROBLEM ABSTRACT 2:

The BASIC-PLUS program:

```
1Ø A%(2) = Ø%
2Ø PRINT A%(2) + (32767 + 2)
```

does not produce the expected result (32769).

BASIC-PLUS will print -32767 when the given program is run. The reason for this is that the constants 2 and 32767 are ambiguous constants. To force a floating point representation of an integer constant, end it with a decimal point. To force an integer representation of a constant, end it with a % character. Constants without a decimal point or % character are termed ambiguous.

To detemine if an ambiguous constant will have an integer or a floating point representation, BASIC-PLUS uses the following rule. If an integer variable or constant occurs anywhere to the left of the ambiguous constant, BASIC-PLUS represents it in integer format. Otherwise, BASIC-PLUS treats the ambiguous constant as a floating point number.

In the example program the integer variable A%() does occur to the left of the ambiguous constants 2 and 32767, and as a result they are treated as integers. Because they are integers, the entire expression is evaluated as an integer, and the result is -32767.

This topic is covered in more detail in the BASIC-PLUS Language Manual under the topic of "Mixed Mode Arithmetic".

2.11 Documentation

The following manuals have been updated or revised for RSTS/E V8.Ø:

RSTS/E System Generation Manual
RSTS/E System Manager's Guide
RSTS/E DCL User's Guide
RSTS/E Task Builder Reference Manual
RSTS/E Programming Manual
RSTS/E System Directives Manual

Refer to the RSTS/E Documentation Directory and the Summary of Technical Changes in each updated or revised manual for more information.

2.12 Upgrading From RSTS/E V7.2 to RSTS/E V8.Ø

When upgrading from RSTS/E V7.2 or earlier releases to RSTS/E V8.Ø, the on-line SYSGEN feature of RSTS/E cannot be used on a V7.2 system to generate your first V8.Ø system. Due to differences between earlier releases of RSTS/E and RSTS/E V8.Ø, the first RSTS/E V8.Ø system generated must be generated off-line on a newly initialized disk.

Because of changes to the ERRLOG.FIL file required due to the increased function of the error-logging programs, earlier versions of ERRLOG.FIL must be deleted prior to the initial running of the RSTS/E V8.Ø ERRINT/ERRCPY. ERRCPY will create and initialize a new ERRLOG.FIL when it determines that the file is not present.

Because of changes to the ERRDAT.FIL file, it must be updated by running RSTS/E V8.Ø ERRBLD prior to the initial running of the RSTS/E V8.Ø ERRINT.

These procedures will avoid any conflict that may be caused by inconsistencies between the two versions.

NOTE

> When installing FMS V1.5 on RSTS/E V8.Ø, do not use the FMS patch entitled "FMSTIO" which is described in the FMS-11/RSTS Installation Guide and Release Notes, March 1982, AA-L324A-TC. It is also in the FMS distribution kit.
>
> This patch is not necessary when installing FMS with RSTS/E V8.Ø.

2.13 Old Problems Corrected

Various system problems found after the release of RSTS/E V7.2 were fixed in RSTS/E V8.Ø. All applicable mandatory RSTS/E Software Dispatch articles published through March 1983 have been incorporated into RSTS/E V8.Ø.

2.14 Distribution Media

RSTS/E V8.Ø is distributed on the following media:

9-Track Magnetic Tape - 8ØØ BPI (TU1Ø/TE1Ø/TSØ3/TU16/TE16/TU45/TU77)
9-Track Magnetic Tape - 16ØØ BPI (TS11/TE16/TU77)
RKØ5 Disk Cartridge
RKØ6 Disk Cartridge
RKØ7 Disk Cartridge
RLØ1 Disk Cartridge
RLØ2 Disk Cartridge

Please refer to Section 3.Ø of these Release Notes for more information on the distribution kits.

2.15 Optional Software

Optional software supported by RSTS/E V8.Ø includes the following:

ADE V2.Ø/V2.1

BASIC-PLUS-2 V2.Ø/V2.1

COBOL-11 V4.4

COBOL-81 V1.Ø/V1.1/V1.2

DATATRIEVE V2.Ø/V2.4

DECAL V2.Ø

DECMAIL/RSTS V1.Ø

DECnet/E V2.Ø

DECWORD/DP V1.1/V1.2

DIBOL V4.5/V5.Ø

DIBS/CTS-5ØØ V1.Ø

DMS-5ØØ V2.1

DX/RSTS V3.1

FMS-11 V1.5

FORTRAN IV V2.5/V2.6

FORTRAN-77 V4.Ø/V4.1

GIGI CAI PRIMER V1.Ø

GIGI SOFTWARE V1.Ø

INDENT V1.1/V1.2

MENU-11/RSTS V1.Ø

PLXY-11/RSTS V1.1

REGIS CAI PRIMER V1.Ø

RSTS/E-278Ø (CTS-5ØØ/278Ø) V3.Ø

RSTS/E 3271 Protocol Emulator V2.1

RSTS/E High Performance 278Ø/378Ø Protocol Emulator V1.1

SPM-RSTS/E V1.Ø

NOTE

Earlier versions of the above products are <u>not</u> supported on RSTS/E V8.Ø.

2.16 Support for RSTS/E V7.2

RSTS/E V7.2 will be supported for 6 months after the release of RSTS/E V8.Ø.

3.Ø RSTS/E V8.Ø Distribution Kits

3.1 Distribution Kit Order Numbers

This chapter provides you with the most current order numbers for the RSTS/E distribution kits and the RSTS/E update kits. It also provides you with the most current order numbers for the DECnet/E V2.Ø and RJ278Ø V8.Ø software distribution kits. For the most current order numbers of other optional software kit(s) see the bill of materials. Your RSTS/E System Generation Manual tells you to check these order numbers prior to beginning the actual system generation.

DIGITAL sends you distribution kits on one of the following:

- 9-track magnetic tape (8ØØbpi)
- 9-track magnetic tape (16ØØbpi)
- RKØ5 disk cartridge
- RLØ1 disk cartridge
- RLØ2 disk cartridge
- RKØ6 disk cartridge
- RKØ7 disk cartridge

3.2 RSTS/E V8.Ø Distribution Kit

The RSTS/E V8.Ø distribution kit consists of the tapes or disks that contain the programs and files you need to perform a system generation. This section provides the order numbers for the RSTS/E V8.Ø distribution kits. You should compare the numbers on the tapes or disks that make up your kit to the numbers shown here, when you are instructed to do so by the RSTS/E System Generation Manual.

DEC Order Number	Pack ID	Label Title
9-Track Magnetic Tape (8ØØ bpi)		
AP-2773K-BC	---	RSTS/E V8.Ø SYSGNK 8MT9
AP-2753J-BC	---	RSTS/E V8.Ø CSP18Ø 8MT9
AP-C725J-BC	---	RSTS/E V8.Ø CSP28Ø 8MT9
AP-L93ØC-BC	---	RSTS/E V8.Ø CSP38Ø 8MT9
AP-C883E-BC	---	RSTS/E V8.Ø CSP48Ø 8MT9
AP-V35ØA-BC	---	RSTS/E V8.Ø CSP58Ø 8MT9
AP-V351A-BC	---	RSTS/E V8.Ø CSP68Ø 8MT9
9-Track Magnetic Tape (16ØØ bpi)		
BB-H751K-BC	---	RSTS/E V8.Ø SYSGNK 16MT9
BB-V353A-BC	---	RSTS/E V8.Ø CSP18Ø 16MT9
RKØ5 Disk Cartridge		
AN-2771K-BC	SYSGNK	RSTS/E V8.Ø SYSGNK RK5
AN-2751J-BC	CSP18Ø	RSTS/E V8.Ø CSP18Ø RK5
AN-5444J-BC	CSP28Ø	RSTS/E V8.Ø CSP28Ø RK5
AN-M276C-BC	CSP38Ø	RSTS/E V8.Ø CSP38Ø RK5
AN-C885E-BC	CSP48Ø	RSTS/E V8.Ø CSP48Ø RK5
AN-V487A-BC	CSP58Ø	RSTS/E V8.Ø CSP58Ø RK5
AN-V488A-BC	CSP68Ø	RSTS/E V8.Ø CSP68Ø RK5
RKØ6 Disk Cartridge		
AM-2774K-BC	SYSGNK	RSTS/E V8.Ø SYSGNK RK6
AM-V356A-BC	CSP18Ø	RSTS/E V8.Ø CSP18Ø RK6
RKØ7 Disk Cartridge		
AY-D526K-BC	SYSGNK	RSTS/E V8.Ø SYSGNK RK7
RLØ1 Disk Cartridge		
AX-D527K-BC	SYSGNK	RSTS/E V8.Ø SYSGNK RL1
AX-D528J-BC	CSP18Ø	RSTS/E V8.Ø CSP18Ø RL1
AX-V354A-BC	CSP28Ø	RSTS/E V8.Ø CSP28Ø RL1
AX-V355A-BC	CSP38Ø	RSTS/E V8.Ø CSP38Ø RL1
RLØ2 Disk Cartridge		
BC-JØ26E-BC	SYSGNK	RSTS/E V8.Ø SYSGNK RL2
BC-V352A-BC	CSP18Ø	RSTS/E V8.Ø CSP18Ø RL2

3.1.2 RSTS/E V8.Ø Update Kit

This section provides the order numbers for the RSTS/E V8.Ø Update Kit A. This kit consists of the tape(s) or disk(s) that contain feature and mandatory patches and replacement modules. You receive this kit approximately every three months between releases, and the kits are labeled A, B, C, and so forth. Because each kit is cumulative, use the most current one. For example, Update Kit C contains all of the patches and replacement modules that were on Update Kits A and B plus some new patches and replacement modules. Pack IDs for disk update kits are always in the form "Patchz", where "z" is the version of the update kit.

DEC Order Number	Pack ID	Label Title
<u>9-Track Magnetic Tape (8ØØ bpi)</u>		
AP-PØ13A-BC	---	RSTS/E V8.Ø Update A 8MT9
<u>9-Track Magnetic Tape (16ØØ bpi)</u>		
BB-PØ16A-BC	---	RSTS/E V8.Ø Update A 16MT9
<u>RKØ5 Disk Cartridge</u>		
AN-PØ14A-BC	PATCHA	RSTS/E V8.Ø Update A RK5
<u>RKØ6 Disk Cartridge</u>		
AM-PØ18A-BC	PATCHA	RSTS/E V8.Ø Update A RK6
<u>RKØ7 Disk Cartridge</u>		
AY-PØ19A-BC	PATCHA	RSTS/E V8.Ø Update A RK7
<u>RLØ1 Disk Cartridge</u>		
AX-PØ17A-BC	PATCHA	RSTS/E V8.Ø Update A RL1
<u>RLØ2 Disk Cartridge</u>		
BC-PØ15A-BC	PATCHA	RSTS/E V8.Ø Update A RL2

3.1.3 DECnet/E V2.0 Software Kit

This section provides the order numbers for the DECnet/E V2.0 Software Kit. The reason for providing you with the order numbers for this particular optional software is that one of the configuration questions asks if you want to support DECnet/E. If you answer YES to this question, then a later task asks you to mount the correct tape or disk on which you received the DECnet/E programs and files. (See Phase 4 in the _RSTS/E System Generation Manual_.)

DEC Order Number	Pack ID	Contents
9-Track Magnetic Tape (800 bpi)		
AP-M478A-BC	---	DECnet/E Programs and Files
9-Track Magnetic Tape (1600 bpi)		
BB-M481A-BC	---	DECnet/E Programs and Files
RK05 Disk Cartridge		
AN-M479A-BC	DECNTC	DECnet/E Programs and Files
RK06 Disk Cartridge		
AM-M483A-BC	DECNTC	DECnet/E Programs and Files
RK07 Disk Cartridge		
MY-M484A-BC	DECNTC	DECnet/E Programs and Files
RL01 Disk Cartridge		
MX-M482A-BC	DECNTC	DECnet/E Programs and Files
RL02 Disk Cartridge		
BC-M480A-BC	DECNTC	DECnet/E Programs and Files

3.1.4 RJ2780 Software Kit Order Numbers

This section provides the order numbers for the RJ2780 V3.0 Software Kits.

DEC Order Number	Pack ID	Contents
9-Track Magnetic Tape (800 bpi)		
AP-2636D-BC	---	RJ2780 Programs and Files
9-Track Magnetic Tape (1600 bpi)		
BB-K173A-BC	---	RJ2780 Programs and Files
RK05 Disk Cartridge		
AN-2633D-BC	ORCOAD	RJ2780 Programs and Files
RK06 Disk Cartridge		
AM-C729D-BC	ORCOAD	RJ2780 Programs and Files
RK07 Disk Cartridge		
AY-H340D-BC	ORCOAD	RJ2780 Programs and Files
RL01 Disk Cartridge		
AX-H339D-BC	ORCOAD	RJ2780 Programs and Files
RL02 Disk Cartridge		
BC-K172A-BC	ORCOAD	RJ2780 Programs and Files

After you check the order numbers for all your kits, return to Phase 1, Task 2 of your RSTS/E System Generation Manual to find out the next task you should perform.

3.1.5 Location of Control Files

DIGITAL distributes RSTS/E system programs, bundled software, and optional software on control files. These control files are located on the distribution tapes or disks. You process these control files during Phase 8 of system generation to build the system programs, bundled software, and optional software your installation needs.

This section shows where each of the control files is located. Be sure to keep this section of the release notes open while you are performing Phase 8 of system generation.

The control files used to build the system library and program packages are listed on the next page.

The name immediately above each list is the Pack ID. For example, SYSGNK is the pack ID of the RLØ1 distribution disk that contains the RSX.CTL, TECO.CTL, and MICRO.CTL control files. Likewise, CSP18Ø is the name of the 8ØØ BPI distribution tape that contains the MICRO.CTL, RSX.CTL, and TECO.CTL control files.

In the case of magnetic tape distribution media, the control files have been arranged to limit the number of times that the tape must rewind if the control files are specified in the order in which they are listed on the next page.

Note that if more than one control file is used, the files must be specified in the order listed on the next page. For example, if you are building the Spooling and Operator Services Package, the Device Exerciser Package, and some of the unsupported programs (and your distribution tape is 8ØØ BPI), the order in which you should specify the control files is SPLER.CTL, DEVTST.CTL, and UNSUPP.CTL.

9-Track Magnetic Tape (8ØØ BPI) or RKØ5 Disk Cartridge:

```
        SYSGNK:       CSP18Ø:         CSP28Ø:         CSP38Ø:
        -------       -------         -------         --------
                      BUILD .CTL      MICRO .CTL      BIGPRG.CTL
                                      RSX   .CTL      DCL   .CTL
                                      TECO  .CTL      HELP  .CTL
                                                      EDT   .CTL


        CSP48Ø:       CSP58Ø:         CSP68Ø:
        -------       -------         -------
        SPLER .CTL    BACKUP.CTL      RMS11 .CTL
        DEVTST.CTL    SORT  .CTL      SPL   .CTL
        UNSUPP.CTL
```

RLØ1 Disk Cartridge:

```
        SYSGNK:         CSP18Ø:         CSP28Ø:         CSP38Ø:
        -------         -------         -------         -------
        RSX   .CTL      BUILD .CTL      SPLER .CTL      RMS11 .CTL
        TECO  .CTL      BIGPRG.CTL      BACKUP.CTL      SORT  .CTL
        MICRO .CTL      DCL   .CTL      DEVTST.CTL      SPL   .CTL
                        HELP  .CTL      UNSUPP.CTL
                        EDT   .CTL
```

9-Track Magnetic Tape (16ØØ BPI)

```
        SYSGNK:         CSP18Ø:
        -------         -------
        RSX   .CTL      BACKUP.CTL
        BUILD .CTL      RSX   .CTL
        BIGPRG.CTL      RMS11 .CTL
        DCL   .CTL      SORT  .CTL
        HELP  .CTL      TECO  .CTL
        EDT   .CTL      DEVTST.CTL
        SPLER .CTL      UNSUPP.CTL
        MICRO .CTL      SPL   .CTL
```

RLØ2 Disk Cartridge

```
SYSGNK:         CSP18Ø:
------          ------
RSX   .CTL      BACKUP.CTL
BUILD .CTL      RSX   .CTL
BIGPRG.CTL      RMS11 .CTL
DCL   .CTL      SORT  .CTL
HELP  .CTL      TECO  .CTL
EDT   .CTL      DEVTST.CTL
SPLER .CTL      UNSUPP.CTL
MICRO .CTL      SPL   .CTL
```

RKØ6 Disk Cartridge:

```
SYSGNK:         CSP18Ø:
------          ------
RSX   .CTL      BACKUP.CTL
BUILD .CTL      RSX   .CTL
BIGPRG.CTL      RMS11 .CTL
DCL   .CTL      SORT  .CTL
HELP  .CTL      TECO  .CTL
EDT   .CTL      SPL   .CTL
SPLER .CTL
MICRO .CTL
DEVTST.CTL
UNSUPP.CTL
```

RKØ7 Disk Cartridge:

```
SYSGNK:
------
RSX   .CTL
BUILD .CTL
BIGPRG.CTL
DCL   .CTL
HELP  .CTL
EDT   .CTL
SPLER .CTL
BACKUP.CTL
MICRO .CTL
RMS11 .CTL
SORT  .CTL
TECO  .CTL
DEVTST.CTL
UNSUPP.CTL
SPL   .CTL
```

3.1.6 Contents of Control Files

This section explains the contents of each of the control files distributed on your RSTS/E distribution kit. 'See Phase 8 of the RSTS/E System Generation Manual for instructions on how to process these control files.

BUILD .CTL, BACKUP.CTL, BIGPRG.CTL, SPLER .CTL, DEVTST.CTL

Contain the commands necessary to build the standard RSTS/E system library. The exact contents of these files are described in Appendix D of the RSTS/E System Generation Manual.

RSX .CTL

Contains the commands necessary to build the RSX Run-Time System and support utilities. Note that you must build RSX before building EDT and most optional layered products.

SORT .CTL

Contains the commands necessary to build the PDP-11 SORT package. Note that you must build RSX and then RMS before building the SORT package.

Note that all patches published for SORT-11 through September 1982 have already been applied to the version of SORT-11 included on the distribution tape or disk.

DCL .CTL

Contains the commands necessary to build DCL and its associated programs and files.

HELP .CTL

Contains the commands necessary to build the HELP system program and its associated data files.

EDT .CTL

Contains the commands necessary to build EDT V2.Ø. Note that EDT uses the RSX run-time system and utilities, so you must build RSX before you can build EDT.

SPL .CTL

Contains the commands necessary to build the new micro-RSTS Spooling package. Note that the new spooling package cannot be started until you build the RSX run-time system.

TECO .CTL

Contains the commands necessary to build a number of TECO programs. The BUILD program should be used with this file if you plan to install TECO on your system. In addition to the TECO Run-Time System, this command file will install several TECO programs which we have found to be generally useful.

For documentation on the new features of TECO, the new TECO programs, and instructions for tailoring the TECO Run-Time System for your system, see the file TECORN.DOC on the distribution kit.

UNSUPP.CTL

Contains the commands necessary to build a number of programs which, though unsupported, are used extensively by the RSTS/E development group. They are included because you may find these programs, or modified versions of them, to be useful on your system.

Unsupported BASIC-PLUS programs are not supplied in pre-compiled form (.TSK), but only in source form (.BAS). Be sure to build the unsupported components using the BASIC-PLUS run-time system.

NOTE

The programs built by UNSUPP.CTL are NOT released products supported by DIGITAL for use on RSTS/E, and their inclusion on the distribution kit does not imply recommendation or endorsement by DIGITAL regarding their use. There is no commitment by DIGITAL to continue to include any of these modules in future distributions of RSTS/E. Each of these modules is to be considered "user supported".

The RSTS/E Development Group would like to be informed of any problems with or suggestions concerning these components as supplied with RSTS/E V8.Ø. A reasonable effort will be made to respond to such feedback, although problem fixes cannot be guaranteed. If you choose to provide reports on unsupported components via the SPR mechanism, please check Priority 5, "Documentation/ Suggestion".

RMS11.CTL

Contains the commands necessary to build RMS V2.Ø.

3.2 SORT-11

SORT-11 V2.Ø is distributed as a part of RSTS/E V8.Ø. Note that as in RSTS/E V7.2, all output files will be made non-contiguous by default. (Contiguous input files by default will produce non-contiguous output files, unless the /CO switch is specified for the output file.)

3.3 EDT V2.Ø

EDT V2.Ø is included as part of RSTS/E V8.Ø. This version of EDT must be built by each site, since you have a choice of 3 configurations of the editor. See Phase 8 of the RSTS/E System Generation Manual for installation instructions.

3.4 Unsupported Components

NOTE

> The following modules are NOT released products supported by DIGITAL for use on RSTS/E, and their inclusion on the distribution kit does not imply recommendation or endorsement by DIGITAL regarding their use. There is no commitment by DIGITAL to continue to include any of these modules in future distributions of RSTS/E. Each of these modules is to be considered "user supported".
>
> The RSTS/E Development Group would like to be informed of any problems with or suggestions concerning these components, as supplied with RSTS/E V8.Ø. A reasonable effort will be made to respond to such feedback, although problem fixes cannot be guaranteed. If you choose to provide reports on unsupported components via the SPR mechanism, please check Priority 5, "Documentation/Suggestion."

3.4.1 RNO

RNO is available from the DECUS library and is widely used by many users of RSTS/E. It is included on the distribution kit for the convenience of those users who wish access to this program.

RNO is a variant of the RUNOFF program (which is part of the RSTS/E library of system programs). RNO is written in MACRO-11 and generally executes much faster than RUNOFF, which is written in BASIC-PLUS. However, RNO is not completely compatible with RUNOFF, and may sometimes produce different results. This version of RNO was originally developed on RSX-11M and was distributed with the V3 release of that system.

For those users wishing to make their own modifications to RNO, sources are included with this distribution. The RSTS/E Development Group would appreciate hearing from any user making modifications in the hope that such feedback will help to improve the program for future releases.

3.4.2 TECO

TECO is a character-oriented text editor that is implemented on all major DIGITAL operating systems. TECO has many powerful features as a text editor and is especially popular among programmers. Many users, however, find TECO's set of commands rather cryptic, while others prefer line-oriented editors. For these users, an editor like the DEC standard editor, EDT, is more appropriate.

A copy of the PDP-11 TECO User's Guide is included in the documentation set and a document file is included on the distribution kit explaining the use of the TECO Video Terminal Editor (VTEDIT). For installation instructions, see Phase 8 of the RSTS/E System Generation Manual.

3.4.3 SRCCOM.SAV and DUMP.SAV

RSTS/E users who are also users of RT11 may find SRCCOM.SAV (a text comparison program) and DUMP.SAV (a file dump/display program) useful on RSTS/E, although such use is not supported by DIGITAL.

The SRCCOM.SAV and DUMP.SAV files on the RSTS/E distribution kits are identical to the SRCCOM.SAV and DUMP.SAV supplied with RT11 V4.Ø. Installation of each program would consist of copying it from the distribution kit with PIP and insuring that it is named to the RT11 run-time system. This has to be done manually, as no control file for automatic installation is provided.

3.4.4 RSTSFLX

RSTSFLX is a program which will run on any VAX/VMS system and read files from a RSTS disk and copy them to a VMS disk. This allows for easy interchange from RSTS to VMS. More information is available on the distribution kit in RSTSFL.DOC and RSTSFL.HLP.

4.Ø RSTS/E V8.Ø Sample Installation Procedure

The following outlines the steps required to build a new RSTS/E V8.Ø operating system. Once these steps have been performed, the system is ready for timesharing.

The time estimates given are for a RSTS/E V8.Ø system on an 11/7Ø with 2 RPØ6 disks, using BASIC-PLUS as the primary run-time system and IMAGE copying the system RPØ6 to the non-system RPØ6. They do not include time for troubleshooting hardware or for pattern checking large disks. The use of a smaller processor will, of course, increase the time required.

NOTE

> It is recommended that all steps be performed from a hard-copy terminal and that the terminal printout be saved for future reference.

1. System Generation 2 - 6 Hours

Bootstrap the distribution medium, tailor the SYSGEN monitor, configure the RSTS/E monitor, tailor the RSTS/E monitor, start the RSTS/E monitor, compile and patch, if necessary, the CUSPs, and install and patch optional software included on the RSTS/E distribution kit.

(For more detailed instructions on how to build a RSTS/E monitor, see the RSTS/E System Generation Manual.)

The monitor/CUSP system generation demonstrates that the hardware supports RSTS/E; that MACRO programs can be assembled, linked, patched, and executed; that the primary run-time system is operational; that the standard library programs can be compiled (and patched, if necessary); and that certain library programs, especially file transfer utilities, are operational.

2. Install Optional Software (Installation Dependent)

If optional software (e.g., COBOL) needs to be installed, you should instal and patch it at this time. Sample procedures for optional software products should be run to indicate that the hardware supports the product and that the installation is complete and correct.

3. Configure Control Files and Re-Start the System 1 - 4 Hours

Configure the standard start control files, START.CTL and CRASH.CTL, set up the standard account file, ACCT.SYS, establish the system message file, NOTICE.TXT, shut down the system, use the INIT HARDWR LIST suboption to print a System Configuration List, use the INIT SET MODEM suboption to enable dial-up lines, and restart the system.

This demonstrates that the system can be started, shut down, and re-started. At this time it can be demonstrated that all local terminals are accessible and, if feasible, dial-up lines can be checked. The System Configuration List should be posted on the system for future reference.

4. Save and Restore the System Using SAVE/RESTORE 30 - 40 Minutes

Save the system disk using the SAVE or IMAGE option of SAVE/RESTORE, boot the saved media, and, if SAVE was used, restore to another disk, boot the new system disk, re-install the monitor SIL, and start the system.

This demonstrates SAVE/RESTORE's ability to backup and restore the system and exercises the disk I/O capability of the system. It also leaves you with a full, bootable, system backup.

Note that you must supply the additional media needed to perform SAVE/RESTORE. If additional media are not available to perform this step, it may be omitted, but it is strongly recommended that this procedure be followed.

5. Test Hardware Devices With DEVTST (Configuration Dependent)

Run DEVTST exercisers for the console terminal and any other devices configured for the system. Refer to the System Configuration List created in Step 3, above, and the Appendix B of the RSTS/E System Generation Manual.

This demonstrates that the devices configured for the system are functional.

At the conclusion of the above procedure, the installation is complete and most hardware will have been exercised sufficiently to determine that it operates within the system environment. Note, however, that the sample procedure is not designed to be a hardware exerciser or a test of system performance.

5.0 RMS-11 V2.0 FOR RSTS/E V8.0, MAY 1983

5.1 RMS V2.0 ACCESS METHODS FEATURES

The following new facilities appear in the RMS-11 V2.0 Access Methods.

5.1.1 Support for Clustering and Task Independence

The memory-resident library RMSRES can be clustered with any other resident library which supports clustering; this allows RMS to share task virtual memory with other libraries. See your FMS and language documentation to determine if a particular library supports clustering.

Task-independence means that once a program is linked with this library, the library can be rebuilt or replaced without requiring that the task linked to it be rebuilt.

Linking with this library requires use of the RMSRLX.ODL ODL file. Include the element RMSROT in the .ROOT statement of your ODL, and the indirect reference @LB:RMSRLX in your ODL file. If you used RMSRLX.ODL in this manner with V1.8, no change should be required (if you made reference to the RMSALL factor, this is no longer necessary but will do no harm).

To use this library as a cluster library, your task simply needs to specify the CLSTR option when linking the task.

5.1.2 Segmented Resident Library

RMSRES has been partitioned into seven separate library segments, not all of which need to be physically resident in memory.

This allows "demand-paging" of the library segments: only those segments containing code which is, or has most recently been, in use must be resident, leaving the balance of the library segments eligible for replacement by tasks which can make use of the physical memory.

Each of the seven segments is 8 KB or less in size, with not more than two segments ever required simultaneously by a given user task.

- RMSRES.TSK is the "root" of the segmented library. As before, this TSK and its associated STB file are the only files needed to link user tasks against RMS.

- RMSLBA.TSK is the library segment that supports $DISPLAY, $ERASE, $PARSE, $SEARCH, $RENAME, sequential file $CREATE and $OPEN, common portions of relative and indexed file $CREATE and $OPEN, common $CLOSE (and file operation exit) processing.

- RMSLBB.TSK is the library segment that supports sequential file $CLOSE and all sequential file record operations, $FREE, magtape functions, block I/O record operations, and the common internal EXTEND function.

- RMSLBC.TSK is the library segment that supports all relative file operations, plus the $EXTEND function.
- RMSLBD.TSK is the library segment that supports indexed file $OPEN, $CLOSE, $CONNECT, $DISCONNECT, $FIND, and $GET operations.
- RMSLBE.TSK is the library segment that supports indexed file $DELETE/$PUT/$UPDATE operations.
- RMSLBF.TSK is the library segment that supports data record insertion and index-update for indexed $PUT/$UPDATE, plus the bucket allocation routine, and indexed $CREATE.

Note that although the library segments do not have to be physically resident, **all** library segments must be installed before any use of the resident library (including use of RMS utilities which are built against the resident library).

Installation of the resident library is done during V2.0 installation by the autoconfigure program, RMSACF, and subsequently by your system manager through the use of the RMSACF program or the use of UTI ADD LIBRARY commands. See Section 5.8, "RMS V2.0 Installation," for more details.

5.1.3 Disk-Overlaid RMS

New versions of the V1.8 RMS ODLs are provided on the distribution medium. These ODLs are:

- RMS11S.ODL

  Designed to use minimal virtual address space for support of sequential and relative file organizations, this ODL file adds about 6.5KB to the task size.

- RMS11X.ODL

  Designed to use minimal virtual address space for full support of all file organizations, this ODL is structured to add about 10KB to the task size.

- RMS12X.ODL

  This ODL adds about 12KB to the task size, supports all file organizations, and offers a good compromise between performance for record operations on indexed files and use of task virtual address space.

- RMS11.ODL.

  This is a prototype ODL file that you can tailor to your own needs.

The V1.8 versions of these ODLs will still work with RMS V2.0 but the new versions of the ODLs will be more efficient, generally requiring less memory and less disk space for the task image. For more information on V1.8 comparisons, see Section 5.3.

In addition, two new ODLs are provided with V2.0:

- RMS12S.ODL

  Supporting only sequential and relative file organizations, this ODL adds about 9KB to the task size and is designed to offer a good compromise between performance and use of task virtual address space.

- DAP11X.ODL

  Structured to add about 14KB to the task size, this ODL is designed to use minimal virtual address space for support of all file organizations and (on systems with the required DECnet support) remote access facilities.

5.1.4 Remote File and Record Access via DECNET

If suitable DECNET facilities exist on your system and on the target system, RMS-11 V2.0 will allow file and record access to files residing on other network nodes, provided that such nodes include an RMS-based file access listener (FAL).

For most purposes, remote access is indistinguishable from local access, although performance may not be equivalent. The following general limitations apply:

- RMS-11 generally does not support remote functions which are not supported locally.

- Certain RMS functions (wildcard support, $PARSE, $SEARCH, $RENAME, and transmission of device, directory, and file identifiers) are not supported by the data access protocol (DAP) used, and hence cannot be executed remotely.

- Certain FALs do not support the full set of RMS functions expressible via DAP, and hence further limit remote access. For example, the current RSTS/E RMS FAL does not support record access to indexed files.

- High-level languages may not allow expression of the file specification necessary to establish contact with a remote node.

To operate upon a remote file, your program must include a node specification for the remote file and you must include the RMS remote access code when you build your task.

5.1.4.1 **Remote Node Specification** - You must include a remote node specification at the start of the file name string or the default name string you provide to an $OPEN, $CREATE, or $ERASE operation. In addition, your file name string and default name string must conform to the DIGITAL Command Language (DCL) file specification syntax rules, and the file specification resulting from the merge of these two strings must conform to the file specification conventions on the destination system as well.

In general, a file specification consisting of the elements:

```
nodespec::devicespec:[directoryspec]filename.filetype;fileversion
```

in the order (and with the delimiters) presented is appropriate, where the elements beyond the nodespec conform to the conventions of the target system. Elements not present will be defaulted according to the conventions of the target system.

The nodespec:: element is of the form:

```
node"user password"::
```

where **node** is the (required) destination node name and **"user password"** is an (optional) access control string containing login information acceptable to the remote system.

If **"user password"** is provided, the device and directory defaults and access privileges of that remote account are acquired; if not, the device and directory defaults and access privileges of the default DECNET account on the remote system are acquired.

5.1.4.2 **Linking Programs with RMSDAP Support** - To link your program with remote RMSDAP modules, you can use either the disk-overlaid version, or reference the DAPRES resident library:

- To link with the disk-overlaid version, reference the DAP11X.ODL file in your ODL. And, as with other RMS ODLs, you reference the RMS root modules as RMSROT, and specify other RMS modules as the cotree RMSALL.
- To link with the DAPRES resident library, use the ODL file DAPRLX, and specify RMSRES and DAPRES in your TKB command file. RMSRES and DAPRES can be specified as single resident libraries, or as clustered resident libraries.

Note that DAPRES is a clusterable, task-independent library as defined previously for RMSRES. It is not a segmented library however.

5.1.4.3 **RMSDAP Environments** - RMS-based FALs are currently available on VAX/VMS, RSTS/E, RSX-11M, and RSX-11M-PLUS systems. The version of data access protocol which is used must be at least Version 5.6 or greater; this means that you must have at least Version 2.0 of DECNET/E, Version 3.1 of DECNET for 11M, version 1.1 of DECNET for 11MPLUS, or Version 2.0 of DECNET for VAX.

Note for RSTS users: you will need patch numbers 31.23.1 and 31.23.4 to DECNET/E before using the RMS remote access features. (These patches will automatically be applied if you install/patch DECNET/E when installing RSTS/E 8.0.)

For more information on remote access using RMS-11, see Appendix B of the <u>RMS-11 User's Guide</u>.

5.1.5 New File and Directory Operations

RMS-11 V2.0 supports several new file and directory operations.

5.1.5.1 $RENAME - $RENAME is a new RMS operation macro which can be used to replace the directory entry for a file (change the name of a file). The file specification for the new directory entry must not specify a new device or node for the file, but can specify a different file name and file extension.

5.1.5.2 $PARSE - $PARSE is a new RMS operation macro which can be used to analyze a file specification.

The PARSE operation merges the file specification with the default file specification and with the system defaults, and analyzes the result. If you specify wildcarding, the $PARSE operation constructs a match-pattern; this match-pattern can be used by later wildcard SEARCH operations.

5.1.5.3 $SEARCH - $SEARCH is a new RMS-11 operation macro which can be used to find the next (or first) file that matches a match-pattern (constructed as the expanded string by a previous PARSE operation).

5.1.6 Wildcarding

You specify wildcarding by including wildcard characters in the file specification (or the default file specification): wildcards can be specified in the directory entry, file name, or type. The guidelines for wildcarding match those described in the RSTS/E System User's Guide.

See the RMS-11 Macro Programmer's Guide for a full description of how to use the new directory operations and wildcarding. In addition, you may wish to examine the demo programs which are provided.

5.1.7 Random Access to Fixed-Format Sequential Files

For sequential files with records of fixed format, random access for $GET, $FIND, and $PUT may be specified: it is very similar to random access to relative files, with some exceptions.

Your program specifies keyed access by setting the RB$KEY code in the 1-byte RAC field of the RAB, and by specifying the address of a "relative record number" in the KBF field of the RAB. Key size must be specified as 4 in the KSZ field of the RAB.

Some differences from relative file handling are:

- There is NO initialization of the space in the file. If the first operation is a random $PUT to the file at relative record number 5, the "cells" preceding record number 5 may contain garbage: whatever data happened to be there.

  (This is similar to RMS-32.)

- There is also no maximum record number (except as limited by the 4 byte specification of the key). If your program specifies a random $PUT at relative record number 999, the file will be extended to the appropriate size (but not initialized).

- $PUT overwrites any existing record in the target location.

- There is no support for $DELETE to a sequential file.

5.1.8 Increased Device Independence

STREAM and VFC files may now be created on unit-record devices to avoid the need for special-case code in copy-type operations.

- In the case of VFC files, the record header is thrown away on output unless it is a "print format" file, which RMS-11 now supports as RMS-32 does.

- For stream files, if none of the 3 carriage control bits is set (print file format, carriage control or FORTRAN carriage control), and if the last character is not a line feed, form feed or vertical tab, then carriage-return/line feed (CR/LF) is appended at the end of the record.

- For stream files, where either of the carriage control or FORTRAN carriage control attributes is set, then if the last two characters of the record are CR/LF, the trailing CR/LF is stripped off, and then definition of the carriage control (CR or FTN) attribute is applied.

For similar ease-of-copying reasons, RMS-11 now allows creation of relative and indexed files for output to non-disk devices (though for tape, record format must be VAR or FIX).

5.1.9 More Usable Block I/O Facilities

The following enhancements have been made for block I/O:

- Block I/O may now be used to copy files without the need to "fix up" the file header attributes.

  Block I/O $READ operations now generally respect the logical EOF header attribute (exceptions are documented in the _RMS-11 User's Guide_ and in the _RMS-11 Macro Programmer's Guide_).

- Block I/O may also be used for sequential access, by zeroing the RAB BKT field, allowing sequential copy and update operations.

5.2 RMS V2.0 UTILITIES FEATURES

5.2.1 New File Design Utility

The RMS-11 File Design Utility (RMSDES) is a new utility that allows you to interactively design and create files. RMSDES supersedes the RMS-11 File Definition Utility (RMSDEF).

Some of the features of RMSDES are:

- Full RMS-11 file structure capabilities -- you can design and create any type of RMS-11 file to contain your data records, tailoring it to meet your program's data processing requirements exactly.

- Error detection -- RMSDES can detect many kinds of errors and omissions in the attribute information you supply. Because RMSDES is interactive, you can correct the errors immediately, before you create and load records into the file.

- Saved file description -- You can save a description of a file design for future use, whether or not you have created a file based on the description. This is useful if you want to complete an unfinished file design at a later date, or modify a file design to accommodate a new use.

- Copied file description -- RMSDES can construct a file description based on the attributes of an existing data file. You can either create a new file like the existing file, or modify the file design before creating a new file.

- Default calculations -- RMSDES can calculate defaults for many attributes.

- Extensive help facilities are provided for RMSDES.

5.2.2 New Features for RMSDSP

RMSDSP has a completely reorganized and much more readable format. It also includes the following new features:

- The combination of options /BP/FU now displays more complete information about the contents of backup container files.

- The size field lists high block as well.

- DSP displays the clustersize of a file on disk.

- The protection information is given as a number (for example, <60>), and is also "decomposed" into Owner, Group, World protection fields.

- If RMSDSP is given a privileged protection code, it will now drop privilege around wildcard operations, creation of the output file, and around command file processing. In this case, users of RMSDSP cannot look at files for which they do not normally have access privileges.

  However, note that if the system manager patches the FIP to allow nonprivileged users to see all files with the directory lookup, then RMSDSP users will also have read access to those files, regardless of protection code.

5.2.3 New Features for RMSCNV

RMSCNV has been rewritten and is smaller and generally faster than the 1.8 version.

Other features of RMSCNV are:

- The /CA switch can be used to dynamically create an output file, by copying file attributes from another file. (That is, you do not need to run RMSDES or RMSDEF to create the output file before doing the convert, if you already have a prototype file whose attributes can be copied for the output file.)

- The /IM switch can be used to copy files in IMAGE (block) mode.

- The /EO switch can be used to change ASCII stream files that end with a CTRL/Z character to end with null characters. This switch is provided for help in converting certain non-RMS stream files into a format readable by RMS V2. (See also Section 5.3.6, "Other V1.8 Compatibility Notes.")

- The /KN:keyname switch can be used to convert files based on a keyname (as opposed to a key number, as in the /KR switch).

- The /ML switch allows you to explicitly set the limit of memory allocation.

- The /SL switch, summary listing, has been significantly enhanced. It now produces several summaries: summaries of options used for the convert, summaries of the file attributes for the files involved in the convert, a message summary of errors and warnings, and a processing summary.

- Remote access capability is provided for suitable configurations. Your system and the target system must include DECNET support (at least V2.0 of DECNET/E). The target system must also include an RMS-based file access listener (FAL). And you must rebuild RMSCNV to include the remote access capability (see Section 5.8.2, "Utility Configurations").

5.2.4 Additional Utility Features

- RMSIFL has been rewritten and incorporates a number of bug fixes.

  In addition, the algorithms for allocating work file space for sorting have been changed and in most cases, the amount of space needed for the sort should be decreased.

  The /DE switch has been enhanced to allow specification of a different device for IFL's temporary files.

- Command and ODL files are provided for rebuilding the RMS utilities, to use different forms of the RMS-11 access methods. See Section 5.8.2, "Utility Configurations."

- All the utilities have been changed to use lowercase format.

- The error message format has been changed for all the utilities except RMSDEF and RMSIFL.

5.3 COMPARISONS WITH V1.8

Conversion from RMS-11 V1.8 to RMS-11 V2.0 has been made transparent where possible. Structural changes required by feature additions, however, may be significant to some programs.

5.3.1 Memory-Resident Library (RMSRES)

RMSRES now contains all RMS support code save for minor linkage code in the task. Further, it has been configured as multiple segments, not all of which need be resident in memory simultaneously, thus dramatically reducing instantaneous physical memory overhead. The library is also "task-independent," in that it can be rebuilt or replaced without having to rebuild tasks which use it, and "clusterable," which means that it can share task virtual memory with other libraries that support "clustering" (see your FMS and language documentation).

These features required changes in the file RMSRLX.ODL, the file normally used to build tasks using RMSRES. Tasks which used RMSRLX.ODL with RMS-11 V1.8 should continue to build properly using the new RMSRLX.ODL and RMSRES. Build files which used RMSRES but did not use RMSRLX.ODL with RMS-11 V1.8 must be changed to use (or incorporate the contents of) the new RMSRLX.ODL file.

5.3.2 Memory-Resident Library (RMSSEQ)

RMS-11 V2.0 contains no library equivalent to the RMSSEQ memory-resident library included with RMS-11 V1.8. RMSSEQ offered no advantages in use of task virtual memory, and no advantages in use of system physical memory which are not equally attainable through use of disk-overlaid RMS at negligible performance cost. In addition, structural constraints made it impossible for RMSSEQ to be clusterable or task-independent.

The improved packaging of the full-function library RMSRES should make that a suitable replacement for RMSSEQ in most environments. Where this is inappropriate, disk-overlaid or non-overlaid RMS can be used.

5.3.3 Memory-Resident Library (DAPRES)

If your system includes the necessary DECNET facilities, the new resident library DAPRES provides support for remote file and record access as well as local access. Like RMSRES, DAPRES is both task-independent and clusterable. In particular, a task using RMSRES may include remote-access support via DAPRES with negligible increase in task virtual address space requirement by clustering the two libraries: the file DAPRLX.ODL should be used to generate this configuration.

5.3.4 Disk-Overlaid RMS

Structural changes to the RMS code were performed such that the RMS-11 V1.8 ODL files RMS11X, RMS12X, and RMS11S would remain valid. Most other ODL structures will probably continue to be valid, although this cannot be guaranteed for all cases. Use of V1.8 ODL structures with V2.0 code may require more task virtual address space than with V1.8 code.

New versions of RMS11X, RMS12X, and RMS11S are provided which use approximately the same (or slightly less) task virtual address space as their V1.8 counterparts used with the V1.8 code. If your system includes the necessary DECNET facilities, the new file DAP11X.ODL provides support for remote file and record access as well as local access.

An extensively-commented "prototype" ODL file RMS11.ODL is also included. This file is not meant to be used directly, but may be helpful in designing your own ODL structures or in modifying the structure of the above ODL files to suit your special needs.

A restriction on ODL structures involves the RMS entry point modules R0xxxx (where xxxx represents the leading characters of the RMS operation name, for example, R0PUT). These modules, which were not explicitly referred to in previous RMS ODL files, MUST not be referred to in RMS ODLs for V2.0; if they are, a multiple definition for symbols of the form $RMxxx will occur.

5.3.5 Non-Overlaid RMS

While the total end-to-end size of RMS has decreased despite addition of new features, the space savings and feature additions have not been spread evenly throughout. Most of the added features have occurred in the areas of file/directory operations and record operations for sequential files.

The result is that applications using non-overlaid RMS for sequential-file-only access may experience an increase in virtual address space requirements for the RMS code (especially in RSTS/E environments). Where this presents a problem, it should normally be possible to select a suitable RMS disk overlay structure without noticeable performance impact.

5.3.6 Other V1.8 Compatibility Notes

- The utility RMSDFN is no longer supported.

- The RMS-11 run-time system (RMS11.RTS) is no longer supported.

- During system generation of your RSTS system, you must answer "YES" to the question about including support for RSX directives.

- Programs which have been compiled under RMS V1.8 should still run under V2.0, without recompilation (they must be relinked). Of course, such programs will not be able to take advantage of any new features in Version 2.0 of RMS.

- In general, RMS V2.0 cannot be run on a previous version of RSTS/E.

  RMS V2.0 depends on changes to the monitor and on the new structure of TKB autoload vectors in V8.0 of RSTS/E; do not try to use old versions of TKB to link RMS V2.0 programs.

- With V2.0, some changes to the SYSLIB modules have been made; several modules have been placed in SYSLIB to maintain compatibility with previous releases.

  As a result, you must be careful to reference RMSLIB as a "search" library before referring to SYSLIB when building tasks which use non-overlaid RMS. (This is only necessary if you need to make an explicit reference to SYSLIB as a "search" library when taskbuilding.)

- MACRO-11 programs using RMS should always check for errors by checking the value of the appropriate STS field.

  With V1.8, the C-bit may have been returned on RMS errors. However, this is not a defined error return and has not been maintained in RMS V2.0.

- ^Z and ESC are no longer recognized as record terminators for stream files, and ^Z is no longer recognized as a file terminator for stream files.

  Note that RMSCNV has a new switch, /EO, which can be used to convert stream files which were created by a non-RMS program, and which used ^Z as the file terminator.

- The expanded string is now an output from $OPEN and $ERASE by file-id.

- RMS now makes a check that at least 2 keys fit into a bucket. Therefore, with a bucket size of 1, the maximum key size is 243.

- When block I/O is used, $OPEN and $DISPLAY operations no longer return relative or indexed file key XAB, allocation XAB, or summary XAB information.

5.4 MISCELLANEOUS INFORMATION

- Note that all the new features provided in RMS-11 are fully accessible only to the MACRO-11 programmer; for high-level languages, check the language documentation for supported features.

- RMS-11 will process switches included in file specifications when the switch does not conflict with RMS-11's use of the RSTS/E file system or with information explicitly provided via the normal RMS-11 interface. These switches include the /FILESIZE, /CLUSTERSIZE, /POSITION, /MODE, and /PROTECT switches.

  These switches are documented in the RMS-11 User's Guide, Appendix A.

- For magtape, RMS now allows ANSI fixed format tape records to be less than 18 bytes.

- Several RMS demonstration programs have been included on the distribution. These sample programs are provided as MACRO-11 sources and task images; they illustrate how to use the new $PARSE/$SEARCH/$RENAME/$ERASE and wildcarding facilities. Also, a GSA source routine is included as illustration of how to extend the task in the event of RMS pool exhaustion.

  The names of the programs are: PARSE, SEARCH, RENAME, ERASE, GSA.

5.5 PROBLEMS FIXED WITH THIS VERSION

- The problem which was reported on V1.8 of RMS on GET after FIND on alternate keys (Dispatch article number 48.2.20), has been fixed in this version.

- The V1.8 version of RMSIFL did not load indexed files correctly if the key type was integer. (Not all records could be accessed by key value.) This problem has been fixed in the new version of RMSIFL.

- The V1.8 (and prior) difficulties associated with unwanted write-accessors being allowed access when only read-sharing was specified have been corrected in this version.

- On V1.8, RMS-11 would create temporary files using an octal (not decimal) representation of the job number in the filename. This caused problems when the job number was greater than 7 or when the decimal job number happened to correspond to the octal representation of another job number.

  This problem has been fixed in V2.0 of RMS; RMS now creates temporary files using the correct RSTS temporary file mechanism.

- RMS now pads stream files with nulls, to the high block of the file (not just to the end of the current block). This will make RMS-created stream files readable by programs which do not recognize the EOF value from the file header.

- RMSBCK now saves the clustersize of the original file; RMSRST will first attempt to use that original clustersize when restoring the file. If there are enough clusters of the correct size, then the file will be restored to that clustersize. Otherwise, RMSRST will restore the file using the default pack clustersize.

5.6 KNOWN PROBLEMS WITH THIS VERSION

- Under some circumstances, the deferred write option may be ignored.

- Use of the mass insert feature may occasionally cause the error ER$DME.

- RMSDES and RMSIFL will incorrectly exit with an error status following a command line consisting only of a ^Z.

- When using RMSDES and specifying a GET operation, it incorrectly requests write access on the GET file. In some cases, this may result in an ER$FLK or ER$PRV error if your task does not have write access to the file.

- RMSIFL will not reassign the sort work file devices if the /DE switch specifies a logical name instead of a device name.

5.7 RMS FILES AND PLACEMENT ON THE KIT

The RSTS/E V8.0 kit contains the following RMS elements in [1,2]; note that the "destination" column describes the destination of files for the default configuration (see Section 5.8).

File Name	Destination	Comments
RMS11.CTL	SY:/deleted	Input to the Build program to do the RMS installation.
RMSMAC.MLB	LB:	Can be deleted if you are not using MACRO RMS programs.
RMSLIB.OLB	LB:	Object library for RMS local access.
RMSDAP.OLB	LB:	Object library for RMS remote access. Can be deleted if you do not need remote access.
RMSACF.TSK	[1,2]	Used during installation to install RMS resident libraries.
RMSBCK.TSK	[1,2]	BCK utility; uses RMSRES.
RMSRST.TSK	[1,2]	RST utility; uses RMSRES.
RMSDEF.TSK	[1,2]	DEF utility; uses RMSRES.
RMSDSP.TSK	[1,2]	DSP utility; uses RMSRES.
RMSCNV.TSK	[1,2]	CNV utility; uses RMSRES.
RMSDES.TSK	[1,2]	DES utility; uses RMSRES.
RMSIFL.TSK	[1,2]	IFL utility; uses RMSRES.
DES.HLP	SY:/deleted	Sequential file used to build RMSDES indexed help file
DESHLP.DES	SY:/deleted	DES file used to build RMSDES indexed help file
RMSFUN.OBJ	SY:/deleted	Concatenated object module containing objects to be put in LB:SYSLIB
RMS11.ODL	LB:	Prototype ODL file
R0RMS1.MAC	LB:	For use with the prototype ODL
RMS11S.ODL	LB:	ODL file for sequential
RMS12S.ODL	LB:	ODL file for sequential
RMS11X.ODL	LB:	Standard indexed file ODL
RMS12X.ODL	LB:	Indexed file ODL
RMSRLX.ODL	LB:	ODL for use with RMSRES
DAP11X.ODL	LB:	ODL for use with overlaid DAP
DAPRLX.ODL	LB:	ODL for use with DAPRES
RMSRES.LIB	[0,1]	Library image for RMSRES "root"
RMSRES.TSK	LB:	TSK image for linking against RMSRES

(Continued on next page)

File Name	Destination	Comments
RMSLBA.LIB	[0,1]	Library image for RMSRES segment
RMSLBB.LIB	[0,1]	Library image for RMSRES segment
RMSLBC.LIB	[0,1]	Library image for RMSRES segment
RMSLBD.LIB	[0,1]	Library image for RMSRES segment
RMSLBE.LIB	[0,1]	Library image for RMSRES segment
RMSLBF.LIB	[0,1]	Library image for RMSRES segment
RMSRES.STB	LB:	STB file for RMSRES
DAPRES.LIB	[0,1]	Library image for RMSDAP library
DAPRES.TSK	LB:	TSK file for linking against DAPRES
DAPRES.STB	LB:	STB file for DAPRES
RMSRES.MAP	Not transferred	
RMSLBA.MAP	Not transferred	
RMSLBB.MAP	Not transferred	
RMSLBC.MAP	Not transferred	
RMSLBD.MAP	Not transferred	
RMSLBE.MAP	Not transferred	
RMSLBF.MAP	Not transferred	
DAPRES.MAP	Not transferred	
DAPRES.CMD, DAPRES.ODL	Not transferred	For rebuilding DAPRES
RMSRES.CMD, RMSRES.ODL, RMSRES.ONL, RMSFAK.CMD	Not transferred	For rebuilding RMSRES
RMSLBA.CMD, RMSLBA.ODL	Not transferred	For rebuilding RMSLBA
RMSLBB.CMD, RMSLBB.ODL	Not transferred	For rebuilding RMSLBB
RMSLBC.CMD, RMSLBC.ODL	Not transferred	For rebuilding RMSLBC
RMSLBD.CMD, RMSLBD.ODL	Not transferred	For rebuilding RMSLBD
RMSLBE.CMD, RMSLBE.ODL	Not transferred	For rebuilding RMSLBE
RMSLBF.CMD, RMSLBF.ODL	Not transferred	For rebuilding RMSLBF
RMSROT.CMD, RMSROT.ODL, RMSROT.STB	Not transferred	For rebuilding segmented library
BCKNON.CMD, BCKNON.ODL	Not transferred	For rebuilding the overlaid version of BCK
BCKNRN.CMD, BCKNRN.ODL	Not transferred	For rebuilding the resident library version of BCK
CNVNON.CMD, CNVNON.ODL	Not transferred	For rebuilding the overlaid version of CNV
CNVNRN.CMD, CNVNRN.ODL	Not transferred	For rebuilding the resident library version of CNV
CNVNOO.CMD, CNVNOO.ODL	Not transferred	For rebuilding the overlaid version of CNV, with RMSDAP
CNVNRR.CMD, CNVNRR.ODL	Not transferred	For rebuilding the resident library version of CNV,with RMSDAP
DEFNON.CMD, DEFNON.ODL	Not transferred	For rebuilding the overlaid version of DEF
DEFNRN.CMD, DEFNRN.ODL	Not transferred	For rebuilding the resident library version of DEF

(Continued on next page)

File Name	Destination	Comments
DESNON.CMD, DESNON.ODL, RMSDES.ODL	Not transferred	For rebuilding the overlaid version of DES
DESNRN.CMD, DESNRN.ODL	Not transferred	For rebuilding the resident library version of DES
DSPNON.CMD, DSPNON.ODL	Not transferred	For rebuilding the overlaid version of DSP
DSPNRN.CMD, DSPNRN.ODL	Not transferred	For rebuilding the resident library version of DSP
IFLNON.CMD, IFLNON.ODL	Not transferred	For rebuilding the overlaid version of IFL
IFLNRN.CMD, IFLNRN.ODL	Not transferred	For rebuilding the resident library version of IFL
RSTNON.CMD, RSTNON.ODL	Not transferred	For rebuilding the overlaid version of RST
RSTNRN.CMD, RSTNRN.ODL	Not transferred	For rebuilding the resident library version of RST
RMSUTL.OLB, RMSODL.ODL	Not transferred	For rebuilding several utilities
GSA.MAC	Not transferred	Demonstration program included as an illustration of how to extend an RMS task in the event of pool exhaustion.
PARSE.MAC, SEARCH.MAC, RENAME.MAC, ERASE.MAC, PARSE.TSK, SEARCH.TSK, RENAME.TSK, ERASE.TSK	Not transferred	Demonstration programs for the new directory and wildcarding facilities
RMSV2.DOC	Not transferred	RMS release notes (this chapter).

Note that all RMSDAP files can be deleted if you are not using RMS to access files on remote nodes.

5.8 RMS V2.0 INSTALLATION

The control file RMS11.CTL is used as the input control file to the $BUILD program to do the installation of RMS V2. In general, this procedure is very similar to the V1.8 installation.

NOTE

> In order to install RMS V2.0, you must have specified YES to the "RSX directives?" question when doing the RSTS V8.0 sysgen.

First, you must physically mount the kit media, and log in under a privileged account. Then, if the kit media is a disk, logically mount the kit as:

```
     MOUNT dev:  name/RO
```

where **dev:** specifies the physical name and number of the drive with the installation disk, and **name** is the packid of the disk that contains the RMS files.

For all kit media, invoke the BUILD procedure, specifying the control file as [1,2]RMS11.CTL. The control file executes the following steps:

- Asks if you want the default RMS configuration which includes the segmented resident library(s) and DAPRES, and the RMS utilities built against the segmented resident library.

  This is the recommended configuration, but you must have at least 22KW of memory available to install the RMS segmented library. And, you must always ensure that all segments of the RMS segmented library are installed before trying to use any of the RMS utilities.

  The optional RMS configuration is one that does not include any resident libraries, and the RMS utilities are rebuilt to use disk-overlaid RMS.

- Removes libraries and deletes V1.8-specific files that will not be replaced by the installation.

  This is done to maintain consistency with resident libraries: if you need to save V1.8 files, you must make provisions for this before doing a V2.0 installation.

- Transfers the files from the kit media to the destination accounts on the designated disk. The destination accounts are described in Section 5.7.

- Replaces modules in LB:SYSLIB with modules included in the concatenated object module RMSFUN.

  Note that the module SIORTN is a new module added to SYSLIB; it is used by RMSIFL and SORT-11. And, the modules R1GET and R1GSET are deleted from SYSLIB, as they are no longer used.

- Replaces any existing CCL definitions for RMS utilities with CCLs for the V2.0 RMS utilities. Note that a new CCL for the RMSDES utility is added; a CCL for RMSDFN is no longer needed, since RMSDFN is not supported with V2.0.

- If you have selected the default configuration using the resident libraries, the control file runs the RMS autoconfigure task, RMSACF. This task will remove any existing RMS resident libraries from memory and will install the RMS V2.0 resident libraries in memory with the NOERROR option.

  Note that the RMS resident library RMSRES now consists of 7 library images instead of one library image. (See Section 5.1.2 for a discussion on the segmented resident library.) RMSACF installs all 7 segments for RMSRES and, if it detects the presence of DECNET on your system, it installs DAPRES.

  RMSACF can fail if you do not have enough memory for the resident library. If that happens, the subsequent processing by RMSDES and RMSCNV will also fail; you should in this case restart the installation and specify the optional configuration which does not include the RMS resident libraries.

- If you have selected the optional configuration without resident libraries, the control file retaskbuilds all the RMS utilities to use disk-overlaid RMS.

- Runs the RMSDES and RMSCNV utilities to create an indexed file in LB:RMSDES.HLP. This file will be the HELP file used by the RMSDES utility.

- Terminates after displaying the line:

  ```
  ! END OF INSTALLATION PROCEDURE
  ```

At this point, you can logically dismount the installation media with a DISMOUNT command, and physically dismount the volume.

5.8.1 Startup Command Procedures

To examine the results of the installation autoconfigure program, run SYSTAT.

Note that the autoconfigure program, RMSACF, inserts the resident library segments in the order: RMSRES, RMSLBB, RMSLBA, RMSLBD, RMSLBC, RMSLBE, RMSLBF. This is an attempt to order the segments in order of "most-use" by applications, to avoid leaving "holes" in memory when unused segments are displaced from memory. It is recommended that you use a similar algorithm, based on your application's use of RMS, when positioning the library segments in memory.

That is, it is the system manager's responsibility to modify the system startup procedures to install the RMS resident libraries. The startup procedures can use the RMSACF program to install the libraries every time, or UTI ADD LIBRARY commands can be used, if more flexibility is needed.

For example, you may wish to use the /STAY switch for segments which will be used frequently (RMSRES is a very good candidate for this). Using the /STAY switch will mean the segment stays in memory even when it is not mapped by a task; for heavily-used segments, this could cut down on possible thrashing.

However, you should **not** use the /REMOVE switch on the UTI ADD LIBRARY command. This would force the library segment to go out of memory whenever a task was done using a particular library segment.

Note that although the library segments do not have to be physically resident, **all** library segments must be installed before any use of the resident library (including use of RMS utilities which are built against the resident library). Failure to install the root segment RMSRES will cause an error "?Resident library not found" when the task or utility is invoked; however, failure to install any of the remaining resident library segments will not give an error at invocation, but may cause a BPT trap when the missing segment is subsequently needed by RMS.

NOTE

> There are several circumstances involving "inconsistent or incomplete resident libraries" which can cause a BPT trap to be generated, with R0 containing the error code ER$LIB. This can happen if not all the 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.

You may also wish to include the new RMS CCL definitions in the system start-up control file. RMSDES is a new V2.0 utility which needs a new CCL definition.

5.8.2 Utility Configurations

The RMS utilities which are provided on the kit are built to use the segmented resident library. At installation time, you can choose to use these versions of the utilities or request that the utilities be rebuilt to use disk-overlaid RMS. See the discussion on RMS V2.0 installation for more details of this process.

However, for selective rebuilding and optional configurations (such as rebuilding RMSCNV to include remote access support), all the command and ODL files are provided.

5.8.2.1 Utility Command and ODL Files

- For each utility, a command and ODL file is provided which can be used to build the utility using disk-overlaid RMS. The names of the files are:

For RMSBCK:	BCKNON.CMD,BCKNON.ODL,RMSODL.ODL
For RMSRST:	RSTNON.CMD,RSTNON.ODL,RMSODL.ODL
For RMSCNV:	CNVNON.CMD,CNVNON.ODL
For RMSDSP:	DSPNON.CMD,DSPNON.ODL,RMSODL.ODL
For RMSDES:	DESNON.CMD,DESNON.ODL,RMSDES.ODL
For RMSDEF:	DEFNON.CMD,DEFNON.ODL,RMSODL.ODL
For RMSIFL:	IFLNON.CMD,IFLNON.ODL

For CNV, two additional configurations are available, if you wish to use CNV to access files on remote nodes. To build CNV using the clustered RMS and RMSDAP resident libraries, you can use:

For RMSCNV: CNVNRR.CMD,CNVNRR.ODL

To build CNV using disk-overlaid RMS and RMSDAP, you can use:

For RMSCNV: CNVNOO.CMD,CNVNOO.ODL

For each utility, a command and ODL file are provided which can be used to build the utility using the resident library RMSRES. The names of the files are:

For RMSBCK:	BCKNRN.CMD,BCKNRN.ODL
For RMSRST:	RSTNRN.CMD,RSTNRN.ODL
For RMSCNV:	CNVNRN.CMD,CNVNRN.ODL
For RMSDSP:	DSPNRN.CMD,DSPNRN.ODL
For RMSDES:	DESNRN.CMD,DESNRN.ODL
For RMSDEF:	DEFNRN.CMD,DEFNRN.ODL
For RMSIFL:	IFLNRN.CMD,IFLNRN.ODL

5.8.2.2 **Rebuilding the Utilities** - If you do decide to rebuild the utilities, you must take the following steps:

- Use PIP to transfer the appropriate CMD and ODL files to your account. The files are in [1,2] on the distribution media.

- Use PIP to transfer the file RMSUTL.OLB from [1,2] on the distribution media, to your account.

- Use TKB to build the utility(s). The utilities and corresponding map files will be built in your account; you may then wish to put the utility in a system account.

Each command file for a particular utility creates a utility of the correct name. So, for example, CNVNRN.CMD and CNVNON.CMD both produce task images called RMSCNV.TSK; one is built against the resident library and one is built with disk-overlaid RMS.

Finally, note that the IFL utility command files include an EXTTSK value which is used to increase the size of internal buffers, thereby allowing IFL to load larger-sized files and files with many areas and key definitions.

The default EXTTSK which is specified is fairly large. If you want a smaller IFL, you can edit the EXTTSK value and rebuild IFL. The EXTTSK value however, must be greater than 2048 (decimal) words. If the value is too small, you may get the error "IFL memory exhausted," or a SORTs error code.

HOW TO ORDER ADDITIONAL DOCUMENTATION

DIRECT TELEPHONE ORDERS

In Continental USA and Puerto Rico call **800–258–1710**	In Canada call **800–267–6146**	In New Hampshire, Alaska or Hawaii call **603–884–6660**

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

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

DIRECT MAIL ORDERS (Canada)

DIGITAL EQUIPMENT OF CANADA LTD.
940 Belfast Road
Ottawa, Ontario, Canada K1G 4C2
Attn: A&SG Business Manager

INTERNATIONAL

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

Internal orders should be placed through the Software Distribution Center (SDC), Digital Equipment Corporation, Northboro, Massachusetts 01532

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

RSTS/E V8.0
Release Notes
AA–5246F–TC

Reader's Comments

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

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

____

____

____

____

____

____

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

____

____

____

____

____

____

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

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

Name ______________________ Date ______________________

Organization ______________________________

Street ______________________________

City ______________________ State ____________ Zip Code or Country ____________

Do Not Tear - Fold Here and Tape

digital

No Postage
Necessary
if Mailed in the
United States

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

POSTAGE WILL BE PAID BY ADDRESSEE

ATTN: Commercial Engineering Publications MK01–2/E06
RSTS/E Documentation
DIGITAL EQUIPMENT CORPORATION
CONTINENTAL BOULEVARD
MERRIMACK, N.H. 03054

Do Not Tear - Fold Here and Tape

Cut Along Dotted Line