September 1979

This document describes the differences between RSTS/E releases and establishes a maintenance notebook for documentation corrections, published software problems and solutions, and programming notes.

RSTS/E V7.0
Release Notes

Order No. AA-5246C-TC

SUPERSESSION/UPDATE INFORMATION: This contains information concerning RSTS/E V7.0 and replaces the previous document titled RSTS/E V06C Release Notes, Order No. AA-5246B-TC.

OPERATING SYSTEM AND VERSION: RSTS/E V7.0

SOFTWARE VERSION: RSTS/E V7.0

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

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.

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

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

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

The following are trademarks of Digital Equipment Corporation:

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

TABLE OF CONTENTS

1.Ø Introduction

The RSTS/E V7.Ø Release Notes establish a maintenance notebook for documentation corrections, published software problems and solutions, and programming notes. Subordinate software information is also included. The RSTS/E Software Dispatch is designed to be merged into the maintenance notebook. It is provided for one year after each Category A software installation, and is addressed to the software contact established by the DIGITAL field office. After one year, the customer may purchase the Binary Program Update Service to continue receiving the RSTS/E Software Dispatch.

Anyone who is about to generate RSTS/E V7.Ø must incorporate all "Mandatory" and appropriate optional "Feature" patches as part of the system generation and BUILD procedures. All articles published are summarized in the Cumulative Index. There is also an index of the version and edit levels for all the Commonly Used System Programs (CUSPs). These indices are updated each month in the RSTS/E Software Dispatch.

For further information on RSTS/E documentation, refer to the RSTS/E V7.Ø Documentation Directory (AA-2642D-TC).

NOTE

References in this manual to RSTS/E VØ6C will be assumed to refer to both VØ6C-Ø3 and VØ6C-Ø4, the latter being a special release to provide RLØ1 and RKØ7 support. These two releases are functionally identical with the exception of the added device support. There is no such distinction, however, in RSTS/E V7.Ø.

Please note also that all numeric references in these Release Notes will refer to decimal numbers unless specified as octal numbers in the format 177777(8). This convention does NOT apply, however, to the patch articles.

2.Ø Differences Between RSTS/E V7.Ø and VØ6C

RSTS/E V7.Ø, along with becoming even more reliable and maintainable, provides new features that improve overall system performance and extend the capabilities of the disk file system and the monitor. Additionally, installation of RSTS/E V7.Ø has been streamlined by reducing the number of steps required to install a system with all known modifications. Upward compatibility is maintained for all non-privileged user software and all privileged user software which does not make assumptions about the internal structure of the monitor or the on-disk file structure. Upward compatibility is maintained for optional software products as well. RSTS/E V7.Ø also supports new versions of optional software products and the latest hardware devices available from DIGITAL. All CUSPs (Commonly Used System Programs) can now be used with either BASIC-PLUS or BASIC-PLUS-2 (V1.6 or later). Most of the existing documentation has undergone revision and new documentation has been created for RSTS/E V7.Ø.

2.1 New Features of RSTS/E V7.Ø

2.1.1 RMS is bundled

Beginning with RSTS/E V7.Ø, all RMS software, including multi-keyed indexed access, is bundled into the RSTS/E system. This provides users of RSTS/E with a powerful, general-purpose data management subsystem which supports sequential, relative, and indexed file organizations.

2.1.2 Files Larger than 65535 Blocks

An upward-compatible extension of the RSTS/E file structure is available to include support for disk files larger than 65535 blocks. The upper limit for a disk file on RSTS/E V7.Ø is increased to the capacity of the largest disk pack supported by RSTS/E V7.Ø. This capability is selectable at SYSGEN time and will require more memory (about 6ØØ words) in the RSTS/E Monitor, and will also require more small buffers, one additional for each open file (i.e., if a file is open simultaneously 5 times, 6 small buffers are used).

Large file support also provides added performance benefits even if files larger than 65535 blocks are not created. For further information on these benefits, refer to sections 2.2.5 and 2.4.3 of this chapter.

2.1.3 Disk File Placement

A new option to the "create a file" Monitor call is available that allows the placement of the beginning of a file at the first free cluster at or after a specified disk location.

2.1.4 Disk Data Cache

An extension of the disk cache mechanism is available that allows data from disk files to be put in the cache. This augments the capability in VØ6C of retaining disk directory information in the cache. Furthermore, the disk cache mechanism can store multiple blocks at a time, thus providing a "look-ahead" capability when a file is accessed in a sequential manner. The disk data cache capability is selectable during SYSGEN and requires approximately 1K additional words in the RSTS/E Monitor.

2.1.5 New Device Support

2.1.5.1 VT1ØØ

The VT1ØØ is an entry-level terminal replacement for the VT52 with a variety of proprietary features designed into its custom-built LSI chips. These features include two screen widths, three character sizes, four character attributes, smooth scroll capability, split screen capability, reverse screen, settable tabs, 7x9 dot matrix character font, and a special graphics character set. While the terminal is fully supported, none of the RSTS/E supplied software utilizes any of the terminal specific features.

2.1.5.2 LA12Ø

The LA12Ø is a hard-copy terminal for high-speed communications applications, featuring intelligent bi-directional printing, a 1ØØØ character buffer, 18Ø character per second print speed, and user selectable baud rate up to 96ØØ baud. The LA12Ø has more than 45 user selectable features relating to communications, printing, and formatting.

2.1.5.3 LA34 and LA38

The LA34 and LA38 are 3ØØ-baud desk-top terminals which are smaller, lighter, and quieter than most typewriters. All features are set from the keyboard, including four character width adjustments. The LA34 has roll paper feed while the LA38 comes with tractor feed for multi-part forms. Numeric keypads are optionally available on both.

2.1.5.4 RXØ2

The RXØ2 is a random access, mass memory device storing data in fixed length blocks on a flexible diskette with industry standard headers. The device will support existing single density, RXØ1/IBM compatible diskettes, and a non-IBM compatible double density mode which allows diskettes to be recorded with twice the linear density.

2.1.5.5 RLØ2

The RLØ2 is a 1Ø 1/2 inch high disk drive which can be universally rack mounted on the slides provided in any DIGITAL cabinet. The drive operates at 24ØØ RPM on either 5Ø Hz or 6Ø Hz and transfers data at a rate of 512.5 K bytes per second. The RLØ2 uses a top loading disk cartridge which has one platter with two recording surfaces. Each surface contains 512 tracks, with 4Ø sectors per track, each containing 128 (16 bit) words. The RLØ2 disk cartridge is factory formatted and cannot be formatted by the user.

2.1.5.6 TS11

The TS11 is a 45 ips, 16ØØ BPI only, microprocessor controlled tape drive utilizing industry standard phase-encoding recording techniques and tension arms instead of vacuum columns. The microprocessor also offers multi-level diagnostic support through off-line tests run from a maintenance panel, on-line tests run from the front panel, and in-line tests run automatically whenever the drive is in stand-by state. The TS11 is UNIBUS compatible only.

NOTE

> The TS11 is referred to as the TSØ4 in all RSTS/E documentation except these Release Notes.

2.1.5.7 TU77

The TU77 is a 125 ips, dual-density, fully integrated tape storage sub-system which is packaged with its associated interface and power supply in a single cabinet. The TU77 offers a peak transfer rate of 2ØØ,ØØØ bps, making it ideal for high performance applications such as disk backup.

2.1.5.8 TU58

The TU58 DECtape II is a random-access, fixed-length-block, mass-storage tape cartridge system. It uses preformatted tape cartridges which store 262 kilobytes of data in 512-byte blocks. There are 256 blocks on each of two tracks. Data stored on these cartridges may be accessed similar to data stored on non-file structured disks.

2.1.6 Resident Libraries

Jobs may access one or more resident libraries, up to a maximum of 5 per job at any one time. Resident libraries are segments of shared code or data handled by the system in a manner similar to run-time systems. However, unlike run-time systems, resident libraries can have "memory-resident overlays," thus reducing the usage of job address space without incurring disk accesses due to overlay activity. Resident libraries may be used in addition to the run-time system of a job. A resident library can also overlap the user address space used by the RSX Run-Time System, provided that at system generation time, the RSTS/E monitor was specified to have the RSX directive emulation capability. (See section 2.2.4 of this chapter for further information.) The resident library capability is selectable at system generation time and requires approximately 1.5K additional words in the RSTS/E Monitor.

2.1.7 High Speed Disk Volume SAVE/RESTORE Utility

SAVE/RESTORE is a new utility for saving, restoring, and image copying all of the files on a RSTS/E disk volume. (See section 2.8.1 of this chapter for more information.)

2.1.8 FORTRAN IV-PLUS Compatibility

RSTS/E FORTRAN IV-PLUS V2.5 will operate on RSTS/E V7.Ø. It is distributed as Category C software.

2.1.9 New Manuals

Most RSTS/E manuals have been completely re-written and several new manuals have been added. Two new manuals of special interest are:

2.1.9.1 RSTS/E System Directives Manual

Following a brief general description of run-time systems, this manual describes directives to the RSTS/E Monitor and RSX Emulator that can be used in MACRO programs. (See also the RSTS/E V7.Ø Documentation Directory.)

2.1.9.2 RSTS/E Programmer's Utilities Manual

This manual describes the RSX-based utility programs available to RSTS/E users: MAC, LBR, and PAT. The manual also describes the Save Image Library (SIL) formatting program, MAKSIL. (See also the RSTS/E V7.Ø Documentation Directory.)

2.1.1Ø SORT-11

SORT-11 V2.Ø is now included as part of RSTS/E V7.Ø. Before attempting to install SORT-11, however, please refer to article Seq 19.2.1 N in these Release Notes.

2.1.11 BASIC-PLUS-2 as System-Default Run-Time System

All BASIC-PLUS CUSPs are also runnable in BASIC-PLUS-2. This allows the use of BASIC-PLUS-2 exclusively, if so desired, and eliminates the memory requirements for the BASIC-PLUS Run-Time System, if it is not otherwise needed. The system library programs are coded in such a way that they can be compiled by the BASIC-PLUS-2 compiler, without having to be translated by the the BASIC-PLUS-2 translator.

2.1.12 RSX as System-Default Run-Time System

All BASIC-PLUS CUSPs can be compiled with CSPCOM (see section 2.8.2 of this chapter) and linked against the RSX Run-Time System. This eliminates the memory requirements for the BASIC-PLUS Run-Time System, if it is not otherwise needed. This has been implemented for installations using primarily COBOL. (NOTE: The DEVTST program CPEXER cannot be built with CSPCOM. It is the only exception.)

2.1.13 Build Procedures

The BUILD procedure for RSTS/E V7.Ø has been modified to provide automated patching of system library files at the time they are installed on your system. This procedure is described in the RSTS/E V7.Ø System Generation Manual.

2.1.14 CUSP Packages

Some CUSP packages (e.g., BACKUP Package, Error Logging Package, Spooling and Operator Services Package), can be stored in and executed from accounts other than [1,2] to reduce the overhead to this account.

2.1.15 ONLCLN

This new, fast, on-line program duplicates the functions of the CLEAN sub-option of INIT.SYS for cleaning disk packs, and should be used instead of the "CLEAN" command of UTILTY. (See article Seq 1.2.1 N published in these Release Notes.)

2.2 System Generation

The system generation procedure for V7.Ø is similar to that of VØ6C. Several new questions have been added to the SYSGEN dialogue, however, which reflect new or expanded features of V7.Ø. These questions and their ramifications are described in the following sections.

2.2.1 Extended Data Buffering

This is the V7.Ø data caching feature. This feature makes use of the system extended buffer space (XBUF), a region of memory optionally specified at system start up time, and dynamically allocated by the RSTS/E Monitor for several uses including large message SEND/RECEIVE, caching directories, and caching data if specified. Extended Data Buffering is not recommended for systems with less than 256K words of memory because such systems usually cannot afford to allocate a large enough XBUF to make it effective. In general, systems with 64K to 192K words should use directory caching (FIP buffering) but not data caching. Between 193K and 256K words, experimentation with the individual system should be performed to determine if data caching will be effective.

The best size for XBUF can only be determined by experimentation. For directory caching, a good starting point is about 1Ø% of memory; for data caching, a good starting point is 15% of memory.

General guidelines for use of data caching are presented in the RSTS/E V7.Ø System Manager's Guide, the RSTS/E V7.Ø System Generation Manual, and section 2.3.1.1 of this chapter.

2.2.2 Extended Buffering For Line Printer Driver

If this question is answered "Yes", the line printer driver will store data in XBUF as well as in small buffers. Each line printer will use a maximum of 2.5K words of XBUF while it is running. The driver is also about 7Ø words larger.

This option is intended for customers with large systems who want to improve line printer performance and reduce line printer impact on the small buffer pool. It is not recommended for systems with less than 1ØK words of XBUF because use of XBUF by the line printer driver will significantly reduce the hit rate for directory caching on small systems.

2.2.3 Disk Driver Phase

For some large systems (i.e., systems with many devices and new features installed), the number of small buffers that may be configured is restricted by the space occupied by device drivers. These systems may increase the number of small buffers by answering "Yes" to this question. The actual number of small buffers that are made available by selecting this option depends on the types of disks and other devices for which support is generated.

About 15% of the time, putting disk drivers in a separate phase will make the monitor 1K words longer, independent of any size increase from added small buffers. There is no difference in performance, however, when the disk drivers are put in a different phase.

2.2.4 RSX Directives

If, during system generation, the RSX Directives question is answered "Yes", code is included in the monitor which duplicates some of the functions of the RSX Run-Time System. The RSX Run-Time System is then used only as a keyboard monitor and for loading programs. Once the program starts running, the run-time system, which normally occupies the top 4K words of the user job address space, is removed from the user job address space. This allows the space to be used for expanding the user program (up to 31K words) or for mapping a resident library. This option should be selected if the RMS Resident Library, RMSRES, is to be used. Future versions of RMS and optional software products dependent on RMS will probably require the use of this RSTS/E monitor feature. Therefore, this option should be selected unless it is absolutely necessary to minimize the size of the monitor. The RSX directives capability requires approximately 1K words in the RSTS/E monitor.

Note that user programs which expect to be able to read data from the RSX Run-Time System will fail with "Memory Management Violation" on systems with the RSX Directives option selected, or will get erroneous data if the address space is used for something else.

2.2.5 Large Files

If, during system generation, the Large Files question is answered "Yes," a version of the File Processor (FIP) is included in the system which will support disk files larger than 65535 blocks. Even if you do not have a need for files of this size, there are certain performance benefits which are derived from the use of this processor. These benefits include reduced directory I/O required to do window turns, reduced directory I/O to open a file already open, decreased directory search time, and decreased load on disk I/O drivers. These benefits are discussed more fully in the RSTS/E V7.Ø System Generation Manual.

2.3 Initialization Code

2.3.1 DEFAULT Option

2.3.1.1 Cache Cluster Size

This question appears if "Extended Data Buffering" was selected during system generation and a minimum of 2K words was alloted to XBUF. Data in the cache is organized in clusters of 1, 2, 4, or 8 blocks. If the cache cluster size is greater than the file cluster size of a non-contiguous file or a directory that is being cached, some of the blocks in the cluster may contain unneeded data. On the other hand, if the cache cluster size is too small, the performance benefits of caching will not be available because a read may have to be split into two separate reads to install the data in two different cache clusters. RSTS/E will not split a read if doing so would cause more than two physical disk operations. Furthermore, if a file is being read sequentially, one block at a time, each time a cluster is installed, the remainder of the blocks in the cluster can be read without a disk access.

One way to resolve these conflicts is to consider the number of cache clusters that will fit in XBUF. If XBUF is, for example, 2ØK words long, and the cache cluster size is 2 blocks, almost 4Ø clusters will fit - probably reasonable for a system that typically runs about 2Ø jobs. If the cache cluster size were increased to 8 blocks, fewer than 1Ø cache clusters would fit in cache, and performance might suffer.

Note that the cache cluster size can also be changed on-line for the current time-sharing session with the UTILTY program.

2.3.2 DSKINT Option

The DSKINT option of INIT.SYS will now, if requested, pre-extend the directories of accounts [Ø,1] - the System Files account, [1,1] - the Master File Directory, and [1,2] - the System Library account. This pre-allocation of directory clusters eliminates fragmentation of directories and reduces directory search time.

The DSKINT option will, as a default, place the SATT.SYS file and the Library account [1,2] near the center of the disk to reduce head movement when accessing these commonly used accounts. Previous releases required the person performing the system generation to perform a manual calculation to achieve this optimization.

The DSKINT option will also identify a disk, if it is RSTS/E file-structured, by printing the pack-id and certain of the pack attributes.

2.3.3 SAVRES Option

This new option performs the same functions as the SAVE/RESTORE utility described in Section 2.8.1 of this chapter, i.e., backing up (SAVE) to disk or tape, reconstructing (RESTORE), copying (IMAGE) data disk to disk, and extracting (IDENTIFY) label information.

2.4 Monitor

Several changes and additions have been made to the RSTS/E Monitor for V7.Ø.

2.4.1 256-Message Error File

The system error file is now 16 blocks long, allowing 256 error messages. BASIC-PLUS uses only the first 128 messages. The remainder are used by BASIC-PLUS-2, when present on the system.

2.4.2 Changed Error Messages

ERR=33	?UNIBUS timeout fatal trap	becomes:	?Odd address trap
ERR=36	?SP(R6) stack overflow	becomes:	?SP stack overflow

2.4.3 Large Files

There are restrictions placed on the use of large files (i.e., more than 65535 blocks in length). Large files can be used only as data files and are not executable. These restrictions are documented in the RSTS/E Programming Manual and in the BASIC-PLUS Language Manual.

2.4.4 Placed Files

The first cluster of a file can be placed at or after a specified device cluster. Documentation for this feature exists in the RSTS/E System User's Guide, the RSTS/E Programming Manual, and the BASIC-PLUS Language Manual.

2.4.5 Resident Libraries

You may select this feature during system generation so that jobs may share common code, such as RMS routines, and to minimize usage of user job address space. Information about resident libraries appears in several manuals. The RSTS/E Programmer's Utilities Manual and the update to the RSTS/E Task Builder Reference Manual contain information on creating and linking resident libraries. The RSTS/E V7.Ø System Manager's Guide contains information on adding and removing resident libraries with UTILTY, a procedure which is similar to adding and removing run-time systems. The RSTS/E System Directives Manual and the RSTS/E Programming Manual contain information about how RSTS/E supports resident libraries.

2.4.6 Make A Contiguous File Non-Contiguous

A contiguous file may be made non-contiguous by using the /NOCTG switch with the FLAG command of UTILTY. This feature allows you to extend files that were contiguous. The /NOCTG switch is documented in article Seq 19.3.1 N published in these Release Notes. The FLAG command and its format are described in the RSTS/E V7.Ø System Manager's Guide.

2.4.7 New Optional "Feature" Patches

Several optional "Feature" patches, if installed, enable directory lookup protection, cross-account file renames, and allow any privileged job to give privilege to any disk file. These patches are included in these Release Notes.

2.4.8 Monitor Statistics

The format of the monitor disk statistics has been changed to include more information about the effects of directory and data caching. Six monitor tables store data on swapping, user I/O, SATT.SYS, monitor overlays, directory I/O, and DECtape buffering respectively. Each table now contains three 4-word entries for each disk unit: physical disk reads, disk writes, and read requests satisfied from cache. The first two words of an entry count access and the second two words count total blocks.

STATUS.BAS has been updated to present this information.

Note that the Monitor Statistics Package is not a fully supported feature in that there is no documentation as to its use and not all field Software Support Specialists have received formal training in its use. This package should also be regarded as a relative, rather than an absolute, measurement tool.

2.5 File Processing

2.5.1 New Modes For Disk File OPENs

Detailed documentation for these features is included in the RSTS/E Programming Manual.

1. Create tentative file - a newly created file is not made permanent until it is closed.

2. Create conditionally contiguous - a file is created contiguous if sufficient contiguous space is available, otherwise it is created non-contiguous and no error is returned.

3. Create but do not supersede - an error will be returned if a file being created already exists.

4. Data caching - data transfers are eligible for caching in memory if data caching is generated in the monitor and is enabled, and the job is privileged.

5. Sequential data caching - if data transfers are cached, caching will be in sequential mode.

6. Non-file structured block mode - if large file support has been generated, blocks, rather than clusters, can be addressed.

2.5.2 New Features

Detailed documentation for these features appears in the RSTS/E Programming Manual and RSTS/E System User's Guide.

1. Overwrite file (with zeros) before it is deleted - note that FIP will be locked up during the overwrite. The privilege bit (bit 128) in the file's protection code enables this feature.

   NOTE

   Use this feature very carefully! FIP can be tied up for many minutes overwriting files of any appreciable size. (On an 11/34 system, it takes about 35 minutes to overwrite a 1ØØ,ØØØ block file.) The use of the PIP utility is recommended for deleting such files because PIP by-passes the FIP mechanism and avoids tying up system resources.

2. Create with privilege bit - permanently privileged jobs can now create a file with the privileged bit (bit 128) set in its protection code.

3. Extending UPDATE files - if your system was generated to include support of large files, a non-contiguous file open in UPDATE mode can be extended.

2.5.3 Modified SYS Calls

Detailed documentation for these calls appears in the RSTS/E Programming Manual.

1. Date/Time - You can specify no change to either value by specifying a value of zero (Ø).

2. Assign/Deassign - these calls can now be used to enter and remove user logical names, user logical PPNs, or user default protection codes. User logicals can include PPNs.

3. File string scan - string to be translated can specify that the device name must not be translated. Documentation for this also appears in the RSTS/E System User's Guide. In addition, a new switch, "/POSITION:n", can be used for placement of files, and the "/SIZE:n" switch can handle "large" (i.e., greater than 65535) arguments. File string scan will also now return PPNs stored in user logicals as well as system wide logicals.

4. Get monitor tables (part 2) - if large file support has been generated, this SYS call will also return a pointer to the file control block (FCB) linked lists. (The FCB on a system with large file support is totally unlike the FCB in previous versions of RSTS/E. The FCB on a system without large file support is called an SCB, and is compatible with the FCB in previous releases.)

5. Directory lookup - rather than returning the file size which is stored on disk, sometimes incorrect if the file is open and being extended by a job, the correct size is always returned if large file support is included in the system.

6. Seizing an assigned device - a privileged user can assign or reassign a device already assigned to another user as long as the device is not open.

The following changes were made after the completion of the documentation. They will be included in the next release of the RSTS/E Programming Manual.

7. Exit to Keyboard Monitor - this call has been extended to allow you to switch to a run-time system without making it your private default run-time system. (See article Seq 19.6.1 N published in these Release Notes.)

8. Create a Job - this call has been extended to allow you to specify that the job should be created even if logins are disabled. (See article Seq 19.6.1Ø N published in these Release Notes.)

2.5.4 New SYS Calls

Detailed documentation for the following SYS calls appears in the RSTS/E Programming Manual.

1. Wild card PPN lookup - used to determine which PPNs match a specific wild-card specification.

2. File control and Data caching - can be used to change a file's run-time system name, backup statistics, placed bit setting, and to reset the contiguous bit. It can be used to obtain an unopened file's run-time system name and attribute data. It can also set a file to always be eligible for caching (if caching is enabled) regardless of privilege and designate random or sequential caching.

3. Disk block locking - The SPEC% function can change an implicit disk block lock into an explicit lock and release the implicit lock.

4. Return job status SYS call - returns current statistics on the state of a job and its resource statistics.

The following changes were made after the completion of the documentation. They will be included in the next release of the RSTS/E Programming Manual.

1. CCL SYS call - allows user BASIC-PLUS programs to execute CCL calls as though they had been typed directly to a keyboard monitor. (See article Seq 19.6.2 N published in these Release Notes.)

2. Date and Time Conversion SYS call - allows BASIC-PLUS programs to convert internal format date and time to the specified alphanumeric or numeric format. (See article Seq 19.6.3 N published in these Release Notes.)

3. Snap Shot Dump SYS call - allows a privileged user to take an on-line "snap shot dump." (See article Seq 19.6.4 N published in these Release Notes.)

4. Spooling SYS call - allows a user program to spool one (or more if wildcard specification) file(s) to any spooler without CHAINing to QUE. While not privileged, protection codes are enforced. (See article Seq 19.6.5 N published in these Release Notes.)

2.6 Device Drivers

1. Parity checking on paper tape - new mode bits can be used to enable parity checking and generating for the paper tape reader and punch. Detailed documentation appears in the RSTS/E Programming Manual.

2. KMC11 Driver - This driver allows access to the KMC microcode RSTS/3271 Protocol Emulator V2.Ø. This driver is documented in the RSTS/3271 Protocol Emulator User's Guide.

3. IBM Interconnect Driver - This driver is the user interface with the KMC11 Driver. It is documented in the RSTS/3271 Protocol Emulator User's Guide.

4. DX Driver - The DX Driver has been modified to include support for the RXØ2. This Driver is discussed in the RSTS/E Programming Manual.

5. DD Driver - This driver supports the new TU58 DECtape II. The DD Driver is discussed in the RSTS/E Programming Manual.

6. MM Driver - This driver has been modified to include support for the TU77. The MM Driver is discussed in the RSTS/E Programming Manual.

7. MS Driver - This driver supports the new TS11 tape drive. The MS Driver is discussed in the RSTS/E Programming Manual.

8. DL Driver - This driver now supports overlapped seek and the RLØ2.

2.7 BASIC-PLUS Language and Run-Time System

Documentation for these functions appears in the BASIC-PLUS Language Manual.

1. CCPOS - identical to POS.

2. CHAIN...LINE... - the optional keyword LINE has been added.

3. GET/PUT with , BLOCK - large file access.

4. Virtual arrays - can be large files and can have an I/O buffer (RECORDSIZE) of 512, 1Ø24, 2Ø48, 4Ø96, 8192, or 16384 bytes. Note that the maximum valid subscript in an array is still 32767.

5. DEF* - identical to DEF.

6. CLOSE - "reset" type close can be used by specifying a channel number between -1 and -12 inclusive. Some devices do special handling of this type of close. Note that the final buffer, if any, is not written. This can also be used to prevent making a tentative file permanent.

7. COMPILE command - now uses the conditionally contiguous mode for creating a .BAC file in addition to any other mode specified. The protection code can include the privileged protection code bit.

8. ASSIGN command - can also be used by privileged users to seize a device by including "/MODE:1", and to create user logicals with or without PPNs.

9. RUN NH command - In EXTEND mode, "RUN NH" is no longer equivalent to "RUNNH". Instead, "RUN NH" will attempt to execute the program "NH". The effect of "RUN NH" is still equivalent to "RUNNH" in NOEXTEND mode.

Several features were added to BASIC-PLUS after the documentation was completed. These features are explained in various "Undocumented Feature" note articles in the Documentation section of these Release Notes.

2.8 RSTS/E Commonly Used System Programs

This section of the Release Notes describes the changes made to the RSTS/E CUSPs (Commonly Used System Programs) for RSTS/E V7.Ø.

All CUSPs, with the exception of the CPEXER program in the DEVTST package, can be compiled with any of the three variants of BASIC-PLUS: BASIC-PLUS, BASIC-PLUS-2, or CSPCOM which is described in section 2.8.2 of this chapter. 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.

2.8.1 Changes to the RSTS/E CUSPs for RSTS/E V7.Ø

ANALYS

The ANALYS program has been upgraded to support resident libraries, DECnet/E, and all the new features of SYSTAT except "/C". (See SYSTAT in this section.)

Installed patches to the monitor are displayed, and the crash dump may, optionally, include an annotated listing.

ATPK

The ATPK (AT Pseudo Keyboard) program is an indirect command file processor used in the BUILD process. Since ATPK controls a job through a pseudo-keyboard, it is able to monitor the job's terminal output to determine if any errors are encountered. A log file for the job may optionally be generated.

ATPK may also be used as a general purpose command file processor, invoked through the CCL commands "AT" or "@". The ability to detach and report to a log file allows the program to be used as a very simple batch processor.

A system startup procedure can take advantage of ATPK to provide a faster and "silent" startup for many systems. INIT.BAC may pass control to ATPK for most of the system startup procedure. ATPK can then detach and report the system startup dialogue in a log file. Although not as flexible as INIT.BAC, ATPK should satisfy most of the needs for this type of system startup.

A detailed description of ATPK can be found in article Seq 17.1.1 N published in these Release Notes.

AUTOPATCH

The autopatching mechanism for the CUSPs has been enhanced to allow:

1. Autopatching of the CUSPs when the library is built (via BUILD), rather than the old sequence of OLD-COMPILE-PATCH-OLD-COMPILE.
2. Checksum verification of source-level patches.
3. Comments in all phases, except within the text of a patch.
4. The use of comments with ONLPAT.

BACKUP

The BACKUP package has the following new features:

1. A LOAD INDEX command has been added, allowing you to create an index file on disk from the index of a BACKUP set.
2. A LIST command has been added to re-create the listing file from a BACKUP run.
3. BACKUP can be run under BATCH. Note that you must correctly anticipate all of BACKUP's requests for a BACKUP BATCH job to run successfully.

NOTE

The BACKUP package cannot backup files longer than 65535 blocks. If such a file is encountered, BACKUP will print a warning message to indicate that it is skipping the file, and proceed. The only supported way to backup "large" files is with the PIP.SAV utility program or as part of a disk volume SAVE operation.

BACKUP will transfer "placed" files that are shorter than 65535 blocks, but it is unable to maintain the placement of those files. A warning message will be printed when BACKUP encounters a "placed" file.

BATCH

BATCH now enforces the default elapsed time limit of 1Ø minutes if no other limits are specified in the $JOB statement. The SUBMIT CCL command, if available on your system, may be used to submit jobs for batch processing.

BPCREF

The optional patch published in the RSTS/E VØ6C Software Dispatch that allows you to cross-reference "local" and "global" variables has been implemented as a standard feature.

BUILD

The BUILD package has been completely re-written to provide the following enhancements:

1. Autopatching can be specified as an integral part of the BUILD procedure.
2. "Packages" of programs, such as the Backup and Spooling and Operator Services packages, can be created in, and run from, accounts other than [1,2].
3. CUSPs can be built by using BASIC-PLUS, BASIC-PLUS-2, or CSPCOM with the same set of control files.
4. Multiple control files can be specified during a single execution of BUILD.

To assist you when BUILDing the CUSPs, the locations of the System Build Control Files on the distribution media are listed in note article Seq 1Ø.2.1 N in these Release Notes.

COPY

The COPY program will identify the device which has caused an I/O error.

DEVTST - Device Exerciser Package

The UETP package has been replaced with a set of device exercisers known collectively as DEVTST. Documentation for this new package is included in the RSTS/E V7.Ø System Manager's Guide.

DIRECT

The DIRECT program has been enhanced to print additional information on request.

1. File Placement: An "L" (for "Located") is printed next to the file's size if the file is placed as the result of "/POSITION" switch when the file was created or by flagging the file as being placed with UTILTY.
2. The disk block number of the first block of any file may be listed.
3. File attributes can be printed either symbolically or as octal numbers.
4. If the terminal is less than 8Ø characters wide, the OPEN count will be removed from "Full" and "Slow" listings.
5. Feature patch Seq 1Ø.3.1 F may be applied to allow cross-account directory protection.

Use the /HE (help) option to list the new commands.

DSKINT

The DSKINT program supports all new disk types, and reports the current Pack ID and Pack Cluster Size if the disk to be initialized is already a RSTS/E file structured disk.

ERROR PACKAGE

The system error-logging package has been enhanced to support new devices, and will always report the block number of the disk block which may have caused an error.

FILCOM

The FILCOM program has been enhanced as follows:

1. The output file is not opened until both of the input files have been opened (this should make it more difficult to accidentally delete one of the input files).

2. A single-line command format has been added, of the form:

   <outfile>=<infile-1>,<infile-2>[/switches]

3. Wildcards are allowed in file specifications.

FIT

The FIT (FIle Transfer) program is a new program that allows the RT11 directory structure to be maintained on many devices under RSTS/E. FIT permits file structured processing of flexible diskettes (RXØ1 and RXØ2) and TU58 DECtape II under RSTS/E. It will also read and write files on RT11-format RKØ5, RKØ6, RKØ7, RLØ1, RLØ2, and RPØ2 disks. FIT will also transfer files from DOS disks to RSTS/E devices, superseding the PIPEXT program (no longer distributed) in this function. FIT will not support RT11-format disks that have more than 65535 blocks. This program is documented in the RSTS/E System User's Guide and in article Seq 11.15.1 N published in these Release Notes.

FLINT

The FLINT program continues to support the RXØ1 and now supports the RXØ2 in an RXØ1 compatible mode, i.e., as a single-sided, single-density device. In addition, two commands have been added: ZERO and ERASE. The /NH switch appended to the file specification entered in response to the "output to ?" prompt will suppress the header which FLINT puts on RSTS/E files.

GRIPE

The GRIPE program has been modified to allow you to terminate gripes with CTRL/Z, as well as with ESCAPE.

HELP.BAC

This new utility has been added to RSTS/E V7.Ø to provide on-line documentation of many system features. See article Seq 19.4.6 N for a complete description of the HELP package.

INIT.BAC

No changes.

INUSE

No changes.

LOGIN

A spawned job entry point has been added for use by ATPK, and thus by the new BUILD package.

LOGOUT

No changes.

MAKSIL

This is a new program that will make Resident Library SILs from task images. The program is documented in the RSTS/E Programmer's Utilities Manual. This program supersedes MAKRTS.

MONEY

If an account is using 65535 blocks or more on the public structure, the "blocks used" column will show ">=65535."

ODT

The ODT program has been enhanced to allow access to large files and editing of block zero of a disk.

OPSER

The Operator Services package reports the keyboard number from which a message to the operator has been sent.

PIPEXT

The PIPEXT program has been superseded by the PIP.SAV program and will no longer be distributed. (Note that DOS-format disks can be read with the new FIT program.) The PIP.SAV program is documented in the RSTS/E System User's Guide and article Seq 19.4.3 N published in these Release Notes.

PIPSML

The PIPSML program has been superseded by the PIP.SAV program and will no longer be distributed. The PIP.SAV program is documented in the RSTS/E System User's Guide and article Seq 19.4.3 N published in these Release Notes.

PMDUMP

No changes. The content of the dump, however, is explained in the RSTS/E System User's Guide.

PRIOR

The functions of the PRIOR program have been integrated into the UTILTY and SYSTAT programs. PRIOR will no longer be distributed.

QUE

A short-form list command (/S) has been added. The FLUSH command has been documented, and is protected from non-privileged users.

The /AFTER switch will accept relative time specifications, e.g., /AFTER:+2 DAYS. (See article Seq 19.4.5 N published in these Release Notes.)

QUEMAN

The QUEMAN program allows any valid operator to enable and disable queueing and spooling, using INTERRUPT commands of the form:

```
                      DIS[ABLE]   QUE[UE] <message>
PLEASE/INT QUEMAN:                SPO[OL] <message>
                      ENA[BLE]    ALL
```

In other words, the QUEUE, SPOOL, or ALL commands can be used with either ENABLE or DISABLE.

QUOLST

The QUOLST program reports disk blocks used as ">=65535" if the account owns 65535 disk blocks or more.

REACT

The REACT program sets the protection code of the ACCT.SYS file to include the <128> bit, causing RSTS/E to erase the file anytime that it is deleted.

REORDR

No change. However, please read article Seq 1Ø.2Ø.1 N published in these Release Notes.

RUNOFF

The PRINT INDEX feature, which requires an obsolete SORT package, is no longer supported.

SAVE/RESTORE

SAVE/RESTORE is a new utility for copying all allocated clusters on a disk volume to a like disk (IMAGE), or to an intermediate set of volumes (SAVE - disk or tape), which can later be restored to another disk of the same size as the original input disk (RESTORE).

SAVE/RESTORE may be operated off-line or on-line. Off-line, it is available as an option of INIT.SYS. On-line, it is available as the MACRO-11 program SAVRES.SAV which runs under the RT-11 Run-Time System.

Since SAVE/RESTORE uses the disk's storage allocation table ([Ø,1]SATT.SYS) to determine which disk blocks should be transferred, it is essential that the disk be CLEAN (either properly dismounted or cleaned with ONLCLN or the REFRESH/CLEAN suboption of INIT.SYS) before it is copied. If the disk is not clean, there is a good chance that some allocated blocks will not be transferred to the output volume. SAVE/RESTORE will tell you if a disk is not clean, but it will <u>not</u> prevent you from continuing with the operation.

Because the storage allocation table is critical to the operation of SAVE/RESTORE, the input disk must be logically dismounted when SAVE/RESTORE is used on-line to prevent the destruction of critical directory pointers or the loss of any data. This precludes the on-line use of SAVE/RESTORE for the system disk.

SAVE/RESTORE normally transfers each allocated cluster to the same cluster on the output disk. If the output cluster is found to contain bad blocks the data will be relocated if possible, the directory structure modified accordingly, and the bad blocks added to BADB.SYS if not already there.

SAVE/RESTORE was designed with two goals: reliability and speed. Because of this, no directory optimization is attempted. In addition, all allocated input clusters, including those listed in BADB.SYS (in case there are clusters doubly allocated to BADB.SYS and some other file), are transferred. While this may cause errors to be logged, SAVE/RESTORE will determine whether the bad block was previously known or just encountered and proceed accordingly. New input bad blocks and the files in which they occurred will be reported to the user; output bad blocks will only be reported if they result in a file having to be un-placed or made non-contiguous.

SAVE/RESTORE always attempts to create bootable media. The output of a RESTORE or IMAGE copy operation will be bootable if the original disk contained an INIT.SYS file in [Ø,1] (even if the original disk was not bootable).

NOTE

> If the output of a RESTORE or IMAGE copy operation is to be used as a system disk, the INIT INSTALL option must be used to re-install the desired monitor Save Image Library (SIL). In addition, any necessary HARDWR sub-options must be re-done because the INSTALL option clears various items set up by the HARDWR sub-options. This procedure is necessary due to the fact that a disk can be SAVED from one system disk and then RESTOREd on a system with a different hardware configuration. Likewise, a disk which has been IMAGE copied may also be used on a different system.

The first output volume produced by a SAVE operation is also bootable. If a SAVE volume is booted, a portion of INIT.SYS is loaded, allowing you to perform some of INIT's operations. On completion of the RESTORE operation, you must boot the system disk or reboot the first SAVE volume. The recommended procedure is to use DSKINT to initialize the output disk followed by SAVRES to IDENTIFY and/or RESTORE to the disk when booting from a SAVE volume.

Note that the SAVE option is illegal when SAVRES is running from a booted SAVE volume or a booted magnetic tape, regardless of its format.

Refer to the RSTS/E V7.Ø System Manager's Guide for a complete description of SAVE/RESTORE.

SPOOLING PACKAGE

The QUEUE.SYS file used for RSTS/E VØ6C will be deleted with the initial execution of the spooling package for RSTS/E V7.Ø.

BASIC-PLUS programs may now request that files be queued to any valid spooling device by using the Spooling SYS call. (See article Seq 19.6.5 N published in these Release Notes.) MACRO programs may execute a similar request by issuing the UU.SPL UUO. (See article Seq 19.24.1 N published in these Release Notes.)

SHUTUP

The SHUTUP program has been modified to remove resident libraries, and to include the current date and time in the "shutup" messages.

SWITCH

No changes.

SYSCAT

The SYSCAT program is no longer distributed. DIRECT and PIP can be used to display the information formerly supplied by SYSCAT.

SYSTAT

The SYSTAT program has been upgraded to show resident library status and the priority and the run burst of all jobs on the system.

On systems generated with large file support, SYSTAT can be used by privileged users to determine the name, account, status, cluster size, and user of each open file on the disk structure.

SYSTAT also indicates to the privileged user that a job is running with temporary privileges by printing a plus sign (+) next to the job number. A minus sign (-) is used to indicate that a job has temporarily dropped its temporary privileges.

SYSTAT flags jobs operating on dial-up lines by printing an asterisk (*) following the keyboard number.

Disks marked as "Pri, NFS" will also indicate the job number of the job that has the disk assigned.

A new SYSTAT switch, "/C", has been added which displays memory allocation. This feature, which can only be invoked by a privileged user, facilitates the location and placement of resident libraries in memory. (See article Seq 19.4.7 N published in these Release Notes.)

TALK

No changes.

TTYSET

Macro commands have been added for the VT1ØØ, LA34, LA38, and LA12Ø terminals. (See article Seq 1Ø.25.1 N published in these Release Notes.)

UETP

The UETP package has been superseded by the DEVTST package. UETP will no longer be distributed with RSTS/E. The DEVTST Package is described in the RSTS/E V7.Ø System Manager's Guide.

UMOUNT

The UMOUNT program has been enhanced to support new magnetic tape devices, and to allow you to specify "ANSI" format for any tape.

UTILTY

Commands have been added to the UTILTY program to allow you to control system-wide data caching parameters, resident libraries, and the job-oriented commands from PRIOR.

A "SEIZE <device>" command has also been added to UTILTY, allowing privileged users to assign devices that are currently assigned to another user, so long as the devices are not open.

Commands have been added to flag files to be cached and to remove a file's "contiguous" flag, allowing a contiguous file to be extended.

A "SNAP" command has been added which will dump the image of the currently running monitor into the system file CRASH.SYS. The ANALYS program can then be run to examine the state of the system at the time the SNAP command was invoked. Note that the monitor image will overwrite any data which may currently be in CRASH.SYS.

The "DATE" command will accept a date in either DD-MMM-YY or YY.MM.DD format.

UTILTY is now a single program.

VT??PY (The DISPLAY Programs)

The VT??PY programs have been enhanced to show the status and location of resident libraries.

2.8.2 The CUSP Compiler

A new feature of RSTS/E V7.Ø, 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 not 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.

2.8.3 Hints For BASIC-PLUS/BASIC-PLUS-2 Compatibility

In our efforts to make the BASIC-PLUS CUSPs compatible with BASIC-PLUS-2, the RSTS/E Development Group has followed the guidelines outlined below.

1. CCPOS(x) has been added to BASIC-PLUS as a synonym for POS(x). CCPOS(x) should be used.

2. DEF* has been added to BASIC-PLUS as a synonym for DEF. DEF* should be used.

3. BASIC-PLUS recognizes LINE as an optional keyword in the CHAIN statement. The form

   ```
   CHAIN <string> LINE <line #>
   ```

   should be used instead of

   ```
   CHAIN <string> <line #>
   ```

4. All programs should be written in EXTEND mode.

5. All programs should use the "ampersand - CARRIAGE RETURN" construct for continuation lines. Ampersands should be aligned one space after the last significant character on the line, or after one tab on otherwise blank lines. (From TECO, this alignment can be invoked by appending the /B2 switch to the file specification.)

6. Functions with no arguments should never be DEFined or invoked with parentheses. In other words, you should use:

   ```
   DEF* FNA%
   ```

   rather than:

   ```
   DEF* FNA%()
   ```

7. All string literals must be properly delimited with matching quote characters.

8. Implied semicolons in PRINT statements are illegal. In other words, you should use:

```
PRINT "CUSTOMER ADDRESS "; A$
```

rather than:

```
PRINT "CUSTOMER ADDRESS " A$
```

2.9 RMS Software Package

The RMS software bundled with RSTS/E is Version 1.8. The primary emphasis of RMS V1.8 is enhanced performance in two major areas: 1) load times for indexed files, and 2) execution speed of application programs using various indexed file functions (Indexed files, previously available optionally on RSTS/E with RMS-11K, are now standard with RSTS/E V7.Ø).

In the area of load times for indexed files, a completely new RMS utility called Indexed File Load (RMSIFL) is now available. Unlike the existing CONVERT utility (RMSCNV), RMSIFL processes blocks, not records, and utilizes internal knowledge of the actual structure of indexed files to build them "from the bottom up" at enhanced processing speeds.

The second major area of performance concern, the execution speed of user applications, was typically affected by the fact that RMS code in the user task was fully overlayed (4.5KW overlayed versus up to 22KW non-overlayed). To address this issue, a sharable memory-resident library, RMSRES, is included in RMS V1.8. This library requires 23KW of physical memory and 8KW of virtual address space. Use of RMSRES reduces taskbuild time, and, during task execution, this library nearly eliminates the costly I/O overhead associated with disk overlayed tasks.

Additionally, RMS V1.8 provides:

1. Three new disk overlay ODL files, which cause more optimal overlaying of RMS code. Also, an improved prototype ODL (RMS11.ODL) with extensive internal comments is provided.

2. A mass insert mode capability for indexed files. This mode optimizes the loading of a collated series of records in a single-key ISAM file.

3. Deferred-write capability for relative and indexed files. With this capability, I/O buffers are written only when they must be replaced. This eliminates, particularly at initial file load time, the equivalent of one physical write operation for each logical write.

4. Modification of the CONVERT utility (RMSCNV) to take advantage of the mass insert and deferred write capabilities as well as enhanced buffer management.

Another major enhancement to the RMS Software Package is the inclusion of the RMS-11 User's Guide. This manual contains conceptual and technical information, not previously available, on file structure, file design, application optimization techniques, and utilities usage.

Finally, RMS V1.8 contains the following additional items:

1. Five new indexed key data types -- packed decimal, signed and unsigned one-word and two-word integers.

2. Upgrade of DISPLAY (RMSDSP), DEFINE (RMSDEF), BACKUP (RMSBCK), and RESTORE (RMSRST) utilities to reflect new RMS features.

As was true with previous versions of RMS, MACRO programs can access all of the new V1.8 features. COBOL V4.Ø programs can utilize the mass insert and deferred write capabilities. The new ODL files and resident library capability are available to all.

2.1Ø RSX Emulator and Utilities Package

The RSX Emulator is designed to allow several standard RSX-11 CUSPs (i.e., MAC, TKB, LBR, and PAT) to run under RSTS/E. This is done by intercepting the calls to the RSX-11 monitor (EMTs) and translating them into the corresponding calls to the RSTS/E Executive. The commands and EMTs supported are described in the RSTS/E System Directives Manual.

The RSX Emulator and Utilities Package contains the following components:

RSX.RTS - the RSX Emulator Run-Time System which emulates a subset of the RSX-11M system directives. The utilities supplied with this package run under this run-time system. This run-time system can serve as any job's keyboard monitor, and, in fact, may be selected as a system's default run-time system.

TKB.TSK - the RSX-11M Task Builder which links .OBJ files to create runnable tasks.

MAC.TSK - the RSX-11M MACRO assembler which assembles MACRO assembly language source code into linkable object modules.

LBR.TSK - the RSX-11M Librarian program which performs object and macro library maintenance functions.

PAT.TSK - a software maintenance tool which merges original .OBJ files with correction .OBJ files to produce patched .OBJ files.

SYSLIB.OLB - an object module library containing system support subroutines, accessible to the MACRO programming language through the Task Builder. It is also accessible to other programming languages.

RSXMAC.SML - a system library of MACRO routines which form the basis of the calls to SYSLIB.OLB. Standard macros can also be placed here.

EDT.TSK - the DEC Standard editor which edits normal RSTS/E ASCII stream files and RMS-compatible sequential files. The editing commands are primarily line-oriented and operate on an entire file (i.e., no page buffering needs to be taken into account). An updated version is included with this release.

CSPCOM.TSK - the CUSP compiler which generates object modules from the BASIC-PLUS system library programs (CUSPs) that can be Task Built to run under the RSX Run-Time System. (See section 2.8.2 of this chapter.)

CSPCOM.OLB - an object module library which contains system support subroutines available to the CUSPs through CSPCOM.

CRF.TSK - the RSX-11M cross-reference utility which produces cross-referenced listings appended to MAC listing files and TKB map files.

SLOTKB.TSK - a slower version of TKB to be used when TKB's virtual memory space for work files is exceeded.

2.11 RT-11 Emulator and Utilities Package

The RT-11 Emulator is designed to allow several standard RT-11 CUSPs (i.e., MACRO and LINK) to run under RSTS/E. This is done by intercepting the calls to the RT-11 Monitor (EMTs) and translating them into corresponding calls to the RSTS/E Executive. The environment provided is similar to an RT-11 single job system. The commands and EMTs supported are described in article Seq 22.1.1 N published in these Release Notes.

The RT-11 Emulator and Utilities Package contains the following components:

RT11.RTS - the RT-11 Emulator run-time system which emulates a subset of the RT-11 system directives. The utilities supplied with this package run under this run-time system. This run-time system can serve as any job's keyboard monitor, and, in fact, is used as the system default run-time system for the initial installation of RSTS/E V7.Ø. However, some of the CUSPs essential to system operation, such as INIT, ATPK, and SHUTUP, are not available for running under this run-time system.

LINK.SAV - the RT-11 linkage editor which links .OBJ files to create runnable save images.

MACRO.SAV - the RT-11 MACRO assembler which assembles MACRO assembly language source code into linkable object modules.

LIBR.SAV - the RT-11 librarian program which performs object and MACRO library maintenance functions.

PATCH.SAV - the RT-11 patching utility which performs minor modifications to linked .SAV files.

PAT.SAV - the RT-11 object module patching program which performs minor modifications to files in object format.

CREF.SAV - the RT-11 cross-reference utility which produces cross-referenced listings of MACRO source code files.

SYSGEN.SAV - the RSTS/E system generation dialogue program which creates the SYSGEN.CTL and CONFIG.MAC control files.

SYSBAT.SAV - a batch processor runs under RT11.RTS which controls the RSTS/E system generation.

HOOK.SAV - the RSTS/E system program which writes appropriate boot blocks on bootable disks and tapes.

SILUS.SAV - the RSTS/E system program which creates Save Image Library format files (.SILs) from .SAV files.

LOGIN.SAV - a system access utility provided primarily for use during system generation.

LOGOUT.SAV - a system egress utility provided primarily for use during system generation.

PIP.SAV - a system file transfer and maintenance utility that can copy files from one RSTS/E device to another, concatenate, delete, and rename files, transfer protection codes and file attributes, zero accounts, and list directories. This is the only version of PIP distributed with V7.Ø.

ONLCLN.SAV - an on-line version of the CLEAN sub-option of the INIT REFRESH option. This program does all the checking of the INIT CLEAN sub-option while using its own caching mechanism, greatly reducing the time it requires to run.

SAVRES.SAV - a new system program which is a disk backup and copy utility providing SAVE, RESTORE, IMAGE, and IDENTIFY functions.

UTILTY.SAV - a system utility program providing the system manager on-line control of system operations.

ONLPAT.SAV - an on-line version of the INIT PATCH sub-option which can be used to patch monitor SILs, run-time systems, and RT-11 utility programs. (ONLPAT.SAV is listed in the Component.Subcomponent Index as part of the Automated Patching Facility Package.)

2.12 Documentation

1. The following manuals have been re-written for V7.Ø:

   RSTS/E V7.Ø Documentation Directory (AA-2642D-TC)
   RSTS/E V7.Ø System Generation Manual (AA-2669E-TC)
   RSTS/E V7.Ø System Manager's Guide (AA-2762C-TC)
   RSTS/E System User's Guide (AA-5133B-TC)
   BASIC-PLUS Language Manual (AA-2623C-TC)
   RSTS/E Programming Manual (AA-2726C-TC)
   RSTS/E V7.Ø Release Notes (AA-5246C-TC)

2. The following manuals have been added to the RSTS/E Documentation Set for V7.Ø:

   Beginner's Guide to DEC Editor (AA-D374A-TC)
   IAS/RSX ODT Reference Manual (DEC-11-OIODA-A-D)
   RSTS/E System Directive's Manual (AA-D748A-TC)
   RSTS/E Programmer's Utilities Manual (AA-D749A-TC)
   RSTS/E FORTRAN IV Utilities Manual (AA-2140B-TC)
   PDP-11 SORT Manual (AA-3341C-TC)
   PDP-11 TECO User's Guide (DEC-11-UTECA-A-D)
   PDP-11 TECO User's Guide Update (DEC-11-UTECA-A-DN1)
   RMS-11 Installation Guide (AA-H235A-TC)
   RMS-11 User's Guide (AA-D538A-TC)

3. The following manual has been updated for V7.Ø.

   RSTS/E Task Builder Reference Manual (AA-5Ø72A-TC)
   RSTS/E Task Builder Reference Manual Update (AD-5Ø72A-T1)

4. The following manuals and reference materials are also included in the V7.Ø Documentation Set:

   DEC Editor Reference Manual (AA-5789A-TC)
   PDP-11 MACRO-11 Language Reference Manual (AA-5Ø75A-TC)
   RSTS/E Text Editor Manual (DEC-11-UTEMA-A-D)
   RSTS/E RUNOFF User's Guide (DEC-11-URUNA-B-D)
   RMS-11 MACRO-11 Programmer's Manual (AA-H683A-TC)
   Introduction To BASIC (AA-Ø155A-TC)
   RSTS/E Pocket Guide (DEC-11-ORFCA-A-D)

2.13 RNO

RNO has been available from the DECUS library and is widely used by many users of RSTS/E. It is included on the distribution kit of V7.Ø for the convenience of those users who wish access to this program.

NOTE

> RNO is NOT a released product supported by DIGITAL, and the inclusion of this program on the distribution kit does not imply recommendation or endorsement by DIGITAL regarding its use. There is no commitment by DIGITAL to continue to include this program in future distributions of RSTS/E. RNO is to be considered a "user supported" utility.

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 have been 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.

2.14 TECO

TECO is a character-oriented text editor which has been 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 has been added to the documentation set and a document file is included on the distribution kit explaining the use of the TECO Video Terminal Editor (VTEDIT).

The RSTS/E Development Group would like to be informed of any problems with or suggestions concerning TECO V35, as supplied with RSTS/E V7.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 TECO via the SPR mechanism, please check Priority 4, "No significant impact", or Priority 5, "Documentation/Suggestion."

For installation instructions, see article Seq 9.1.1 N published in these Release Notes.

2.15 MACRO Programming

Under RSTS/E V7.Ø, programs as well as subroutines may be written in the MACRO-11 assembly language. To support this capability, four new manuals have been added to the RSTS/E document set for V7.Ø: the RSTS/E System Directives Manual, the RSTS/E Programmer's Utilities Manual, the RSTS/E FORTRAN IV Utilities Manual, and the IAS/RSX ODT Reference Manual.

The RSTS/E System Directives Manual describes how a programmer coding in MACRO-11 can make calls to the RSTS/E Monitor. (By comparison, the RSTS/E Programming Manual describes how a programmer coding in BASIC-PLUS can use SYS calls to the monitor.) The RSTS/E System Directives Manual also describes the subset of RSX-11M directives which are emulated by the RSX Run-Time System, and by the RSTS/E Monitor if that capability has been generated into the system. However, RT-11 system directives are not documented in the manual. Article Seq 22.1.1 N, published in these Release Notes, contains a summary of the directives recognized by the RT-11 Run-Time System. RT-11 system documentation, which may be purchased from Digital, contains detailed descriptions of these directives.

The RSTS/E System Directives Manual also contains a description of the general structure of run-time systems. This description provides a general understanding of run-time systems; the manual is by no means a "How To" manual for building run-time systems. Writing a run-time system should not be attempted by programmers who are not expert in both MACRO-11 programming and the RSTS/E system.

The RSTS/E Programmer's Utilities Manual provides instructions for the MACRO-11 programmer in the use of the RSX utilities: the assembler (MAC), the librarian utility (LBR), the object module patch utility (PAT), and the utility for making a Save Image Library (MAKSIL). Generally, programs prepared with MAC are run under the RSX Run-Time System.

The RSTS/E FORTRAN IV Utilities Manual provides instructions for the FORTRAN and/or MACRO programmer in the use of the RT-11 utilities: the assembler (MACRO), the linker (LINK), the librarian (LIBR), the object module patching utility (PAT), and the Octal Debugging Tool (ODT). Generally, programs prepared with MACRO and LINK are run under the RT-11 Run-Time System.

The IAS/RSX ODT Reference Manual provides instructions for using the Octal Debugging Tool (ODT) when it is linked into a task by the Task Builder.

2.16 Distribution Media

RSTS/E V7.Ø will be distributed on the following media:

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

Please refer to Section 3.Ø of these Release Notes for the contents of the distribution kits.

2.17 Optional Software

Optional software supported by V7.Ø includes the following:

BASIC-PLUS-2	V1.6
COBOL	V4.Ø
DATATRIEVE	V1.1
DECAL	V2
DECnet/E	V1.Ø, V1.1
DIBOL/DECFORM	V4C, V4D
DMS-5ØØ	VØ2
FORTRAN IV	V2.1
RMS-11K	V1.5
RPG II	V8.Ø2
RSTS/E 278Ø	V3.Ø
RSTS/3271 Protocol Emulator	V2.Ø
WISE	V1

NOTE

Prior to installing the optional software products listed immediately below, please refer to the related articles published in these Release Notes for special V7.Ø installation instructions.

COBOL	V4.Ø	Seq 43.1.1 N
DATATRIEVE	V1.1	Seq 42.1.1 N
DECnet/E	V1.Ø	Seq 31.1.1 N
DIBOL/DECFORM	V4D	Seq 36.3.3 N
DIBOL/DECFORM	V4C	Seq 39.3.3 N
DMS-5ØØ	VØ2	Seq 24.1.3 N
RPG II	V8.Ø2	Seq 32.1.3 N
WISE	V1	Seq 35.1.1 N
DX/RSTS	V3.Ø	Seq 6Ø.1.1 N

The following software is not fully supported:

APL-11	V1	(No support services)
DX/RSTS	V3	(No support services)
FORTRAN IV-PLUS	V2.5	(No support services)
NTR/CTS-5ØØ	V1	(No support services outside of Europe)
BASIC-PLUS-2	V1.5	(However, a currently patched version will execute on RSTS/E V7.Ø)

The following software is no longer supported:

BASIC-PLUS-2	V1.Ø4
COBOL	V3.Ø, V3.1
DATATRIEVE	V1
DIBOL/DECFORM	V3C
FORTRAN IV	V2.Ø

2.18 Old Problems Corrected

Various system problems found after the release of RSTS/E VØ6C have been fixed in V7.Ø. All applicable mandatory RSTS/E Software Dispatch articles published through September, 1979 have been incorporated into V7.Ø.

2.19 Upgrading From VØ6C to V7.Ø

When upgrading from VØ6C to V7.Ø, the on-line SYSGEN feature of VØ6C cannot be used. Due to differences in the INIT.SYS program between VØ6C and V7.Ø, the first V7.Ø system generated must be generated OFF-LINE.

3.Ø Contents of RSTS/E V7.Ø Distribution Kits

3.1 Magnetic Tape Distribution Kits

DEC Order No.	BPI	Contents
7-Track	**8ØØ**	**(12ØØ ft reel)**
AP-2772G-BC		System Generation
AP-2725G-BC		System Library I
AP-C726G-BC		System Library II
AP-C881B-BC		RSX Run-Time System and Library
AP-DØ47C-BC		RMS Run-Time System and Library
AP-H465A-BC		RSTS/E V7.Ø Patch Kit "A"
9-Track	**8ØØ**	**(6ØØ ft reel)**
AP-2773G-BC		System Generation
AP-2753G-BC		System Library I
AP-C725G-BC		System Library II
AP-C883B-BC		RSX Run-Time System and Library
AP-5226D-BC		RMS Run-Time System and Library
AP-H466A-BC		RSTS/E V7.Ø Patch Kit "A"
9-Track	**16ØØ**	**(24ØØ ft reel)**
BB-H751G-BC		System Generation, System Library I, System Library II, RSX Run-Time System and Library
BB-H752G-BC		RMS Run-Time System and Library
BB-H753A-BC		RSTS/E V7.Ø Patch Kit "A"

3.2 Disk Distribution Kits

DEC Order No.	Pack ID	Contents
RKØ5		
AN-2771G-BC	SYSGNG	System Generation
AN-2751G-BC	SYSL1G	System Library I
AN-5444G-BC	SYSL2G	System Library II
AN-C885B-BC	RSXLBB	RSX Run-Time System and Library
AN-5227D-BC	RMSKIT	RMS Run-Time System and Library
AN-H467A-BC	PATCHA	RSTS/E V7.Ø Patch Kit "A"
RKØ6		
AM-2774G-BC	SYSGNG	System Generation, RSX Run-Time System and Library
AM-5445G-BC	SYSL1G	System Library I, System Library II
AM-5228D-BC	RMSKIT	RMS Run-Time System and Library
AM-H468A-BC	PATCHA	RSTS/E V7.Ø Patch Kit "A"
RKØ7		
AY-D526G-BC	SYSGNG	System Generation, System Library I, System Library II, RSX Run-Time System and Library
AY-D946C-BC	RMSKIT	RMS Run-Time System and Library
AY-H469A-BC	PATCHA	RSTS/E V7.Ø Patch Kit "A"
RLØ1		
AX-D527G-BC	SYSGNG	System Generation, RSX Run-Time System and Library
AX-D528G-BC	SYSL1G	System Library I, System Library II
AX-D945C-BC	RMSKIT	RMS Run-Time System and Library
AX-H47ØA-BC	PATCHA	RSTS/E V7.Ø Patch Kit "A"

4.Ø RSTS/E V7.Ø Sample Installation Procedure

The following list of tasks is an outline of the steps performed to accomplish the complete installation of a new RSTS/E V7.Ø operating system. Once these steps have been performed, the software specialist can turn the system over to the customer, ready for time sharing.

The time estimates given are for a RSTS/E V7.Ø system on an 11/7Ø with 2 RPØ6 disks, using BASIC-PLUS as the system default 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.

1. **System Generation** **2 - 4 Hours**

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

   (For a complete overview of the system generation process, refer to chapter one of the RSTS/E V7.Ø 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 system default 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 (and if DIGITAL is responsible for the installation) it should be installed and patched at this time. Sample procedures for optional software products should be run at this time 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** **3Ø - 4Ø 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 the user with a full, bootable, system backup.

Note that the user 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 RSTS/E V7.Ø System Manager's Guide.

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.

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

5.Ø Notebook Maintenance

Each month you should take apart the RSTS/E Software Dispatch and merge the new articles by sequence number in your notebook. If an article is a replacement, discard the superseded article. Each article addresses a single topic and its title gives the symptom or a one-line problem description. This information appears at the top of each page. For example:

RSTS/E V7.Ø Release Notes, September 1979

RSTS/E V7.Ø
BASIC-PLUS
BASIC-PLUS Patches

Seq 4.1.4 F

Page 1 of 2

DEFAULT TO EXTEND MODE - BASIC-PLUS FEATURE PATCH

The system components and subcomponents are listed in Section 5.1. Articles for each subcomponent are assigned a sequencing number which uniquely identifies the article within that component.subcomponent category. As each new article is published for a subcomponent, it is assigned the next higher sequence number. The entire number is defined as the COMPONENT.SUBCOMPONENT.ISSUE number, an example being the "4.1.4" shown in the upper right corner of the sample heading. Replacement articles are identified as such.

Articles for subordinate software products that are not unique to RSTS/E are to be filed by software product and subcomponent. Articles are sequenced by subcomponent after this sorting process. In some cases, there are so few articles that they are sequenced only by product. Missing numbers indicate problems which only apply to other operating systems. The cumulative index is published monthly in the RSTS/E Software Dispatch to help you maintain the notebook for the complete RSTS/E system.

A flag will appear after the sequencing number:

M = MANDATORY:	These patches correct errors in the software product. All users are required to apply these patches unless the accompanying article specifies otherwise.
F = FEATURE:	These patches extend or configure non-standard capability into the product. These functions will be treated as a supported part of the product for the duration of the current release.
R = RESTRICTION:	These articles discuss areas that will not be patched in the current release because they require major modifications or because they are not consistent with the design of the product. Restrictions, except those described as permanent, are reviewed and modified when possible as part of the normal release cycle.
N = NOTE:	These articles provide explanatory information that supplements the documentation set and provides more detailed information about a program or package. They also provide procedural information to make it easier to use a program or package.

5.1 RSTS/E Component.Subcomponent Assignments

The component.subcomponent numbers outlined below appear on articles to be filed following Section 8.3 of this notebook.

RSTS/E MONITOR

1.Ø Initialization
- 1.1 INIT.SYS Program Patches
- 1.2 INIT.SYS Program Notes

2.Ø System Generation
- 2.1 SYSGEN Patches
- 2.2 SYSGEN Notes

3.Ø Executive
- 3.1 Monitor Patches
- 3.2 Monitor Notes
- 3.3 Terminal Service Patches
- 3.4 Terminal Service Notes
- 3.5 File Processor Patches
- 3.6 File Processor Notes
- 3.7 Device Driver Patches
- 3.8 Device Driver Notes

4.Ø BASIC-PLUS
- 4.1 BASIC-PLUS Patches
- 4.2 BASIC-PLUS Notes
- 4.3 Programming Hints
- 4.4 Math Packages
- 4.5 Matrix Functions
- 4.6 PRINT-USING
- 4.7 Record I/O
- 4.8 Virtual Core
- 4.9 String Arithmetic
- 4.1Ø Non-Standard BASIC-PLUS Features

5.Ø RSTS/E 278Ø Device Driver
- 5.1 278Ø Device Driver Patches
- 5.2 278Ø Device Driver Notes

6.Ø DECnet/E Package V1.Ø
- 6.1 DECnet/E Patches
- 6.2 DECnet/E Package Notes

7.Ø (Reserved)

8.Ø (Reserved)

9.Ø TECO
- 9.1 TECO Notes
- 9.2 TECO.RTS
- 9.3 TECO.TEC
- 9.4 VTEDIT.TEC
- 9.5 TECO.INI
- 9.6 TYPE.TEC
- 9.7 SQU.TEC
- 9.8 LOCAL.TEC
- 9.9 TECKBM.TEC

RSTS/E UTILITIES

1Ø.Ø System Utilities Package
- 1Ø.1 Package Notes
- 1Ø.2 BUILD
- 1Ø.3 DIRECT
- 1Ø.4 DISPLY
- 1Ø.5 VTØ5 (DISPLY)
- 1Ø.6 VT5Ø (DISPLY)
- 1Ø.7 (Reserved)
- 1Ø.8 DSKINT
- 1Ø.9 GRIPE
- 1Ø.1Ø INIT
- 1Ø.11 INUSE
- 1Ø.12 LOGIN
- 1Ø.13 LOGOUT
- 1Ø.14 MONEY
- 1Ø.15 ODT
- 1Ø.16 PLEASE
- 1Ø.17 (Reserved)
- 1Ø.18 QUOLST
- 1Ø.19 REACT
- 1Ø.2Ø REORDR
- 1Ø.21 SHUTUP
- 1Ø.22 (Reserved)
- 1Ø.23 SYSTAT
- 1Ø.24 TALK
- 1Ø.25 TTYSET
- 1Ø.26 UMOUNT
- 1Ø.27 UTILTY
- 1Ø.28 (Reserved)
- 1Ø.29 SWITCH
- 1Ø.3Ø VT55
- 1Ø.31 DEMO55
- 1Ø.32 HELP

11.Ø Data Manipulation Package

12.Ø Error Control Package

13.Ø Backup Package

14.Ø Spooling and Operator Services Package
- 14.1 Package Notes
- 14.2 OPSER
- 14.3 OPSRUN
- 14.4 QUE
- 14.5 QUEMAN
- 14.6 QUMRUN
- 14.7 CHARS
- 14.8 SPOOL
- 14.9 SPLIDL
- 14.1Ø SPLRUN
- 14.11 BATDCD
- 14.12 BATCH
- 14.13 BATIDL
- 14.14 BATDEC
- 14.15 BATRUN

15.Ø (Reserved)

16.Ø System Management Guidelines

17.Ø Automated Patching Facility Package
- 17.1 Package Notes
- 17.2 PATCPY
- 17.3 PBUILD
- 17.4 CPATCH
- 17.5 AUTOED
- 17.6 ONLPAT.SAV
- 17.7 ATPK

18.Ø Device Test Package
- 18.1 Package Notes
- 18.2 DSKSEK
- 18.3 DSKEXR
- 18.4 LPEXER
- 18.5 PPEXER
- 18.6 PREXER
- 18.7 DXEXER
- 18.8 DTEXER
- 18.9 CPEXER
- 18.1Ø CPUTST
- 18.11 KBEXER
- 18.12 MTEXER

RSTS/E DOCUMENTATION

19.Ø	**Documentation**	
19.1	RSTS/E V7.Ø Documentation Directory	AA-2642D-TC
19.2	RSTS/E V7.Ø System Generation Manual	AA-2669E-TC
19.3	RSTS/E V7.Ø System Manager's Guide	AA-2762C-TC
19.4	RSTS/E System User's Guide	AA-5133B-TC
19.5	BASIC-PLUS Language Manual	AA-2623C-TC
19.6	RSTS/E Programming Manual	AA-2726C-TC
19.7	RSTS/E RUNOFF User's Guide	DEC-11-URUNA-B-D
19.8	RSTS/E Text Editor Manual	DEC-11-UTEMA-A-D
19.9	RSTS/E V7.Ø Release Notes	AA-5246C-TC
19.1Ø	RSTS/E Pocket Guide	DEC-11-ORFCA-A-D
19.11	(Reserved)	
19.12	(Reserved)	
19.13	RSTS/E Task Builder Reference Manual	AA-5Ø72A-TC
	RSTS/E Task Builder Reference Manual Update	AD-5Ø72A-T1
19.14	(Reserved)	
19.15	RMS-11 MACRO Programmer's Reference Manual	AA-H683A-TC
19.16	(Reserved)	
19.17	DEC Editor Reference Manual	AA-5789A-TC
19.18	PDP-11 MACRO-11 Language Reference Manual	AA-5Ø75A-TC
19.19	(Reserved)	
19.2Ø	(Reserved)	
19.21	(Reserved)	
19.22	Beginner's Guide to DEC Editor	AA-D374A-TC
19.23	IAS/RSX ODT Reference Manual	DEC-11-OIODA-A-D
19.24	RSTS/E System Directives Manual	AA-D748A-TC
19.25	RSTS/E Programmer's Utilities Manual	AA-D749A-TC
19.26	RSTS/E FORTRAN IV Utilities Manual	AA-2140B-TC
19.27	PDP-11 SORT Reference Manual	AA-3341C-TC
19.28	PDP-11 TECO User's Guide	DEC-11-UTECA-A-D(+DN1)
19.29	RMS-11 Installation Guide	AA-H235A-TC
19.3Ø	RMS-11 User's Guide	AA-D538A-TC

RUN-TIME SYSTEMS SUPPORT

2Ø.Ø (Reserved)

21.Ø RSX Emulator and Utilities Package

21.1	Package Notes
21.2	(Reserved)
21.3	RSX.RTS
21.4	TKB.TSK
21.5	MAC.TSK
21.6	LBR.TSK
21.7	SYSLIB.OLB
21.8	RSXMAC.SML
21.9	PAT.TSK
21.1Ø	EDT.TSK
21.11	CSPCOM.TSK
21.12	CSPCOM.OLB
21.13	CRF.TSK
21.14	RNO.TSK
21.15	SLOTKB.TSK

22.Ø RT-11 Emulator and Utilities Package

22.1	Package Notes
22.2	RT11.RTS
22.3	CREF.SAV
22.4	(Reserved)
22.5	HOOK.SAV
22.6	LIBR.SAV
22.7	LINK.SAV
22.8	LOGIN.SAV
22.9	LOGOUT.SAV
22.1Ø	MACRO.SAV
22.11	PAT.SAV
22.12	PATCH.SAV
22.13	PIP.SAV
22.14	SILUS.SAV
22.15	SYSBAT.SAV
22.16	SYSGEN.SAV
22.17	UTILTY.SAV
22.18	ONLCLN.SAV
22.19	SAVRES.SAV

5.2 Subordinate Software Component.Subcomponent Assignments

23.Ø (Reserved)

24.Ø DMS-5ØØ VØ2
- 24.1 Package Notes
- 24.2 DMSUTL
- 24.3 DMSFSS
- 24.4 DMSISR
- 24.5 DMSISC
- 24.6 DMSRAR
- 24.7 DMSISA
- 24.8 DMSRAA
- 24.9 DMSFSU
- 24.1Ø DMSISO
- 24.11 DMSRAO
- 24.12 DMSFSX
- 24.13 IAMGEN
- 24.14 IAMCRI
- 24.15 IAMDMP
- 24.16 IAMVFY
- 24.17 IAMPRI
- 24.18 IAMFNS
- 24.19 DSORT
- 24.2Ø DSORTD
- 24.21 DSORTF
- 24.22 DSORTX
- 24.23 DSORTS
- 24.24 DSORTM
- 24.25 DSORTO
- 24.26 DMS-5ØØ User's Guide (DEC-11-ORDUA-B-D)

25.Ø RSTS/E 278Ø PACKAGE V3.Ø
- 25.1 Package Notes
- 25.2 RJ278Ø
- 25.3 RSTS/E 278Ø User's Guide (DEC-11-ORJEA-B-D)

26.Ø (Reserved)

27.Ø (Reserved)

28.Ø (Reserved)

29.Ø RMS-11 V1.5
- 29.1 Package Notes
- 29.2 RMS11.RTS
- 29.3 RMS11.ODL
- 29.4 RMSLIB.OLB
- 29.5 RMSMAC.MLB
- 29.6 RMSUTL.OLB
- 29.7 RSTS/E RMS-11K Installation Guide (AA-5185B-TC)
- 29.8 (Reserved)
- 29.9 (Reserved)
- 29.1Ø RMS-11K V1.5
- 29.11 RMS-11K Package Notes
- 29.12 RMS11.RTS
- 29.13 RMS11.ODL
- 29.14 RMSLIB.OLB
- 29.15 RMSMAC.MLB
- 29.16 RMSUTL.OLB

3Ø.Ø SORT-11 VØ2
- 3Ø.1 Package Notes
- 3Ø.2 SORT Utility
- 3Ø.3 (Reserved)

31.Ø DECnet/E Utilities V1.Ø
- 31.1 Package Notes
- 31.2 NCP
- 31.3 NCU
- 31.4 TLK
- 31.5 LSN
- 31.6 NIP
- 31.7 NETCPY
- 31.8 NCROOT
- 31.9 NETFNC
- 31.1Ø NFT
- 31.11 FAL
- 31.12 RSTS/E DECnet/E System Manager's Guide (AA-5666A-TC)
- 31.13 RSTS/E DECnet/E User's Guide (AA-5667A-TC)
- 31.14 RSTS/E DECnet/E Release Notes (AA-D462A-TC)

32.Ø RPG II V8.Ø2
- 32.1 Package Notes
- 32.2 COMPILER
- 32.3 OTS
- 32.4 RPGFNS
- 32.5 RPGASN
- 32.6 RPGDCV
- 32.7 RPGDMP
- 32.8 RPGESP
- 32.9 RPGFCV
- 32.1Ø RPGPCV
- 32.11 RPG II Report Program Generator Language Manual (AA-5371A-TC)
- 32.12 RPG II Report Program Generator User's Guide (AA-C866A-TC)

33.Ø APL-11 V1
- 33.1 Package Notes
- 33.2 Run-Time System Patches
- 33.3 Documentation

34.Ø (Reserved)

35.Ø WISE V1
- 35.1 Package Notes
- 35.2 UPDATE
- 35.3 SKLTON
- 35.4 REQUES
- 35.5 RGRADE
- 35.6 REGSTR
- 35.7 NEWFIL

36.Ø DIBOL-11/DECFORM V4D
- 36.1 Package Notes
- 36.2 CTSPIP
- 36.3 Build Procedures
- 36.4 DICOMP
- 36.5 DIBOLR
- 36.6 DECFORM
- 36.7 RMS Notes
- 36.8 DIBOL-11 Language Reference Manual (AA-176ØD-TC)
- 36.9 CTS-5ØØ DIBOL User's Guide (AA-52Ø5B-TC)
- 36.1Ø DECFORM User's Guide (AA-5792C-TC)

37.Ø DECAL V2
- 37.1 Package Notes
- 37.2 QUIZ
- 37.3 CREATE

38.Ø (Reserved)

39.Ø DIBOL-11/DECFORM V4C
- 39.1 Package Notes
- 39.2 CTSPIP
- 39.3 Build Procedures
- 39.4 DICOMP
- 39.5 DIBOL
- 39.6 DECFORM
- 39.7 ISMUTL
- 39.8 SORT
- 39.9 DIBOL-11 Language Reference Manual (AA-176ØD-TC)
- 39.1Ø CTS-5ØØ DIBOL User's Guide (AA-52Ø5B-TC)
- 39.11 DECFORM User's Guide (AA-5792C-TC)

4Ø.Ø (Reserved)

41.Ø FORTRAN IV V2.1
- 41.1 Package Notes
- 41.2 COMPILER
- 41.3 OTS
- 41.4 Run-Time System

42.Ø DATATRIEVE V1.1
- 42.1 Package Notes
- 42.2 DTR Patches
- 42.3 QCPRS Patches
- 42.4 QDEXT Patches
- 42.5 User's Guide to DATATRIEVE-11 (AA-C742A-TC)
- 42.6 DATATRIEVE-11(V1.1) IAS, RSX-11M, RSTS/E Installation Guide (AA-H169A-TC)
- 42.7 DATATRIEVE-11 V1.1 Release Notes (AA-D11ØB-TC)

43.Ø COBOL-11 V4.Ø
- 43.1 Package Notes
- 43.2 COBOL Compiler
- 43.3 COBOL Object Library - Overlaid
- 43.4 COBOL ODL
- 43.5 COBOL Utilities
- 43.6 PDP-11 COBOL Language Reference Manual (AA-1749D-TC)
- 43.7 PDP-11 COBOL User's Guide (AA-1757C-TC)
- 43.8 PDP-11 COBOL Installation Guide (AA-1745C-TC)
- 43.9 COBOL Object Library - Non-Overlaid
- 43.1Ø COBOL Interactive Debugger

44.Ø (Reserved)

45.Ø BASIC-PLUS-2 V1.6
- 45.1 Package Notes
- 45.2 BASIC2 Compiler Patches
- 45.3 BASIC2 Run-Time System Patches
- 45.4 BP2COM Run-Time System Patches
- 45.5 BASIC2 Object Library Patches
- 45.6 BP2COM Object Library Patches
- 45.7 BASRMS Object Library Patches
- 45.8 TRANSLATOR Utility
- 45.9 PDP-11 BASIC-PLUS-2 Language Reference Manual (AA-HØ74A-TC)
- 45.1Ø BASIC-PLUS-2 RSTS/E User's Guide (AA-Ø254B-TC)
- 45.11 BASIC-PLUS-2 RSTS/E Installation Guide (AA-Ø156B-TC)
- 45.12 BASICS Shared Library Patches

46.Ø DECnet/E Package V1.1
- 46.1 DECnet/E Patches
- 46.2 DECnet/E Package Notes

47.Ø DECnet/E Utilities V1.1
- 47.1 Package Notes
- 47.2 NCP
- 47.3 NCU
- 47.4 TLK
- 47.5 LSN
- 47.6 NIP
- 47.7 NETCPY
- 47.8 NCROOT
- 47.9 NET
- 47.1Ø NFT
- 47.11 FAL
- 47.12 NETACT
- 47.13 NPKDVR
- 47.14 MSRSUB (RT-11)
- 47.15 MSRSUB (RSX-11)
- 47.16 DNEHHL (RT-11)
- 47.17 DNEHHL (RSX-11)
- 47.18 DNECCI
- 47.19 RSTS/E DECnet/E System Manager's Guide (AA-H5Ø5A-TC)
- 47.2Ø RSTS/E DECnet/E Release Notes (AA-D426B-TC)
- 47.21 RSTS/E DECnet/E Network Programming In BASIC-PLUS and BASIC-PLUS-2 (AA-H5Ø1A-TC)
- 47.22 RSTS/E DECnet/E Network Programming In MACRO-11 and FORTRAN (AA-H5Ø2A-TC)
- 47.23 RSTS/E DECnet/E Network Programming in COBOL (AA-H5Ø3A-TC)
- 47.24 RSTS/E DECnet/E Utilities Manual (AA-H5Ø4A-TC)

48.Ø (Reserved)

49.Ø (Reserved)

5Ø.Ø KMC Drivers
- 5Ø.1 Package Notes
- 5Ø.2 IBM Interconnect (II) Driver
- 5Ø.3 KMC11 (XK) Driver

51.Ø RSTS/3271 Protocol Emulator V1.1
- 51.1 Package Notes
- 51.2 KMCUT
- 51.3 3271 Microcode (C-RAM)
- 51.4 3271 Microcode (RAM)
- 51.5 MTO
- 51.6 RSTS/3271 Protocol Emulator Release Notes
- 51.7 RSTS/3271 Protocol Emulator User's Guide

52.Ø (Reserved)

53.Ø (Reserved)

54.Ø (Reserved)

55.Ø (Reserved)

56.Ø (Reserved)

57.Ø (Reserved)

58.Ø (Reserved)

59.Ø (Reserved)

6Ø.Ø DX/RSTS V3.Ø
- 6Ø.1 Package Notes

9Ø.Ø User Assigned Components - For Development Use
- 9Ø.1 Package Notes

91.Ø - 99.Ø User Assigned Components - For User Use

6.Ø Software Performance Reports

Each new installation is provided with Software Performance Report (SPR) forms. The SPR form enables the user to suggest enhancements to, or report problems, with DIGITAL software or documentation. When a problem is encountered, an SPR should be completed and mailed to the local SPR Center (see the inside back cover of the SPR form).

Responses will be sent to the name and address appearing on the form. Additional SPR forms may be obtained by writing to the local SPR Center. SPR response is provided at no charge for one year after installation and may be continued by subscription thereafter.

6.1 Software Performance Report Guidelines

These guidelines for SPR completion ensure that adequate information is included to prevent delays in processing.

For all types of actual or suspected software problems, the following should be included with the SPR:

1. A complete description of the problem.
2. The CPU type.
3. The system disk type.
4. The amount and type of memory.
5. If possible, a simple program or procedure which can be used to reproduce the problem.
6. Any additional information which you may think is relevant.

For problems associated with the RSTS/E Monitor, also include:

1. The crash dump printed by ANALYS and the CRASH.SYS file (dump only if no magnetic tape). Be sure to use the "/WIDE" switch when running ANALYS to produce an annotated listing.
2. The map files created during SYSGEN (i.e., files created with ".MAP" extension).
3. A list of any optional feature patches installed.

4. A machine readable copy of your monitor.

5. A listing of your CONFIG.MAC and SYSGEN.CTL files created during SYSGEN.

When submitting an SPR concerning a DIGITAL supplied BASIC-PLUS program (CUSP), also include:

1. The name of the CUSP.

2. A "RUN" of the CUSP which displays the header line, version number, and the problem which is occurring, if possible.

3. A list of any optional feature patches installed.

4. Whether the program was compiled under BASIC-PLUS, BASIC-PLUS-2, or CSPCOM. In the latter two cases, also include the name of the run-time system under which the program was running.

5. Under what account(s) the problem occurs, distinguishing between privileged and non-privileged behavior.

6. A complete listing of the CUSP (preferably on magnetic tape) if your installation has made any non-standard changes.

7. Listings of all relevant input and output files.

NOTE

SPR turnaround time for CUSPs running under the BASIC-PLUS Run-Time System can be reduced by including a "DUMP" of the CUSP. See articles Seq 4.1Ø.2 N, Seq 4.1Ø.4 F, and Seq 4.1Ø.6 F in these notes.

NOTE ALSO

No SPR, monitor or CUSP, will be considered unless ALL relevant Mandatory patches published to date in the RSTS/E Software Dispatch have been applied. Before submitting an SPR, the user should review the relevant section(s) of the Maintenance Notebook, containing Release Notes and Software Dispatch articles, to insure that the problem has not already been solved and a solution published.

7.Ø Patching Procedure

This section briefly describes the patching procedures to be used for maintenance of RSTS/E V7.Ø. Programs included in the patching package are ONLPAT for the monitor, and ATPK, BUILD, PBUILD, PATCPY, CPATCH, and AUTOED for the CUSPs. The procedures to be followed for patching the monitor and CUSPs when generating a new RSTS/E system are described in the RSTS/E V7.Ø System Generation Manual.

Some patches to the monitor or INIT code must be applied manually with the INIT PATCH option before the system can be started for autopatching. Any patches requiring such special attention will be clearly marked in their accompanying article.

The programs used for automated patching can also be used for manual patching. The procedures for manual patching with these programs can be found in the RSTS/E V7.Ø System Manager's Guide.

Two types of patching will be used for RSTS/E software: 1) Monitor patching and 2) CUSP patching. Other optional software supplied may require different patching procedures. In all cases, however, the patching procedure will be described in the accompanying patch article.

Unless otherwise stated, all patches published in the Release Notes and the RSTS/E Software Dispatch are MANDATORY patches which must be made to the affected component as soon as possible. Since not all components will be included at each installation, patches to unused components will fail when the attempt is made to apply them. The system manager should file all articles, however, even for unused components, since the installation might choose at some future time to install additional components and these components will then have to be brought up to the latest support level.

Some articles will detail the procedure to patch a component in order for the installation to use an optional feature of that component. These optional "Feature" patches are not required to keep a component at its current support level. If problems occur with a component containing optional features, however, the installation must report which optional feature patches are in use when filing the Software Performance Report (SPR).

Next, some discussion of documentation conventions needs to be presented. The following list of syntactic elements provides a description of certain terminal keys which have non-printing functions:

^C	up-arrow/C or Control-C

NOTE

The syntactic element "^C" refers only to "up-arrow/C" (i.e., typing upper-shift 6 followed by the character "C") when used in patches using the program ONLPAT.

CTRL/C	Control-C
<tab>	Horizontal tab
<lf>	Line feed
<ff>	Form feed
<cr>	Carriage return
^Z or CTRL/Z	Control-Z
<esc>	Escape (Alt Mode or SEL)

When one of these syntactic elements appears in the documentation, it signifies that the associated key(s) is (are) pressed if input is desired or the appropriate terminal action occurs if an output operation is being described.

An optional item will be indicated by enclosing it in square brackets. Note that nesting may take place, as in the following example illustrating the use of the "/DETACH" switch:

```
PIP[/DET[A[C[H]]]] filespec=filespec
```

The modifier /DETACH is optional, as are the last three letters of DETACH. In other words, /DET, /DETA, /DETAC, and /DETACH are acceptable, if this modifier is used.

NOTE

Whenever a project-programmer number is called for in the text, the square brackets [and] which may appear are explicitly stated and do not signify that the enclosed quantity is optional.

7.1 Monitor Patching

Changes to the RSTS/E monitor will be made via the PATCH option of INIT.SYS and/or the program ONLPAT, as described in the RSTS/E V7.Ø System Generation Manual.

Release Note articles discussing various monitor options and problem corrections will contain the exact operator action required when using the PATCH option. The expected dialogue will be shown in the article as it should appear when the patch is made. If any deviation occurs, the process should be halted, any changes already made returned to their former values via the PATCH option, and a determination made as to where the discrepancy occurred.

7.2 Commonly Used System Program (CUSP) Patching

The RSTS/E CPATCH program will serve as the CUSP program patching tool for maintenance purposes. The program uses a subset of the RSTS/E EDIT program commands to perform patching of one line at a time.

Refer to the RSTS/E V7.Ø System Manager's Guide for complete information on the commands and capabilities of CPATCH and PBUILD.

Each patch notice will be self-contained as a convenience to the person who will be making the change. It is recommended that the date of the change and the name of the person making the patch be written on the article at the time the program modification is performed in order to keep an accurate record of the status of the system library software.

8.Ø Release Notes and RSTS/E Software Dispatch Articles

Section 8.Ø is the repository for all Release Notes and RSTS/E Software Dispatch articles. This section will include problem solutions, optional "Feature" patches, documentation errata, programming hints, and tips for better operation of RSTS/E.

Anyone who generates a RSTS/E system (whether on-line under time sharing or as the initial version of the system) should become familiar with these articles. This should be a continuing process of education as sub-sections are added and updated.

Beginning on the next page is an index to all Release Note articles. This index will be updated monthly as new articles are published in the RSTS/E Software Dispatch.

8.1 Cumulative Index of All Articles

8.2 Patch Levels for Commonly Used System Programs (CUSPs)

This report reflects the current Version/Edit level of all of the Commonly Used System Programs (CUSPs) supplied as part of the standard RSTS/E V7.Ø-Ø7 System Library Programs release.

System Utilities Package

BUILD V7.Ø-Ø7	DIRECT V7.Ø-Ø7	DISPLY V7.Ø-Ø7	DSKINT V7.Ø-Ø7
GRIPE V7.Ø-Ø7	INIT V7.Ø-Ø7	INUSE V7.Ø-Ø7	LOGIN V7.Ø-Ø7
LOGOUT V7.Ø-Ø7	MONEY V7.Ø-Ø7	ODT V7.Ø-Ø7	PLEASE V7.Ø-Ø7
QUOLST V7.Ø-Ø7	REACT V7.Ø-Ø7	REORDR V7.Ø-Ø7	SHUTUP V7.Ø-Ø7
SWITCH V7.Ø-Ø7	SYSTAT V7.Ø-Ø7	TALK V7.Ø-Ø7	TTYSET V7.Ø-Ø7
UMOUNT V7.Ø-Ø7	VTØ5 V7.Ø-Ø7	VT5Ø V7.Ø-Ø7	

Data Manipulation Package

BPCREF V7.Ø-Ø7	BPCRF1 V7.Ø-Ø7	COPY V7.Ø-Ø7	EDIT V7.Ø-Ø7
EDITCH V7.Ø-Ø7	FILCOM V7.Ø-Ø7	FIT V7.Ø-Ø7	FLINT V7.Ø-Ø7
MAKSIL V7.Ø-Ø7	PMDUMP V7.Ø-Ø7	RUNOFF V7.Ø-Ø7	

Error Control Package

ANALYS V7.Ø-Ø7	ANALY1 V7.Ø-Ø7	ERRBLD V7.Ø-Ø7	ERRCPY V7.Ø-Ø7
ERRDET V7.Ø-Ø7	ERRDIS V7.Ø-Ø7	ERRINT V7.Ø-Ø7	

Backup Package

BACCOM V7.Ø-Ø7	BACCON V7.Ø-Ø7	BACDEL V7.Ø-Ø7	BACDIR V7.Ø-Ø7
BACDMP V7.Ø-Ø7	BACDSK V7.Ø-Ø7	BACENT V7.Ø-Ø7	BACFRM V7.Ø-Ø7
BACLAB V7.Ø-Ø7	BACLOD V7.Ø-Ø7	BACLST V7.Ø-Ø7	BACMNT V7.Ø-Ø7
BACPRM V7.Ø-Ø7	BACKTO V7.Ø-Ø7	BACKUP V7.Ø-Ø7	

Spooling and Operator Services Package

BATCH V7.Ø-Ø7	BATDCD V7.Ø-Ø7	BATDEC V7.Ø-Ø7	BATIDL V7.Ø-Ø7
BATRUN V7.Ø-Ø7	CHARS V7.Ø-Ø7	OPSER V7.Ø-Ø7	OPSRUN V7.Ø-Ø7
QUE V7.Ø-Ø7	QUEMAN V7.Ø-Ø7	QUMRUN V7.Ø-Ø7	SPLIDL V7.Ø-Ø7
SPLRUN V7.Ø-Ø7	SPOOL V7.Ø-Ø7		

Automated Patching Facility Package

ATPK V7.Ø-Ø7	AUTOED V7.Ø-Ø7	CPATCH V7.Ø-Ø7	ONLPAT V7.Ø-Ø7
PATCPY V7.Ø-Ø7	PBUILD V7.Ø-Ø7		

Device Test Package

CPEXER V7.Ø-Ø7	CPUTST V7.Ø-Ø7	DSKEXR V7.Ø-Ø7	DSKSEK V7.Ø-Ø7
DTEXER V7.Ø-Ø7	DXEXER V7.Ø-Ø7	KBEXER V7.Ø-Ø7	LPEXER V7.Ø-Ø7
MTEXER V7.Ø-Ø7	PPEXER V7.Ø-Ø7	PREXER V7.Ø-Ø7	

DECnet/E Utilities V1.1

NCP V7.Ø-Ø7	NCU V7.Ø-Ø7	TLK V7.Ø-Ø7	LSN V7.Ø-Ø7
NIP V7.Ø-Ø7	NETCPY V7.Ø-Ø7	NCROOT V7.Ø-Ø7	NET V7.Ø-Ø7
NETWRK V7.Ø-Ø7	NPKDVR V7.Ø-Ø7	MSRSUB(RT) V7.Ø-Ø7	MSRSUB(RSX) V7.Ø-Ø7
DNEHLL(RT) V7.Ø-Ø7	DNEHLL(RSX) V7.Ø-Ø7	DNECCI V7.Ø-Ø7	

DECnet/E Utilities V1.Ø

LSN VØ6C-Ø3	NCP VØ6C-Ø3A	NCROOT VØ6C-Ø3AB	NCU VØ6C-Ø3AB
NETFNC VØ6C-Ø3	NIP VØ6C-Ø3	TLK VØ6C-Ø3A	

8.3 Protection Codes for Commonly Used System Programs (CUSPs)

This table lists the protection codes for the Commonly Used System Programs (CUSPs) supplied as part of the standard RSTS/E V7.Ø System Library Program release.

System Utilities Package

BUILD <124>	DIRECT <232>	DSKINT <124>	GRIPE <232>
INIT <124>	INUSE <1Ø4>	LOGIN <232>	LOGOUT <232>
MONEY <1Ø4>	ODT <124>	PLEASE <232>	QUOLST <232>
REACT <124>	REORDR <124>	SHUTUP <124>	SWITCH <232>
SYSTAT <232>	TALK <232>	TTYSET <232>	UMOUNT <232>
VT5DPY <232>	VT5ØPY <232>		

Data Manipulation Package

BPCREF <1Ø4>	BPCRF1 <1Ø4>	COPY <1Ø4>	EDIT <1Ø4>
EDITCH <1Ø4>	FILCOM <1Ø4>	FIT <232>	FLINT <1Ø4>
MAKSIL <1Ø4>	PMDUMP <1Ø4>	RUNOFF <1Ø4>	

Error Control Package

ANALYS <124>	ANALY1 <124>	ERRBLD <124>	ERRCPY <124>
ERRDET <124>	ERRDIS <124>	ERRINT <124>	

Backup Package

BACCOM <232>	BACCON <232>	BACDEL <232>	BACDIR <232>
BACDMP <232>	BACDSK <232>	BACENT <124>	BACFRM <232>
BACLAB <232>	BACLOD <232>	BACLST <232>	BACMNT <232>
BACPRM <124>	BACKTO <232>	BACKUP <232>	

Spooling and Operator Services Package

BATCH <124>	BATDCD <124>	BATDEC <124>	BATIDL <124>
BATRUN <124>	CHARS <124>	OPSER <124>	OPSRUN <124>
QUE <232>	QUEMAN <124>	QUMRUN <124>	SPLIDL <124>
SPLRUN <124>	SPOOL <124>		

Automated Patching Facility Package

ATPK <124>	AUTOED <124>	CPATCH <124>	ONLPAT <124>
PATCPY <124>	PBUILD <124>		

Device Test Package

CPEXER <124>	CPUTST <124>	DSKEXR <124>	DSKSEK <124>
DTEXER <124>	DXEXER <124>	KBEXER <124>	LPEXER <124>
MTEXER <124>	PPEXER <124>	PREXER <124>	

DECnet/E Utilities V1.1

NCP <124>	NCU <124>	TLK <232>	LSN <232>
NIP <232>	NETCPY <232>	NCROOT <232>	NET <232>
NETACT <1Ø4>	NPKDVR <124>		

DECnet/E Utilities V1.Ø

LSN <232>	NCP <124>	NCU <124>	NETCPY <232>
NIP <232>	TLK <232>		

RSX Emulator Package

CRF <1Ø4>	CSPCOM.OLB <4Ø>	CSPCOM.TSK <1Ø4>	EDT <1Ø4>
LBR <1Ø4>	MAC <1Ø4>	PAT <1Ø4>	RNO <1Ø4>
RSXMAC <4Ø>	SLOTKB <1Ø4>	SYSLIB <4Ø>	TKB <1Ø4>

RT-11 Emulator Package

CREF <1Ø4>	HOOK <1Ø4>	LIBR <1Ø4>	LINK <1Ø4>
LOGIN <232>	LOGOUT <232>	MACRO <1Ø4>	ONLCLN <124>
PAT <1Ø4>	PATCH <1Ø4>	PIP <232>	SAVRES <124>
SILUS <1Ø4>	SYSBAT <124>	SYSGEN <124>	UTILTY <1Ø4>

8.4 Compiling CUSPs under BASIC-PLUS-2 or RSX (CSPCOM)

The patching procedure included in those articles that patch BASIC-PLUS source code refer to this section for the appropriate procedure to be followed if BASIC-PLUS is not your system default run-time system. (The BUILD/AUTOPATCH will perform all of these steps for you if you are installing patches from a patch kit.) The sequence of commands for compiling BASIC-PLUS programs under the BASIC-PLUS Run-Time System is (items in square brackets are patch or program dependent):

```
OLD PRGNAM.BAS
[APPEND APPEND.BAS]
COMPILE [SYØ:]$PRGNAM[<prot>]
```

The paragraphs which follow describe the comparable commands if BASIC-PLUS-2 or RSX (CSPCOM) is your system default run-time system.

Note that the following CUSPs must always reside on SYØ:, regardless of your system default run-time system:

ATPK INIT LOGIN LOGOUT SHUTUP UTILTY

BASIC-PLUS-2

If an article indicates that the program "should be compiled under the BASIC2 Run-Time System", the compile sequence is:

```
SWITCH BP2COM
SCALE Ø
OLD PRGNAM.BAS
[APPEND APPEND.BAS]
COMPILE [SYØ:]$PRGNAM[<prot>]/TSK/CHA/LIN/NODEB
```

If an article indicates that the program "must be task built against the BP2COM Run-Time System", use the following sequence of commands:

```
SWITCH BP2COM
SCALE Ø
OLD PRGNAM.BAS
[APPEND APPEND.BAS]
COMPILE PRGNAM.OBJ/CHA/LIN/NODEB/OBJ
RUN SY:[1,2]TKB.TSK
[SYØ:]$PRGNAM.TSK/FP=PRGNAM.OBJ,LB:BP2COM.OLB/LB
/
HISEG=BP2COM
UNITS=12
ASG=SY:5:6:7:8:9:1Ø:11:12
//
RUN SY:[1,2]PIP.SAV
PRGNAM.OBJ/DE:NO
[[SYØ:]$PRGNAM.TSK<prot>/RE]
^Z
```

Programs which must be task built include those which must reside on SYØ:, as well as FILCOM and FIT.

RSX (CSPCOM)

If RSX is your system default Run-Time System, use the following sequence of commands:

```
SWITCH RSX
RUN $CSPCOM
PRGNAM.OBJ/OBJ=PRGNAM.BAS[,APPEND.BAS]
^Z
RUN SY:[1,2]TKB.TSK
PRGNAM.TSK/FP=PRGNAM.OBJ,LB:CSPCOM.OLB/LB
/
UNITS=12
ASG=SY:5:6:7:8:9:1Ø:11:12
//
RUN SY:[1,2]PIP.SAV
PRGNAM.OBJ/DE:NO
[[SYØ:]$PRGNAM.TSK<prot>/RE]
^Z
```

RSTS/E V7.Ø
Initialization
INIT.SYS Program Patches

Seq 1.1.1 F

CHANGE SAVRES DENSITY DEFAULT TO 16ØØ BPI - INIT.SYS FEATURE PATCH

PROBLEM:

SAVRES, as released, defaults to the lowest density (of 8ØØ BPI and 16ØØ BPI) allowed by a given tape drive (normally 8ØØ BPI) for all tape I/O. It may be desirable at some installations to have SAVRES default to the highest density allowed by a given tape drive for more compact storage of data. This is especially true if all drives on the system support 16ØØ BPI and the problem described below is not applicable.

CAUTION

If the tape drive being used has a TMØ2 formatter, the hardware bootstrap of a 16ØØ BPI tape will not be possible. (You can determine the formatter type by using the HARDWR LIST suboption of INIT.) Therefore, if it is desirable for SAVRES to create tapes which are bootable on such a drive it is recommended that this patch not be installed. Normally, TU16 and TU45 drives use a TMØ2 formatter, TE16 and TU77 drives do not.

SOLUTION:

The following feature patch will cause the SAVRES option of INIT.SYS to default to the highest density allowed by a given tape drive for all tape I/O. This default can be overridden by attaching the /DENSITY:8ØØ switch to the device specification. Refer also to article Seq 22.19.1 F, which supplies a similar patch for SAVRES.SAV.

PROCEDURE:

1. This is a feature patch to the RSTS/E INIT.SYS Program. Since patching the distribution medium is not recommended, the patch must be installed every time the COPY option is used to copy INIT.SYS from the distribution medium. Any copy of the patched INIT.SYS will propagate the feature.

2. The patch described in Step 4 below can be installed using the PATCH option of INIT.SYS:

   ```
   Option: PATCH
   File to patch? INIT.SYS
   ```

3. This patch can be installed manually using ONLPAT, the on-line patching program:

   ```
   RUN $ONLPAT
   Command File Name? <cr>                (RETURN for manual patch installation)
   File to patch? INIT.SYS
   File found in account [Ø,1]
   ```

   The patch is also contained in a patch file appearing in patch kit version "A" or later.

4. The patch is as follows:

```
Base address? DIACTL
Offset address? 31Ø
 Base   Offset  Old     New?
??????  ØØØ31Ø  Ø3ØØ7Ø  ? "16
??????  ØØØ312  177460  ? "ØØ
??????  ØØØ314  Ø33Ø61  ? Ø34377
??????  ØØØ316  Ø3ØØ6Ø  ? <lf>          (no change; verify only)
??????  ØØØ32Ø  ØØØ377  ? <lf>          (no change; verify only)
??????  ØØØ322  ØØ144Ø  ? 16ØØ.
??????  ØØØ324  ØØ31ØØ  ? 8ØØ.
??????  ØØØ326  ØØØØØØ  ? ^Z           (CTRL/Z for new offset)
Offset address? ^Z                     (CTRL/Z for new base)
Base address? SAVCTL
Offset address? 42
 Base   Offset  Old     New?
??????  ØØØØ42  Ø3ØØ7Ø  ? "16
??????  ØØØØ44  177460  ? "ØØ
??????  ØØØØ46  Ø33Ø61  ? Ø34377
??????  ØØØØ5Ø  Ø3ØØ6Ø  ? <lf>          (no change; verify only)
??????  ØØØØ52  ØØØ377  ? <lf>          (no change; verify only)
??????  ØØØØ54  ØØ144Ø  ? 16ØØ.
??????  ØØØØ56  ØØ31ØØ  ? 8ØØ.
??????  ØØØ6Ø  ØØØØØØ  ? ^Z           (CTRL/Z for new offset)
Offset address? ^Z                     (CTRL/Z for new base)
Base address? RESCTL
Offset address? 3Ø
 Base   Offset  Old     New?
??????  ØØØØ3Ø  Ø3ØØ7Ø  ? "16
??????  ØØØØ32  177460  ? "ØØ
??????  ØØØØ34  Ø33Ø61  ? Ø34377
??????  ØØØØ36  Ø3ØØ6Ø  ? <lf>          (no change; verify only)
??????  ØØØØ4Ø  ØØØ377  ? <lf>          (no change; verify only)
??????  ØØØØ42  ØØ144Ø  ? 16ØØ.
??????  ØØØØ44  ØØ31ØØ  ? 8ØØ.
??????  ØØØØ46  ØØØØØØ  ? ^C           (up-arrow/C to exit;CTRL/C for INIT)
```

RSTS/E V7.Ø
Initialization
INIT.SYS Program Patches

Seq 1.1.2 F

MAKING /NOERROR THE DEFAULT - INIT.SYS FEATURE PATCH

PROBLEM:

The /NOERROR switch indicates that SAVRES should abort under the following conditions:

1. A contiguous file is made non-contiguous
2. A placed file is 'unplaced'
3. A bad comparison occurs
4. An unexpected bad block is encountered on the input RSTS/E disk in a SAVE or IMAGE operation

Some installations, however, may always want SAVRES to abort under these circumstances.

SOLUTION:

The following feature patch will make /NOERROR, rather than /ERROR, the default in the SAVRES option of INIT.SYS. Note that the user may override the default for a particular operation by specifying /ERROR. Refer also to article Seq 22.19.2 F, which supplies a similar patch for SAVRES.SAV.

PROCEDURE:

1. This is a feature patch to the RSTS/E INIT.SYS Program. Since patching the distribution medium is not recommended, the patch must be installed every time the COPY option is used to copy INIT.SYS from the distribution medium. Any copy of the patched INIT.SYS will propagate the feature.

2. The patch described in Step 4 below can be installed using the PATCH option of INIT.SYS:

   ```
   Option: PATCH
   File to patch? INIT.SYS
   ```

3. This patch can be installed manually using ONLPAT, the on-line patching program:

   ```
   RUN $ONLPAT
   Command File Name? <cr>                (RETURN for manual patch installation)
   File to patch? INIT.SYS
   File found in account [Ø,1]
   ```

   The patch is also contained in a patch file appearing in patch kit version "A" or later.

4. The patch is as follows:

```
Base address? ..NOER
Offset address? 2
 Base   Offset  Old     New?
??????  ØØØØØ2  ØØØØ1Ø  ? 4
??????  ØØØØØ4  ??????  ? <lf>          (no change; verify only)
??????  ØØØØØ6  ??????  ? <lf>          (no change; verify only)
??????  ØØØØ1Ø  ??????  ? <lf>          (no change; verify only)
??????  ØØØØ12  ØØØØØ4  ? 1Ø
??????  ØØØØ14  ??????  ? ^C            (up-arrow/C to exit;CTRL/C for INIT)
```

RSTS/E V7.Ø
Initialization
INIT.SYS Program Patches

Seq 1.1.3 F

MAKING /NOSTATS THE DEFAULT - INIT.SYS FEATURE PATCH

PROBLEM:

SAVRES always prints a summary report after completing a transfer unless the /NOSTATS (no statistics) switch is specified. Certain installations may prefer that this report NOT be printed unless specifically requested.

SOLUTION:

The following feature patch will make /NOSTATS, rather than /STATS, the default in the SAVRES option of INIT.SYS. Note that the user may override this default for a particular operation by specifying /STATS. Refer also to article Seq 22.19.3 F, which supplies a similar patch for SAVRES.SAV.

PROCEDURE:

1. This is a feature patch to the RSTS/E INIT.SYS Program. Since patching the distribution medium is not recommended, the patch must be installed every time the COPY option is used to copy INIT.SYS from the distribution medium. Any copy of the patched INIT.SYS will propagate the feature.

2. The patch described in Step 4 below can be installed using the PATCH option of INIT.SYS:

```
Option: PATCH
File to patch? INIT.SYS
```

3. This patch can be installed manually using ONLPAT, the on-line patching program:

```
RUN $ONLPAT
Command File Name? <cr>                (RETURN for manual patch installation)
File to patch? INIT.SYS
File found in account [Ø,1]
```

   The patch is also contained in a patch file appearing in patch kit version "A" or later.

4. The patch is as follows:

```
Base address? ..NOST
Offset address? 2
 Base   Offset  Old     New?
??????  ØØØØØ2  ØØØØ4Ø  ? 2Ø
??????  ØØØØØ4  ??????  ? <lf>          (no change; verify only)
??????  ØØØØØ6  ??????  ? <lf>          (no change; verify only)
??????  ØØØØ1Ø  ??????  ? <lf>          (no change; verify only)
??????  ØØØ12   ØØØØ2Ø  ? 4Ø
??????  ØØØ14   ??????  ? ^C            (up-arrow/C to exit;CTRL/C for INIT)
```

CLEANING DISKS AFTER SYSTEM CRASHES

In RSTS/E V7.Ø, system crash processing operates as follows:

1. Volatile memory is written to [Ø,1]CRASH.SYS after every crash if the crash dump facility was enabled at system startup, regardless of the setting of the console switches.

2. The system performs an automatic restart if the console switches were set to 177777, if the switch register was disabled with the HARDWARE option of INIT, or if the switch register was volatile and the crash was due to a power failure.

If the conditions for performing an automatic restart are not met, or if the system crashed twice within one minute, the CPU halts at address 54. If auto-restart is enabled, the initialization program INIT.SYS is booted, the system disk cleaned, and the INIT library program entered to perform the commands in the $CRASH.CTL file.

On a multi-disk system, commands in $CRASH.CTL should attempt to mount the remaining disks of the public structure. INIT.BAC automatically executes the CLEAN SYS function call as needed.

There is one disadvantage in letting INIT.BAC automatically execute the CLEAN SYS function when it mounts a "dirty" disk. The CLEAN SYS call is limited and does not perform all the validation and correction of the disk file structure that the CLEAN function of the INIT.SYS REFRESH option does.

DIGITAL, therefore, recommends that you add commands to run $ONLCLN.SAV to the $CRASH.CTL file before the commands that attempt to mount any disks. $ONLCLN.SAV will perform all the consistency checks and file structure corrections that the CLEAN function of the INIT.SYS REFRESH option provides.

ADDING BAD BLOCKS TO THE BAD BLOCK FILE

The ERRDIS program prints a list of potential bad blocks. If you decide to add a block to the bad block file on a disk, you may use the BADS suboption of the REFRESH option during initialization as described in the RSTS/E V7.Ø System Generation Manual.

DIGITAL recommends that you follow the procedure below:

1. Use PIP to copy the data in the bad file to a new file. Specify the /GO switch (along with other necessary switches, such as /CL, etc.) to ignore possible USER DATA ERROR ON DEVICE errors. Rename the file which contains the bad block, but do not delete it:

```
RUN $PIP
*FILE.TMP=FILE.OLD/GO
*FILE.BAD=FILE.OLD/RE
*FILE.OLD=FILE.TMP/RE
```

2. When convenient, shut down the system and use the BADS suboption of REFRESH to add any bad blocks to the disk's bad block file. Each new bad block is now allocated to two files ([Ø,1]BADB.SYS and FILE.BAD in the above example).

3. The CLEAN suboption automatically rebuilds the allocation tables. CLEAN will indicate that the file has a bad block and allow you to delete the file (FILE.BAD in the above example). This is described in the RSTS/E V7.Ø System Generation Manual.

When CLEAN deletes the file, it frees the non-bad blocks in the file.

The above procedure must be modified slightly if the bad block is in a file directory. In this case, use the BACKUP or SAVRES programs to copy all accessible files in the account (or disk) to an archive medium. When you add the bad block and clean the disk, the CLEAN suboption will allow you to delete the account whose directory contains the bad block. After starting time sharing, you should use SAVRES to restore the entire disk or use REACT to enter the account on the disk and BACKUP to restore the files from the archive medium.

RSTS/E V7.Ø
Initialization
INIT.SYS Program Notes

DEVICES WITH NON-STANDARD UNIBUS ADDRESSES AND VECTORS

Some installations have devices which are installed at non-standard UNIBUS addresses or which vector to non-standard locations. INIT does not automatically recognize these devices. Of course, the best solution is to have field service re-configure these devices so that they conform to the standard. Appendix F of the RSTS/E V7.Ø System Generation Manual contains the information you need to determine the correct UNIBUS addresses and vectors for all devices supported by RSTS/E. Note that extra units such as the second RX, second PR, etc., do not have "homes" and must use the CSR option.

If re-configuration is not possible, you can use the HARDWR option of INIT. The CSR suboption sets non-standard UNIBUS addresses and the VECTOR suboption sets non-standard vectors. DIGITAL strongly recommends that you use only the CSR suboption. This tells INIT to find the device at its non-standard UNIBUS address, but lets INIT determine the device's vector. If the device is functioning properly, INIT will find the device's non-standard vector. The only device for which INIT cannot determine a vector is the card reader. If a card reader has a non-standard vector, you must use the VECTOR suboption. The PA611 paper tape reader must be powered on for INIT to find its vector. Use the VECTOR suboption if you intend to start your system with a PA611 paper tape reader powered off.

Be careful when you assign non-standard vectors to devices. INIT checks all device vectors (assigned through the VECTOR suboption or automatically determined) against a table of reserved locations. INIT also checks to see if any vector location is used by more than one device. Any conflict will print the message

Vector for Device XXn: (vvv) already in use - device disabled.

where vvv is the octal address of the erring vector.

The RSTS/E reserved locations, which may not be used as device vectors, are:

Addresses	RSTS/E usage
Ø-2	Detection of jumps to Ø and traps to Ø
4-36	System trap vectors
4Ø-56	Reload start addresses, failure HALT
1ØØ-1Ø2	KW11-L line frequency clock vector
1Ø4-1Ø6	KW11-P crystal clock vector
11Ø-112	Jump to Ø handling
114-116	Memory parity trap vector
144-146	Crash dump handling
234-236	Statistics handling
24Ø-242	PIRQ trap vector
244-246	FPP or FIS exception trap vector
25Ø-252	Memory Management Unit trap vector

POWER-FAIL HARDWARE OPTION REQUIRED BY RSTS/E

When power fails, a PDP-11 processor traps through the vector at location 24. When power is restored, the system may be configured to halt, boot a specific device, or trap again through the vector at location 24. For RSTS/E to restart automatically after a power failure, the hardware must be configured to trap to 24 when power is restored. If the hardware is configured to boot the system disk, INIT.SYS will come to its 'OPTION:' prompt and wait.

RSTS/E V7.Ø
Initialization
INIT.SYS Program Notes

BOOTING 16ØØ BPI MAGNETIC TAPES

In RSTS/E V7.Ø, it is possible to create bootable 16ØØ BPI magnetic tapes. There are three types of these tapes: 24ØØ ft. RSTS/E distribution tapes, tapes created by SAVE/RESTORE with the /DEN:16ØØ switch, and tapes created by HOOK. You will be able to boot these tapes only on TS11, TU45, TU16, TU77, or TE16 tape drives. However, you must be careful when booting 16ØØ BPI tapes on TU45, TU16, TU77, or TE16 drives to make sure that they have a TMØ3 formatter. (You can determine this by using the HARDWR LIST suboption of INIT.) If, instead, you have a TMØ2 formatter, you will NOT be able to use the console bootstrap to boot the tape. Instead, you must use the BOOT option of INIT.SYS V7.Ø to boot the tape. Note that the BOOT option will print about 16 warning messages during the boot procedure if the TMØ2 formatter is present. These messages may be ignored; type CTRL/O to suppress them.

RSTS/E V7.Ø
Initialization
INIT.SYS Program Notes

HARDWARE TU58 SUBOPTION

The TU58 suboption of the HARDWR option of INIT.SYS has been removed. Therefore, section 3.6.8 of the RSTS/E V7.Ø System Generation Manual which describes this suboption is no longer applicable.

The purpose of the TU58 option was to allow you to declare a DL11 controller to have a TU58 connected to it, even if the startup code did not find one. Changes in INIT.SYS have allowed the TU58 to be treated as any other controller. Thus, if a TU58 is turned off when the system starts up, the DL11 line will be listed as a DL11 by HARDWR. To specify that the DL11 controller actually has a TU58 connected to it, you should use the HARDWR CSR suboption and specify DDn: as having the CSR of the DL11 controller it is connected to.

The process of starting up (the time between booting and printing the option prompt) may take more time in V7.Ø than in VØ6C, especially if the system has many DL11 style controllers. This is because the start up code must try sending a break sequence to each DL11-style controller to determine if a TU58 is connected to it.

RSTS/E V7.Ø
Executive
Monitor Patches

MAKING RECEIVER DECLARATION NON-PRIVILEGED - MONITOR FEATURE PATCH

The RSTS/E monitor normally requires that a job have permanent or temporary privileges in order to declare itself to be a message receiver. There are some production environments in which it is useful for non-privileged jobs to have this capability. The following procedure allows you to patch the RSTS/E monitor to remove the privileged status of the message receiver declaration SYS call.

PROCEDURE:

1. This is a feature patch to the RSTS/E V7.Ø executive. It may be installed in any target monitor SIL.

2. The patch described in Step 4 below can be installed using the PATCH option of INIT.SYS:

```
Option: PATCH
File to patch? <lf>                    (LINE FEED for installed monitor SIL)
```

3. This patch can be installed manually using ONLPAT, the on-line patching program. Note, however, that it can not be applied online to the installed monitor SIL.

```
RUN $ONLPAT
Command File Name? <cr>                (RETURN for manual patch installation)
File to patch? ??????.SIL              (SIL name)
File found in account [Ø,1]
```

   The patch is also contained in a patch file appearing in patch kit version "A" or later.

4. If you made the send/receive code resident when you generated the monitor SIL, answer the first 'Module name?' question with "FIP". Otherwise, answer with "OVR" as shown. The patch is as follows:

```
Module name? OVR          (or FIP)
Base address? SND1ST
Offset address? ..RCVP-SND1ST
 Base   Offset  Old       New?
??????  ØØØØ66  ØØ1445  ? 24Ø
??????  ØØØØ7Ø  Ø1Ø446  ? ^Z            (CTRL/Z for new offset)
Offset address? ^Z                     (CTRL/Z for new base)
Base address? ^Z                       (CTRL/Z for new module)
Module name? RSTS
Base address? $$Ø3Ø1
Offset address? Ø
 Base   Offset  Old       New?
??????  ØØØØØØ  ??????  ? Q!1
??????  ØØØØØ2  ??????  ? ^C            (up-arrow/C to exit;CTRL/C for INIT)
```

RSTS/E V7.Ø
Executive
Monitor Patches

Seq 3.1.2 F
Page 1 of 1

FIRST FIT MEMORY ALLOCATION - MONITOR FEATURE PATCH

When a residency request is made for a job or for a run-time system residency without a specific load address, the RSTS/E monitor does a best fit allocation: the job or run-time system is loaded into the smallest free area in memory in which it fits. For some systems, especially those with a large user memory region, a first fit allocation may result in better performance and less memory fragmentation.

This monitor feature patch changes the memory allocation from best fit to first fit.

PROCEDURE:

1. This is a feature patch to the RSTS/E V7.Ø executive. It may be installed in any target monitor SIL.

2. The patch described in Step 4 below can be installed using the PATCH option of INIT.SYS:

```
Option: PATCH
File to patch? <lf>                    (LINE FEED for installed monitor SIL)
```

3. This patch can be installed manually using ONLPAT, the on-line patching program:

```
RUN $ONLPAT
Command File Name? <cr>                (RETURN for manual patch installation)
File to patch? <lf>                    (LINE FEED for installed monitor SIL)
File found in account [Ø,1]
```

   The patch is also contained in a patch file appearing in patch kit version "A" or later.

4. The patch is as follows:

```
Module name? EMT
Base address? ..FFIT
Offset address? Ø
 Base   Offset  Old       New?
??????  ØØØØØØ  ØØ1627  ? 627
??????  ØØØØØ2  Ø2Ø2Ø3  ? ^Z            (CTRL/Z for new offset)
Offset address? ^Z                     (CTRL/Z for new base)
Base address? ^Z                       (CTRL/Z for new module)
Module name? RSTS
Base address? $$Ø3Ø1
Offset address? Ø
 Base   Offset  Old       New?
??????  ØØØØØØ  ??????  ? Q!2
??????  ØØØØØ2  ??????  ? ^C          (up-arrow/C to exit;CTRL/C for INIT)
```

RSTS/E V7.Ø
Executive
Monitor Patches

CACHE REPLACEMENT TIME - MONITOR FEATURE PATCH

When data is installed in the disk cache, it will not be removed until a certain minimum residency time has expired (unless the last block of a cluster is read in sequential mode). The minimum residency time is meant to keep some useful data in the cache even when the cache is very small, and prevent thrashing. For systems with large caches or rapidly changing disk access patterns, it may be advantageous to reduce the minimum residency time from its default value of 3 minutes. The residency time may be changed to Ø to 65535 inclusive. Refer to the RSTS/E V7.Ø System Manager's Guide for a discussion of caching tradeoffs.

PROCEDURE:

1. This is a feature patch to the RSTS/E V7.Ø executive. It may be installed in any target monitor SIL.

2. Determine the new minimum cache residency time in seconds. Use this value as n in Step 5 below. Be sure to include a period after the number so that it is interpreted as a decimal number.

3. The patch described in Step 5 below can be installed using the PATCH option of INIT.SYS:

```
Option: PATCH
File to patch? <lf>                    (LINE FEED for installed monitor SIL)
```

4. This patch can be installed manually using ONLPAT, the on-line patching program:

```
RUN $ONLPAT
Command File Name? <cr>                (RETURN for manual patch installation)
File to patch? <lf>                    (LINE FEED for installed monitor SIL)
File found in account [Ø,1]
```

The patch is also contained in a patch file appearing in patch kit version "A" or later.

NOTE

The patch file for this patch requires manual editing to include installation specific parameters before it can be successfully installed.

5. The patch is as follows:

```
Module name? RSTS
Base address? ..CAGE
Offset address? Ø
 Base   Offset  Old        New?
??????  ØØØØØØ  ØØØ264  ? n.          (from step 2)
??????  ØØØØØ2  ??????  ? ^Z          (CTRL/Z for new offset)
Offset address? ^Z                    (CTRL/Z for new base)
Base address? $$Ø3Ø1
Offset address? Ø
 Base   Offset  Old        New?
??????  ØØØØØØ  ??????  ? Q!4
??????  ØØØØØ2  ??????  ? ^C          (up-arrow/C to exit;CTRL/C for INIT)
```

RSTS/E V7.Ø
Executive
Terminal Service Patches

Seq 3.3.1 F

RESTRICTING TERMINAL MODES - TERMINAL SERVICE FEATURE PATCH

The RSTS/E monitor normally allows any user to OPEN a terminal in binary mode or echo control mode (if configured). There is a danger in the use of these terminal modes, since all characters (including CTRL/C) are passed to the user program. A novice user could easily find himself in a situation in which his terminal does not respond. A malicious user could "lock up" a terminal in a way which required the system manager to kill his job. The following procedure allows you to patch the RSTS/E monitor to specify which terminal OPEN modes require privileges. Privileged users and non-privileged users running privileged programs would still have full access to all available terminal modes. Attempts by non-privileged users to use restricted modes would be rejected with no error message.

PROCEDURE:

1. This is a feature patch to the RSTS/E V7.Ø executive. It may be installed in any target monitor SIL.

2. Decide which terminal modes you wish to make privileged. Consult the RSTS/E Programming Manual for the various modes which are available, and determine the associated MODE values. Compute the sum, n, of the MODE values corresponding to the modes you wish to restrict. If you calculate the sum of the mode values in decimal, include a decimal point after the value of n in the patch below so that it is interpreted as a decimal number.

3. The patch described in Step 5 below can be installed using the PATCH option of INIT.SYS:

```
Option: PATCH
File to patch? <lf>                    (LINE FEED for installed monitor SIL)
```

4. This patch can be installed manually using ONLPAT, the on-line patching program:

```
RUN $ONLPAT
Command File Name? <cr>                (RETURN for manual patch installation)
File to patch? <lf>                    (LINE FEED for installed monitor SIL)
File found in account [Ø,1]
```

The patch is also contained in a patch file appearing in patch kit version "A" or later.

NOTE

The patch file for this patch requires manual editing to include installation specific parameters before it can be successfully installed.

5. The patch is as follows:

```
Module name? TER
Base address? ..KBMP
Offset address? -2
 Base   Offset  Old       New?
??????  177776  Ø42702  ? <lf>          (no change; verify only)
??????  ØØØØØØ  ØØØØØØ  ? n             (from step 2)
??????  ØØØØØ2  ??????  ? ^Z            (CTRL/Z for new offset)
Offset address? ^Z                      (CTRL/Z for new base)
Base address? ^Z                        (CTRL/Z for new module)
Module name? RSTS
Base address? $$Ø3Ø3
Offset address? Ø
 Base   Offset  Old       New?
??????  ØØØØØØ  ??????  ? Q!1
??????  ØØØØØ2  ??????  ? ^C            (up-arrow/C to exit;CTRL/C for INIT)
```

RSTS/E V7.Ø
Executive
Terminal Service Patches

IMMEDIATE MODEM HANGUP - TERMINAL SERVICE FEATURE PATCH

When the carrier frequency drops on a modem line connecting a terminal to RSTS/E, the monitor normally waits approximately 5 seconds before hanging up the line. This allows recovery from transient line failures or accidental unseating of a phone from an acoustic coupler. In certain areas, such as the United Kingdom, computer systems are required to disconnect from such a line within milliseconds. The following patch causes RSTS/E to hang up a modem line as soon as it detects loss of carrier signal on that line. Refer also to a LOGIN patch, Seq 1Ø.12.2 F, which causes LOGIN to hang up after an unsuccessful log-in attempt.

PROCEDURE:

1. This is a feature patch to the RSTS/E V7.Ø executive. It may be installed in any target monitor SIL that supports modem control.

2. The patch described in Step 4 below can be installed using the PATCH option of INIT.SYS:

```
Option: PATCH
File to patch? <lf>                    (LINE FEED for installed monitor SIL)
```

3. This patch can be installed manually using ONLPAT, the on-line patching program:

```
RUN $ONLPAT
Command File Name? <cr>                (RETURN for manual patch installation)
File to patch? <lf>                    (LINE FEED for installed monitor SIL)
File found in account [Ø,1]
```

   The patch is also contained in a patch file appearing in patch kit version "A" or later.

4. The patch is as follows:

```
Module name? TER
Base address? ..UKMD
Offset address? Ø
 Base   Offset  Old       New?
??????  ØØØØØØ  ØØØ4Ø4   ? 24Ø
??????  ØØØØØ2  Ø127Ø4   ? ^Z              (CTRL/Z for new offset)
Offset address? ^Z                        (CTRL/Z for new base)
Base address? ^Z                          (CTRL/Z for new module)
Module name? RSTS
Base address? $$Ø3Ø3
Offset address? Ø
 Base   Offset  Old       New?
??????  ØØØØØØ  ??????   ? Q!2
??????  ØØØØØ2  ??????   ? ^C              (up-arrow/C to exit;CTRL/C for INIT)
```

RSTS/E V7.Ø
Executive
Terminal Service Patches

CHANGING SPECIAL LOGIN TERMINAL - TERMINAL SERVICE FEATURE PATCH

The number of jobs that can log in to a RSTS/E system is limited by the swapping space available, the JOB MAX set at system start-up time, and the login setting (set by the LOGINS, NO LOGINS, and SET LOGINS functions of UTILTY). However, the console terminal (KBØ:) is a special terminal, and can log in regardless of the login setting, provided that swapping space and JOB MAX permit.

With the patch given below, one or two special keyboards can be selected, or this feature can be disabled entirely.

PROCEDURE:

1. This is a feature patch to the RSTS/E V7.Ø executive. It may be installed in any target monitor SIL.

2. Decide which terminals should be allowed to log in regardless of the login setting, and use those numbers as m and n in the patch below. If you wish to allow only one terminal, use its number as m and use -1 in place of n. If you wish to disable this feature, use -1 in place of both m and n. Be sure you include the decimal point after the new value of m or n so that it is interpreted as a decimal number.

3. The patch described in Step 5 below can be installed using the PATCH option of INIT.SYS:

```
Option: PATCH
File to patch? <lf>                    (LINE FEED for installed monitor SIL)
```

4. This patch can be installed manually using ONLPAT, the on-line patching program:

```
RUN $ONLPAT
Command File Name? <cr>                (RETURN for manual patch installation)
File to patch? <lf>                    (LINE FEED for installed monitor SIL)
File found in account [Ø,1]
```

The patch is also contained in a patch file appearing in patch kit version "A" or later.

NOTE

The patch file for this patch requires manual editing to include installation specific parameters before it can be successfully installed.

5. The patch is as follows:

```
Module name? TER
Base address? ..CTZ.
Offset address? Ø
 Base   Offset  Old       New?
??????  ØØØØØØ  ØØØØØØ  ? m.*2 (or -1)  (from step 2)
??????  ØØØØØ2  ØØ14Ø7  ? ^Z            (CTRL/Z for new offset)
Offset address? ^Z                     (CTRL/Z for new base)
Base address? ..CTY.
Offset address? Ø
 Base   Offset  Old       New?
??????  ØØØØØØ  177777  ? n.*2 (or -1)  (from step 2)
??????  ØØØØØ2  ØØ14Ø4  ? ^Z            (CTRL/Z for new offset)
Offset address? ^Z                     (CTRL/Z for new base)
Base address? ^Z                       (CTRL/Z for new module)
Module name? RSTS
Base address? $$Ø3Ø3
Offset address? Ø
 Base   Offset  Old       New?
??????  ØØØØØØ  ??????  ? Q!4
??????  ØØØØØ2  ??????  ? ^C          (up-arrow/C to exit;CTRL/C for INIT)
```

RSTS/E V7.Ø
Executive
Terminal Service Patches

CHANGING A TERMINAL'S INPUT BUFFER QUOTA - TERMINAL SERVICE FEATURE PATCH

Inside every terminal's DDB (Device Data Block) is an "input buffer quota" value. The default value is 6. Since there are 3Ø characters in a buffer, this is equal to 6 times 3Ø or 18Ø characters. The terminal service will attempt to buffer that many characters before telling the terminal to stop (by sending it an XOFF). You can increase the input buffer quota value for one or more high speed input terminals by using this feature patch.

NOTE

There is no guarantee that a terminal will be able to get its full buffer quota. A heavy system load may leave less than the terminal's full buffer quota available. In addition, while the installation of this patch will not have any adverse effect on a system, it does cause more small buffers to be used.

PROCEDURE:

1. This is a feature patch to the RSTS/E V7.Ø executive. It may be installed in any target monitor SIL.

2. Decide which terminal will have its buffer quota altered, and use that number as n in the patch below. Be sure you include the decimal point after the value of n so that it is interpreted as a decimal number. Also decide what the new buffer quota for that terminal should be, and use that number as m in the patch below. The legal values for m are 6. through about 2Ø. inclusive.

3. The patch described in Step 5 below can be installed using the PATCH option of INIT.SYS:

```
Option: PATCH
File to patch? <lf>                    (LINE FEED for installed monitor SIL)
```

4. This patch can be installed manually using ONLPAT, the on-line patching program:

```
RUN $ONLPAT
Command File Name? <cr>                (RETURN for manual patch installation)
File to patch? <lf>                    (LINE FEED for installed monitor SIL)
File found in account [Ø,1]
```

The patch is also contained in a patch file appearing in patch kit version "A" or later.

NOTE

The patch file for this patch requires manual editing to include installation specific parameters before it can be successfully installed.

5. The patch is as follows:

```
Module name? RSTS
Base address? KBDDDB
Offset address? DDS.KB*n.+26          (n is the KB number)
 Base   Offset  Old       New?
??????  ??????  ØØØØØ6  ? m.          (from step 2)
??????  ??????  ØØØØØØ  ? ^Z          (CTRL/Z for new offset)
Offset address? ^Z                    (CTRL/Z for new base)
Base address? $$Ø3Ø3
Offset address? Ø
 Base   Offset  Old       New?
??????  ØØØØØØ  ??????  ? Q!1Ø
??????  ØØØØØ2  ??????  ? ^C          (up-arrow/C to exit;CTRL/C for INIT)
```

INPUT ESCAPE SEQUENCE HANDLING - TERMINAL SERVICE FEATURE PATCH

RSTS/E allows multiple character incoming escape sequences. The RSTS/E Programming Manual fully describes the escape sequence handling.

One of the multi-character escape sequences is ESCape followed by "P" followed by one modifier character. On the VT52 terminal, the upper left hand key on the auxiliary keypad (usually blue) generates ESC P. When the VT1ØØ terminal is operating in VT52 mode, the upper lefthand key on the auxiliary keypad (usually labelled "PF1") generates ESC P. If the terminal user types this key and then any other graphic key, the whole sequence is gathered by the terminal service and delivered to the application program. None of the sequence is echoed. Thus, the blue key can be considered a "function" key to be followed by some function code character.

On the other hand, some existing applications using the VT52 may expect this key to generate a standalone escape sequence with no required (or desired) modifier character. You may install the following patch to change the terminal service so that the ESCape, "P" sequence does not expect a modifier character. Note that this patch affects all terminals on the system.

PROCEDURE:

1. This is a feature patch to the RSTS/E V7.Ø executive. It may be installed in any target monitor SIL.

2. The patch described in Step 4 below can be installed using the PATCH option of INIT.SYS:

```
Option: PATCH
File to patch? <lf>                    (LINE FEED for installed monitor SIL)
```

3. This patch can be installed manually using ONLPAT, the on-line patching program:

```
RUN $ONLPAT
Command File Name? <cr>                (RETURN for manual patch installation)
File to patch? <lf>                    (LINE FEED for installed monitor SIL)
File found in account [Ø,1]
```

   The patch is also contained in a patch file appearing in patch kit version "A" or later.

4. The patch is as follows:

```
Module name? TER
Base address? ..ESCP
Offset address? Ø
 Base   Offset  Old        New?
??????  ØØØØØØ  Ø5ØØØ1  ? Q&177400
??????  ØØØØØ2  ??????  ? ^Z          (CTRL/Z for new offset)
Offset address? ^Z                    (CTRL/Z for new base)
Base address? ^Z                      (CTRL/Z for new module)
Module name? RSTS
Base address? $$Ø3Ø3
Offset address? Ø
 Base   Offset  Old        New?
??????  ØØØØØØ  ??????  ? Q!2Ø
??????  ØØØØØ2  ??????  ? ^C          (up-arrow/C to exit;CTRL/C for INIT)
```

RSTS/E V7.Ø
Executive
Terminal Service Patches

Seq 3.3.6 F
Page 1 of 2

RESTRICTING PSEUDO KEYBOARD MODES - TERMINAL SERVICE FEATURE PATCH

The RSTS/E monitor normally allows any user to OPEN a pseudo keyboard with mode 1%, which causes the controlled job to detach rather than be killed if the controlling job closes the pseudo keyboard prematurely. There is a danger in the use of this mode, since a non-privileged user could fill the system with detached jobs. The following procedure allows you to patch the RSTS/E monitor to specify which pseudo keyboard OPEN modes require privileges. Privileged users and non-privileged users running privileged programs would still have full access to all available pseudo keyboard modes. If a non-privileged user specifies a restricted mode, the mode value will be ignored but no error message will be printed.

PROCEDURE:

1. This is a feature patch to the RSTS/E V7.Ø executive. It may be installed in any target monitor SIL.

2. Decide which pseudo keyboard modes you wish to make privileged. Consult the RSTS/E Programming Manual for the various modes which are available, and determine the associated MODE values. Compute the sum, n, of the MODE values corresponding to the modes you wish to restrict. If you calculate the sum of the mode values in decimal, include a decimal point after the value of n in the patch below so that it is interpreted as a decimal number.

3. The patch described in Step 5 below can be installed using the PATCH option of INIT.SYS:

```
Option: PATCH
File to patch? <lf>                    (LINE FEED for installed monitor SIL)
```

4. This patch can be installed manually using ONLPAT, the on-line patching program:

```
RUN $ONLPAT
Command File Name? <cr>                (RETURN for manual patch installation)
File to patch? <lf>                    (LINE FEED for installed monitor SIL)
File found in account [Ø,1]
```

The patch is also contained in a patch file appearing in patch kit version "A" or later.

NOTE

The patch file for this patch requires manual editing to include installation specific parameters before it can be successfully installed.

5. The patch is as follows:

```
Module name? TER
Base address? ..PKMP
Offset address? -2
 Base   Offset  Old        New?
??????  177776  142761  ? <lf>      (no change; verify only)
??????  ØØØØØØ  ØØØØØØ  ? n         (from step 2)
??????  ØØØØØ2  ??????  ? ^Z        (CTRL/Z for new offset)
Offset address? ^Z                  (CTRL/Z for new base)
Base address? ^Z                    (CTRL/Z for new module)
Module name? RSTS
Base address? $$Ø3Ø3
Offset address? Ø
 Base   Offset  Old        New?
??????  ØØØØØØ  ??????  ? Q!4Ø
??????  ØØØØØ2  ??????  ? ^C      (up-arrow/C to exit;CTRL/C for INIT)
```

RSTS/E V7.Ø
Executive
Terminal Service Patches

DISABLE BREAK = CTRL/C - TERMINAL SERVICE FEATURE PATCH

Currently, the BREAK key generates a CTRL/C. This could cause compatibility problems for some users. This feature patch causes the BREAK key to generate a "null" character for every terminal on the system, as it did in VØ6C and prior versions of RSTS/E.

PROCEDURE:

1. This is a feature patch to the RSTS/E V7.Ø executive. It may be installed in any target monitor SIL.

2. The patch described in Step 4 below can be installed using the PATCH option of INIT.SYS:

```
Option: PATCH
File to patch? <lf>                    (LINE FEED for installed monitor SIL)
```

3. This patch can be installed manually using ONLPAT, the on-line patching program:

```
RUN $ONLPAT
Command File Name? <cr>                (RETURN for manual patch installation)
File to patch? <lf>                    (LINE FEED for installed monitor SIL)
File found in account [Ø,1]
```

   The patch is also contained in a patch file appearing in patch kit version "A" or later.

4. The patch is as follows:

```
Module name? TER
Base address? TTINØ9
Offset address? -4Ø
 Base   Offset  Old     New?
??????  177740  ØØ1ØØ2  ? ØØØ4Ø2
??????  177742  Ø12702  ? ^Z            (CTRL/Z for new offset)
Offset address? ^Z                     (CTRL/Z for new base)
Base address? ^Z                       (CTRL/Z for new module)
Module name? RSTS
Base address? $$Ø3Ø3
Offset address? Ø
 Base   Offset  Old     New?
??????  ØØØØØØ  ??????  ? Q!1ØØ
??????  ØØØØØ2  ??????  ? ^C            (up-arrow/C to exit;CTRL/C for INIT)
```

RSTS/E V7.Ø
Executive
Terminal Service Notes

2741 HARDWARE OPTIONS REQUIRED BY RSTS/E

RSTS/E supports 2741-compatible terminals connected through DL11D, DL11E, and DC11 single-line interfaces and through DH11 and DZ11 multiplexors. Certain 2741 hardware options are required for proper operation under RSTS/E.

The ability to detect "reverse break" is an option on 2741 terminals supplied by some manufacturers, including IBM. A reverse break is a control sequence sent by the computer to a terminal which is currently transmitting (keyboard is unlocked). The purpose of the reverse break is to force the terminal into receive state (keyboard locked). RSTS/E sends a reverse break any time the system has output for the 2741 terminal but internal status tables indicate that the terminal is in transmit state. If the terminal does not recognize the reverse break sequence, lock the keyboard, and switch to receive state, the output will be lost and the terminal can end up in a strange state.

IBM refers to the reverse break detection capability on the IBM 2741 Model 1 Communications Terminal as Feature #4708 - Receive Interrupt. This option must be installed on all IBM 2741 terminals for correct operation under RSTS/E.

The ATTENTION key is also optional on IBM 2741 terminals and is required for operation under RSTS/E. The ATTN key generates a break which is interpreted in several ways by RSTS/E software. The RSTS/E System User's Guide describes the various functions of the ATTN key. IBM refers to the ATTN key and the associated break generation hardware as Feature #7900 - Transmit Interrupt. This option is commonly included on 2741 terminals supplied by IBM and is a standard feature on many 2741-compatible terminals supplied by other manufacturers.

2741 TERMINAL INTERFACE WIRING

The DL11D and DL11E terminal line interfaces require a special configuration for supporting the 2741. The configuration requires 6 data bits, 1 start bit, 1 stop bit, odd parity, 134.5 baud, and rotary switch position 3 for both input and output (full counter-clockwise is position 1). The following jumpers apply to each condition:

Condition	Jumper	
6 Data bits	NB1	out
	NB2	in
1 Stop bit	2SB	in
	J9	out
	J1Ø	in
	J11	out
Odd parity	NP	in
	EPS	in

The 134.5 baud specification requires a 1.Ø3296M crystal (DEC part #18-Ø55Ø2-6). Refer to the DL11 Installation Procedure for specifications pertinent to the DL11D and DL11E.

UNDERSTANDING SYNCHRONIZATION DELAYS

DIGITAL's video terminals (VT5Ø, VT52, VT1ØØ, etc.) use two special control characters (commonly called XOFF and XON) to request the host computer to suspend and resume data transmission. One use of this synchronization protocol allows the terminal to "freeze" a screen of data giving the user a chance to read it before it has scrolled off the screen. On VT5Ø-series terminals, this special mode, called HOLD SCREEN MODE, is enabled and disabled with escape sequences. Once enabled, the terminal detects when a line is about to be scrolled from the screen. The character received from the host computer that would cause the scroll is line feed (LF). At this point, the terminal sends an XOFF to the host and waits for the user to type the SCROLL key. During this time, the terminal will accept and buffer in a temporary storage area (called a silo) further characters received from the host. Even if the host computer can cease transmission with no software delays, there are delays in line transmission that cause a determinable number of characters to be received by the terminal after it has sent the XOFF.

HOLD SCREEN MODE is just one use of this synchronization protocol. Another use is in the handling of the hard copy option of the VT5Ø series terminals. The copier is slow (at least compared to the video screen) and the terminal must ask the host computer to suspend transmission so that it doesn't miss data when the copier is running.

The LA34, LA38, and LA12Ø terminals use this synchronization protocol, and when the LA18Ø DECprinter I is connected to the computer through a serial line interface (referred to as the LA18ØS), it also uses this protocol. If the printer "gets behind" due to high line transmission speed or carriage restore time, it requests the host to suspend data transmission. The printer will also send an XOFF if its ONLINE/OFFLINE switch is placed into the OFFLINE position or if it runs out of paper.

The RSTS/E terminal service processing of the synchronization protocol is enabled and disabled by the TTYSET commands STALL and NO STALL.

Below is the formula used for calculating the largest number of characters that would be received after the terminal has sent the XOFF:

$$1 + (3*Y/X) + 2*D*Y$$

> Note: This formula assumes that there is no host computer software delay. In reality there is always some delay at the host computer. This delay might be very small or possibly large depending on system loading and other I/O interrupt activity.

In this formula, X is the terminal's transmission rate in characters per second (CPS), Y is the host computer's transmission rate in CPS, and D is the delay time in seconds for a character to travel from one end of the line to the other.

Considering the formula from left to right:

The 1 is the character storage requirement for the line feed that triggered the XOFF in the first place.

The 3*Y/X is due to the nature of the UART (Universal Asynchronous Receiver/Transmitter) in the terminal. Since the UART is normally double buffered, there can be up to two characters already in the UART's buffers when the terminal decides to send the XOFF. Those two characters plus the XOFF correspond to three character times of terminal to host transmit during which the host may be sending to the terminal. Therefore, it is the ratio of the host and terminal transmission rates that is important for three terminal transmission times.

The 2*D is the total time delay for the line in both directions (the 2) to clear. During this time the host computer may have sent up to 2*D*Y characters to the terminal.

The formula presented here is not completely correct for the VT5Ø series of video terminals. The VT5Ø does not double buffer transmission to the host computer in its UART. On the other hand, the formula can be used as a worst case formula to ensure correct operation of the XON/XOFF Synchronization Protocol.

The following tables give the required size of the temporary storage area (silo) given: 1) the transmission speed (baud rate) from the host computer to the terminal, 2) the transmission speed from the terminal to the host computer, and 3) the physical delay of the transmission line. Local terminal connections have essentially a zero delay factor. The Telephone Company specifies a nominal worst case delay of 5Øms in a coast-to-coast connection, assuming it is not via satellite.

The horizontal scale of baud rates is the baud rate from the host computer to the terminal.

The vertical scale of baud rates is the baud rate from the terminal to the host computer. Baud rates can be converted to characters per second (CPS) by dividing the baud rate by 1Ø (1 start bit + 8 data bits + 1 stop bit = 1Ø bits) with the exception of 11Ø baud which corresponds to 1Ø CPS (1 start bit + 8 data bits + 2 stop bits).

Øms Line Delay

Transmit Speed to Terminal (Y)

Receive Speed from Terminal (X)	11Ø	15Ø	3ØØ	6ØØ	12ØØ	24ØØ	48ØØ	96ØØ
11Ø	4	6	1Ø	19	37	73	145	289
15Ø	3	4	7	13	25	49	97	193
3ØØ	2	3	4	7	13	25	49	97
6ØØ	2	2	3	4	7	13	25	49
12ØØ	1	1	2	3	4	7	13	25
24ØØ	1	1	1	2	3	4	7	13
48ØØ	1	1	1	1	2	3	4	7
96ØØ	1	1	1	1	1	2	3	4

50ms Line Delay

Transmit Speed to Terminal (Y)

Receive Speed from Terminal (X)	11Ø	15Ø	3ØØ	6ØØ	12ØØ	24ØØ	48ØØ	96ØØ
11Ø	5	7	13	25	49	97	193	385
15Ø	4	6	1Ø	19	37	73	145	289
3ØØ	3	4	7	13	25	49	97	193
6ØØ	3	3	6	1Ø	19	37	73	145
12ØØ	2	3	5	9	16	31	61	121
24ØØ	2	3	4	8	15	28	55	1Ø9
48ØØ	2	3	4	7	14	27	52	1Ø3
96ØØ	2	3	4	7	13	26	51	1ØØ

By referring to the manual that came with your terminal you can determine which baud rate combinations are legal for your terminal.

RSTS/E V7.Ø
Executive
Terminal Service Notes

UNDERSTANDING '?I/O TO DETACHED KEYBOARD' ERRORS

The error '?I/O to detached keyboard' (ERR = 27) occurs only when:

1. one of the simple terminal SYS calls (function codes Ø, 1, 2, 3, 4, and 11) is attempted for the job's console terminal (i.e., the job's KB: device) and the job is detached, or

2. an open is attempted using the device name "KB:" and the job is detached, or

3. an I/O operation (INPUT, PRINT, GET, PUT, etc.) is attempted to a modem line terminal which is neither the job's console terminal nor the terminal from which the job detached, and that line is hung up. (Note: Opening a modem line which is currently hung up does not produce any error.)

If an I/O request is issued to a detached job's KB: terminal or any channel on which that terminal is open, the job is put into the hibernate (HB) state. This can happen using local terminals if the job detaches. It can happen on modem lines if the job detaches or the line is hung up and the terminal service automatically detaches the job.

RSTS/E V7.Ø
Executive
Terminal Service Notes

UNDERSTANDING TERMINAL PARITY AND STOP BIT SETTINGS

Terminal Parity

The TTYSET commands "NO PARITY", "EVEN PARITY", and "ODD PARITY" control the format in which data characters are sent to a terminal. NO PARITY (the default) sends the full 8 bits of a byte out to a terminal. EVEN PARITY and ODD PARITY trim the byte to 7 bits of data and then set or clear the eighth bit to achieve the correct parity setting. (Note: Binary Output Mode is "Data Binary Output Mode". NO PARITY yields 8 bit data, EVEN PARITY and ODD PARITY yield 7 bit data.)

The KL11, LC11, DL11A/B/C/D/E, DC11, and DJ11 interfaces do not have programmable hardware parity generation. For these interfaces, the terminal service software generates the output parity bit. The DH11 and DZ11 have programmable hardware parity generation, which the terminal service uses.

Stop Bit Settings

Terminal interfaces with programmable baud rate settings (DC11, DH11, and DZ11) also have programmable stop bit settings. RSTS/E will automatically select 2 stop bits for speed settings of 11Ø baud or less and 1 stop bit for speeds greater than 11Ø baud. Older mechanical terminals such as the ASR33 (which runs at 11Ø baud) require the time delay of 2 stop bits for synchronization. Modern terminals need only 1 stop bit.

When converting from baud rates to characters per second (CPS), the correct number of stop bits becomes clear. Asynchronous communication consists of 1 start bit, 8 data bits, and 1 or 2 stop bit(s).

(11Ø bits per second)/(1+8+2 bits per character)=11Ø/11=1Ø characters per second

(3ØØ bits per second)/(1+8+1 bits per character)=3ØØ/1Ø=3Ø characters per second

AUXILIARY LIBRARY ACCOUNTS - FILE PROCESSOR FEATURE PATCH

The standard system library for RSTS/E systems is account [1,2]. This account is normally referenced by the $ character in file specifications. The special characters ! % and & can also be used to reference three auxiliary libraries. The default assignments for these characters are [1,3], [1,4], and [1,5] respectively. Although the association of character with account is defined, the three accounts must be created with REACT before the special characters can be used successfully.

In some installations, the system manager may want to change the account numbers referenced by the three special characters. The account numbers are defined by four words in the RSTS/E monitor. The default account numbers are altered with a simple patch as described below.

PROCEDURE:

1. This is a feature patch to the RSTS/E V7.Ø executive. It may be installed in any target monitor SIL.

2. Decide which account numbers you wish to use for the auxiliary library characters. You may change any or all of the default assignments (although changing the assignment of the $ library character is strongly discouraged). To specify a new project-programmer number of [p,pn], type a new value at the appropriate offset. The new value has the form:

   p.*256.+pn.

   where p is the new project number (followed by a decimal point to ensure that it is interpreted as a decimal number), the multiplication shifts the project to the high byte (again, do not omit the decimal point after the 256), and pn is the new programmer number (followed by a decimal point). p and pn must be between Ø and 254, and p may not be zero.

3. The patch described in Step 5 below can be installed using the PATCH option of INIT.SYS:

   ```
   Option: PATCH
   File to patch? <lf>                (LINE FEED for installed monitor SIL)
   ```

4. This patch can be installed manually using ONLPAT, the on-line patching program:

   ```
   RUN $ONLPAT
   Command File Name? <cr>            (RETURN for manual patch installation)
   File to patch? <lf>                (LINE FEED for installed monitor SIL)
   File found in account [Ø,1]
   ```

   The patch is also contained in a patch file appearing in patch kit version "A" or later.

NOTE

The patch file for this patch requires manual editing to include installation specific parameters before it can be successfully installed.

5. In this example, the ! account is patched to [1ØØ,25Ø], the % account is left unchanged, and the & account is changed to [1,99]. Of course, you should type in appropriate values for your system. The patch is as follows:

```
Module name? RSTS
Base address? ..PPN.
Offset address? Ø
 Base   Offset  Old       New?
??????  ØØØØØØ  ØØØ4Ø2  ? <lf>                    (No change to $)
??????  ØØØØØ2  ØØØ4Ø3  ? 1ØØ.*256.+25Ø.         (New value for !)
??????  ØØØØØ4  ØØØ4Ø4  ? <lf>                    (No change to %)
??????  ØØØØØ6  ØØØ4Ø5  ? 1.*256.+99.            (New value for &)
??????  ØØØØ1Ø  ??????  ? ^Z              (CTRL/Z for new offset)
Offset address? ^Z                       (CTRL/Z for new base)
Base address? $$ØØ3Ø5
Offset address? Ø
 Base   Offset  Old       New?
??????  ØØØØØØ  ??????  ? Q!1
??????  ØØØØØ2  ??????  ? ^C              (up-arrow/C to exit;CTRL/C for INIT)
```

RSTS/E V7.Ø
Executive
File Processor Patches

Seq 3.5.2 F

CHANGING DEFAULT PROTECTION CODE - FILE PROCESSOR FEATURE PATCH

When a user creates a file without specifying a protection code for that file, and if the personal default protection code is not set, the RSTS/E monitor creates that file with the system default protection code. This is normally 6Ø., which allows the user read/write access to the file, but restricts all non-privileged users from any access. The legal value range for the default protection code is Ø. to 63. inclusive. The following procedure allows you to patch the RSTS/E monitor to change your system's default protection code.

PROCEDURE:

1. This is a feature patch to the RSTS/E V7.Ø executive. It may be installed in any target monitor SIL.

2. The patch described in Step 4 below can be installed using the PATCH option of INIT.SYS:

```
Option: PATCH
File to patch? <lf>                    (LINE FEED for installed monitor SIL)
```

3. This patch can be installed manually using ONLPAT, the on-line patching program. Note, however, that it can not be applied online to the installed monitor SIL.

```
RUN $ONLPAT
Command File Name? <cr>                (RETURN for manual patch installation)
File to patch? ??????.SIL              (LINE FEED for installed monitor SIL)
File found in account [Ø,1]
```

The patch is also contained in a patch file appearing in patch kit version "A" or later.

NOTE

The patch file for this patch requires manual editing to include installation specific parameters before it can be successfully installed.

4. If you chose "resident disk handling" when you generated the monitor SIL, answer the first 'Module name?' question with "FIP"; otherwise, answer with "OVR" as shown. Be sure you include the decimal point after the new value so that it is interpreted as a decimal number.

   If your system is configured with large file support, you should use the patch described in Step 5. Otherwise, the patch is as follows:

```
Module name? OVR        (or FIP)
Base address? OPN4ST
Offset address? ..DPRT-OPN4ST
 Base   Offset  Old       New?
??????  ØØØ411     Ø74  ? n.          (the new default)
??????  ØØØ412     364  ? ^Z          (CTRL/Z for new offset)
Offset address? ^Z                    (CTRL/Z for new base)
Base address? ^Z                      (CTRL/Z for new module)
Module name? RSTS
Base address? $$Ø3Ø5
Offset address? Ø
 Base   Offset  Old       New?
??????  ØØØØØØ  ??????  ? Q!2
??????  ØØØØØ2  ??????  ? ^C          (up-arrow/C to exit;CTRL/C for INIT)
```

5. The patch for a large file system is as follows:

```
Module name? OVR        (or FIP)
Base address? OPL4ST
Offset address? ..DPRT-OPL4ST
 Base   Offset  Old       New?
??????  ØØØ415     Ø74  ? n.          (the new default)
??????  ØØØ416     364  ? ^Z          (CTRL/Z for new offset)
Offset address? ^Z                    (CTRL/Z for new base)
Base address? ^Z                      (CTRL/Z for new module)
Module name? RSTS
Base address? $$Ø3Ø5
Offset address? Ø
 Base   Offset  Old       New?
??????  ØØØØØØ  ??????  ? Q!2
??????  ØØØØØ2  ??????  ? ^C          (up-arrow/C to exit;CTRL/C for INIT)
```

RSTS/E V7.Ø
Executive
File Processor Patches

Seq 3.5.3 F

RESTRICTING DISK MODES - FILE PROCESSOR FEATURE PATCH

The RSTS/E monitor normally allows any user to OPEN a disk file using any mode except 4Ø96. (decimal) (the read regardless mode) or 256. (data caching), and requires that a user have privileges to open a disk file with MODE 4Ø96. or 256.

You may wish to allow non-privileged users to use mode 4Ø96. or 256. or to restrict one or more of the other disk modes (see the RSTS/E Programming Manual for a complete list). The following procedure allows you to patch the RSTS/E monitor to specify which disk file OPEN modes require privileges. Privileged users and non-privileged users running privileged programs would still have full access to all available modes. If a non-privileged user specifies a restricted mode, the mode value will be ignored but no error message will be printed.

PROCEDURE:

1. This is a feature patch to the RSTS/E V7.Ø executive. It may be installed in any target monitor SIL.

2. Decide which disk file modes you wish to make privileged. Consult the RSTS/E Programming Manual for the various modes which are available, and determine the associated MODE values. Compute the sum, n, of the MODE values corresponding to the modes you wish to restrict. If you wish to restrict special update mode, but leave normal update mode unrestricted, use a value of 4 (not 5). If you wish to restrict both update modes, use 4+1. You may type the addition directly to the new value question of PATCH. Be sure to include a decimal point after each number so that it is interpreted as decimal.

3. The patch described in Step 5 below can be installed using the PATCH option of INIT.SYS:

```
Option: PATCH
File to patch? <lf>                    (LINE FEED for installed monitor SIL)
```

4. This patch can be installed manually using ONLPAT, the on-line patching program. Note, however, that it can not be applied online to the installed monitor SIL.

```
RUN $ONLPAT
Command File Name? <cr>                (RETURN for manual patch installation)
File to patch? ??????.SIL              (LINE FEED for installed monitor SIL)
File found in account [Ø,1]
```

The patch is also contained in a patch file appearing in patch kit version "A" or later.

NOTE

The patch file for this patch requires manual editing to include installation specific parameters before it can be successfully installed.

5. If you made the disk utility code resident when you generated the monitor SIL, answer the first 'Module name?' question with "FIP"; otherwise, answer with "OVR" as shown.

   If your system is configured with large file support, you should use the patch described in Step 6. Otherwise, the patch is as follows:

```
Module name? OVR          (or FIP)
Base address? OPNØST
Offset address? ..DKNP-OPNØST
 Base   Offset  Old        New?
??????  ØØØ314  Ø1Ø4ØØ   ? n.          (from step 2)
??????  ØØØ316  Ø1ØØ37   ? ^Z          (CTRL/Z for new offset)
Offset address? ^Z                    (CTRL/Z for new base)
Base address? ^Z                      (CTRL/Z for new module)
Module name? RSTS
Base address? $$Ø3Ø5
Offset address? Ø
 Base   Offset  Old        New?
??????  ØØØØØØ  ??????   ? Q!4
??????  ØØØØØ2  ??????   ? ^C          (up-arrow/C to exit;CTRL/C for INIT)
```

6. The patch for a large file system is as follows:

```
Module name? OVR          (or FIP)
Base address? OPLØST
Offset address? ..DKNP-OPLØST
 Base   Offset  Old        New?
??????  ØØØ226  Ø1Ø4ØØ   ? n.          (from step 2)
??????  ØØØ23Ø  Ø1ØØ37   ? ^Z          (CTRL/Z for new offset)
Offset address? ^Z                    (CTRL/Z for new base)
Base address? ^Z                      (CTRL/Z for new module)
Module name? RSTS
Base address? $$Ø3Ø5
Offset address? Ø
 Base   Offset  Old        New?
??????  ØØØØØØ  ??????   ? Q!4
??????  ØØØØØ2  ??????   ? ^C          (up-arrow/C to exit;CTRL/C for INIT)
```

RSTS/E V7.Ø
Executive
File Processor Patches

MAKING FIP SYS CALLS PRIVILEGED - FILE PROCESSOR FEATURE PATCH

The following FIP SYS calls are normally non-privileged:

-25. File Attribute Read/Write
-17. Change File's Run-Time System Name
+1Ø. Assign a Device
+13. Zero a Device or Account
+14. Read Accounting Data
+15. Indexed Directory Lookup
+17. Wildcard Directory Lookup

It is possible to make one or more of these SYS calls privileged. Doing so will restrict the use of that SYS call to only privileged users and/or privileged programs.

CAUTION

The standard CUSP programs assume that the above SYS calls are non-privileged; unexpected results may occur using the CUSPs if one or more of the above are made privileged.

PROCEDURE:

1. This is a feature patch to the RSTS/E V7.Ø executive. It may be installed in any target monitor SIL.

2. Decide which SYS call code you wish to make privileged. Be sure to include a decimal point after the SYS call code to ensure that it is interpreted in decimal.

3. The patch described in Step 5 below can be installed using the PATCH option of INIT.SYS:

```
Option: PATCH
File to patch? <lf>                (LINE FEED for installed monitor SIL)
```

4. This patch can be installed manually using ONLPAT, the on-line patching program:

```
RUN $ONLPAT
Command File Name? <cr>            (RETURN for manual patch installation)
File to patch? <lf>                (LINE FEED for installed monitor SIL)
File found in account [Ø,1]
```

The patch is also contained in a patch file appearing in patch kit version "A" or later.

NOTE

The patch file for this patch requires manual editing to include installation specific parameters before it can be successfully installed.

5. The patch is as follows:

```
Module name? RSTS
Base address? $UUOTB
Offset address? n.-UU$MIN*2                (from step 2)
 Base   Offset  Old        New?
??????  ??????  ??????  ? Q!1
??????  ??????  ??????  ? ^Z               (CTRL/Z for new offset)
Offset address? ^Z                         (CTRL/Z for new base)
Base address? $$Ø3Ø5
Offset address? Ø
 Base   Offset  Old        New?
??????  ØØØØØØ  ??????  ? Q!1Ø
??????  ØØØØØ2  ??????  ? ^C               (up-arrow/C to exit;CTRL/C for INIT)
```

RSTS/E V7.Ø
Executive
File Processor Patches

Seq 3.5.5 F

NON-PRIVILEGED ACCESS TO NON-FILE STRUCTURED DISKS - FILE PROCESSOR FEATURE PATCH

Normally, a non-privileged user is given read and write access to a non-file structured disk if it was that user's open request which mounted the disk as non-file structured. Certain installations may wish to prohibit non-privileged users from ever getting any access to a non-file structured disk.

PROCEDURE:

1. This is a feature patch to the RSTS/E V7.Ø executive. It may be installed in any target monitor SIL.

2. The patch described in Step 4 below can be installed using the PATCH option of INIT.SYS:

```
Option: PATCH
File to patch? <lf>                (LINE FEED for installed monitor SIL)
```

3. This patch can be installed manually using ONLPAT, the on-line patching program. Note, however, that it can not be applied online to the installed monitor SIL.

```
RUN $ONLPAT
Command File Name? <cr>            (RETURN for manual patch installation)
File to patch? ??????.SIL          (LINE FEED for installed monitor SIL)
File found in account [Ø,1]
```

   The patch is also contained in a patch file appearing in patch kit version "A" or later.

4. If you made the disk utility code resident when you generated the monitor SIL, answer the first 'Module name?' question with "FIP"; otherwise, answer with "OVR" as shown.

   If your system is configured with large file support, you should use the patch described in Step 5. Otherwise, the patch is as follows:

```
Module name? OVR        (or FIP)
Base address? OPN2ST
Offset address? ..NFSP-OPN2ST
 Base   Offset  Old       New?
??????  ØØØØ3Ø  ØØ1414  ? 1426
??????  ØØØØ32  1Ø1Ø23  ? ^Z          (CTRL/Z for new offset)
Offset address? ^Z                   (CTRL/Z for new base)
Base address? ^Z                     (CTRL/Z for new module)
Module name? RSTS
Base address? $$Ø3Ø5
Offset address? Ø
 Base   Offset  Old       New?
??????  ØØØØØØ  ??????  ? Q!2Ø
??????  ØØØØØ2  ??????  ? ^C          (up-arrow/C to exit;CTRL/C for INIT)
```

5. The patch for a large file system is as follows:

```
Module name? OVR        (or FIP)
Base address? OPL2ST
Offset address? ..NFSP-OPL2ST
 Base   Offset  Old       New?
??????  ØØØØ3Ø  ØØ1414  ? 1423
??????  ØØØØ32  1Ø1Ø2Ø  ? ^Z          (CTRL/Z for new offset)
Offset address? ^Z                   (CTRL/Z for new base)
Base address? ^Z                     (CTRL/Z for new module)
Module name? RSTS
Base address? $$Ø3Ø5
Offset address? Ø
 Base   Offset  Old       New?
??????  ØØØØØØ  ??????  ? Q!2Ø
??????  ØØØØØ2  ??????  ? ^C          (up-arrow/C to exit;CTRL/C for INIT)
```

NON-PRIVILEGED WRITES TO NON-FILE STRUCTURED DISKS - FILE PROCESSOR FEATURE PATCH

Normally, a non-privileged user is given write access to a non-file structured disk if it was that user's open request which mounted the disk as non-file structured. Certain installations may wish to prohibit non-privileged users from ever obtaining write access to a non-file structured disk.

PROCEDURE:

1. This is a feature patch to the RSTS/E V7.Ø executive. It may be installed in any target monitor SIL.

2. The patch described in Step 4 below can be installed using the PATCH option of INIT.SYS:

   ```
   Option: PATCH
   File to patch? <lf>                    (LINE FEED for installed monitor SIL)
   ```

3. This patch can be installed manually using ONLPAT, the on-line patching program. Note, however, that it can not be applied online to the installed monitor SIL.

   ```
   RUN $ONLPAT
   Command File Name? <cr>                (RETURN for manual patch installation)
   File to patch? ??????.SIL              (LINE FEED for installed monitor SIL)
   File found in account [Ø,1]
   ```

   The patch is also contained in a patch file appearing in patch kit version "A" or later.

4. If you made the disk utility code resident when you generated the monitor SIL, answer the first 'Module name?' question with "FIP"; otherwise, answer with "OVR" as shown.

   If your system is configured with large file support, you should use the patch described in Step 5. Otherwise, the patch is as follows:

```
Module name? OVR        (or FIP)
Base address? OPN2ST
Offset address? ..NFSW-OPN2ST
 Base   Offset  Old       New?
??????  ØØØ166  ØØ1ØØØ  ? 1ØØ1
??????  ØØØ17Ø  ØØ5Ø16  ? ^Z          (CTRL/Z for new offset)
Offset address? ^Z                   (CTRL/Z for new base)
Base address? ^Z                     (CTRL/Z for new module)
Module name? RSTS
Base address? $$Ø3Ø5
Offset address? Ø
 Base   Offset  Old       New?
??????  ØØØØØØ  ??????  ? Q!4Ø
??????  ØØØØØ2  ??????  ? ^C          (up-arrow/C to exit;CTRL/C for INIT)
```

5. The patch for a large file system is as follows:

```
Module name? OVR        (or FIP)
Base address? OPL2ST
Offset address? ..NFSW-OPL2ST
 Base   Offset  Old       New?
??????  ØØØ16Ø  ØØ1ØØØ  ? 1ØØ1
??????  ØØØ162  ØØ5Ø16  ? ^Z          (CTRL/Z for new offset)
Offset address? ^Z                   (CTRL/Z for new base)
Base address? ^Z                     (CTRL/Z for new module)
Module name? RSTS
Base address? $$Ø3Ø5
Offset address? Ø
 Base   Offset  Old       New?
??????  ØØØØØØ  ??????  ? Q!4Ø
??????  ØØØØØ2  ??????  ? ^C          (up-arrow/C to exit;CTRL/C for INIT)
```

RSTS/E V7.Ø
Executive
File Processor Patches

Seq 3.5.7 F

ALLOWING CROSS ACCOUNT CREATIONS - FILE PROCESSOR FEATURE PATCH

Normally, a non-privileged job cannot create files in any account other than its own. Some installations may wish to allow a non-privileged job to create files in any account within its project number (i.e., create files in any [x,*] account if the job's account is [x,y]).

This feature patch changes the File Processor's cross account check to allow any job to create files within its project. A non-privileged job still cannot create files in accounts outside of its project number group.

PROCEDURE:

1. This is a feature patch to the RSTS/E V7.Ø executive. It may be installed in any target monitor SIL.

2. The patch described in Step 4 below can be installed using the PATCH option of INIT.SYS:

   ```
   Option: PATCH
   File to patch? <lf>                    (LINE FEED for installed monitor SIL)
   ```

3. This patch can be installed manually using ONLPAT, the on-line patching program. Note, however, that it can not be applied online to the installed monitor SIL.

   ```
   RUN $ONLPAT
   Command File Name? <cr>                (RETURN for manual patch installation)
   File to patch? ??????.SIL              (LINE FEED for installed monitor SIL)
   File found in account [Ø,1]
   ```

   The patch is also contained in a patch file appearing in patch kit version "A" or later.

4. If you made the disk utility code resident when you generated the monitor SIL, answer the first 'Module name?' question with "FIP"; otherwise, answer with "OVR" as shown.

   If your system is configured with large file support, you should use the patch described in Step 5. Otherwise, the patch is as follows:

```
Module name? OVR        (or FIP)
Base address? OPN4ST
Offset address? ..XGRP-OPN4ST
 Base   Offset  Old       New?
??????  ØØØ366  ØØ1334  ? 24Ø
??????  ØØØ37Ø  124237  ? ^Z          (CTRL/Z for new offset)
Offset address? ^Z                    (CTRL/Z for new base)
Base address? ^Z                      (CTRL/Z for new module)
Module name? RSTS
Base address? $$Ø3Ø5
Offset address? Ø
 Base   Offset  Old       New?
??????  ØØØØØØ  ??????  ? Q!1ØØ
??????  ØØØØØ2  ??????  ? ^C          (up-arrow/C to exit;CTRL/C for INIT)
```

5. The patch for a large file system is as follows:

```
Module name? OVR        (or FIP)
Base address? OPL4ST
Offset address? ..XGRP-OPL4ST
 Base   Offset  Old       New?
??????  ØØØ374  ØØ1334  ? 24Ø
??????  ØØØ376  124237  ? ^Z          (CTRL/Z for new offset)
Offset address? ^Z                    (CTRL/Z for new base)
Base address? ^Z                      (CTRL/Z for new module)
Module name? RSTS
Base address? $$Ø3Ø5
Offset address? Ø
 Base   Offset  Old       New?
??????  ØØØØØØ  ??????  ? Q!1ØØ
??????  ØØØØØ2  ??????  ? ^C          (up-arrow/C to exit;CTRL/C for INIT)
```

RSTS/E V7.Ø
Executive
File Processor Patches

Seq 3.5.8 F

ALLOWING CROSS ACCOUNT RENAME OPERATIONS - FILE PROCESSOR FEATURE PATCH

Normally, a non-privileged job cannot rename files in any account other than its own. Some installations may wish to allow a non-privileged job to rename files in any account within its project number (i.e., rename files in any [x,*] account if the job's account is [x,y]).

This feature patch changes the File Processor's cross account check to allow any job to rename files within its project. A non-privileged job still cannot rename files in accounts outside of its project number group.

PROCEDURE:

1. This is a feature patch to the RSTS/E V7.Ø executive. It may be installed in any target monitor SIL.

2. The patch described in Step 4 below can be installed using the PATCH option of INIT.SYS:

```
Option: PATCH
File to patch? <lf>                    (LINE FEED for installed monitor SIL)
```

3. This patch can be installed manually using ONLPAT, the on-line patching program. Note, however, that it can not be applied online to the installed monitor SIL.

```
RUN $ONLPAT
Command File Name? <cr>                (RETURN for manual patch installation)
File to patch? ??????.SIL              (LINE FEED for installed monitor SIL)
File found in account [Ø,1]
```

   The patch is also contained in a patch file appearing in patch kit version "A" or later.

4. If you made the delete/rename code resident when you generated the monitor SIL, answer the 'Module name?' question with "FIP"; otherwise, answer with "OVR" as shown. The patch is as follows:

```
Module name? OVR          (or FIP)
Base address? DLNØST
Offset address? ..XGRR-DLNØST
 Base   Offset  Old       New?
??????  ØØØ762  ØØ136Ø  ? 24Ø
??????  ØØØ764  126Ø37  ? ^Z            (CTRL/Z for new offset)
Offset address? ^Z                     (CTRL/Z for new base)
Base address? ^Z                       (CTRL/Z for new module)
Module name? RSTS
Base address? $$Ø3Ø5
Offset address? Ø
 Base   Offset  Old       New?
??????  ØØØØØØ  ??????  ? Q!2ØØ
??????  ØØØØØ2  ??????  ? ^C          (up-arrow/C to exit;CTRL/C for INIT)
```

RSTS/E V7.Ø
Executive
File Processor Patches

RESTRICTING DIRECTORY LOOKUP - FILE PROCESSOR FEATURE PATCH

Normally, the FIP calls 15 (Directory look up on Index) and 17 (Directory look up by File Name/Wildcard directory look up) will allow any user to find information about any file on the system. Some installations, particularly those with 'hostile users', may find it desirable to restrict non-privileged users' from looking up files which they may not access. If this patch is applied, privileged users and programs with temporary privileges will still be able to look up any file, but non-privileged use of FIP calls 15 and 17 will return information about only those files that could be read by the user. Since PIP.SAV (for the /L command) and BASIC-PLUS (for the CATALOG command) use these calls, these commands would be restricted for non-privileged users. The DIRECT program does not use these calls; another feature patch, Seq 1Ø.3.1 F, may be usèd to restrict its use.

PROCEDURE:

1. This is a feature patch to the RSTS/E V7.Ø executive. It may be installed in any target monitor SILs.

2. The patch described in Step 4 below can be installed using the PATCH option of INIT.SYS:

```
Option: PATCH
File to patch? <lf>                    (LINE FEED for installed monitor SIL)
```

3. This patch can be installed manually using ONLPAT, the on-line patching program. Note, however, that it can not be applied online to the installed monitor SIL.

```
RUN $ONLPAT
Command File Name? <cr>                (RETURN for manual patch installation)
File to patch? <lf>                    (LINE FEED for installed monitor SIL)
File found in account [Ø,1]
```

   The patch is also contained in a patch file appearing in patch kit version "A" or later.

4. The patch is as follows:

```
Module name? OVR          (or FIP)
Base address? DIRØST
Offset address? ..DIRP-DIRØST
 Base   Offset  Old       New?
??????  ØØØ62Ø  ØØØ415  ? 1415
??????  ØØØ622  Ø21Ø37  ? ^Z            (CTRL/Z for new offset)
Offset address? ^Z                     (CTRL/Z for new base)
Base address? ^Z                       (CTRL/Z for new module)
Module name? RSTS
Base address? $$Ø3Ø5
Offset address? 1
 Base   Offset  Old       New?
??????  ØØØØØ1     ???  ? Q!1
??????  ØØØØØ2     ???  ? ^C          (up-arrow/C to exit;CTRL/C for INIT)
```

WRITE PROTECT AGAINST OWNER FILE PROTECTION CODE HANDLING

The "write protect" file protection code bits (owner, group, and world) mean both write protection and deletion protection. Write protect against owner (Bit 1, value 2) has a special meaning in two kinds of implicit file deletion:

1. When a file is created (e.g., OPEN FOR OUTPUT), any file of the same name is implicitly deleted unless that existent file's protection code says write (delete) protect. Normally, a privileged user has the right to override the delete protect status. But, in this implicit delete case, the write (delete) protect against owner bit is not overridden by privileged users. This ensures that a file marked as non-deletable is not deleted mistakenly. Of course, a privileged user can alter the protection code of any file and, therefore, remove the write protect against owner status.

   Note that explicit deletion of files (e.g., UNSAVE, KILL) will always proceed if the user is privileged.

2. When an account is ZEROed, any file that is marked as write (delete) protected against owner is not deleted. Such files are skipped by the zeroing code; all other files are still deleted. Once again, a privileged user can change the protection code of any file and then delete it or zero the account.

UNDERSTANDING NO ROOM FOR USER ON DEVICE ERRORS IN MESSAGE SEND/RECEIVE

The "?No room for user on device" error, error code 4, occurs when sending a message to a receiver whose pending message quota has been reached. In RSTS/E VØ6C this problem occurs when the receiver cannot process incoming messages fast enough to keep the number of pending messages below its quota. To ease the impact on system resources, once a sender tries to exceed the pending message quota for a receiver all further message sends to a receiver fail with error 4, until all outstanding messages are received. If the receiver goes into a hibernation state in RSTS/E VØ6C, messages can still be sent to the receiver until its quota has been reached, at which time all further sends to the receiver will return error code 4.

When a receiver goes into a hibernation state and messages are sent to it, significant degradation of the system may occur. This is due to the number of small buffers which are allocated as pending message blocks. If the receiver is unable to service the pending messages, due to hibernation, these small buffers are essentially lost from the system until the receiver starts to service its pending message queue again. To eliminate this problem, the send/receive code in RSTS/E V7.Ø has been modified to return error code 4 if the receiver is hibernating. This change will result in less degradation to the system if the receiver goes into a hibernation state and will result in fewer lost messages when the receiver is re-initialized to a runnable state.

SPECIAL LINE PRINTER DRUMS - DEVICE DRIVER FEATURE PATCH

Most DEC LP11 line printer drums contain a character which corresponds to octal code Ø4Ø (the ASCII SPACE character). This character may be a blank, the British Pound Sterling character, or some other optional non-standard character. Normally, this character is irrelevant, since the LP11 is usually wired to print only ASCII codes Ø41 through 137 (octal) for the 64 character set, or Ø41 through 176 for the 96 character set. The SPACE character (Ø4Ø), "non-printing" characters (codes ØØØ to Ø11, Ø13, and Ø16 to Ø37), and character codes greater than 137 (on printers with the 64 character set) all print as spaces. The space is created by suppressing printer action rather than actually having a hammer strike a character position on the printer drum.

By making a simple wiring change to the LP11, the range of printing characters can be extended to include ASCII code Ø4Ø. If the printer drum contains a special character in the Ø4Ø position, sending an Ø4Ø code to the printer will cause that special character to be printed.

In conjunction with this change, you can patch the RSTS/E line printer driver to allow special handling of ASCII Ø4Ø on a unit by unit basis. Normally, the driver sends Ø4Ø codes (SPACEs) unchanged to the printer. If the wiring change is not installed in the printer, this character will appear as a space regardless of the symbol on the line printer drum. The patch allows you to specify an ASCII code which the driver will translate to Ø4Ø before sending it to the printer. The special character will thus be printed. In addition, the driver will translate all Ø4Ø codes to Ø37, which is a non-printing character and appears as a space. Note that translation of Ø4Ø is performed only if the special character translation is enabled.

Ideally, you should choose a character which is not commonly used for other purposes but which can be typed and printed on any standard terminal. This allows you to use any standard editor to insert the special character into text destined for the printer. Remember that the conversion in the printer driver overrides the standard symbol (if any) for the selected code. For example, if you select the code 1Ø1 (the letter A), the special character is printed in place of any A sent to the driver. It would therefore be impossible to print an A.

All normal printing characters have some meaning in RSTS/E or BASIC-PLUS. For example, the & character ends a comment field in the EXTEND mode of BASIC-PLUS, is an abbreviation for account [1,5], and is a special character in RUNOFF source files. The @ character is seldom used in printed material, but RSTS/E uses it as an assignable account designator.

If none of the printable characters is suitable, one of the ASCII control characters (codes ØØ1 to Ø37) may be used. For example, CTRL/F (ØØ6) can be generated with most terminals and will echo and print as ^F if UP ARROW mode is selected with TTYSET. It has no meaning to any standard DEC terminal.

As system manager, you must choose the character which is appropriate for your installation by studying the ASCII code table and considering your users' needs. The procedure for installing the patch is described below.

PROCEDURE:

1. This is a feature patch to the RSTS/E V7.Ø executive. It may be installed in any target monitor SIL that includes line printer support.

2. Decide which line printer unit should have special character handling. If more than one unit is to be patched, repeat the steps below for each unit you wish to modify.

3. Select the character to represent the special printer symbol. If this character is a printable character (e.g., &), specify the new value in the patch by simply typing

   ```
   '&
   ```

   and replacing the & with the character you have chosen. If the character is a control character, specify its ASCII code as an octal number (or decimal number followed by a decimal point), or as 'x-1ØØ to represent CTRL/x. For example, CTRL/N may be specified as

   ```
   16
   ```

   or as

   ```
   14.
   ```

   or as

   ```
   'N-1ØØ
   ```

   in the patch.

4. The patch described in Step 6 below can be installed using the PATCH option of INIT.SYS:

   ```
   Option: PATCH
   File to patch? <lf>                    (LINE FEED for installed monitor SIL)
   ```

5. This patch can be installed manually using ONLPAT, the on-line patching program:

   ```
   RUN $ONLPAT
   Command File Name? <cr>                (RETURN for manual patch installation)
   File to patch? <lf>                    (LINE FEED for installed monitor SIL)
   File found in account [Ø,1]
   ```

   The patch is also contained in a patch file appearing in patch kit version "A" or later.

NOTE

The patch file for this patch requires manual editing to include installation specific parameters before it can be successfully installed.

6. In this example, the symbol n is used to denote the line printer unit being modified. You should type the actual unit number (between zero and the number of printers -1). The patch is as follows:

```
Module name? RSTS
Base address? LPDDDB
Offset address? DDS.LP*n+3Ø              (n is the unit number)
 Base   Offset  Old       New?
??????  ??????  ØØØØØØ  ? (new value from step 3)
??????  ??????  ØØØØØØ  ? ^Z             (CTRL/Z for new offset)
Offset address? ^Z                       (CTRL/Z for new base)
Base address? $$Ø3Ø7
Offset address? Ø
 Base   Offset  Old       New?
??????  ØØØØØØ  ??????  ? Q!1
??????  ØØØØØ2  ??????  ? ^C             (up-arrow/C to exit;CTRL/C for INIT)
```

7. Install the hardware modification to the printer. The modification is different for the several printer models, as listed below:

LPØ1 (231Ø) Wiring change required to card cage backplane. Remove wire A3A4-36 and connect this wire through a 1K pullup resistor to +5 Volts.

LPØ2 (241Ø) Change to AR16 Data Register Card. Remove card from slot A3A24. Lift Pin 5 on Z15 and tie Pin 5 to ground.

LPØ4 (247Ø) Wiring change required to card cage backplane. Remove wire from A3A24-Ø4 and connect this wire through a 1K pullup resistor to +5 Volts.

LPØ5 (223Ø) Insert jumper W1 on the Logic Control Board 29-21112.

RSTS/E V7.Ø
Executive
Device Driver Patches

CHANGING DEFAULT PRINTER FORM LENGTH - DEVICE DRIVER FEATURE PATCH

When a line printer is opened, the default hardware form length is set to 66. This is the actual form length for most applications. You can change the default form length by installing the patch given below. The legal form length values are between 7. and 127. inclusive. Note that this patch changes the default form length for all line printers on the system.

PROCEDURE:

1. This is a feature patch to the RSTS/E V7.Ø executive. It may be installed in any target monitor SIL which includes line printer support.

2. The patch described in Step 4 below can be installed using the PATCH option of INIT.SYS:

```
Option: PATCH
File to patch? <lf>                    (LINE FEED for installed monitor SIL)
```

3. This patch can be installed manually using ONLPAT, the on-line patching program:

```
RUN $ONLPAT
Command File Name? <cr>                (RETURN for manual patch installation)
File to patch? <lf>                    (LINE FEED for installed monitor SIL)
File found in account [Ø,1]
```

The patch is also contained in a patch file appearing in patch kit version "A" or later.

NOTE

> The patch file for this patch requires manual editing to include installation specific parameters before it can be successfully installed.

4. Be sure you include the decimal point after the new value so that it is interpreted as a decimal number. The patch is as follows:

```
Module name? RSTS
Base address? ..LP66
Offset address? Ø
 Base    Offset  Old        New?
??????  ØØØØØØ  ØØØ1Ø2   ? n.          (the new default)
??????  ØØØØØ2  112712   ? ^Z          (CTRL/Z for new offset)
Offset address? ^Z                     (CTRL/Z for new base)
Base address? $$Ø3Ø7
Offset address? Ø
 Base    Offset  Old        New?
??????  ØØØØØØ  ??????   ? Q!2
??????  ØØØØØ2  ??????   ? ^C          (up-arrow/C to exit;CTRL/C for INIT)
```

5. If you are using the line printer spooling package, you must change the control file (or files) which start the spoolers. Include the /DFLENGTH:n switch in the command line for each spooler to specify the new default form length.

RSTS/E V7.Ø
Executive
Device Driver Patches

TU16/TE16/TU45/TU77 MAGNETIC TAPE DENSITY/PARITY DEFAULT - DEVICE DRIVER FEATURE PATCH

The default density and parity settings for TU16, TE16, TU45, or TU77 magnetic tape (device MM) under RSTS/E V7.Ø are 8ØØ BPI and ODD parity. A program can change the density and/or parity value for either file structured or non-file structured operations by using the MAGTAPE function.

You can change your system's default settings for density and/or parity by installing the patch given below. This is useful if you frequently interchange tapes with other systems which use 16ØØ BPI phase encoded defaults. It is also useful if you want to use the 16ØØ BPI recording mode as your installation's standard.

NOTE

All RSTS/E V7.Ø distribution magnetic tapes for TU16, TE16, TU45, and TU77 (including optional languages such as COBOL and FORTRAN) are written at 8ØØ BPI with ODD parity (except for the 24ØØ ft. distribution). Do not install this patch until after you have completed the system library build procedures, including any optional software.

PROCEDURE:

1. This is a feature patch to the RSTS/E V7.Ø executive. It may be installed in any target monitor SIL which is configured for TU16, TE16, TU45, or TU77 magnetic tape.

2. Select the desired magnetic tape density and parity. To use 16ØØ BPI phase encoded mode, specify a new value of Ø in the patch. To return to 8ØØ BPI and ODD parity, specify a new value of 14Ø. Any other value will cause file structured operations to fail.

3. The patch described in Step 5 below can be installed using the PATCH option of INIT.SYS:

```
Option: PATCH
File to patch? <lf>                    (LINE FEED for installed monitor SIL)
```

4. This patch can be installed manually using ONLPAT, the on-line patching program:

```
RUN $ONLPAT
Command File Name? <cr>                (RETURN for manual patch installation)
File to patch? <lf>                    (LINE FEED for installed monitor SIL)
File found in account [Ø,1]
```

The patch is also contained in a patch file appearing in patch kit version "A" or later.

NOTE

The patch file for this patch requires manual editing to include installation specific parameters before it can be successfully installed.

5. The patch is as follows:

```
Module name? RSTS
Base address? ..MTDP
Offset address? 1
 Base   Offset  Old       New?
??????  ØØØØØ1     14Ø  ? n              (from step 2)
??????  ØØØØØ2     ???  ? ^Z             (CTRL/Z for new offset)
Offset address? ^Z                       (CTRL/Z for new base)
Base address? $$Ø3Ø7
Offset address? Ø
 Base   Offset  Old       New?
??????  ØØØØØØ  ??????  ? Q!4
??????  ØØØØØ2  ??????  ? ^C             (up-arrow/C to exit;CTRL/C for INIT)
```

RSTS/E V7.Ø
Executive
Device Driver Patches

ALLOW SPECIAL CONTROL CHARACTERS - DEVICE DRIVER FEATURE PATCH

All line printers supported by RSTS/E software use only 7-bit characters; the sign bit of the character is always Ø. In fact, the line printer driver interprets characters with the sign bit "on" (ASCII values greater than 127 decimal) as forms control characters if the special OPEN MODE 512%+n% is used. (See the RSTS/E Programming Manual.) The line printer driver also uses character values greater than 127 internally for space compression purposes and to suppress trailing blanks.

Some RSTS/E installations have special line printers that require the use of control characters with values greater than 127, even though the use of such equipment is not supported. The feature patch descibed below will allow full 8-bit characters to be passed unmodified to the line printer.

CAUTION

> The patch described below produces some side effects which must be recognized before installing the patch. First, if your line printer cannot interpret 8-bit characters (all supported printers fall into this category), do not install this patch or unpredictable results may occur. Second, this patch prevents the use of software formatting (MODE 512%) and prevents internal space compression, i.e., trailing blanks will be printed. Third, this patch affects every line printer on your system.
>
> The patch also has some effect on the spooling package. Incorrect page counts will probably result if the RESTART or REQUE options are used. In addition, unless the line printer spooler start-up option /LPFORM:NO is used, a user specifying LPFORM will probably crash the spooling package.

PROCEDURE:

1. This is a feature patch to the RSTS/E V7.Ø executive. It may be installed in any target monitor SIL.

2. The patch described in Step 4 below can be installed using the PATCH option of INIT.SYS:

```
Option: PATCH
File to patch? <lf>                    (LINE FEED for installed monitor SIL)
```

3. This patch can be installed manually using ONLPAT, the on-line patching program:

```
RUN $ONLPAT
Command File Name? <cr>                (RETURN for manual patch installation)
File to patch? <lf>                    (LINE FEED for installed monitor SIL)
File found in account [Ø,1]
```

The patch is also contained in a patch file appearing in patch kit version "A" or later.

4. The patch is as follows:

```
Module name? RSTS
Base address? ..LPSP
Offset address? Ø
 Base   Offset  Old       New?
??????  ØØØØØØ  ØØØØØØ  ? ØØ2ØØØ
??????  ØØØØØ2  ØØ5Ø61  ? ^Z          (CTRL/Z for new offset)
Offset address? ^Z                    (CTRL/Z for new base)
Base address? $$Ø3Ø7
Offset address? Ø
 Base   Offset  Old       New?
??????  ØØØØØØ  ??????  ? Q!1Ø
??????  ØØØØØ2  ??????  ? ^C          (up-arrow/C to exit;CTRL/C for INIT)
```

RSTS/E V7.Ø
Executive
Device Driver Patches

USING THE CSS PAPER TAPE READER ON RSTS/E V7.Ø - DEVICE DRIVER FEATURE PATCH

PROBLEM:

The paper tape reader device driver included in the RSTS/E V7.Ø object library does not work with the CSS paper tape reader (PA611).

SOLUTION:

The paper tape reader driver for RSTS/E V7.Ø is for Digital Equipment Corporation models PC11 and PR11. The following optional patch will make the driver compatible with the CSS paper tape reader model PA611. Note that after applying this patch the device driver does not work with models PC11 and PR11.

PROCEDURE:

1. This is a feature patch to the RSTS/E V7.Ø executive. It may be installed in any target monitor SILs.

2. The patch described in Step 4 below can be installed using the PATCH option of INIT.SYS:

```
Option: PATCH
File to patch? <lf>                    (LINE FEED for installed monitor SIL)
```

3. This patch can be installed manually using ONLPAT, the on-line patching program:

```
RUN $ONLPAT
Command File Name? <cr>                (RETURN for manual patch installation)
File to patch? <lf>                    (LINE FEED for installed monitor SIL)
File found in account [Ø,1]
```

   The patch is also contained in a patch file appearing in patch kit version "A" or later.

4. The patch is as follows:

```
Module name? RSTS
Base address? ..PCSS
Offset address? Ø
 Base   Offset  Old      New?
??????  ØØØØØØ  1ØØ2ØØ  ? 1ØØ1ØØ
??????  ØØØØØ2  ??????  ? ^Z              (CTRL/Z for new offset)
Offset address? ^Z                       (CTRL/Z for new base)
Base address? $$Ø3Ø7
Offset address? Ø
 Base   Offset  Old        New?
??????  ØØØØØØ  ??????  ? Q!2Ø
??????  ØØØØØ2  ??????  ? ^C              (up-arrow/C to exit;CTRL/C for INIT)
```

CR11 CARD READER NOTES

The CR11 card reader normally performs a dark check at column 81 of each card. Cards which have been verified using the IBM 129 punch/verifier or similiar devices will have a verification punch in column 81. This will result in a READ CHECK condition.

There is no way to fix this problem in software since the READ CHECK must be cleared manually at the card reader. There is, however, a hardware fix.

The presence or absence of a wire jumper between pad A and pad B on the "CLOCK" card inside the CR11 reader selects either column Ø or column 81 for the dark check.

Refer to the Documation M2ØØ Card Reader Manual for the clock card schematic and the location of this printed circuit card in the reader.

Jumper A-B

In	Selects dark check at Column Ø
Out	Selects dark check at Column 81

The jumper should be present (in) if cards are read which have verification punches in column 81.

RSTS/E SUPPORT OF MIXED TYPE DRIVES ON SAME DISK CONTROLLER

RSTS/E V7.Ø supports many types of disks. Usually, each type of disk requires a separate controller. Frequently, two similar types of disks may share the same controller. RSTS/E supports many of these mixed configurations. The following sections describe the level of support for each possible mixing.

MIXING RPØ4, RPØ5, AND RPØ6 DRIVES ON THE SAME CONTROLLER

For example, the RPØ4 and the RPØ6 both use the RH11 interface (RH7Ø for PDP-11/7Ø's). As a result, on a given system one or more RPØ4 drives may share an RH with one or more RPØ6 drives. The only restrictions are that each drive in the combined set must have a unique drive number, and that the total number of drives may not exceed eight. The RSTS/E Monitor distinguishes which type of disk is on a given controller and uses it appropriately. The RPØ5 is functionally identical to the RPØ4, and the above comments apply equally to the RPØ5.

MIXING RKØ6 AND RKØ7 DRIVES ON THE SAME CONTROLLER

RSTS/E V7.Ø supports both RKØ6 and RKØ7 disks. The RK611 and RK711 controllers are both capable of supporting RKØ6 and RKØ7 drives in any combination. (Some older RK611 controllers need an ECO in order to have this capability.) The same rules apply to mixing RKØ6 and RKØ7 drives as stated above for RPØ4 and RPØ6 drives.

MIXING RLØ1 AND RLØ2 DRIVES ON THE SAME CONTROLLER

RSTS/E V7.Ø supports both RLØ1 and RLØ2 disks. The same rules apply to mixing RLØ1 and RLØ2 drives as stated above for RPØ4 and RPØ6 drives, except that the number of RLØ1's and RLØ2's may not exceed four.

MIXING RMØ2 AND RMØ3 DRIVES ON THE SAME CONTROLLER

The RMØ2 and RMØ3 are very similar to the RPØ4, RPØ5 and RPØ6 disks, and also use the RH controllers. The RMØ2 disk uses the RH7Ø (for PDP-11/7Ø's) or the RH11 interface (for other PDP-11's), while the RMØ3 disk uses only the RH7Ø interface and is supported only on PDP-11/7Ø systems. On a PDP-11/7Ø system, RMØ2 and RMØ3 disks may share the same RH7Ø. This combination is analogous to the mixing of RPØ4 and RPØ6 drives described above.

MIXING RMØ2/RMØ3 DRIVES AND RPØ4/RPØ5/RPØ6 DRIVES ON SEPARATE CONTROLLERS

When you want to use both the RM and the RP disks on a single system configuration, Digital recommends that you use a separate RH controller for each kind of disk. The normal arrangement is to have the RH controller for the RP disks at the standard CSR address of 176700, and the RH controller for the RM disks at the non-standard CSR address of 176300. However, with this arrangement you cannot bootstrap the RM disks with any of the existing hardware bootstraps, although you can use the BOOT option of INIT.

If you would rather use the hardware bootstrap with the RM disks, you should reverse the CSR assignments so that the RP disk controller is at the non-standard address (176300), and the RM disk controller is at the standard address (176700).

Note that if the hardware bootstrap requires you to type a device name at the console keyboard, then typing either DB (the name used for RP disks) or DR (the name used for RM disks) will bootstrap a disk controlled through the standard CSR address, regardless of which type it actually is.

MIXING RMØ2/RMØ3 DRIVES AND RPØ4/RPØ5/RPØ6 DRIVES ON THE SAME CONTROLLER

It is possible to mix RMØ2/RMØ3 drives with RPØ4/RPØ5/RPØ6 drives on the same RH controller. Up to eight drives are allowed per RH, and each drive must have a unique drive number. The Monitor knows which type of disk is on a given drive, and uses it appropriately. Note that RMØ3's are still supported only on PDP-11/7Ø's

The unusual aspect of this arrangement is the device naming scheme. The lowest numbered drive on the RH determines whether all drives on this RH will be known as DBn: or DRn:. For example, if the lowest numbered drive is an RPØ4, RPØ5 or RPØ6, then the device name DB is used for all drives. If the lowest numbered drive is an RMØ2 or RMØ3, then the device name DR is used. The RSTS/E Monitor is not confused about the identity of the disks. Only the naming scheme is not obvious.

MULTIPLE CONTROLLERS FOR THE SAME TYPE OF DISK

Prior to VØ6C, RSTS/E supported only one RH controller per system for RPØ4/RPØ5/RPØ6 disks. This limited each system to a maximum of eight drives of this type. Since VØ6C, the support of RMØ2/RMØ3 disks, which also use RH controllers, allows you to use two similar RH controllers on the same system by specifying both RP and RM disks during System Generation. These two RH controllers can both be controlling the same kind of disk, either RP or RM. Of course, only one of these two RH's may be at the standard CSR address. The other must be at the non-standard CSR address. Each RH can control up to eight drives, giving a possible combined total of sixteen drives.

The naming scheme for the above arrangement is as follows: The drives interfaced at the standard CSR address will be known by the device name that corresponds to the lowest numbered drive type (DB or DR), while the drives interfaced at the non-standard CSR address will be known by the other name.

RANDOM MIXING OF RP AND RM DISKS ON MULTIPLE CONTROLLERS

RSTS/E V7.Ø allows you to have two RH controllers and place a mixture of RP and RM disks on both RH's. If the respective lowest numbered drives on the two RH's are of different types, then the previous rule of "lowest numbered drive determines device name" holds. If both lowest numbered drives are of the same type, then the tie-breaking rule is: The RH at the standard CSR address gets the device name that corresponds to the lowest numbered drive type and the RH at the non-standard CSR address gets the other device name.

BOOTING MIXED RMØ2/RMØ3 AND RPØ4/RPØ5/RPØ6 DISKS

RSTS/E will support the types of disk combinations described above. However, if you place two disks of the same type (RMØ2/RMØ3 or RPØ4/RPØ5/RPØ6) at the same unit number but different controllers then you will only be able to boot (with INIT.SYS or the hardware bootstrap) the one at the primary (176700) controller address.

ANSWERING SYSGEN QUESTIONS FOR RP AND RM DISKS

In all of the above cases where a mixture of RP and RM disks are controlled by one or more controllers, or where there is more than one controller for the same kind of disk, the naming scheme for each case affects how you answer the questions regarding the number of RP disks and the number of RM disks. Your answer to the "RPØ4/RPØ5/RPØ6's?" question determines how many drives are supported on the controller called DB, regardless of the actual types of those drives. Similarly, the answer to the "RMØ2/RMØ3's?" question determines how many drives are supported on the controller called DR.

RSTS/E V7.Ø
BASIC-PLUS
BASIC-PLUS Patches

SPECIAL PRINT-USING CHARACTERS - BASIC-PLUS FEATURE PATCH

The Print-Using option of BASIC-PLUS provides for floating dollar sign, comma insertion every three digits to the left of the decimal point, and, or course, printing of the decimal point. The $-sign is used for the currency symbol in the United States and Canada, but various other symbols are used for other monetary systems. In the United Kingdom it would be useful to float the Pound Sterling symbol (or whatever ASCII character is used to represent that symbol) instead of the $-sign. In France a floating Franc symbol (or F) would be generally more useful than the floating $-sign. The role of the comma and period are also reversed in France (e.g., 9,999,999.Ø2 should be printed as 9.999.999,Ø2).

To accommodate these differences in monetary systems, Print-Using is coded to allow substitution of any character for the floating currency symbol, the decimal point character, and the "every three digits" character. These three characters are defined by three words in the BASIC-PLUS Run-Time System. System defaults and several suggestions are shown in the table below:

Location	Usage	Default	England	France
..PUCH+Ø	Floating currency symbol	$ (Ø44)	& (Ø46)	F (1Ø6)
..PUCH+2	Decimal point character	. (Ø56)	. (Ø56)	, (Ø54)
..PUCH+4	Every three digits character	, (Ø54)	, (Ø54)	. (Ø56)

Substitution of different characters, such as those listed under England and France above, is done with a simple patch. Any change will affect both the Print-Using format string and the output produced. For example, if the suggestions under "France" were installed, the following Print-Using statement would result in the output shown:

```
PRINT USING "FF#.######,##", 3673298.Ø2, 4545.2Ø

F3.673.298,Ø2
    F4.545,2Ø
```

Since any change in this area does render the Print-Using documentation slightly incorrect, an appropriate note should be published for users of the system.

PROCEDURE:

1. This is a feature patch to the BASIC-PLUS Run-Time System. It may be installed in any BASIC-PLUS Run-Time System configured with Print-Using.

2. Determine the three special characters you wish to use for the floating currency symbol, the decimal point character, and the "every three digits" character. The chosen characters can be entered as either:

   'x where x is the chosen printable character
   n where n is the octal value of the chosen character
   n. where n is the decimal value of the chosen character

3. The patch described in Step 5 below can be installed using the PATCH option of INIT.SYS:

```
Option: PATCH
File to patch? BASIC.RTS                (BASIC-PLUS Run-Time System name)
```

4. This patch can be installed manually using ONLPAT, the on-line patching program:

```
RUN $ONLPAT
Command File Name? <cr>                 (RETURN for manual patch installation)
File to patch? [Ø,1]BASIC.RTS           (BASIC-PLUS Run-Time System name)
File found in account [Ø,1]
```

The patch is also contained in a patch file appearing in patch kit version "A" or later.

NOTE

> The patch file for this patch requires manual editing to include installation specific parameters before it can be successfully installed.

5. The patch is as follows:

```
Base address? $$Ø4Ø1
Offset address? Ø
 Base   Offset  Old     New?
??????  ØØØØØØ  ??????  ? Q!1
??????  ØØØØØ2  ??????  ? ^Z           (CTRL/Z for new offset)
Offset address? ^Z                      (CTRL/Z for new base)
Base address? ..PUCH
Offset address? Ø
 Base   Offset  Old     New?
??????  ØØØØØØ  ØØØØ44  ? 'f           (currency sign from step 2)
??????  ØØØØØ2  ØØØØ56  ? ',           (decimal point from step 2)
??????  ØØØØØ4  ØØØØ54  ? '.           ("every three" from step 2)
??????  ØØØØØ6  ??????  ? ^C          (up-arrow/C to exit;CTRL/C for INIT)
```

6. If the above patch was installed using ONLPAT, it will take effect the next time the Run-Time System is reloaded. If the Run-Time System has been ADDed, or if it is your system default Run-Time System, execute the following commands:

```
RUN $UTILTY<cr>
<UTILTY's header line>
#UNLOAD BASIC<cr>                       (BASIC-PLUS Run-Time System name)
#EXIT<cr>

Ready
```

NOTE: The UNLOAD command will not remove the Run-Time System, but simply instructs the monitor to reload it the next time a job requests it.

If, in addition to being previously ADDed, the Run-Time System had been ADDed with the /STAY switch, also reload it as follows:

```
RUN $UTILTY<cr>
<UTILTY's header line>
#LOAD BASIC/STAY/ADDR:nnn<cr>          (BASIC-PLUS Run-Time System name)
#EXIT<cr>

Ready
```

DEFAULT SCALE FACTOR - BASIC-PLUS FEATURE PATCH

The BASIC-PLUS default SCALE factor may be modified by utilizing the procedure in this article. A complete description of SCALE factors is included in the BASIC-PLUS Language Manual.

PROCEDURE:

1. This is a feature patch to the BASIC-PLUS Run-Time System. It may be installed in any BASIC-PLUS Run-Time System configured with the 4-word, scaled math package.

2. Determine the new BASIC-PLUS default SCALE factor. This default SCALE factor must be between Ø and 6 inclusive.

3. The patch described in Step 5 below can be installed using the PATCH option of INIT.SYS:

```
Option: PATCH
File to patch? BASIC.RTS                (BASIC-PLUS Run-Time System name)
```

4. This patch can be installed manually using ONLPAT, the on-line patching program:

```
RUN $ONLPAT
Command File Name? <cr>                 (RETURN for manual patch installation)
File to patch? [Ø,1]BASIC.RTS           (BASIC-PLUS Run-Time System name)
File found in account [Ø,1]
```

The patch is also contained in a patch file appearing in patch kit version "A" or later.

NOTE

> The patch file for this patch requires manual editing to include installation specific parameters before it can be successfully installed.

5. The patch is as follows:

```
Base address? $$Ø4Ø1
Offset address? Ø
 Base   Offset  Old     New?
??????  ØØØØØØ  ??????  ? Q!2
??????  ØØØØØ2  ??????  ? ^Z              (CTRL/Z for new offset)
Offset address? ^Z                      (CTRL/Z for new base)
Base address? ..SCA.
Offset address? Ø
 Base   Offset  Old     New?
??????  ØØØØØØ  ØØØØØØ  ? n               (from step 2)
??????  ØØØØØ2  ??????  ? ^C              (up-arrow/C to exit;CTRL/C for INIT)
```

6. If the above patch was installed using ONLPAT, it will take effect the next time the Run-Time System is reloaded. If the Run-Time System has been ADDed, or if it is your system default Run-Time System, execute the following commands:

```
RUN $UTILTY<cr>
<UTILTY's header line>
#UNLOAD BASIC<cr>                    (BASIC-PLUS Run-Time System name)
#EXIT<cr>

Ready
```

NOTE: The UNLOAD command will not remove the Run-Time System, but simply instructs the monitor to reload it the next time a job requests it.

If, in addition to being previously ADDed, the Run-Time System had been ADDed with the /STAY switch, also reload it as follows:

```
RUN $UTILTY<cr>
<UTILTY's header line>
#LOAD BASIC/STAY/ADDR:nnn<cr>        (BASIC-PLUS Run-Time System name)
#EXIT<cr>

Ready
```

RSTS/E V7.Ø
BASIC-PLUS
BASIC-PLUS Patches

OMITTING SCALE FACTOR WARNING MESSAGE - BASIC-PLUS FEATURE PATCH

Normally, if a BASIC-PLUS program is compiled with one SCALE factor and then is run under a different job SCALE factor, the "%SCALE Factor Interlock" warning message is printed and execution proceeds using the program's SCALE factor. It may be desirable to omit the printing of this message.

PROCEDURE:

1. This is a feature patch to the BASIC-PLUS Run-Time System. It may be installed in any BASIC-PLUS Run-Time System configured with the 4-word, scaled math package.

2. The patch described in Step 4 below can be installed using the PATCH option of INIT.SYS:

```
Option: PATCH
File to patch? BASIC.RTS                (BASIC-PLUS Run-Time System name)
```

3. This patch can be installed manually using ONLPAT, the on-line patching program:

```
RUN $ONLPAT
Command File Name? <cr>                 (RETURN for manual patch installation)
File to patch? [Ø,1]BASIC.RTS           (BASIC-PLUS Run-Time System name)
File found in account [Ø,1]
```

   The patch is also contained in a patch file appearing in patch kit version "A" or later.

4. The patch is as follows:

```
Base address? $$Ø4Ø1
Offset address? Ø
 Base   Offset  Old     New?
??????  ØØØØØØ  ??????  ? Q!4
??????  ØØØØØ2  ??????  ? ^Z           (CTRL/Z for new offset)
Offset address? ^Z                      (CTRL/Z for new base)
Base address? ..SCE.
Offset address? Ø
 Base   Offset  Old     New?
??????  ØØØØØØ  1Ø4577  ? 24Ø          (for no warning message)
??????  ØØØØØ2  ØØØ2Ø7  ? ^C           (up-arrow/C to exit;CTRL/C for INIT)
```

5. If the above patch was installed using ONLPAT, it will take effect the next time the Run-Time System is reloaded. If the Run-Time System has been ADDed, or if it is your system default Run-Time System, execute the following commands:

```
RUN $UTILTY<cr>
<UTILTY's header line>
#UNLOAD BASIC<cr>                    (BASIC-PLUS Run-Time System name)
#EXIT<cr>

Ready
```

NOTE: The UNLOAD command will not remove the Run-Time System, but simply instructs the monitor to reload it the next time a job requests it.

If, in addition to being previously ADDed, the Run-Time System had been ADDed with the /STAY switch, also reload it as follows:

```
RUN $UTILTY<cr>
<UTILTY's header line>
#LOAD BASIC/STAY/ADDR:nnn<cr>        (BASIC-PLUS Run-Time System name)
#EXIT<cr>

Ready
```

RSTS/E V7.Ø
BASIC-PLUS
BASIC-PLUS Patches

DEFAULT TO EXTEND MODE - BASIC-PLUS FEATURE PATCH

Normally, BASIC-PLUS defaults to NO EXTEND mode. To change this default to EXTEND mode, apply the patch in this article. Digital recommends that you install this patch, because the use of EXTEND mode will ease any transition to other versions of BASIC-PLUS or BASIC-PLUS-2. (See section 2.8.3 of these Release Notes for a list of the conventions used by the RSTS/E group to ensure compatibility with both BASIC-PLUS and BASIC-PLUS-2.)

PROCEDURE:

1. This is a feature patch to the BASIC-PLUS Run-Time System. It may be installed in any BASIC-PLUS Run-Time System.

2. The patch described in Step 4 below can be installed using the PATCH option of INIT.SYS:

```
Option: PATCH
File to patch? BASIC.RTS                (BASIC-PLUS Run-Time System name)
```

3. This patch can be installed manually using ONLPAT, the on-line patching program:

```
RUN $ONLPAT
Command File Name? <cr>                 (RETURN for manual patch installation)
File to patch? [Ø,1]BASIC.RTS           (BASIC-PLUS Run-Time System name)
File found in account [Ø,1]
```

   The patch is also contained in a patch file appearing in patch kit version "A" or later.

4. The patch is as follows:

```
Base address? $$Ø4Ø1
Offset address? Ø
 Base   Offset  Old     New?
??????  ØØØØØØ  ??????  ? Q!1Ø
??????  ØØØØØ2  ??????  ? ^Z          (CTRL/Z for new offset)
Offset address? ^Z                     (CTRL/Z for new base)
Base address? ..XTN.
Offset address? Ø
 Base   Offset  Old     New?
??????  ØØØØØØ  ØØØØØØ  ? 2ØØØØ
??????  ØØØØØ2  ??????  ? ^C          (up-arrow/C to exit;CTRL/C for INIT)
```

5. If the above patch was installed using ONLPAT, it will take effect the next time the Run-Time System is reloaded. If the Run-Time System has been ADDed, or if it is your system default Run-Time System, execute the following commands:

```
RUN $UTILTY<cr>
<UTILTY's header line>
#UNLOAD BASIC<cr>                    (BASIC-PLUS Run-Time System name)
#EXIT<cr>

Ready
```

NOTE: The UNLOAD command will not remove the Run-Time System, but simply instructs the monitor to reload it the next time a job requests it.

If, in addition to being previously ADDed, the Run-Time System had been ADDed with the /STAY switch, also reload it as follows:

```
RUN $UTILTY<cr>
<UTILTY's header line>
#LOAD BASIC/STAY/ADDR:nnn<cr>        (BASIC-PLUS Run-Time System name)
#EXIT<cr>

Ready
```

RSTS/E V7.Ø
BASIC-PLUS
BASIC-PLUS Patches

NO IMMEDIATE MODE FROM .BAS FILE - BASIC-PLUS FEATURE PATCH

Normally, BASIC-PLUS allows immediate mode commands to appear in the file accessed by the 'OLD' command. This may be undesirable in hostile environments where, for example, students could write programs which included 'UNSAVE' commands. To prevent BASIC-PLUS from accepting immediate mode commands in files, apply the patch in this article.

PROCEDURE:

1. This is a feature patch to the BASIC-PLUS Run-Time System. It may be installed in any BASIC-PLUS Run-Time System.

2. The patch described in Step 4 below can be installed using the PATCH option of INIT.SYS:

```
Option: PATCH
File to patch? BASIC.RTS                (BASIC-PLUS Run-Time System name)
```

3. This patch can be installed manually using ONLPAT, the on-line patching program:

```
RUN $ONLPAT
Command File Name? <cr>                 (RETURN for manual patch installation)
File to patch? [Ø,1]BASIC.RTS           (BASIC-PLUS Run-Time System name)
File found in account [Ø,1]
```

   The patch is also contained in a patch file appearing in patch kit version "A' or later.

4. The patch is as follows:

```
Base address? $$Ø4Ø1
Offset address? Ø
 Base   Offset  Old     New?
??????  ØØØØØØ  ??????  ? Q!2Ø
??????  ØØØØØ2  ??????  ? ^Z          (CTRL/Z for new offset)
Offset address? ^Z                     (CTRL/Z for new base)
Base address? ..IBAS
Offset address? Ø
 Base   Offset  Old     New?
??????  ØØØØØØ  ØØØ4Ø1  ? 14Ø1
??????  ØØØØØ2  1Ø4767  ? ^C          (up-arrow/C to exit;CTRL/C for INIT)
```

5. If the above patch was installed using ONLPAT, it will take effect the next time the Run-Time System is reloaded. If the Run-Time System has been ADDed, or if it is your system default Run-Time System, execute the following commands:

```
RUN $UTILTY<cr>
<UTILTY's header line>
#UNLOAD BASIC<cr>                    (BASIC-PLUS Run-Time System name)
#EXIT<cr>

Ready
```

NOTE: The UNLOAD command will not remove the Run-Time System, but simply instructs the monitor to reload it the next time a job requests it.

If, in addition to being previously ADDed, the Run-Time System had been ADDed with the /STAY switch, also reload it as follows:

```
RUN $UTILTY<cr>
<UTILTY's header line>
#LOAD BASIC/STAY/ADDR:nnn<cr>        (BASIC-PLUS Run-Time System name)
#EXIT<cr>

Ready
```

NO PPN'S IN CATALOG COMMAND - BASIC-PLUS FEATURE PATCH

Normally, BASIC-PLUS allows any user to list the contents of another user's directory with the CATALOG [P,PN] command. The patch in this article prevents BASIC-PLUS from accepting a project, programmer number in the CATALOG command. Thus, only the user's own directory may be listed.

PROCEDURE:

1. This is a feature patch to the BASIC-PLUS Run-Time System. It may be installed in any BASIC-PLUS Run-Time System.

2. The patch described in Step 4 below can be installed using the PATCH option of INIT.SYS:

```
Option: PATCH
File to patch? BASIC.RTS                (BASIC-PLUS Run-Time System name)
```

3. This patch can be installed manually using ONLPAT, the on-line patching program:

```
RUN $ONLPAT
Command File Name? <cr>                 (RETURN for manual patch installation)
File to patch? [Ø,1]BASIC.RTS           (BASIC-PLUS Run-Time System name)
File found in account [Ø,1]
```

The patch is also contained in a patch file appearing in patch kit version "A" or later.

4. The patch is as follows:

```
Base address? $$Ø4Ø1
Offset address? Ø
 Base   Offset  Old     New?
??????  ØØØØØØ  ??????  ? Q!4Ø
??????  ØØØØØ2  ??????  ? ^Z          (CTRL/Z for new offset)
Offset address? ^Z                     (CTRL/Z for new base)
Base address? ..CAT.
Offset address? Ø
 Base   Offset  Old     New?
??????  ØØØØØØ  ØØØ4Ø1  ? 5Ø64
??????  ØØØØØ2  ØØØØØ6  ? ^C          (up-arrow/C to exit;CTRL/C for INIT)
```

5. If the above patch was installed using ONLPAT, it will take effect the next time the Run-Time System is reloaded. If the Run-Time System has been ADDed, or if it is your system default Run-Time System, execute the following commands:

```
RUN $UTILTY<cr>
<UTILTY's header line>
#UNLOAD BASIC<cr>                    (BASIC-PLUS Run-Time System name)
#EXIT<cr>

Ready
```

NOTE: The UNLOAD command will not remove the Run-Time System, but simply instructs the monitor to reload it the next time a job requests it.

If, in addition to being previously ADDed, the Run-Time System had been ADDed with the /STAY switch, also reload it as follows:

```
RUN $UTILTY<cr>
<UTILTY's header line>
#LOAD BASIC/STAY/ADDR:nnn<cr>        (BASIC-PLUS Run-Time System name)
#EXIT<cr>

Ready
```

RSTS/E V7.Ø
BASIC-PLUS
BASIC-PLUS Patches

DISABLING THE CCL SYS CALL - BASIC-PLUS FEATURE PATCH

Normally, BASIC-PLUS allows any user to execute any valid CCL command on the system with SYS Call 14 (described in article seq 19.6.2 n in these notes). However, some installations may choose to disable this feature to discourage users from writing programs that will simulate the features of a standard Run-Time System. The patch in this article prevents BASIC-PLUS from executing SYS Call 14.

PROCEDURE:

1. This is a feature patch to the BASIC-PLUS Run-Time System. It may be installed in any BASIC-PLUS Run-Time System.

2. The patch described in Step 4 below can be installed using the PATCH option of init.sys:

```
Option: PATCH
File to patch? BASIC.RTS                (BASIC-PLUS Run-Time System name)
```

3. This patch can be installed manually using ONLPAT, the on-line patching program:

```
RUN $ONLPAT
Command File Name? <cr>                 (RETURN for manual patch installation)
File to patch? [Ø,1]BASIC.RTS           (BASIC-PLUS Run-Time System name)
File found in account [Ø,1]
```

   The patch is also contained in a patch file appearing in patch kit version "A" or later.

4. The patch is as follows:

```
Base address? $$ØØ4Ø1
Offset address? Ø
 Base   Offset  Old     New?
??????  ØØØØØØ  ??????  ? Q!1ØØ
??????  ØØØØØ2  ??????  ? ^Z           (CTRL/Z for new offset)
Offset address? ^Z                      (CTRL/Z for new base)
Base address? ..CCL.
Offset address? Ø
 Base   Offset  Old     New?
??????  ØØØØØØ  ??????  ? UUOBAD
??????  ØØØØØ2  ??????  ? ^C           (up-arrow/C to exit;CTRL/C for INIT)
```

5. If the above patch was installed using ONLPAT, it will take effect the next time the Run-Time System is reloaded. If the Run-Time System has been ADDed, or if it is your system default Run-Time System, execute the following commands:

```
RUN $UTILTY<cr>
<UTILTY's header line>
#UNLOAD BASIC<cr>                    (BASIC-PLUS Run-Time System name)
#EXIT<cr>

Ready
```

NOTE: The UNLOAD command will not remove the Run-Time System, but simply instructs the monitor to reload it the next time a job requests it.

If, in addition to being previously ADDed, the Run-Time System had been ADDed with the /STAY switch, also reload it as follows:

```
RUN $UTILTY<cr>
<UTILTY's header line>
#LOAD BASIC/STAY/ADDR:nnn<cr>        (BASIC-PLUS Run-Time System name)
#EXIT<cr>

Ready
```

TIME$ FUNCTION

The string produced by the BASIC-PLUS TIME$ function has been implemented to conform with standardized usage. The string is always 8 characters in length. For 24-hour time, the format is "hh:mm ", where midnight is "ØØ:ØØ " and noon is "12:ØØ ". For AM/PM time, the format is "hh:mm xx", where midnight is "12:ØØ PM" and noon is "12:ØØ M ".

The following table summarizes:

24-hour	AM/PM	Description
ØØ:ØØ	12:ØØ PM	Midnight
ØØ:Ø1	12:Ø1 AM	1 minute after midnight
ØØ:59	12:59 AM	59 minutes after midnight
Ø1:ØØ	Ø1:ØØ AM	1 hour after midnight
11:59	11:59 AM	1 minute before noon
12:ØØ	12:ØØ M	Noon
12:Ø1	12:Ø1 PM	1 minute after noon
12:59	12:59 PM	59 minutes after noon
13:ØØ	Ø1:ØØ PM	1 hour after noon
23:59	11:59 PM	1 minute before midnight

According to the Harper Dictionary of Contemporary Usage (Morris, William and Mary, Harper and Row, 1975, p. 3Ø),

> The designations "A.M." for the period between midnight and noon and "P.M." for that between noon and midnight are indisputable and never confused. Some persons are confused, though, as to how to write "12 noon" and "12 midnight."
>
> The key is the letter "M" which appears in "A.M." and "P.M." It is the abbreviation of "meridies", the Latin word for midday. Thus the correct designation for noon is "12 M." (for meridies); midnight is "12 P.M." (for "post meridiem"). Hours between midnight and midday are, of course, "A.M." - "ante meridiem."
>
> Occasionally you will see in print "12 N." for noon and "12 M." for midnight. These abbreviations are wrong.
>
> The forms given are those officially recognized by the U.S. Naval Observatory.

COMMA PRINT ZONE HANDLING

Comma print zones are a convenient method of formatting data in a columnar fashion when the width of the output device is unknown.

Comma print zones are fourteen (14) characters in length. The first comma print zone starts at the left hand margin and succeeding zones start every 14 characters thereafter. If the width of the output device is not a multiple of 14, there are up to 13 character positions not usable in comma print zone processing; partial comma print zones are not allowed.

A comma (,) in a BASIC-PLUS PRINT statement advances the "print head" to the start of the next comma print zone. If, after advancing at least one character position, there would remain a full 14 character comma print zone on the current line, BASIC-PLUS outputs spaces to advance to the start of that zone. Since at least one space is output, adjacent commas in a PRINT statement skip over comma print zones. If no more full comma print zones remain on the current line, BASIC-PLUS advances to the left hand margin of the next line by outputting a carriage return/line feed sequence.

Terminals and Line Printers have a real "line width". For terminals, this width can be set using TTYSET. BASIC-PLUS uses the current width setting for these devices in the comma print zone calculations.

All other devices do not have a "line width". These devices must default "line width" to some reasonable value so that, for example, a file output to disk can later be PIP'd to a terminal or line printer. The default width value used for all "widthless" devices is 72. This corresponds to five (5) comma print zones and ensures that files can be printed on all terminals and line printers without line length problems.

For complete control of output formatting, some other method such as the TAB function or PRINT-USING must be used.

CARRIAGE RETURN HANDLING IN INPUT LINE

This article describes the handling of Carriage Return (ASCII code 13 decimal) by the BASIC-PLUS INPUT LINE statement.

Whenever a Carriage Return (<CR>) is encountered, the next input character is examined. The following three cases exist:

1. <CR> followed by Line Feed (<LF>, ASCII code 1Ø decimal)

   This is the normal ASCII stream line termination case.

   The <CR> and <LF> together serve as the line's delimiter. The string returned by the INPUT LINE ends with <CR><LF>.

2. <CR> followed by Null (<NUL>, ASCII code Ø)

   Both the <CR> and the <NUL> are thrown away; character processing continues as if they never appeared.

3. <CR> followed by anything else

   The <CR> is not used as a delimiting character, but is kept and treated as a normal data character.

   This case occurs when overprint lines are being used (e.g., RUNOFF output with underlining). The line's normal form is:

   <data><CR><more data><CR><LF>

   The full <CR><LF> sequence is the line's terminator.

RSTS/E V7.Ø
BASIC-PLUS
Programming Hints

HANDLING OF INTEGER AND FLOATING POINT CONSTANTS

INTEGER CONSTANTS

Versions of BASIC-PLUS prior to VØ6B generated 5 bytes of code and data storage for every reference to an integer constant. These were: 1 byte of pseudo-opcode, 2 bytes of address, and 2 bytes of data (the constant itself). The current version of BASIC-PLUS generates 3 bytes of code and data for all integer constant references: 1 byte of pseudo-opcode and 2 bytes of data (the constant itself). It also treats the integers zero (Ø%) and one (1%) as special cases. These commonly used integers generate only 1 byte of code, a special pseudo-opcode. One programming technique in earlier versions of BASIC-PLUS was to assign some variable permanently to a constant value and use that variable instead of the literal constant throughout the program (variable references generated only 3 bytes of code). With the enhancement of BASIC-PLUS, this technique is no longer necessary and changing a program to use literal integer constants may well save further space!

FLOATING POINT CONSTANTS

Each reference to a floating point constant normally generates 7 (or 11 for 4-word math) bytes of code and data storage: 1 byte of pseudo-opcode, 2 bytes of address, and 4 (or 8) bytes of data (the constant itself). The current version of BASIC-PLUS checks the value of every floating point constant before generating the above sequence of code and data. If the constant has no non-zero bits in its least significant word (or 3 words for 4-word math), BASIC-PLUS generates an alternate 3 byte sequence: 1 byte of special pseudo-opcode and 2 bytes of data (the leading or most significant word of the constant). Most commonly used floating point numbers fall into this special class. Furthermore, BASIC-PLUS will recognize the floating point constants zero (Ø.) and one (1.) and generate only 1 byte of special pseudo-opcode.

RSTS/E V7.Ø
BASIC-PLUS
Non-standard BASIC-PLUS Features

THE BASIC-PLUS DEBUG FACILITY

A new feature of BASIC-PLUS, the DEBUG facility, is available with RSTS/E V7.Ø as an unsupported feature, and may be changed or removed in future versions of BASIC-PLUS.

The DEBUG commands extend the present BASIC-PLUS immediate mode debugging commands by allowing you to trace the flow of your program and to set breakpoints. These commands provide a subset of the functions provided by the BASIC-PLUS-2 debugging facility, and are only available as immediate mode commands. They cannot be used within BASIC-PLUS programs. The commands are used in one of two ways:

1. Issuing them between the OLD and RUN of a program and, thereafter, whenever a breakpoint is hit.

2. By including STOP statements within the program and then issuing the DEBUG commands when the first STOP statement is encountered and, thereafter, whenever another STOP statement or breakpoint is hit.

Any DEBUG commands are disabled when:

a. A "RUN <program-name>" command is successfully executed,
b. A NEW, OLD, or EXIT command is executed, or
c. Any valid CCL command is executed.

The keywords used with the DEBUG facility have been implemented with the second character of each keyword changed to a question mark ("?"). The patch described in article Seq. 4.1Ø.3 F may be used to change the question marks to the characters required to make the keyword names match the names of their respective functions (for example, "T?ACE" is changed to "TRACE"). Note that, if this patch is applied, any program with a variable or function name that represents a variant of TRACE, UNTRACE, BREAK, or UNBREAK (such as TRACE%, FNBREAK$, etc.) will not compile until the variable name is changed.

For the purposes of this discussion, assume that the patch described above has been installed. If you include the DEBUG features, but choose not to install the patch, replace all occurrences of the keywords below as follows:

With Patch	Without Patch
TRACE	T?ACE
UNTRACE	U?TRACE
BREAK	B?EAK
UNBREAK	U?BREAK

TRACE/UNTRACE

The TRACE command causes BASIC-PLUS to print the message "at line nnn" (where "nnn" represents a line number) each time that a line number is encountered or the current line number changes.

The UNTRACE command disables any previous invocation of the TRACE command.

BREAK/UNBREAK

The BREAK command allows you to specify that execution should stop any time that a specified line number is encountered. Up to 1Ø such breakpoints can be set. The BREAK command is of the form

BREAK [N1, N2, N3 ..., N1Ø]

where "Nn" represents a line number between 1 and 32767.

When a breakpoint is encountered, BASIC-PLUS prints the message "Break at line nnn" and returns to "Ready" as though a STOP statement had been executed. At this point you can examine or change the values of variables, execute the DUMP command (if enabled, see article Seq. 4.1Ø.2 N), or execute any other legal command. Type "CONT" (continue) to resume execution.

If the BREAK command is issued without a line number argument, BASIC-PLUS will BREAK each time that a new line is encountered.

The UNBREAK command is used to disable breakpoints, and is of the form

UNBREAK [N1, N2, N3, ..., N1Ø]

If no line numbers are specified, all breakpoints are disabled.

UNDERSTANDING LINE NUMBERS IN THE DEBUG FACILITY

(This section discusses the actual implementation of the DEBUG facility in the BASIC-PLUS Run-Time System, and should help to explain certain anomalies that you may encounter.)

The DEBUG code is entered whenever an internal construct called a "statement header" is encountered. Statement headers are generated for the following BASIC-PLUS elements:

Line numbers	DIMENSION statements
FNEND statements	NEXT statements
DATA statements	DEF statements
FOR statements	

When a statement header is encountered, BASIC-PLUS first checks to see if the BREAK or TRACE feature has been requested. If not, program execution continues.

If BREAK or TRACE has been requested, BASIC-PLUS now compares the current line number with the previously encountered line number. If they are the same, program execution continues.

If a different line number has been encountered, BASIC-PLUS then checks to see if the statement header represents a function definition ("DEF") statement. If so, program execution continues.

Because of the internal structure of BASIC-PLUS, it is not possible to BREAK at a function definition. As you can see from the algorithm used, any attempt to BREAK at a function definition is rejected while the program is being executed, rather than when the "BREAK" command is specified.

If the BREAK feature has been requested, BASIC-PLUS compares the current line number with the list of breakpoints that have been requested. If the current line number is included in the breakpoint list, BASIC-PLUS prints "Break at line nnn," and returns to keyboard monitor ("Ready") state.

If the TRACE feature has been requested, BASIC-PLUS prints "at line nnn" and proceeds with execution of the program.

SUPPORT POLICY FOR THE BASIC-PLUS DEBUG FEATURE

DIGITAL makes no commitment, expressed or implied, to support the BASIC-PLUS Run-Time System, or any BASIC-PLUS System programs (CUSPs) used with that run-time system, if the DEBUG feature is enabled in the BASIC-PLUS Run-Time System.

If you experience problems with the DEBUG feature, please submit an FYI-type Software Performance Report (SPR). While DIGITAL makes no commitment to fix problems reported with the DEBUG facility, we would like to know about any problems that you encounter.

ENABLING THE DEBUG FACILITY

The DEBUG facility requires approximately 16Ø (decimal) additional words in the BASIC-PLUS Run-Time System. In most cases, this means that one or more of the other optional features (String Arithmetic, Print Using, etc.) must be omitted to prevent the BASIC-PLUS Run-Time System from exceeding 16K words in size.

To include the DEBUG facility in your BASIC-PLUS Run-Time System, respond "Y/D" to SYSGEN's "Generate BASIC-PLUS?" query.

THE BASIC-PLUS DUMP FACILITY

A new feature of BASIC-PLUS, the DUMP facility, is available with RSTS/E V7.Ø as an unsupported feature, and may be changed or removed in future versions of BASIC-PLUS.

The keyword D?MP has been implemented to write a post-mortem dump to the specified file on a random-access device if the patch described in article Seq. 4.1Ø.4 F is installed. To change the keyword to DUMP, also install the patch described in article Seq. 4.1Ø.6 F. Note that, if this patch is installed, any program with a variable or function named DUMP, DUMP%, DUMP$, FNDUMP(), etc. will not compile until the name of the variable is changed.

For the purposes of this discussion, assume that the patch described above has been installed. If you enable the DUMP facility, but choose not to install this patch, replace all occurrences of "DUMP" with "D?MP" in the discussion below.

The DUMP command provides an extension to the present BASIC-PLUS immediate mode debugging commands by allowing you to take a "snap-shot" dump of your current job. After DUMPing your program, you can use the BPDA (BASIC-PLUS Dump Analyzer) program to display the contents of all variables used by the program. In addition, you can continue executing your program after issuing the DUMP command; this allows you to inspect the state of the program at several stages of execution.

The DUMP command is only available as an immediate mode command (i.e., it cannot be used in a BASIC-PLUS program), and has the syntax

DUMP <file-specification>

If no device name is specified, the public disk structure is used. If no filename is specified, the current program name is used. If no extension is specified, ".PMD" is used.

THE BASIC-PLUS DUMP ANALYSIS PROGRAM

After the DUMP command has been issued, you can use the BPDA program to print the contents of each variable that is used in the program, as well as the contents of the buffer of any open files.

The BPDA program asks for an input file name. The default extension is ".PMD". There is no default for the file name. Wildcards are illegal.

When BPDA asks for the output file name, responding with <cr> will direct the output to your terminal. If no filename is specified, the input filename is used with the default extension ".PDA".

You may also respond to the "Input file?" query with a command of the form

<outfile> = <infile>

The DUMP facility has not been implemented as a standard feature because it could be used to obtain confidential information, such as student grades or payroll information, in applications where a user could type CTRL/C and examine the contents of buffers and variables. If the feature is enabled, however, there are two mechanisms available to protect against this kind of situation.

1. Non-privileged users are normally prevented from using the DUMP command on programs that are running from compiled files. This restriction may be removed if desired (see article Seq 4.1Ø.5 F).

2. Whether or not non-privileged users are allowed to use the DUMP command on programs that are executed from compiled files, they still may not use the DUMP command on a program that uses temporary privileges, since the BASIC-PLUS Run-Time System clears the program from memory before a non-privileged user returns to keyboard monitor state.

Some installations may find it desirable to generate two versions of the BASIC-PLUS Run-Time System, one for development and one for production. This would prevent the potential problems described above while providing the DUMP feature for development work.

SUPPORT POLICY FOR THE BASIC-PLUS DUMP FACILITY

DIGITAL makes no commitment, expressed or implied, to support the BASIC-PLUS Run-Time System, or any BASIC-PLUS System programs (CUSPs) used with that Run-Time System, if the DUMP facility is enabled.

If you experience problems with the DUMP feature, please submit an FYI-type Software Performance Report (SPR). While DIGITAL makes no commitment to fix problems reported with the DUMP facility, we would like to know about any problems that you encounter.

RSTS/E V7.Ø
BASIC-PLUS
Non-standard BASIC-PLUS Features

Seq 4.1Ø.3 F

CHANGING THE BASIC-PLUS DEBUG KEYWORDS - BASIC-PLUS FEATURE PATCH

PROBLEM:

If support for the DEBUG facility has been included in the BASIC-PLUS Run-Time System, the keywords BREAK, TRACE, UNBREAK, and UNTRACE are initially enabled with the second character of the keyword changed to a question mark ("?"), i.e., B?EAK, T?ACE, U?BREAK, and U?TRACE.

SOLUTION:

The patching procedure detailed below will replace the question marks with the correct characters, thus changing the keywords to BREAK, TRACE, UNBREAK, and UNTRACE.

NOTE

> If this patch is installed, any program with a variable or function name of one of these keywords, such as BREAK% or FNTRACE$, will fail to compile until the variable name has been changed.

PROCEDURE:

1. This is a feature patch to the BASIC-PLUS Run-Time System. It may be installed in any BASIC-PLUS Run-Time System

2. The patch described in Step 4 below can be installed using the PATCH option of INIT.SYS:

```
Option: PATCH
File to patch? BASIC.RTS                (BASIC-PLUS Run-Time System name)
```

3. This patch can be installed manually using ONLPAT, the on-line patching program:

```
RUN $ONLPAT
Command File Name? <cr>                 (RETURN for manual patch installation)
File to patch? [Ø,1]BASIC.RTS           (BASIC-PLUS Run-Time System name)
File found in account [Ø,1]
```

   This patch is contained in a patch file appearing in patch kit version "A" or later.

4. The patch is as follows:

```
Base address? $$Ø41Ø
Offset address? Ø
 Base   Offset  Old     New?
??????  ØØØØØØ  ??????  ? Q!1
??????  ØØØØØ2  ??????  ? ^Z          (CTRL/Z for new offset)
Offset address? ^Z                     (CTRL/Z for new base)
Base address? ..BEA.
Offset address? Ø
```

```
 Base   Offset  Old     New?
??????  ØØØØØØ     Ø77  ? 'R
??????  ØØØØØ1     ???  ? ^Z          (CTRL/Z for new offset)
Offset address? ^Z                    (CTRL/Z for new base)
Base address? ..TAC.
Offset address? Ø
 Base   Offset  Old     New?
??????  ØØØØØØ     Ø77  ? 'R
??????  ØØØØØ1     ???  ? ^Z          (CTRL/Z for new offset)
Offset address? ^Z                    (CTRL/Z for new base)
Base address? ..UBR.
Offset address? Ø
 Base   Offset  Old     New?
??????  ØØØØØØ     Ø77  ? 'N
??????  ØØØØØ1     ???  ? ^Z          (CTRL/Z for new offset)
Offset address? ^Z                    (CTRL/Z for new base)
Base address? ..UTR.
Offset address? Ø
 Base   Offset  Old     New?
??????  ØØØØØØ     Ø77  ? 'N
??????  ØØØØØ1     ???  ? ^C          (up-arrow/C to exit;CTRL/C for INIT)
```

5. If the above patch was installed using ONLPAT, it will take effect the next time the run-time system is reloaded. If the run-time system has been ADDed, or if it is your system default run-time system, execute the following commands:

```
RUN $UTILTY<cr>
<UTILTY's header line>
#UNLOAD BASIC<cr>                     (BASIC-PLUS Run-Time System name)
#EXIT<cr>

Ready
```

NOTE: The UNLOAD command will not remove the run-time system, but simply instructs the monitor to reload it the next time a job requests it.

If, in addition to being previously ADDed, the run-time system had been ADDed with the /STAY switch, also reload it as follows:

```
RUN $UTILTY<cr>
<UTILTY's header line>
#LOAD BASIC/STAY/ADDR:nnn<cr>         (BASIC-PLUS Run-Time System name)
#EXIT<cr>

Ready
```

RSTS/E V7.Ø
BASIC-PLUS
Non-standard BASIC-PLUS Features

ENABLING THE BASIC-PLUS DUMP FEATURE - BASIC-PLUS FEATURE PATCH

PROBLEM:

The (unsupported) BASIC-PLUS "DUMP" feature is initially disabled. Some installations may choose to enable this feature, even though it is unsupported. (See article Seq. 4.1Ø.2 N for a complete description of the DUMP feature.)

SOLUTION:

The patching procedure detailed below will enable the DUMP feature.

PROCEDURE:

1. This is a feature patch to the BASIC-PLUS Run-Time System. It may be installed in any BASIC-PLUS Run-Time System

2. The patch described in Step 4 below can be installed using the PATCH option of INIT.SYS:

```
Option: PATCH
File to patch? BASIC.RTS                (BASIC-PLUS Run-Time System name)
```

3. This patch can be installed manually using ONLPAT, the on-line patching program:

```
RUN $ONLPAT
Command File Name? <cr>                 (RETURN for manual patch installation)
File to patch? [Ø,1]BASIC.RTS           (BASIC-PLUS Run-Time System name)
File found in account [Ø,1]
```

   This patch is contained in a patch file appearing in patch kit version "A" or later.

4. The patch is as follows:

```
Base address? $$Ø41Ø
Offset address? Ø
 Base    Offset  Old     New?
??????  ØØØØØØ  ??????  ? Q!2
??????  ØØØØØ2  ??????  ? ^Z          (CTRL/Z for new offset)
Offset address? ^Z                     (CTRL/Z for new base)
Base address? ..PMD.
Offset address? Ø
 Base    Offset  Old     New?
??????  ØØØØØØ  1Ø4755  ? BNE+2
??????  ØØØØØ2  ??????  ? ^C          (up-arrow/C to exit;CTRL/C for INIT)
```

5. If the above patch was installed using ONLPAT, it will take effect the next time the run-time system is reloaded. If the run-time system has been ADDed, or if it is your system default run-time system, execute the following commands:

```
RUN $UTILTY<cr>
<UTILTY's header line>
#UNLOAD BASIC<cr>                    (BASIC-PLUS Run-Time System name)
#EXIT<cr>

Ready
```

NOTE: The UNLOAD command will not remove the run-time system, but simply instructs the monitor to reload it the next time a job requests it.

If, in addition to being previously ADDed, the run-time system had been ADDed with the /STAY switch, also reload it as follows:

```
RUN $UTILTY<cr>
<UTILTY's header line>
#LOAD BASIC/STAY/ADDR:nnn<cr>        (BASIC-PLUS Run-Time System name)
#EXIT<cr>

Ready
```

ENABLING DUMP FROM COMPILED FILES FOR NON-PRIVILEGED USERS
- BASIC-PLUS FEATURE PATCH

PROBLEM:

Normally, if the DUMP feature is enabled (see article Seq. 4.1Ø.2 N), BASIC-PLUS will not allow non-privileged users to DUMP programs that are executing from compiled files, even though they may DUMP programs that are executing from source files. This protection helps to prevent unscrupulous users from DUMPing programs which may contain confidential information in their variable strings or file buffers.

Some installations may desire to allow all of their users to DUMP all BASIC-PLUS programs to which they have RUN access, regardless of whether the program is compiled.

SOLUTION:

The patching procedure detailed below will cause BASIC-PLUS to allow all users to DUMP programs that they are executing. (Note that a program whose protection code includes the 128 bit for temporary privileges is always cleared from a non-privileged user's job space before returning the keyboard monitor ("Ready") state. Hence, a non-privileged user can never DUMP such a program.)

PROCEDURE:

1. This is a feature patch to the BASIC-PLUS Run-Time System. It may be installed in any BASIC-PLUS Run-Time System

2. The patch described in Step 4 below can be installed using the PATCH option of INIT.SYS:

```
Option: PATCH
File to patch? BASIC.RTS                (BASIC-PLUS Run-Time System name)
```

3. This patch can be installed manually using ONLPAT, the on-line patching program:

```
RUN $ONLPAT
Command File Name? <cr>                 (RETURN for manual patch installation)
File to patch? [Ø,1]BASIC.RTS           (BASIC-PLUS Run-Time System name)
File found in account [Ø,1]
```

   This patch is contained in a patch file appearing in patch kit version "A" or later.

4. The patch is as follows:

```
Base address? $$Ø41Ø
Offset address? Ø
 Base    Offset  Old      New?
??????  ØØØØØØ  ??????  ? Q!4
??????  ØØØØØ2  ??????  ? ^Z          (CTRL/Z for new offset)
Offset address? ^Z                    (CTRL/Z for new base)
Base address? ..NPD.
Offset address? Ø
 Base    Offset  Old      New?
??????  ØØØØØØ  ØØ4767  ? NOP
??????  ØØØØØ2  ??????  ? NOP
??????  ØØØØØ4  ØØ5Ø46  ? ^C          (up-arrow/C to exit;CTRL/C for INIT)
```

5. If the above patch was installed using ONLPAT, it will take effect the next time the run-time system is reloaded. If the run-time system has been ADDed, or if it is your system default run-time system, execute the following commands:

```
RUN $UTILTY<cr>
<UTILTY's header line>
#UNLOAD BASIC<cr>                     (BASIC-PLUS Run-Time System name)
#EXIT<cr>

Ready
```

NOTE: The UNLOAD command will not remove the run-time system, but simply instructs the monitor to reload it the next time a job requests it.

If, in addition to being previously ADDed, the run-time system had been ADDed with the /STAY switch, also reload it as follows:

```
RUN $UTILTY<cr>
<UTILTY's header line>
#LOAD BASIC/STAY/ADDR:nnn<cr>         (BASIC-PLUS Run-Time System name)
#EXIT<cr>

Ready
```

RSTS/E V7.Ø
BASIC-PLUS
Non-standard BASIC-PLUS Features

Seq 4.1Ø.6 F
Page 1 of 2

CHANGING THE BASIC-PLUS D?MP KEYWORD - BASIC-PLUS FEATURE PATCH

PROBLEM:

The DUMP command, which is disabled unless the patch described in article Seq. 4.1Ø.4 F is installed, has been implemented with the keyword "D?MP." Some installations may desire to change this keyword to "DUMP."

SOLUTION:

The patching procedure detailed below will change the keyword "D?MP" to "DUMP." If this patch is installed, note that any program which uses a variable or function name such as DUMP% or FNDUMP$, will not compile until the variable name is changed.

PROCEDURE:

1. This is a feature patch to the BASIC-PLUS Run-Time System. It may be installed in any BASIC-PLUS Run-Time System

2. The patch described in Step 4 below can be installed using the PATCH option of INIT.SYS:

```
Option: PATCH
File to patch? BASIC.RTS                (BASIC-PLUS Run-Time System name)
```

3. This patch can be installed manually using ONLPAT, the on-line patching program:

```
RUN $ONLPAT
Command File Name? <cr>                 (RETURN for manual patch installation)
File to patch? [Ø,1]BASIC.RTS           (BASIC-PLUS Run-Time System name)
File found in account [Ø,1]
```

   This patch is contained in a patch file appearing in patch kit version "A" or later.

4. The patch is as follows:

```
Base address? $$Ø41Ø
Offset address? Ø
 Base   Offset  Old     New?
??????  ØØØØØØ  ??????  ? Q!1Ø
??????  ØØØØØ2  ??????  ? ^Z           (CTRL/Z for new offset)
Offset address? ^Z                      (CTRL/Z for new base)
Base address? ..DMP.
Offset address? Ø
 Base   Offset  Old     New?
??????  ØØØØØØ     Ø77  ? 'U
??????  ØØØØØ1     ???  ? ^C           (up-arrow/C to exit;CTRL/C for INIT)
```

5. If the above patch was installed using ONLPAT, it will take effect the next time the run-time system is reloaded. If the run-time system has been ADDed, or if it is your system default run-time system, execute the following commands:

```
RUN $UTILTY<cr>
<UTILTY's header line>
#UNLOAD BASIC<cr>                    (BASIC-PLUS Run-Time System name)
#EXIT<cr>

Ready
```

NOTE: The UNLOAD command will not remove the run-time system, but simply instructs the monitor to reload it the next time a job requests it.

If, in addition to being previously ADDed, the run-time system had been ADDed with the /STAY switch, also reload it as follows:

```
RUN $UTILTY<cr>
<UTILTY's header line>
#LOAD BASIC/STAY/ADDR:nnn<cr>        (BASIC-PLUS Run-Time System name)
#EXIT<cr>

Ready
```

RSTS/E 278Ø Device Driver
for RSTS/E V7.Ø
278Ø Device Driver Patches

BUFFER LIMIT CHECKING - MANDATORY 278Ø DEVICE DRIVER PATCH

PROBLEM:

The RSTS/E extended buffering scheme in effect since VØ6B obsoletes the buffer limit checking code in the 278Ø device driver. This buffer limit checking code was removed from the DU11/DUP11 version of the driver, but was left in the DP11 version. Certain valid buffer addresses will cause the DP11 version to fail.

SOLUTION:

This patch eliminates the buffer checking code from the DP11 version of the 278Ø device driver. It should only be installed if you are using the DP11 version of the driver.

PROCEDURE:

1. This is a required patch to the RSTS/E 278Ø Device Driver. It must be installed in all target monitor SILs configured with the DP11 version of 278Ø Device Driver.

2. The patch described in Step 4 below can be installed using the PATCH option of INIT.SYS:

```
Option: PATCH
File to patch? <lf>                    (LINE FEED for installed monitor SIL)
```

3. This patch can be installed manually using ONLPAT, the on-line patching program:

```
RUN $ONLPAT
Command File Name? <cr>                (RETURN for manual patch installation)
File to patch? <lf>                    (LINE FEED for installed monitor SIL)
File found in account [Ø,1]
```

   For automated patching, the patch is contained in a command file ($RJ278Ø.CMD) appearing in patch kit version "A" or later.

4. The patch is as follows:

```
Module name? RJ278Ø
Base address? RJDV3
Offset address? 576
 Base    Offset  Old        New?
135636  ØØØ576  Ø2Ø467   ? 24Ø
135636  ØØØ6ØØ  ??????   ? 24Ø
135636  ØØØ6Ø2  1Ø1Ø33   ? 24Ø
135636  ØØØ6Ø4  Ø1Ø163   ? ^Z            (CTRL/Z for new offset)
Offset address? 622
 Base    Offset  Old        New?
135636  ØØØ622  Ø2Ø467   ? 24Ø
135636  ØØØ624  ??????   ? 24Ø
135636  ØØØ626  1Ø1Ø21   ? 24Ø
135636  ØØØ63Ø  ØØØ767   ? ^C            (up-arrow/C to exit;CTRL/C for INIT)
```

4. The patch is as follows:

```
Module name? RJ278Ø
Base address? RJDV3
Offset address? 1Ø7Ø
 Base   Offset  Old       New?
135636  ØØ1Ø7Ø  Ø42713  ? 4737
135636  ØØ1Ø72  ØØØØØ2  ? RJEPAT
135636  ØØ1Ø74  ØØØ753  ? ^Z            (CTRL/Z for new offset)
Offset address? ^Z                     (CTRL/Z for new base)
Base address? RJEPAT
Offset address? Ø
 Base   Offset  Old       New?
137340  ØØØØØØ  ØØØØØØ  ? 42713
137340  ØØØØØ2  ØØØØØØ  ? 2
137340  ØØØØØ4  ØØØØØØ  ? 52763
137340  ØØØØØ6  ØØØØØØ  ? 4ØØ
137340  ØØØØ1Ø  ØØØØØØ  ? 4
137340  ØØØØ12  ØØØØØØ  ? 2Ø7
137340  ØØØØ14  ??????  ? ^C          (up-arrow/C to exit;CTRL/C for INIT)
```

DISCONNECT COMMAND HANDLING - MANDATORY 278Ø DEVICE DRIVER PATCH

PROBLEM:

If the 278Ø device driver receives a disconnect command (DLE EOT), it mistakenly releases its buffers and line table. A subsequent close will cause modification of critical monitor memory. A system crash may occur thereafter.

SOLUTION:

This patch eliminates the problem.

PROCEDURE:

1. This is a required patch to the RSTS/E 278Ø Device Driver. It must be installed in all target monitor SILs configured with the 278Ø Device Driver.

2. The patch described in Step 4 below can be installed using the PATCH option of INIT.SYS:

```
Option: PATCH
File to patch? <lf>                    (LINE FEED for installed monitor SIL)
```

3. This patch can be installed manually using ONLPAT, the on-line patching program:

```
RUN $ONLPAT
Command File Name? <cr>                (RETURN for manual patch installation)
File to patch? <lf>                    (LINE FEED for installed monitor SIL)
File found in account [Ø,1]
```

   For automated patching, the patch is contained in a command file ($RJ278Ø.CMD) appearing in patch kit version "A" or later.

4. The patch is as follows:

```
Module name? RJ278Ø
Base address? RJDVØ
Offset address? 4266
 Base    Offset  Old       New?
12ØØØØ  ØØ4266  ØØ4767  ? 24Ø
12ØØØØ  ØØ427Ø  174214Ø  ? 24Ø
12ØØØØ  ØØ4272  112777  ? ^C          (up-arrow/C to exit;CTRL/C for INIT)
```

DP11 TRANSMIT HANG CONDITION - MANDATORY 278Ø DEVICE DRIVER PATCH

PROBLEM:

Under certain load conditions, when the 278Ø package issues the error 'Send Error - Transmit Hang - Redial Required', critical locations in the monitor can be modified. This will cause a subsequent system crash.

SOLUTION:

This patch fixes the problem for the DP11 version of the 278Ø Device Driver. It should only be installed if you are using the DP11 version of the driver. Article 5.1.4 M, published in these Release Notes, describes the procedure for correcting this problem on a system configured with the DU11/DUP11 version of the 278Ø Device Driver.

PROCEDURE:

1. This is a required patch to the RSTS/E 278Ø Device Driver. It must be installed in all target monitor SILs configured with the DP11 version of the 278Ø Device Driver.

2. The patch described in Step 4 below can be installed using the PATCH option of INIT.SYS:

```
Option: PATCH
File to patch? <lf>                    (LINE FEED for installed monitor SIL)
File found in account [Ø,1]
```

3. This patch can be installed manually using ONLPAT, the on-line patching program:

```
RUN $ONLPAT
Command File Name? <cr>                (RETURN for manual patch installation)
File to patch? <lf>                    (LINE FEED for installed monitor SIL)
```

   For automated patching, the patch is contained in a command file ($RJ278Ø.CMD) appearing in patch kit version "A" or later.

4. The patch is as follows:

```
Module name? RJ278Ø
Base address? RJDV3
Offset address? 764
 Base    Offset  Old       New?
135636  ØØØ764  Ø42715  ? 5Ø15
135636  ØØØ766  ØØØØØ1  ? 5Ø65
135636  ØØØ77Ø  Ø4276Ø  ? 177774
135636  ØØØ772  ØØ41ØØ  ? 24Ø
135636  ØØØ774  177774  ? 24Ø
135636  ØØØ776  ØØØ763  ? ^C           (up-arrow/C to exit;CTRL/C for INIT)
```

RSTS/E 278Ø Device Driver
for RSTS/E V7.Ø
278Ø Device Driver Patches

DU11/DUP11 TRANSMIT HANG CONDITION - MANDATORY 278Ø DEVICE DRIVER PATCH

PROBLEM:

Under certain load conditions, when the 278Ø package issues the error 'Send Error - Transmit Hang - Redial Required', critical locations in the monitor can be modified. This will cause a subsequent system crash.

SOLUTION:

This patch fixes the problem for the DU11/DUP11 version of the 278Ø Device Driver. It should only be installed if you are using the DU11/DUP11 version of the driver. Article 5.1.3 M, published in these Release Notes, describes the procedure for correcting this problem on a system configured with the DP11 version of the 278Ø Device Driver.

PROCEDURE:

1. This is a required patch to the RSTS/E 278Ø Device Driver. It must be installed in all target monitor SILs configured with the DU11/DUP11 version of the 278Ø Device Driver.

2. The patch described in Step 4 below can be installed using the PATCH option of INIT.SYS:

```
Option: PATCH
File to patch? <lf>                    (LINE FEED for installed monitor SIL)
File found in account [Ø,1]
```

3. This patch can be installed manually using ONLPAT, the on-line patching program:

```
RUN $ONLPAT
Command File Name? <cr>                (RETURN for manual patch installation)
File to patch? <lf>                    (LINE FEED for installed monitor SIL)
```

   For automated patching, the patch is contained in a command file ($RJ278Ø.CMD) appearing in patch kit version "A" or later.

RSTS/E 278Ø Device Driver
for RSTS/E V7.Ø
278Ø Device Driver Patches

LOSS OR DUPLICATION OF DATA - MANDATORY 278Ø DEVICE DRIVER PATCH

PROBLEM:

Occasionally the loss or duplication of data occurs. There is no indication from the 278Ø package that any trouble had occurred.

SOLUTION:

This patch fixes the 278Ø Device Driver's handling of buffer contents.

PROCEDURE:

1. This is a required patch to the RSTS/E 278Ø Device Driver. It must be installed in all target monitor SILs configured with the 278Ø Device Driver.

2. The patch described in Step 4 below can be installed using the PATCH option of INIT.SYS:

```
Option: PATCH
File to patch? <lf>                    (LINE FEED for installed monitor SIL)
```

3. This patch can be installed manually using ONLPAT, the on-line patching program:

```
RUN $ONLPAT
Command File Name? <cr>                (RETURN for manual patch installation)
File to patch? <lf>                    (LINE FEED for installed monitor SIL)
File found in account [Ø,1]
```

For automated patching, the patch is contained in a command file ($RJ278Ø.CMD) appearing in patch kit version "A" or later.

4. The patch is as follows:

```
Module name? RJ278Ø
Base address? RJDV2
Offset address? 1522
 Base   Offset  Old       New?
126510  ØØ1522  ØØ5ØØØ  ? 4737
126510  ØØ1524  152ØØØ  ? RJEPAT+14
126510  ØØ1526  ØØ4771  ? ^Z              (CTRL/Z for new offset)
Offset address? 4776
 Base   Offset  Old       New?
126510  ØØ4776  Ø16767  ? 4537
126510  ØØ5ØØØ  ??????  ? RJEPAT+4Ø
126510  ØØ5ØØ2  ??????  ? 172354
126510  ØØ5ØØ4  11Ø477  ? 172354
126510  ØØ5ØØ6  ??????  ? 172354
126510  ØØ5Ø1Ø  ØØ5267  ? ^Z              (CTRL/Z for new offset)
```

```
Offset address? 7Ø72
 Base   Offset  Old       New?
126510  ØØ7Ø72  Ø10446  ? 4437
126510  ØØ7Ø74  Ø1Ø346  ? RJEPAT+56
126510  ØØ7Ø76  Ø1Ø5Ø4  ? ^Z              (CTRL/Z for new offset)
Offset address? 7122
 Base   Offset  Old       New?
126510  ØØ7122  Ø126Ø4  ? 137
126510  ØØ7124  ØØØ2Ø7  ? RJEPAT+66
126510  ØØ7126  ??????  ? ^Z              (CTRL/Z for new offset)
Offset address? ^Z                       (CTRL/Z for new base)
Base address? RJEPAT
Offset address? 14
 Base   Offset  Old       New?
??????  ØØØØ14  ØØØØØØ  ? 13746
??????  ØØØØ16  ØØØØØØ  ? 172354
??????  ØØØØ2Ø  ØØØØØØ  ? 13737
??????  ØØØØ22  ØØØØØØ  ? SAVR6
??????  ØØØØ24  ØØØØØØ  ? 172354
??????  ØØØØ26  ØØØØØØ  ? 5ØØØ
??????  ØØØØ3Ø  ØØØØØØ  ? 152500
??????  ØØØØ32  ØØØØØØ  ? 12637
??????  ØØØØ34  ØØØØØØ  ? 172354
??????  ØØØØ36  ØØØØØØ  ? 2Ø7
??????  ØØØØ4Ø  ØØØØØØ  ? 13546
??????  ØØØØ42  ØØØØØØ  ? 13735
??????  ØØØØ44  ØØØØØØ  ? SAVR6
??????  ØØØØ46  ØØØØØØ  ? 11Ø477
??????  ØØØØ5Ø  ØØØØØØ  ? SAVADD-.-2
??????  ØØØØ52  ØØØØØØ  ? 12635
??????  ØØØØ54  ØØØØØØ  ? 2Ø5
??????  ØØØØ56  ØØØØØØ  ? 13746
??????  ØØØØ6Ø  ØØØØØØ  ? 172354
??????  ØØØØ62  ØØØØØØ  ? 1Ø346
??????  ØØØØ64  ØØØØØØ  ? 114
??????  ØØØØ66  ØØØØØØ  ? 12637
??????  ØØØ7Ø  ØØØØØØ  ? 172354
??????  ØØØ72  ØØØØØØ  ? 126Ø4
??????  ØØØ74  ØØØØØØ  ? 2Ø7
??????  ØØØ76  ??????  ? ^C            (up-arrow/C to exit;CTRL/C for INIT)
```

DOUBLE CLOSE ALTERS MONITOR - MANDATORY 278Ø DEVICE DRIVER PATCH

PROBLEM:

The 278Ø Device Driver does not protect itself against closing the RJ: device twice in all cases. A double close can alter monitor memory and cause a system crash.

SOLUTION:

This patch adds the necessary protection against a double close.

PROCEDURE:

1. This is a required patch to the RSTS/E 278Ø Device Driver. It must be installed in all target monitor SILs configured with the 278Ø Device Driver.

2. The patch described in Step 4 below can be installed using the PATCH option of INIT.SYS:

   ```
   Option: PATCH
   File to patch? <lf>                    (LINE FEED for installed monitor SIL)
   ```

3. This patch can be installed manually using ONLPAT, the on-line patching program:

   ```
   RUN $ONLPAT
   Command File Name? <cr>                (RETURN for manual patch installation)
   File to patch? <lf>                    (LINE FEED for installed monitor SIL)
   File found in account [Ø,1]
   ```

   For automated patching, the patch is contained in a command file ($RJ278Ø.CMD) appearing in patch kit version "A" or later.

4. The patch is as follows:

```
Module name? RJ278Ø
Base address? RJDVØ
Offset address? 352
 Base   Offset  Old       New?
12ØØØØ  ØØØ352  ØØ4767  ? 4737
12ØØØØ  ØØØ354  ØØØ124  ? RJEPAT+76
12ØØØØ  ØØØ356  ØØ4567  ? ^Z            (CTRL/Z for new offset)
Offset address? 37Ø
 Base   Offset  Old       New?
12ØØØØ  ØØØ37Ø  ØØ4767  ? 4737
12ØØØØ  ØØØ372  ØØØ1Ø6  ? RJEPAT+76
12ØØØØ  ØØØ374  ØØ4567  ? ^Z            (CTRL/Z for new offset)
Offset address? 5Ø6
 Base   Offset  Old       New?
12ØØØØ  ØØØ5Ø6  112767  ? 4737
12ØØØØ  ØØØ51Ø  ØØØ377  ? RJEPAT+116
12ØØØØ  ØØØ512  ??????  ? 1Ø3447
12ØØØØ  ØØØ514  112767  ? ^Z            (CTRL/Z for new offset)
Offset address? ^Z                     (CTRL/Z for new base)
Base address? RJEPAT
Offset address? 76
 Base   Offset  Old       New?
??????  ØØØØ76  ØØØØØØ  ? 5737
??????  ØØØ1ØØ  ØØØØØØ  ? C.STAX
??????  ØØØ1Ø2  ØØØØØØ  ? 14Ø4
??????  ØØØ1Ø4  ØØØØØØ  ? 1Ø137
??????  ØØØ1Ø6  ØØØØØØ  ? CLOSD
??????  ØØØ11Ø  ØØØØØØ  ? 4737
??????  ØØØ112  ØØØØØØ  ? CLS$RJ
??????  ØØØ114  ØØØØØØ  ? 2Ø7
??????  ØØØ116  ØØØØØØ  ? 5737
??????  ØØØ12Ø  ØØØØØØ  ? CLOSD
??????  ØØØ122  ØØØØØØ  ? 1ØØ2
??????  ØØØ124  ØØØØØØ  ? 261
??????  ØØØ126  ØØØØØØ  ? 2Ø7
??????  ØØØ13Ø  ØØØØØØ  ? 112737
??????  ØØØ132  ØØØØØØ  ? 377
??????  ØØØ134  ØØØØØØ  ? GTSEN
??????  ØØØ136  ØØØØØØ  ? 241
??????  ØØØ14Ø  ØØØØØØ  ? 2Ø7
??????  ØØØ142  ??????  ? ^C          (up-arrow/C to exit;CTRL/C for INIT)
```

GET/PUT INTERLOCK PROBLEM - MANDATORY 278Ø DEVICE DRIVER PATCH

PROBLEM:

Due to timing considerations on systems with zero turn-around delay, it is possible for a bid from a remote unit to be processed before the completion of the routine for End Of Transmission. This causes the driver to receive the bid when it "thinks" it is in transmit mode and the GET/PUT INTERLOCK error is triggered.

SOLUTION:

The patching procedure detailed below corrects the above problem by allowing a bid to be processed, even if the End Of Transmission routine is not yet done.

PROCEDURE:

1. This is a required patch to the RSTS/E 278Ø Device Driver. It must be installed in all target monitor SILs configured with the 278Ø Device Driver.

2. The patch described in Step 4 below can be installed using the PATCH option after bootstrapping your system disk:

```
Option: PATCH
File to patch? <lf>                    (LINE FEED for installed monitor SIL)
```

3. This patch can be installed manually using ONLPAT, the on-line patching program:

```
RUN $ONLPAT
Command file name? <cr>                (RETURN for manual patch installation)
File to patch? <lf>                    (LINE FEED for installed monitor SIL)
File found in account [Ø,1]
```

   For automated patching, the patch is contained in a command file ($RJ278Ø.CMD) appearing in patch kit version "A" or later.

4. The patch is as follows:

```
Module name? RJ278Ø
Base address? RJDVØ
Offset address? 2714
 Base   Offset  Old     New?
12ØØØØ  ØØ2714  1Ø5767  ? 137
12ØØØØ  ØØ2716  ??????  ? RJEPAT+142
12ØØØØ  ØØ272Ø  ØØ1733  ? 24Ø
12ØØØØ  ØØ2722  1Ø5767  ? ^Z            (CTRL/Z for new offset)
Offset address? ^Z                      (CTRL/Z for new base)
Base address? RJEPAT
Offset address? 142
 Base   Offset  Old     New?
??????  ØØØ142  ØØØØØØ  ? 1Ø5737
??????  ØØØ144  ØØØØØØ  ? RDWRT
??????  ØØØ146  ØØØØØØ  ? 14Ø1
??????  ØØØ15Ø  ØØØØØØ  ? 4Ø6
??????  ØØØ152  ØØØØØØ  ? 22737
??????  ØØØ154  ØØØØØØ  ? EOTMSG
??????  ØØØ156  ØØØØØØ  ? PTDBF+2
??????  ØØØ16Ø  ØØØØØØ  ? 14Ø2
??????  ØØØ162  ØØØØØØ  ? 137
??????  ØØØ164  ØØØØØØ  ? RJDVØ+261Ø
??????  ØØØ166  ØØØØØØ  ? 137
??????  ØØØ17Ø  ØØØØØØ  ? RJDVØ+2722
??????  ØØØ172  ??????  ? ^C           (up-arrow/C to exit; CTRL/C for INIT)
```

278Ø HANDLER FAILURE PROBLEM - MANDATORY 278Ø DEVICE DRIVER PATCH

PROBLEM:

When the RSTS/E system is under heavy load, it is possible for the notification of the receipt of a bid to be passed back with a receive buffer address of zero. When the driver attempts to requeue the buffer, it is considered a fatal error.

SOLUTION:

The patching procedure detailed below corrects the above problem by making the driver wait for another bid when this condition occurs.

PROCEDURE:

1. This is a required patch to the RSTS/E 278Ø Device Driver. It must be installed in all target monitor SILs configured with the 278Ø Device Driver.

2. The patch described in Step 4 below can be installed using the PATCH option after bootstrapping your system disk:

   ```
   Option: PATCH
   File to patch? <lf>                    (LINE FEED for installed monitor SIL)
   ```

3. This patch can be installed manually using ONLPAT, the on-line patching program:

   ```
   RUN $ONLPAT
   Command file name? <cr>                (RETURN for manual patch installation)
   File to patch? <lf>                    (LINE FEED for installed monitor SIL)
   File found in account [Ø,1]
   ```

   For automated patching, the patch is contained in a command file ($RJ278Ø.CMD) appearing in patch kit version "A" or later.

4. The patch is as follows:

```
Module name? RJ278Ø
Base address? RJDVØ
Offset address? 31Ø2
 Base   Offset  Old     New?
12ØØØØ  ØØ31Ø2  Ø16Ø67  ? 137
12ØØØØ  ØØ31Ø4  ØØØØØ2  ? RJEPAT+172
12ØØØØ  ØØ31Ø6  ??????  ? 24Ø
12ØØØØ  ØØ311Ø  112767  ? ^Z            (CTRL/Z for new offset)
Offset address? ^Z                      (CTRL/Z for new base)
Base address? RJEPAT
Offset address? 172
 Base   Offset  Old     New?
??????  ØØØ172  ØØØØØØ  ? 16Ø37
??????  ØØØ174  ØØØØØØ  ? 2
??????  ØØØ176  ØØØØØØ  ? ASNBF+2
??????  ØØØ2ØØ  ØØØØØØ  ? 14Ø2
??????  ØØØ2Ø2  ØØØØØØ  ? 137
??????  ØØØ2Ø4  ØØØØØØ  ? RJDVØ+311Ø
??????  ØØØ2Ø6  ØØØØØØ  ? 137
??????  ØØØ21Ø  ØØØØØØ  ? RJDVØ+3Ø12
??????  ØØØ212  ??????  ? ^C          (up-arrow/C to exit; CTRL/C for INIT)
```

RSTS/E 278Ø Device Driver
for RSTS/E V7.Ø
278Ø Device Driver Patches

LOCAL BINARY TRANSMIT PROBLEM - MANDATORY 278Ø DEVICE DRIVER PATCH

PROBLEM:

If the default transmit command is 278Ø or GEN, it should be possible to send, as part of a multiple file transfer, a file or files in binary mode. This is done by using the "/B" switch to override the transmit command for those files which should be sent untranslated. Currently, the mode of the first file controls the mode for the complete transfer.

SOLUTION:

The patching procedure detailed below corrects the above problem with the processing of the "/B" switch.

NOTE

The mandatory RJ278Ø.BAS program patch described in article Seq 25.2.2 M, published in these notes, must be installed along with this patch.

PROCEDURE:

1. This is a required patch to the RSTS/E 278Ø Device Driver. It must be installed in all target monitor SILs configured with the 278Ø Device Driver.

2. The patch described in Step 4 below can be installed using the PATCH option after bootstrapping your system disk:

```
Option: PATCH
File to patch? <lf>                    (LINE FEED for installed monitor SIL)
```

3. This patch can be installed manually using ONLPAT, the on-line patching program:

```
RUN $ONLPAT
Command file name? <cr>                (RETURN for manual patch installation)
File to patch? <lf>                    (LINE FEED for installed monitor SIL)
File found in account [Ø,1]
```

For automated patching, the patch is contained in a command file ($RJ278Ø.CMD) appearing in patch kit version "A" or later.

4. The patch is as follows:

```
Module name? RJ278Ø
Base address? RJDVØ
Offset address? 1654
 Base    Offset  Old      New?
12ØØØØ  ØØ1654  ØØ4767  ? 137
12ØØØØ  ØØ1656  ØØØ622  ? RJEPAT+212
12ØØØØ  ØØ166Ø  1ØØ413  ? ^Z              (CTRL/Z for new offset)
Offset address? ^Z                        (CTRL/Z for new base)
Base address? RJEPAT
Offset address? 212
 Base    Offset  Old      New?
??????  ØØØ212  ØØØØØØ  ? 163ØØ
??????  ØØØ214  ØØØØØØ  ? 1Ø
??????  ØØØ216  ØØØØØØ  ? 1Ø5Ø37
??????  ØØØ22Ø  ØØØØØØ  ? NATIV
??????  ØØØ222  ØØØØØØ  ? 42737
??????  ØØØ224  ØØØØØØ  ? 1Ø
??????  ØØØ226  ØØØØØØ  ? PTCPAR+2
??????  ØØØ23Ø  ØØØØØØ  ? 1Ø5237
??????  ØØØ232  ØØØØØØ  ? NATIV
??????  ØØØ234  ØØØØØØ  ? 6ØØØ
??????  ØØØ236  ØØØØØØ  ? 1Ø3ØØ5
??????  ØØØ24Ø  ØØØØØØ  ? 1Ø5Ø37
??????  ØØØ242  ØØØØØØ  ? NATIV
??????  ØØØ244  ØØØØØØ  ? 52737
??????  ØØØ246  ØØØØØØ  ? 1Ø
??????  ØØØ25Ø  ØØØØØØ  ? PTCPAR+2
??????  ØØØ252  ØØØØØØ  ? 127Ø1
??????  ØØØ254  ØØØØØØ  ? PTCPAR
??????  ØØØ256  ØØØØØØ  ? 4737
??????  ØØØ26Ø  ØØØØØØ  ? RJDVØ+42Ø
??????  ØØØ262  ØØØØØØ  ? 1ØØ4Ø4
??????  ØØØ264  ØØØØØØ  ? 4737
??????  ØØØ266  ØØØØØØ  ? RJDVØ+25Ø2
??????  ØØØ27Ø  ØØØØØØ  ? 137
??????  ØØØ272  ØØØØØØ  ? RJDVØ+166Ø
??????  ØØØ274  ØØØØØØ  ? 137
??????  ØØØ276  ØØØØØØ  ? RJDVØ+2634
??????  ØØØ3ØØ  ??????  ? ^C            (up-arrow/C to exit; CTRL/C for INIT)
```

INSTALLING AND TROUBLESHOOTING 278Ø'S

Installing a communication package involves a minimum of two vendors and more likely three or four. The most time-consuming aspect of installation can be pinpointing a problem. When installing A to communicate with B, the problem may be in one of seven areas.

A's software (either code or SYSGEN parameters)
A's hardware
A's modem
phone line
B's modem
B's hardware
B's software

Sometimes a given symptom can have several possible causes. Then it is necessary to go through a step by step elimination procedure.

Here are some ideas and suggestions which may be helpful when difficulties arise in installation of 278Ø packages.

1. What is a 278Ø and what is its protocol?

For those who are unfamiliar with what a 278Ø emulator is emulating, here is a short description of what it is and what its protocol is like.

The 278Ø is an unintelligent, hard-wired terminal which can communicate to an IBM host or to another 278Ø. It reads cards and transmits them over a synchronous link to another computer. It receives data back and prints it. The model 2 278Ø can also receive back data for a card punch.

The 278Ø sends 8Ø character card image records, blocked into a maximum of a 4ØØ character block. For reception, it prints or punchs a file according to a peripheral selection code (basically, an Escape 4 sends a file to the punch). It is able to accept horizontal tabs and handles them according to the spacing of a special HT record which is sent at the beginning of the file. The block size is again a maximum of 4ØØ characters.

A synchronous protocol is used to enable the data to be sent back and forth. The protocol determines who speaks, when, and whether or not the other end heard them correctly. When there is synchronous communication, the two modems sample the line at a preset speed (2ØØØ times a second or 48ØØ, etc.), and the data is sent in blocks with one character directly following another. Every message, therefore, whether a single control character or a 4ØØ character block is preceded by four SYNC characters and followed by a PAD. The SYNC characters are used to allow the hardware to tell when valid data is coming (as opposed to noise on the line), and the PAD is used to make sure that the final significant character is fully received. Every data record sent has a CRC (cyclic redundancy check) computed on its characters, and following each record and block (IUS, ETB, and ETX) is a two-character checksum which is used to make sure the data was received correctly.

The major control characters used in the 278Ø protocol are:

ENQ (enquiry):	(1) Used to request permission to send data (bid); (2) Used to request resending of a missed response.
ACKØ (acknowledge):	(1) Used to give permission to send data; (2) Used to accept every other data block.
ACK1 (acknowledge):	Used to accept every other data block.
NAK (negative acknowledge):	Used to reject a data block.
STX (start of text):	Used to indicate the beginning of a data block.
ETB (end of block):	Used to indicate the end of a data block.
ETX (end of text):	Normally used to indicate the end of the final block of the job.
IUS (intermediate record end):	Used to indicate the end of a record.
EOT (end of transmission):	Used to indicate completed transmission and give up control of line.

A sample job might be sent as follows:

```
A                   B
-                   -

ENQ -->                         ;request permission to send

                    <-- ACKØ        ;permission given

STX DATA1 ETB -->               ;block of data

                    <-- ACK1        ;acknowledge correct reception

STX DATA2 ETB -->               ;another block

                    <-- ACKØ

STX DATA3 ETB -->

                    <-- NAK         ;rejection of data

STX DATA3 ETB -->               ;resending of data block

                    <-- ACK1

STX DATA4 ETX -->

ENQ -->                         ;.3 second silence request for response

                    <-- ACK1        ;response shows did not see block,
                                     since an ACKØ is the expected
                                     response for DATA4

STX DATA4 ETX -->               ;block re-sent

                    <-- ACKØ

EOT -->                         ;transmission complete

                    <-- ENQ         ;other side requests permission to
                                     send data
```

2. What to check before installation of the software.

There are some questions to be asked before 278Ø installation.

What operating system will you be communicating with? If it is not another PDP-11, is it one of the usual IBM systems? If it is a different system (e.g., a Univac 11Ø8 with a Comterm front end, an RPG program running on a System 3, etc.) has a demo test been done with the system, and what problems, if any, were encountered?

What kind of modems are being used? If they are not identical, are they compatible. If they are the same type, are the options the same on both? If it is a local connection are modem eliminators being used, or are null modems being used? Null modems require a clocking option on the computer hardware to provide transmit clocking. Are the modems for a speed higher than is warranted by the software? RSX11M, RSX11D, and RSTS/E are not warranted to work over 48ØØ baud.

Make sure that the modem has been grounded on the same ground as the hardware. The communications interface and many modems can be affected by being plugged into a device which is grounded differently.

Is the DP11, DU11, or DUP11 the furthest forward device on the bus (except the system disk) which operates at hardware level 5? If there is a DH11 or DZ11 on the system, it will probably be necessary to raise the hardware level of the communications interface to 6. This is because the DH11 often is programmed to remain at interrupt level until the silo is emptied (the DZ11 has no silo and must interrupt per character).

Have the diagnostics been run for the communications interface and the KG11? The DP, DU, or DUP should be checked out with the turn-around plug at the end of the modem cable or with the modem in maintenance mode.

Two manuals are shipped with the 278Ø software. They are the 278Ø RCS Users Guide* and the 278Ø RCS Installation Notes (DEC-11-CCDNA-A-D). Relevant chapters should be read in advance of installation. There will be some overlap between this article and the manuals, but in many ways they should complement each other.

---

* DEC-11-ORJEA-B-D is for RSTS/E 278Ø only.
DEC-11-CRCSA-A-D, DN1, DN2, DN3, and DN4 are for the other 2780s.

3. Initial attempt to run

One of the more frustrating circumstances is to bring up the package, issue a transmit command, and have it do absolutely nothing. Silence, however, can be just as informative as a message.

This section will itemize the things that can go wrong in the sequence in which they would be encountered. It is specified when systems react in different ways. The systems involved are RT11, RSX11M, RSX11D, and RSTS/E.

a. Run the program for the first time and
   (1) there is a trap to 4 in RT11 and RSTS/E
   (2) there is an odd address trap abort in RSX11M or D

Probably the device address of the DU11 or DUP11 is incorrect. They are floating device address devices. Check that the device was installed at the correct address (i.e., if the DU11 is the only floating device on the system it will be at 76ØØ4Ø, if the DUP11 is the only floating device it will be at 76ØØ5Ø) and also that the system was told the same address. RSTS/E calculates the address during boot, RT11, RSX11M and D have it specified during taskbuild. The KG11 may also be missing. This is the hardware CRC calculator. If it is missing the software cannot run.

b. Run the program and get ready to make the connect to the other system. Dial the phone, receive the beep from the other end, and push (or pull) the data button and
   (1) the data light does not come on or
   (2) the DSR (or MR) indicator does not light.

Usually a dial-out modem will not give DSR (data set ready) until DTR (data terminal ready) has been presented by the communications interface

   (1) check that the light is not just burned out
   (2) check that you have put the system on-line (all systems but RSTS/E)
   (3) check that the DTR bit is on in the status word
   (4) check that a valid but incorrect device address was not used (RT11, RSX11M and D)
   (5) check that the hardware diagnostics were run, including the one using the modem cable.
   (6) check that the cable is securely plugged into the modem

With a leased line, the modem often has DSR strapped on. In that case, this type of error may not be discovered until later in the procedure (see c. and d.). There are now some dial-out modems (e.g., some MILGO modems) which have DSR strapped on also. They will react the same as the leased line modems.

c. The program has been run and the connection established; a transmit transfer command is issued and a modem (or data set) not ready message is given.
   (1) the data set is not ready (e.g., DSR is not up) or
   (2) a valid, but incorrect, device address has been used.

d. This is at the same stage as c. The first transmit command has been issued, but nothing happens. There are several possible causes.

(1) Incorrect interrupt vector (RSX11M and D). Since in these systems interrupt vector is specified at build time, check that it corresponds to the actual hardware vector.

(2) No response to the bid (RT11, RSX11M and D). These systems do an infinite retry on the bid if they get no response. (RSTS/E does a timeout). Unfortunately, this cause is really a symptom with multiple causes of its own. Some possibilities are:

(a) the remote end software is not up (usually leased line) or the connection was not made (a dial-out modem with DSR strapped on).
(b) the modems have different speeds (e.g., 2ØØØ baud talking to 24ØØ)
(c) the modem is not sending out the signal (this can usually be checked by finding out whether the other side has seen the bid)
(d) the other end is responding, but the turn-around time is too fast (usually on a two-wire dial-out system where there are echo suppressers on the line)
(e) the other end is responding but the modem either does not see it (receive strength too low) or the modem does not pass it back (bad receive leads on the modem), or our hardware does not see it (bad receive chip or modem cable - this should be caught by diagnostics or actual zero turnaround delay (see f.)).

To determine whether or not the bid is actually being sent out, check the receive end to see if it has gone into receive mode. If it is in receive mode, but the transmitting side is still in control mode, the bid is being seen but the response is not (or the response is not considered valid).

e. A transmit command is given and the program aborts. In RT11 if the wrong interrupt vector has been specified, the program is aborted and the monitor reprompts. In RSX11M if the 278Ø is running in a partition that can be 'shuffled' the program may have been moved. This will have completely unpredictable effects, since the interrupt will jump to where the correct code used to be.

f. When attempting to transmit there are frequent transmit abort errors. Each system has its own version of the error message, but they all indicate that either eight attempts to send a data block were NAKed, or eight attempts to get a valid response were ignored.

On RSTS/E VØ6B and later releases, a status request will show whether the data is being NAKed or timed-out. In the other packages an error request will show the number of NAKs. If the transmit is aborting, but there are few or no NAKs the problem is timeouts.

If the first block is not being accepted, the KG11 may not be working correctly. If it is improperly seated the program may be sending out a CRC of zeros. This should be easily caught by diagnostics.

On a multi-user system the package should not be run at a higher speed than is warranted.

On a multi-user system with a DH11 or DZ11, the communications interface should be at hardware bus request level 6.

Other possibilities are: modems with different turn-around delays, bad modems, incompatible modems, and noisy phone lines.

If there is an actual zero turn around delay (not just a modem with no delay, but a system with a front-end which is capable of responding without any processing delays) it is possible to have the response return while our hardware is sending trailing pads. Either a minimum delay time should be installed or the number of trailing pads can be decreased.

g. Transmit works fine, but files cannot be received.

Most of the suggestions for f. should be checked. Two of the most likely ones are running at too high a speed for the system, or having the communication interface at level 5 on a system with high DH11 or DZ11 load.

If no data from the other side is accepted at all it is quite possible that the other system is SYSGEN'd to support either a 378Ø or a HASP Workstation instead of a 278Ø. Our data will be acceptable to their side, since the 278Ø is, in effect, a subset of the other two. In both cases, the data block can be too long (they will both send up to 512 characters and the 278Ø can only accept 4ØØ) and the CRC calculations are done on the complete block instead of on each record. The usual indication is that we consistently NAK the very first block that they attempt to send to us.

If the other side is SYSGEN'd as a 278Ø, but is sending too long a record (over 132 data characters plus a two character escape sequence) most of our packages will now accept the record and truncate it (as long as there were no other problems with the data). The user will be informed as to the number of truncated records at the end of the reception.

h. The package has been transmitting and receiving without difficulty. It is in the middle of transmitting and all activity stops. The package is in transmit mode, but nothing is going out.

This is usually caused by losing DSR on a modem which will not give CTS (clear to send) to a RTS (request to send) if DSR has been lost (e.g., BELL 2Ø8B modems). Check the modem, and if this is correct, try to push the talk button and then the data button to try to re-establish the connection. If that doesn't work you will have to redial. RSTS/E VØ6B (and later releases) will timeout in 3Ø seconds and terminate transmit. The other packages will not terminate until operator intervention has taken place.

Installation is complete when the program has been brought up, a connection made, and files transmitted and received.

DECnet/E Package V1.Ø
for RSTS/E V7.Ø
DECnet/E Patches

ELIMINATE RACE CONDITION - MANDATORY NSP PATCH

PROBLEM:

If a network program which handles incoming CONNECTs kills itself on receiving a DISCONNECT, there is a window between the time the DISCONNECT is received and the job removes itself as a receiver. During this period, the job is an eligible receiver for another CONNECT. If NSP queues a CONNECT INITIATE in this time window and the job does a REMOVE receiver or kills itself, the CONNECT will be rejected.

SOLUTION:

The patching procedure detailed below implements a new DECLARE RECEIVER ACCESS control bit. The "one-shot" bit (value = 8), if set in the DECLARE RECEIVER call, causes NSP to set the receiver's link max to Ø after queueing one CONNECT INITIATE message. This prevents the possibility that NSP will queue a CONNECT INITIATE after its last logical link is disconnected and before it issues a REMOVE RECEIVER.

NOTE

> The mandatory DECnet/E FAL patch described in article Seq 31.11.2 M, published in this issue, must be applied along with this patch.

PROCEDURE:

1. This is a required patch to the RSTS/E V7.Ø executive. It must be installed in all target monitor SILs which include DECnet/E.

2. The patch described in Step 4 below can be installed using the PATCH option of INIT.SYS:

```
Option: PATCH
File to patch? <lf>                    (LINE FEED for installed monitor SIL)
```

3. This patch can be installed manually using ONLPAT, the on-line patching program:

```
RUN $ONLPAT
Command File Name? <cr>                (RETURN for manual patch installation)
File to patch? <lf>                    (LINE FEED for installed monitor SIL)
File found in account [Ø,1]
```

   The patch is also contained in a command file ($DECNTA.CMD) appearing in patch kit version "A" or later.

4. The patch is as follows:

```
Module name? NSP
Base address? NSP
Offset address? 12210
 Base   Offset  Old      New?
120000  012210  105261   ? 4737
120000  012212  000017   ? NSPPAT
120000  012214  016500   ? ^Z          (CTRL/Z for new offset)
Offset address? ^Z                    (CTRL/Z for new base)
Base address? NSPPAT
Offset address? 0
 Base   Offset  Old      New?
137516  000000  000000   ? 105261
137516  000002  000000   ? 17
137516  000004  000000   ? 126527
137516  000006  000000   ? 4
137516  000010  000000   ? 177776
137516  000012  000000   ? 1011
137516  000014  000000   ? 23701
137516  000016  000000   ? NSPLST
137516  000020  000000   ? 1406
137516  000022  000000   ? 132761
137516  000024  000000   ? 10
137516  000026  000000   ? 12
137516  000030  000000   ? 1402
137516  000032  000000   ? 105061
137516  000034  000000   ? 24
137516  000036  000000   ? 207
137516  000040  000000   ? ^C          (up-arrow/C to exit;CTRL/C for INIT)
```

BUILD PROCEDURE FOR DECNET/E V1.Ø ON RSTS/E V7.Ø

To BUILD DECnet/E V1.Ø on RSTS/E V7.Ø, answer the SYSGEN distribution medium question for DECnet/E with xx/A, where xx is the media. For example, answer DK/A for RKØ5 distribution of DECnet/E V1.Ø.

INSTALLING TECO VERSION 35 UNDER RSTS/E V7.Ø

The BUILD program should be used to install TECO on your system. The command file is called TECO.CTL. 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.

RSTS/E V7.Ø
System Utilities Package
BUILD Utility

THE LOCATION OF BUILD CONTROL FILES ON THE SYSTEM DISTRIBUTION MEDIA

The control files that are used to build the system library and program packages are listed below. 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 below. Note that if more than one control file is used, the files must be specified in the order listed below. For example, if you are building the Spooling and Operator Services Package, the Backup Package, and the Device Test Package, the order of the specified control files should be SPLER.CTL, BACKUP.CTL, and DEVTST.CTL.

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

RSXLBB:
RSX.CTL
SORT.CTL
TECO.CTL

SYSL1G:
BUILD.CTL
SPLER.CTL
UNSUPP.CTL

SYSL2G:
BIGPRG.CTL
BACKUP.CTL
DEVTST.CTL
HELP.CTL

RKØ6 Disk Cartridge
RLØ1 Disk Cartridge

SYSGNG:
RSX.CTL
SORT.CTL
TECO.CTL

SYSL1G:
BUILD.CTL
SPLER.CTL
UNSUPP.CTL
BIGPRG.CTL
BACKUP.CTL
DEVTST.CTL
HELP.CTL

9-Track Magnetic Tape (16ØØ BPI)
RKØ7 Disk Cartridge

SYSGNG:
RSX.CTL
BUILD.CTL
SPLER.CTL
UNSUPP.CTL
BIGPRG.CTL
BACKUP.CTL
DEVTST.CTL
HELP.CTL
RSX.CTL
SORT.CTL
TECO.CTL

The RSX.CTL file contains the commands necessary to build the RSX Run-Time System and support utilities.

The SORT.CTL file contains the commands necessary to build the PDP-11 SORT package.

The TECO.CTL file contains the commands necessary to build a number of TECO programs.

The UNSUPP.CTL file 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.

The HELP.CTL file contains the commands necessary to build the HELP system program and its associated data files. (Refer to article Seq 19.4.6 N for a description of the HELP program.)

The BUILD.CTL, BACKUP.CTL, BIGPRG.CTL, SPLER.CTL, and DEVTST.CTL files contain the necessary commands to build the standard RSTS/E system library. The exact contents of these files are described in Appendix D of the RSTS/E V7.Ø System Generation Manual.

RSTS/E V7.Ø
System Utilities Package
DIRECT

RESTRICT NON-PRIVILEGED ACCESS TO CROSS-ACCOUNT DIRECTORIES - DIRECT FEATURE PATCH

PROBLEM:

Non-privileged users are currently allowed to list the directory information of any file on the system. However, for security reasons some installations may find it desirable to restrict non-privileged users' access to those files for which they have read or run access.

SOLUTION:

Another feature patch, Seq 3.5.9 F, restricts the use of FIP calls 15 (Directory look up on Index) and 17 (Directory lookup by File Name/Wildcard directory lookup). Since DIRECT does not use these calls, a separate procedure is necessary to cause DIRECT to restrict non-privileged user's to listing directory information of files only for which they have read or run access.

PROCEDURE:

1. For purposes of this discussion, we will assume that the program to be patched is located in your privileged account on the public disk structure. If this is not the case, replace all program references with suitable text according to the requirements of your installation.

   We assume the executable version of the program will be stored in the System Library Account ($). If this is not the case, replace references to $ in step 3 below with the appropriate package or library account.

   The procedure below assumes that BASIC-PLUS is your system default run-time system.

   If you are using CSPCOM or BASIC-PLUS-2, refer to Section 8.4 of the RSTS/E V7.Ø Release Notes for the correct procedure to compile the program.

   If you are using BASIC-PLUS-2, this program should be compiled under the BASIC2 Run-Time System.

2. This patch is contained in a patch file appearing in patch kit version "A" or later. If you do not have this distribution you can produce a command file by specifying a file for [logfile=] in the procedure below.

   To apply the patch manually, perform the following RSTS/E system commands.

```
RUN $CPATCH<cr>
<CPATCH's header line>

File to patch - DIRECT.BAS=DIRECT.BAS<cr>
#[logfile=]KB:/CS:42649<cr>
*H/2!/V<cr>
2!              PROGRAM         : DIRECT.BAS
```

RSTS/E V7.Ø
System Utilities Package
DIRECT


```
*H/3ØØ<tab>/V<cr>
3ØØ     CROSS.PROTECT%=Ø% &
*G/Ø/-1C/-1/V<cr>
3ØØ     CROSS.PROTECT%=-1% &
*EX<cr>
Patch from KB:[P,PN]CPATCH.CMD complete
#^Z
File to patch - ^Z

Ready
```

3. Use this procedure if your system default Run-Time System is BASIC-PLUS; otherwise, refer to Section 8.4 of the RSTS/E V7.Ø Release Notes for the correct procedure to compile the program.

   To re-compile the program and re-enter it into the system library, type the following RSTS/E commands.

   ```
   OLD DIRECT<cr>

   Ready

   COMPILE $DIRECT<232><cr>

   Ready
   ```

4. The person making the changes to the program should now take whatever steps are necessary, according to installation guidelines, to save the new version of the program.

5. The source (.BAS) version of the program may now be removed from the public structure.

RSTS/E V7.Ø
System Utilities Package
LOGIN

UNSCRUPULOUS PROGRAMS MAY CHAIN TO LOGIN - LOGIN FEATURE PATCH

PROBLEM:

Users, particularly in an educational environment, have been known to write programs that simulate the LOGIN dialogue, store the account number and password, and then CHAIN to LOGIN. In this way, a hostile user can gain access to the accounts of other users without the knowledge (or permission) of those users.

SOLUTION:

The patching procedure detailed below will cause LOGIN to report both the original account and the new account when invoked by a logged-in job. While this will not prevent a user from writing such a program, it will allow the victimized user to report the account that was used to the system manager.

Note that the variable W$ has been defined in the patch as a null string (W$="") at line 28ØØØ. The null string can be replaced with some message such as "Please inform the System Manager" if desired.

Assume that the LOGIN-simulating program resides in account [2,219]. When the patch described below has been installed, a typical run may look like this:

```
HELLO 1,224
Password:

Old Account:        [2,219]

New Account:        [1,224]

Please inform the System Manager.


Ready
```

PROCEDURE:

1. For purposes of this discussion, we will assume that the program to be patched is located in your privileged account on the public disk structure. If this is not the case, replace all program references with suitable text according to the requirements of your installation.

   The procedure below assumes that BASIC-PLUS is your system default run-time system.

   If you are using CSPCOM or BASIC-PLUS-2, refer to Section 8.4 of the RSTS/E V7.Ø Release Notes for the correct procedure to compile the program.

   If you are using BASIC-PLUS-2, this program must be task built against the BP2COM Run-Time System.

2. This patch is contained in a patch file appearing in patch kit version "A" or later. If you do not have this distribution you can produce a command file by specifying a file for [logfile=] in the procedure below.

NOTE

The patch file for this patch requires manual editing to include installation specific parameters before it can be successfully installed. Specifically, have W$ indicate the desired message.

To apply the patch manually, perform the following RSTS/E system commands.

```
RUN $CPATCH<cr>
<CPATCH's header line>

File to patch - LOGIN.BAS=LOGIN.BAS<cr>
#[logfile=]<cr>
*H/2!/V<cr>
2!              PROGRAM         : LOGIN.BAS
*H/15ØØØ<tab>/V<cr>
15ØØØ   I%=-1% &
*15AV<cr>
        !\ GOSUB 28ØØØ IF W% &
*G/!/-1DV<cr>
        \ GOSUB 28ØØØ IF W% &
*H/28ØØØ<tab>/V<cr>
28ØØØ   PRINT &
*7AV<cr>
       \ W$="" &
*G/""/-2C/""/V<cr>              <Insert a message here, if desired.>
       \ W$="" &
*EX<cr>
Patch from KB:[P,PN]CPATCH.CMD complete
#^Z
File to patch - ^Z

Ready
```

3. Use this procedure if your system default Run-Time System is BASIC-PLUS; otherwise, refer to Section 8.4 of the RSTS/E V7.Ø Release Notes for the correct procedure to compile the program.

   To re-compile the program and re-enter it into the system library, type the following RSTS/E commands.

```
OLD LOGIN<cr>

Ready

COMPILE SYØ:$LOGIN<232><cr>

Ready
```

4. The person making the changes to the program should now take whatever steps are necessary, according to installation guidelines, to save the new version of the program.

5. The source (.BAS) version of the program may now be removed from the public structure.

RSTS/E V7.Ø
System Utilities Package
LOGIN

MAKING LOGIN HANG UP A DATASET - LOGIN FEATURE PATCH

PROBLEM:

Some installations may find it desirable to have LOGIN hang up the dataset if a user failed to log in. Systems which might find this patch useful are those in the United Kingdom (because of various telephone rules/restrictions), those where there is a great deal of contention for ports, or those in which users frequently try to dial in and quess passwords.

SOLUTION:

The patching procedure detailed below will cause LOGIN to hang up the dataset approximately 3 seconds after the user has logged out. (The extra 3 seconds are allotted to ensure that the entire "Access Denied" message is printed.)

This patch will have no effect on keyboards that are not connected with datasets.

PROCEDURE:

1. For purposes of this discussion, we will assume that the program to be patched is located in your privileged account on the public disk structure. If this is not the case, replace all program references with suitable text according to the requirements of your installation.

   The procedure below assumes that BASIC-PLUS is your system default run-time system.

   If you are using CSPCOM or BASIC-PLUS-2, refer to Section 8.4 of the RSTS/E V7.Ø Release Notes for the correct procedure to compile the program.

   If you are using BASIC-PLUS-2, this program must be task built against the BP2COM Run-Time System.

2. This patch is contained in a patch file appearing in patch kit version "A" or later. If you do not have this distribution you can produce a command file by specifying a file for [logfile=] in the procedure below.

   To apply the patch manually, perform the following RSTS/E system commands.

```
RUN $CPATCH<cr>
<CPATCH's header line>

File to patch - LOGIN.BAS=LOGIN.BAS<cr>
#[logfile=]KB:/CS:2707<cr>
*H/2!/V<cr>
2!              PROGRAM         : LOGIN.BAS
```

```
*H/32650<tab>/V<cr>
32650   Q$=CHR$(9%) &
*AV<cr>
        \ GOTO 32660 !IF A% &
*G/!/-1DV<cr>
        \ GOTO 32660 IF A% &
*EX<cr>
Patch from KB:[P,PN]CPATCH.CMD complete
#^Z
File to patch - ^Z

Ready
```

3. Use this procedure if your system default Run-Time System is BASIC-PLUS; otherwise, refer to Section 8.4 of the RSTS/E V7.Ø Release Notes for the correct procedure to compile the program.

   To re-compile the program and re-enter it into the system library, type the following RSTS/E commands.

   ```
   OLD LOGIN<cr>

   Ready

   COMPILE SYØ:$LOGIN<232><cr>

   Ready
   ```

4. The person making the changes to the program should now take whatever steps are necessary, according to installation guidelines, to save the new version of the program.

5. The source (.BAS) version of the program may now be removed from the public structure.

RSTS/E V7.Ø
System Utilities Package
LOGIN

MAKING LOGIN INVOKE AN ALTERNATE RUN-TIME SYSTEM - LOGIN FEATURE PATCH

PROBLEM:

Some installations may find it desirable to use an alternate Run-Time System as the default Run-Time System for their users.

SOLUTION:

The best way to establish a non-standard default Run-Time System (RTS) is to re-compile certain system programs, such as INIT, LOGIN, and UTILTY, under the new RTS, bring down the system, and use the DEFAULT option of the INIT code to make the new RTS the default.

An easier way to establish a non-standard default RTS is described below. Note that if the new RTS has not been added with the UTILTY command ADD RTS, LOGIN will exit to the system default RTS as though this patch had not been installed.

The patching procedure described below causes LOGIN to invoke the BP2COM RTS as the user's default RTS. To invoke some other RTS, change the assignment statement at line 32665 from I$="BP2COM" to I$="<newrts>", where <newrts> is the name of the RTS that you wish to use.

PROCEDURE:

1. For purposes of this discussion, we will assume that the program to be patched is located in your privileged account on the public disk structure. If this is not the case, replace all program references with suitable text according to the requirements of your installation.

   The procedure below assumes that BASIC-PLUS is your system default run-time system.

   If you are using CSPCOM or BASIC-PLUS-2, refer to Section 8.4 of the RSTS/E V7.Ø Release Notes for the correct procedure to compile the program.

   If you are using BASIC-PLUS-2, this program must be task built against the BP2COM Run-Time System.

2. This patch is contained in a patch file appearing in patch kit version "A" or later. If you do not have this distribution you can produce a command file by specifying a file for [logfile=] in the procedure below.

   NOTE

   The patch file for this patch requires manual editing to include installation specific parameters before it can be successfully installed. Specifically, change "BP2COM" to the desired RTS.

   To apply the patch manually, perform the following RSTS/E system commands.

```
RUN $CPATCH<cr>
<CPATCH's header line>

File to patch - LOGIN.BAS=LOGIN.BAS<cr>
#[logfile=]<cr>
*H/2!/V<cr>
2!              PROGRAM         : LOGIN.BAS
*H/32665<tab>/V<cr>
32665    I$="" &
*G/""/-2C/"BP2COM"/V<cr>        <Change "BP2COM" to the desired RTS, as required.>
32665    I$="BP2COM" &
*EX<cr>
Patch from KB:[P,PN]CPATCH.CMD complete
#^Z
File to patch - ^Z

Ready
```

3. Use this procedure if your system default Run-Time System is BASIC-PLUS; otherwise, refer to Section 8.4 of the RSTS/E V7.Ø Release Notes for the correct procedure to compile the program.

   To re-compile the program and re-enter it into the system library, type the following RSTS/E commands.

```
OLD LOGIN<cr>

Ready

COMPILE SYØ:$LOGIN<232><cr>

Ready
```

4. The person making the changes to the program should now take whatever steps are necessary, according to installation guidelines, to save the new version of the program.

5. The source (.BAS) version of the program may now be removed from the public structure.

RSTS/E V7.Ø
System Utilities Package
LOGIN

LOGIN CAN PRINT A MESSAGE FOR SPECIFIC ACCOUNTS - LOGIN FEATURE PATCH

PROBLEM:

Some installations may choose, on occasion, to establish messages for the users of specific accounts.

SOLUTION:

The patching procedure detailed below will allow the System Manager, or anyone else with access to a given account, to establish a message file that will be printed anytime that someone logs in under that account. To create the file, use $PIP (or a suitable text editor) to create the file "MESSAG.TXT" in the appropriate account.

Each time that a user logs in, LOGIN will check for the presence of the file "MESSAG.TXT" in that account, and print the contents of that file if it exists. This file will be printed (if present) even if the system message ($NOTICE.TXT) has been suppressed with a slash ("/") in the account specification.

Note that anyone with write-access to the file MESSAG.TXT, including users of that account, will be able to create, change, or delete the message.

PROCEDURE:

1. For purposes of this discussion, we will assume that the program to be patched is located in your privileged account on the public disk structure. If this is not the case, replace all program references with suitable text according to the requirements of your installation.

   The procedure below assumes that BASIC-PLUS is your system default run-time system.

   If you are using CSPCOM or BASIC-PLUS-2, refer to Section 8.4 of the RSTS/E V7.Ø Release Notes for the correct procedure to compile the program.

   If you are using BASIC-PLUS-2, this program must be task built against the BP2COM Run-Time System.

2. This patch is contained in a patch file appearing in patch kit version "A" or later. If you do not have this distribution you can produce a command file by specifying a file for [logfile=] in the procedure below.

   To apply the patch manually, perform the following RSTS/E system commands.

```
RUN $CPATCH<cr>
<CPATCH's header line>

File to patch - LOGIN.BAS=LOGIN.BAS<cr>
#[logfile=]KB:/CS:64Ø49<cr>
*H/2!/V<cr>
2!              PROGRAM         : LOGIN.BAS
```

```
*H/32405<tab>/V<cr>
32405    !RESUME 14020 IF ERL=14000% OR ERL=14010% &
*G/!/-1DV<cr>
32405    RESUME 14020 IF ERL=14000% OR ERL=14010% &
*EX<cr>
Patch from KB:[P,PN]CPATCH.CMD complete
#^Z
File to patch - ^Z

Ready
```

3. Use this procedure if your system default Run-Time System is BASIC-PLUS; otherwise, refer to Section 8.4 of the RSTS/E V7.Ø Release Notes for the correct procedure to compile the program.

   To re-compile the program and re-enter it into the system library, type the following RSTS/E commands.

   ```
   OLD LOGIN<cr>

   Ready

   COMPILE SYØ:$LOGIN<232><cr>

   Ready
   ```

4. The person making the changes to the program should now take whatever steps are necessary, according to installation guidelines, to save the new version of the program.

5. The source (.BAS) version of the program may now be removed from the public structure.

RSTS/E V7.Ø
System Utilities Package
LOGIN

LOGIN CAN CHAIN TO A SPECIFIED PROGRAM - LOGIN FEATURE PATCH

PROBLEM:

Some installations may find it desirable to have LOGIN chain to a specified program when some or all users log in.

SOLUTION:

The patching procedure detailed below will cause LOGIN to chain to the specified program, if it exists. (LOGIN will proceed normally if the program does not exist.)

At line 3266Ø, the variable I$ should be set to some program name, as follows:

START	This will cause LOGIN to chain to the program "START" if it exists in the current user's account.
#START	This will cause LOGIN to chain to the program "START" if it exists in the user's group library ([*,Ø]).
$START	This will always cause LOGIN to chain to the program "START" if it exists in the system library ([1,2]).

PROCEDURE:

1. For purposes of this discussion, we will assume that the program to be patched is located in your privileged account on the public disk structure. If this is not the case, replace all program references with suitable text according to the requirements of your installation.

   The procedure below assumes that BASIC-PLUS is your system default run-time system.

   If you are using CSPCOM or BASIC-PLUS-2, refer to Section 8.4 of the RSTS/E V7.Ø Release Notes for the correct procedure to compile the program.

   If you are using BASIC-PLUS-2, this program must be task built against the BP2COM Run-Time System.

2. This patch is contained in a patch file appearing in patch kit version "A" or later. If you do not have this distribution you can produce a command file by specifying a file for [logfile=] in the procedure below.

NOTE

The patch file for this patch requires manual editing to include installation specific parameters before it can be successfully installed. Specifically, "START" should be replaced with the desired program name.

To apply the patch manually, perform the following RSTS/E system commands.

```
RUN $CPATCH<cr>
<CPATCH's header line>

File to patch - LOGIN.BAS=LOGIN.BAS<cr>
#[logfile=]<cr>
*H/2!/V<cr>
2!              PROGRAM         : LOGIN.BAS
*H/3266Ø<tab>/V<cr>
3266Ø   IF A% THEN &
*AV<cr>
                I$="" &
*G/""/-2C/"START"/V<cr>         <Replace "START" with the desired program name.>
                I$="START" &
*AV<cr>
      !\        CHAIN I$ &
*G/!/-1DV<cr>
       \        CHAIN I$ &
*EX<cr>
Patch from KB:[P,PN]CPATCH.CMD complete
#^Z
File to patch - ^Z

Ready
```

3. Use this procedure if your system default Run-Time System is BASIC-PLUS; otherwise, refer to Section 8.4 of the RSTS/E V7.Ø Release Notes for the correct procedure to compile the program.

   To re-compile the program and re-enter it into the system library, type the following RSTS/E commands.

   ```
   OLD LOGIN<cr>

   Ready

   COMPILE SYØ:$LOGIN<232><cr>

   Ready
   ```

4. The person making the changes to the program should now take whatever steps are necessary, according to installation guidelines, to save the new version of the program.

5. The source (.BAS) version of the program may now be removed from the public structure.

RSTS/E V7.Ø
System Utilities Package
LOGIN

LOGGING ACCESS ATTEMPTS OVER DIAL-UP LINES - LOGIN FEATURE PATCH

PROBLEM:

Some installations may find it desirable to log all access attempts made from dial-up lines. This enables an installation to keep track of which accounts are being used to gain system access over dial-up lines and what type of logged-out requests are made over these lines.

SOLUTION:

The patching procedure detailed below will cause all access attempts made over dial-up lines to be logged to OPSER (KBØ: if OPSER is not running).

PROCEDURE:

1. For purposes of this discussion, we will assume that the program to be patched is located in your privileged account on the public disk structure. If this is not the case, replace all program references with suitable text according to the requirements of your installation.

   The procedure below assumes that BASIC-PLUS is your system default run-time system.

   If you are using CSPCOM or BASIC-PLUS-2, refer to Section 8.4 of the RSTS/E V7.Ø Release Notes for the correct procedure to compile the program.

   If you are using BASIC-PLUS-2, this program must be task built against the BP2COM Run-Time System.

2. This patch is contained in a patch file appearing in patch kit version "A" or later. If you do not have this distribution you can produce a command file by specifying a file for [logfile=] in the procedure below.

   To apply the patch manually, perform the following RSTS/E system commands.

```
RUN $CPATCH<cr>
<CPATCH's header line>

File to patch - LOGIN.BAS=LOGIN.BAS<cr>
#[logfile=]KB:/CS:2931<cr>
*H/2!/V<cr>
2!              PROGRAM         : LOGIN.BAS
*H/22500<tab>/V<cr>
22500    ! &
*12AV<cr>
        \ SEND.OPSER%=0% &
```

```
*G/=/I/(TTINTF% AND 16384%)<>/V<cr>
        \ SEND.OPSER%=(TTINTF% AND 16384%)<>Ø% &
*EX<cr>
Patch from KB:[P,PN]CPATCH.CMD complete
#^Z
File to patch - ^Z

Ready
```

3. Use this procedure if your system default Run-Time System is BASIC-PLUS; otherwise, refer to Section 8.4 of the RSTS/E V7.Ø Release Notes for the correct procedure to compile the program.

   To re-compile the program and re-enter it into the system library, type the following RSTS/E commands.

   ```
   OLD LOGIN<cr>

   Ready

   COMPILE SYØ:$LOGIN<232><cr>

   Ready
   ```

4. The person making the changes to the program should now take whatever steps are necessary, according to installation guidelines, to save the new version of the program.

5. The source (.BAS) version of the program may now be removed from the public structure.

RSTS/E V7.Ø
System Utilities Package
LOGOUT

MAKING LOGOUT HANG UP A DATASET - LOGOUT FEATURE PATCH

PROBLEM:

Some installations may find it desirable to have LOGOUT hang up the dataset after a user has logged out. Systems which might find this patch useful are those in the United Kingdom (because of various telephone rules/restrictions) or those where there is a great deal of contention for ports.

SOLUTION:

The patching procedure detailed below will cause LOGOUT to hang up the dataset approximately 2Ø seconds after the user has logged out. (The extra 2Ø seconds are allotted to ensure that the entire logout message is printed.)

This patch will have no effect on keyboards that are not connected with datasets.

PROCEDURE:

1. For purposes of this discussion, we will assume that the program to be patched is located in your privileged account on the public disk structure. If this is not the case, replace all program references with suitable text according to the requirements of your installation.

   The procedure below assumes that BASIC-PLUS is your system default run-time system.

   If you are using CSPCOM or BASIC-PLUS-2, refer to Section 8.4 of the RSTS/E V7.Ø Release Notes for the correct procedure to compile the program.

   If you are using BASIC-PLUS-2, this program must be task built against the BP2COM Run-Time System.

2. This patch is contained in a patch file appearing in patch kit version "A" or later. If you do not have this distribution you can produce a command file by specifying a file for [logfile=] in the procedure below.

   To apply the patch manually, perform the following RSTS/E system commands.

```
RUN $CPATCH<cr>
<CPATCH's header line>

File to patch - LOGOUT.BAS=LOGOUT.BAS<cr>
#[logfile=]KB:/CS:33Ø9Ø<cr>
*H/2!/V<cr>
2!              PROGRAM          : LOGOUT.BAS
```

```
*H/9Ø3Ø<tab>/V<cr>
9Ø3Ø     CLOSE 1% &
*AV<cr>
         !\ L$=SYS(CHR$(6%)+CHR$(-9%)+CHR$(N%(2%)/2%)+CHR$(2Ø%)) &
*G/!/-1DV<cr>
         \ L$=SYS(CHR$(6%)+CHR$(-9%)+CHR$(N%(2%)/2%)+CHR$(2Ø%)) &
*EX<cr>
Patch from KB:[P,PN]CPATCH.CMD complete
#^Z
File to patch -- ^Z

Ready
```

3. Use this procedure if your system default Run-Time System is BASIC-PLUS; otherwise, refer to Section 8.4 of the RSTS/E V7.Ø Release Notes for the correct procedure to compile the program.

   To re-compile the program and re-enter it into the system library, type the following RSTS/E commands.

   ```
   OLD LOGOUT<cr>

   Ready

   COMPILE SYØ:$LOGOUT<232><cr>

   Ready
   ```

4. The person making the changes to the program should now take whatever steps are necessary, according to installation guidelines, to save the new version of the program.

5. The source (.BAS) version of the program may now be removed from the public structure.

RSTS/E V7.Ø
System Utilities Package
REORDR

A NOTE ON THE SAFE USE OF REORDR

Because of certain restrictions imposed by the RSTS/E file structure, REORDR is unable to prevent access to an account's UFD while it is processing that UFD. The RSTS/E V7.Ø System Manager's Guide states that "To prevent a damaged directory structure, REORDR requires that all files in a UFD must be closed during the entire reordering process". REORDR verifies that no file is open when it starts to process a UFD, but it is unable to detect file opening, creation, or deletion between the initial check and the completion of the processing of that UFD.

The only way to safely reorder a UFD on the public structure is to disable logins and ensure that no other user is currently logged in. A UFD on a private disk should be reordered only when the user is sure that no other job will access the account(s) being reordered on that disk.

RSTS/E V7.Ø
System Utilities Package
SHUTUP

CHANGING 'SHUTUP' SLEEP PARAMETERS - SHUTUP FEATURE PATCH

PROBLEM:

In the "Initial Job Killing Phase", SHUTUP makes two passes through the system job table looking for active jobs which can be logged off the system or KILLed. For those jobs which are attached to a keyboard, the following text string is FORCEd to the terminal's input buffer to simulate the log off sequence:

CTRL/C CTRL/C "BYE/Y" <cr>

Since this method requires the running of the system LOGOUT program, the actual logging off process can consume a fair amount of elapsed time after the text string FORCE is made.

The SHUTUP program is initially set up with the following parameters which control a program pause (SLEEP) at the end of the job table search pass:

```
SLEEP.BASE%      = 1Ø%
SLEEP.INCREMENT% =  2%
```

The length of the SLEEP period in seconds is calculated as:

SLEEP time = SLEEP.BASE% + (SLEEP.INCREMENT% * <number of terminals "killed">)

where the number of terminals "killed" is the total number of jobs sent the FORCE text string during the pass through the job table. For example, for 3 terminals, a period of 16 seconds results; for 1Ø terminals, a period of 3Ø seconds results.

Some installations may desire to change these parameters to alter the SLEEP period during this phase of SHUTUP operation. A case in point is when a system has at least one terminal running on the system at 11Ø baud. At this transmission speed, it requires 25-3Ø seconds just to type out the LOGOUT message which appears after typing BYE/Y. Changing the value of SLEEP.BASE% to 3Ø% should eliminate this problem.

SOLUTION:

Below is presented a procedure for changing the value(s) SLEEP.BASE% and/or SLEEP.INCREMENT%.

Assume the following parameter values for the edit session which follows:

```
xx = new SLEEP.BASE%      value
yy = new SLEEP.INCREMENT% value
```

PROCEDURE:

1. For purposes of this discussion, we will assume that the program to be patched is located in your privileged account on the public disk structure. If this is not the case, replace all program references with suitable text according to the requirements of your installation.

We assume the executable version of the program will be stored in the System Library Account ($). If this is not the case, replace references to $ in step 3 below with the appropriate package or library account.

The procedure below assumes that BASIC-PLUS is your system default run-time system.

If you are using CSPCOM or BASIC-PLUS-2, refer to Section 8.4 of the RSTS/E V7.Ø Release Notes for the correct procedure to compile the program.

If you are using BASIC-PLUS-2, this program must be task built against the BP2COM Run-Time System.

2. This patch is contained in a patch file appearing in patch kit version "A" or later. If you do not have this distribution you can produce a command file by specifying a file for [logfile=] in the procedure below.

NOTE

The patch file for this patch requires manual editing to include installation specific parameters before it can be successfully installed. Specifically, change references to "xx" and "yy" to the desired values.

To apply the indicated corrections manually, perform the following RSTS/E system commands.

```
RUN $CPATCH<cr>
<CPATCH's header line>

File to patch - SHUTUP.BAS=SHUTUP.BAS<cr>
#[logfil=]<cr>
*H/2!/V<cr>
2!              PROGRAM         : SHUTUP
*H/1Ø31<tab>/V<cr>
1Ø31        SLEEP.BASE% = 1Ø% &
*G/1Ø/-2C/xx/V<cr>
1Ø31        SLEEP.BASE% = xx% &
*AV<cr>
        \ SLEEP.INCREMENT% = 2% &
*G/2/-C/yy/V<cr>
        \ SLEEP.INCREMENT% = yy% &
*EX<cr>
Patch from [P,PN]CPATCH.CMD complete
#^Z
File to patch - ^Z

Ready
```

RSTS/E V7.Ø
System Utilities Package
SHUTUP

Seq 1Ø.21.2 R

SHUTUP'S RUN-TIME SYSTEM MUST BE THE SYSTEM DEFAULT - SHUTUP RESTRICTION

The SHUTUP system program can run to completion only if it is compiled (not in .BAS form) and if it is running under the system default run-time system. If a RSTS/E system has two or more run-time systems, the operator must ensure that the SHUTUP program has been compiled under the run-time system which is the system default.

3. Use this procedure if your system default Run-Time System is BASIC-PLUS; otherwise, refer to Section 8.4 of the RSTS/E V7.Ø Release Notes for the correct procedure to compile the program.

   To re-compile the program and re-enter it into the system library, type the following RSTS/E commands.

   ```
   OLD SHUTUP<cr>

   Ready

   COMPILE SYØ:$SHUTUP<cr>

   Ready
   ```

4. The person making the changes to the program should now take whatever steps are necessary, according to installation guidelines, to save the new version of the program.

5. The source (.BAS) version of the program may now be removed from the public structure.

RSTS/E V7.Ø
System Utilities Package
TTYSET

Seq 1Ø.25.1 N

CREATING TTYSET MACROS

As noted in the RSTS/E System User's Guide, TTYSET provides the user with several macros which allow the setting of terminal characteristics via a single command (e.g., LA36 or VT52). If your system has terminals different from those listed in the RSTS/E System User's Guide or if you wish to easily override any of the characteristics automatically set for any of the supplied macros, you may define your own TTYSET macros. This may be done by creating the file $TTYSET.MCM and including entries for each new macro you wish to add. Each entry must be in the form:

```
MACRO,"char1;char2;...;charn"
```

where MACRO is the name of your new command and each charx is a valid TTYSET macro or command. For example, assume your system has terminals which are similar to VT52's but can print up to 12Ø characters per line. The $TTYSET.MCM file you create might contain the following line:

```
VT52X,"VT52;WIDTH 12Ø"
```

Note that each entry must be on a separate line and that $TTYSET.MCM will be examined only if a macro name is specified that does not match one of those provided by TTYSET.BAS.

RSTS/E V7.Ø
System Utilities Package
VT55

VT55 GRAPHICS DISPLAY TERMINAL SUPPORT

Software support for the VT55 Graphics Display Terminal has been included in RSTS/E V7.Ø. The software consists of the following files which may be installed on the library account [1,2]:

VT55.BAS
DEMO55.BAS

VT55.BAS is a collection of BASIC-PLUS callable routines that provide access to the VT55 terminal.

To run the test program and assure correct operation of the graphic functions of the terminal:

1. Login to any account from a VT55
2. OLD $DEMO55
3. APPEND $VT55
4. RUN

The program will prompt for the KB: number of the VT55, and then proceed with a simple but thorough test of the terminal and its software.

A manual presently exists describing methods for programming the VT55 terminal and it may be ordered from the Software Distribution Center.

VT55 Programming Manual Order #: AA-4949A-TC

MODIFYING THE HELP PROGRAM'S DATA BASE

HELP is a data driven utility that provides help messages for users. It initially opens a master HELP file; all other HELP files are found by indirect references from the master HELP file.

All HELP file references are defaulted as:

package's dev:[ppn]	if no explicit device name or ppn
.HLP	if no explicit extension

The initial setting of the "package location" is where HELP itself is located (usually $). An indirect file specification that has an explicit device name or ppn (possibly derived from a logical name) causes the "package location" to be changed to that device/ppn combination for all further levels of indirect file references. When HELP backs out of that level of indirect file(s), the "package location" is restored to its original setting The master HELP file is HELP.HLP, which must be located on the same account as the HELP program.

HELP files can be edited freely by the system manager.

WARNING: Do NOT use TAB's to format your help information. The help information is indented by 2 spaces for every nested level which throws off TAB stops.

Control characters recognized by HELP are:

Line type	Line's first character
Comment	!
Keyword	*
Switch	/
Indirect file	@
Text	other

Comments may be used anywhere; they are totally ignored.

STRUCTURE OF A HELP FILE

There are three parts to a HELP file:

1. Text describing the command (required).
2. A series of "Switch blocks" (see below) describing switches that always apply to the command (optional).
3. A series of "Keyword blocks" (see below) describing options to the command (optional, but usually present).

Since the switches that always apply to the command (the "global switches") can be expressed in one of three ways, there are four formats for a HELP file.

Format of a HELP file without global switches:

1. Text describing the command.
2. An optional series of "Keyword blocks" describing options to the command.

Format of a HELP file with leading global switches:

1. Text describing the command.
2. One or more "Switch blocks" for the global switches.
3. An optional series of "Keyword blocks" describing options to the command.

Format of a HELP file with trailing global switches:

1. Text describing the command.
2. An optional series of "Keyword blocks" describing options to the command.
3. One or more "Switch blocks" for the global switches. Note that the keyword block immediately preceding these switch blocks must use an indirect file specification to distinguish these switch blocks as global switches.

Format of a HELP file with global keyword switches:

1. Text describing the command.
2. An optional series of "Keyword blocks" describing options to the command.
3. A single keyword line (usually "*Switches") announcing the global switches immediately followed by one or more "Switch blocks" for the global switches.

The format of a "Keyword block" is:

1. Start of "Keyword block" flag. This is one or more lines starting with a "*", immediately followed by the keyword(s). For example:

```
 *DELETE
--or--
 *RESUME CTRL/C
 *RESUME ANY
```

2. Either

   a. An indirect file specification (see below).
   --or--
   b. Text describing the option optionally followed by one or more "Switch blocks" describing switches to the option.

The format of a "Switch block" is:

1. Start of "Switch block" flag. This is one or more lines starting with a "/", immediately followed by the switch(es). For example:

```
 /DELETE
--or--
 /LOG
 /NOLOG
```

2. Either

   a. An indirect file specification.
   --or--
   b. Text describing the switch.

The leading text in the master HELP file describes HELP itself. The Keywords in the master HELP file are the commands that HELP can talk about. The global switches are the switches that apply to the HELP command itself (e.g., direct HELP's output to a file).

SPECIAL KEYWORDS

A keyword block line (*XXX) or switch line (/XXX) is available to all but logged out users. To make a line available to logged out users as well, a "`" (accent grave, CHR$(96%)) should be added immediately after the "*" or "/" (e.g., *`XXX). To restrict a line to only logged in privileged users, a "~" (tilde, CHR$(126%)) should be added immediately after the "*" or "/" (e.g., *~XXX).

To allow for common spelling errors, "hidden" keyword and switch lines can be used. A hidden line has a "¦", CHR$(124%), immediately after the "*" or "/" (or the "*~", etc.). Hidden lines should immediately follow the real line they correspond to. For example,

```
*`HELP
*`¦HEPL
```

to allow HEPL as a common mistake when typing HELP. Hidden lines can also be used for synonyms. For example,

```
*FUNCTION Ø
*¦Ø
```

to allow simply Ø as well as FUNCTION Ø. Hidden lines can also be used for keywords or switches that can be matched upon, but that shouldn't appear in a summary for one reason or another. For example, if you have a HELP file about numeric error messages, it could be formatted as:

```
Numeric error message listing.
*Type the numeric error code you want

*¦Ø
Text for error message Ø...
*¦1
Text for error message 1...
```

The blank line just after the "*Type ..." line is important; it makes the "*¦Ø" not a synonym for anything. Hidden lines are never listed in the information available summaries. If a hidden line is matched and the previous non-hidden line is of the correct type (i.e., a keyword or switch) then that non-hidden line is assumed to be the correctly spelled or main line and a message about it is printed.

LIMITED MATCHING

To speed things up a bit, HELP will stop scanning a file if it gets an exact match and there is no more command line to match at the next level. For example, if your HELP file has both "D" and "DATE" as keywords, then if the "D" comes before the "DATE" then a D will only match the "D"; DA, DAT, or DATE will match only "DATE". On the other hand, if the "DATE" comes before the "D" then a D will match both "DATE" and "D". Another case involves multi-word keywords. For example, if your

HELP file has both "ESC" and "ESC SEQ", then if the "ESC" comes before the "ESC SEQ" then a ESC will only match the "ESC" and a ESC SEQ will attempt to find out about SEQ in the next level of the "ESC" as well as match the "ESC SEQ". For this reason, it would be better to place the "ESC" after the "ESC SEQ". This results in a ESC matching both the "ESC SEQ" and then the "ESC" and a ESC SEQ matching the "ESC SEQ" and terminating.

INDIRECT FILE SPECIFICATIONS

An indirect file specification is of the form:

```
@file-specification
```

As described above, the file specification has a default extension of .HLP and will use the current "package location" in the absence of an explicit device name or ppn. If an explicit device name or ppn is specified, the "package location" is changed to that device/ppn combination for further indirect file references. This method allows all of the HELP files for a given set of utilities to be located in the same account (usually the same account as the utilities themselves). The master HELP file has one or more new "Keyword blocks" added to it corresponding to the utilities' command name(s). Each "Keyword block" uses an indirect file specification that explicitly specifys the account of the utilities. The HELP file(s) in the utilities' account can reference their neighbors by simply using "@_file". (Note that "@_file" is used by convention instead of just "@file" to explicitly indicate that no device/ppn is being used.)

OBTAINING HELP

Help can be obtained on a particular topic by typing:

```
HELP  topic  subtopic  subsubtopic  ...
```

A topic can have the following format:

1. an alphanumeric string (e.g., command name, option, etc.)
2. same preceded by a "/" (=> interpreted as a switch)
3. the match-all symbol "*"

Examples:

```
HELP DIRECTORY /S
HELP SET LC
```

Abbreviations result in all matches being displayed.

HELP may be run either by users logged into the system (RUN or CCL entry) or by users not logged into the system (CHAIN entry from LOGIN).

The RUN entry will prompt with "Topic?" and output will always be to the user's terminal. Further prompting will utilize the last command line entered and ask "Subtopic?". A CTRL/Z response to the "Subtopic?" question will revert to the "Topic?" question; a CTRL/Z to the "Topic?" question will exit from HELP.

The CCL entry can be used in one or three ways:

1. HELP command-line

   Output is to the user's terminal. If there is no command-line, HELP lists its command line format and all available topics, then enters prompting mode just like the RUN entry. Otherwise, HELP lists the specified information and exits.

2. HELP/O[UTPUT]:file-name command-line

   Output is to the specified file with a default file extension of ".LST". HELP exits upon command completion.

3. HELP/P[ROMPT] command-line

   Output is to the user's terminal. If there is no command-line, HELP immediately enters prompting mode just like the RUN entry. Otherwise, HELP enters prompting mode after executing the command.

The logged out entry is like the CCL entry, except that HELP's output cannot be directed to a file.

RSTS/E V7.Ø
Data Manipulation Package
COPY

COPY.BAS IS NOT DESIGNED TO HANDLE "BAD BLOCK" DISKS

Many users seem to feel that the COPY program is the best way to back up their RSTS/E disks. This can be an extremely hazardous procedure for any device which allows bad blocks, such as the RKØ6 and the RP series of disks.

The COPY program is designed to work strictly with "perfect" media, i.e., media with no bad blocks. When COPY attempts to read a bad block, it prints an appropriate message and aborts. However, the program may not be able to determine that it has written data to a bad block on the output disk. In that case, a file, an account, or the entire disk might be lost without warning.

Because COPY is not equipped to handle bad blocks, it is an extremely poor substitute for SAVRES or BACKUP on any disks other than RKØ5's, RSØ3/RSØ4's, or RF11's.

CHANGING THE EFFECT OF THE POSITION COMMAND - EDITCH FEATURE PATCH

PROBLEM:

The Position (nP) command, described in Section 2.2.4 of the RSTS/E Text Editor Manual, functions differently than in RSTS/E VØ6A: it does NOT save the contents of the current buffer before beginning its search. Some installations may prefer to make the Position command work as it did in RSTS/E VØ6A.

SOLUTION:

The patching procedure described below will cause the P command to work as follows:

1. The current buffer contents, if any, are output to the primary output device and the buffer is cleared.

2. A page is read into the buffer and searched.

3. If the search is unsuccessful, the page buffer is cleared and step 2 is repeated.

4. If the nth occurrence is found, EDIT returns with the page in which the nth occurrence resides in the buffer, with Dot following the search object.

PROCEDURE:

1. For purposes of this discussion, we will assume that the program to be patched is located in your privileged account on the public disk structure. If this is not the case, replace all program references with suitable text according to the requirements of your installation.

   We assume the executable version of the program will be stored in the System Library Account ($). If this is not the case, replace references to $ in step 3 below with the appropriate package or library account.

   The procedure below assumes that BASIC-PLUS is your system default run-time system.

   If you are using CSPCOM or BASIC-PLUS-2, refer to Section 8.4 of the RSTS/E V7.Ø Release Notes for the correct procedure to compile the program.

   If you are using BASIC-PLUS-2, this program should be compiled under the BASIC2 Run-Time System.

2. This patch is contained in a patch file appearing in patch kit version "A" or later. If you do not have this distribution you can produce a command file by specifying a file for [logfile=] in the procedure below.

   To apply the patch manually, perform the following RSTS/E system commands.

```
RUN $CPATCH<cr>
<CPATCH's header line>

File to patch - EDITCH.BAS=EDITCH.BAS<cr>
#[logfile=]KB:/CS:40437<cr>
*H/2!/V<cr>
2!              PROGRAM         : EDITCH.BAS
*H/2620<tab>/ØAV<cr>
2620    GOSUB 16000 &
*14KV<cr>
2630    GOSUB 19200 &
*4C/2620/V<cr>
2620    GOSUB 19200 &
*EX<cr>
Patch from KB:[P,PN]CPATCH.CMD complete
#^Z
File to patch - ^Z

Ready
```

3. Use this procedure if your system default Run-Time System is BASIC-PLUS; otherwise, refer to Section 8.4 of the RSTS/E V7.Ø Release Notes for the correct procedure to compile the program.

   To re-compile the program and re-enter it into the system library, type the following RSTS/E commands.

   ```
   OLD EDITCH<cr>

   Ready

   COMPILE $EDITCH<40><cr>

   Ready
   ```

4. The person making the changes to the program should now take whatever steps are necessary, according to installation guidelines, to save the new version of the program.

5. The source (.BAS) version of the program may now be removed from the public structure.

RSTS/E V7.Ø
Data Manipulation Package
FIT

USING RT11 BAD BLOCK DISKS WITH FIT

The RT11 operating system follows the convention that files with the extension .BAD are used to cover bad data blocks on a device. Since FIT is primarilly intended for the transfer of files between RSTS/E disks and small RT11 structured devices which are assumed to be free of bad blocks, it was decided that a full implementation of the RT11 bad block convention in FIT was unnecessary and possibly confusing to the RSTS/E user.

Files may be transferred between RSTS/E and RT11 systems on bad block disks, however, by using FIT's /RT11 switch.

First, any disk containing bad blocks that is to be used by FIT must be initialized on an RT11 system. All the bad blocks must be located and files with extension (file type) .BAD created to cover the bad blocks.

Whenever the bad block disk is referenced in a FIT command under RSTS/E, the /RT11 switch, which may be abbreviated to /RT, must be appended to the specification containing the name of the disk. The /RT11 switch, in addition to operating as described in the RSTS/E System User's Guide, produces the following effects:

/RT11 on Transfer

FIT will ignore any files selected for transfer that have an extension of .BAD. This applies on transfers both to and from the RT11 device.

/RT11 on Delete

Again, FIT will ignore any request to delete a .BAD file on an RT11 device when the /RT11 switch is specified.

/RT11 on Squeeze

If the /RT11 switch is present, FIT will first inspect the device to see if it contains any .BAD files. If any are found, FIT will print a warning message and the squeeze operation will be aborted.

NOTE

FIT will not prevent an RT11 disk with bad blocks from being zeroed. Zeroing such a disk will destroy the information it contains about bad blocks. An RT11 bad block disk should never be zeroed under FIT, but all the non-.BAD files may be deleted with a wildcard delete operation (specifing the /RT11 switch).

RSTS/E V7.Ø
Error Control Package
Package Notes

ERRLOG.FIL FILE USED IN VØ6C SHOULD BE DELETED

Because of changes to the ERRLOG.FIL file required due to the increased functionality of the error logging programs, the VØ6C version of ERRLOG.FIL should be deleted prior to the initial running of the V7.Ø ERRINT/ERRCPY. This will avoid any conflict that may be caused by inconsistencies between the two versions. ERRCPY will create and initialize a new ERRLOG.FIL file when it determines that the file is not present.

RSTS/E V7.Ø
Error Control Package
Package Notes

UNDERSTANDING "PA MEMORY SYSTEM" ERROR LOGS

The following information is logged when a memory error causes a trap or an instruction abort. These logs can be recognized by the fact that "MEM ERR" is not zero.

CACHECTL — This is the data RSTS/E sets into the Cache Control Register (777746). It is meaningless for systems without cache memory.

ADDR LSB
ADDR MSB — These words contain the address of the word in memory which failed. On the PDP-11/7Ø, they are the contents of the Low Error Address Register (777740) and High Error Address Register (777742). Bits <15:14> of ADDR MSB indicate the cycle type of the failure. On other processors, these words are assembled from the other logged information.

MEM ERR — On processors with cache, this is the contents of the Memory System Error register (777744). On other processors, this word always contains 1ØØØØØ (CPU Abort, no cache error). If the same cache group fails five times in the same minute, that cache group will be disabled.

The remaining registers are not logged on the 11/7Ø, which does not have accessible memory CSR's. In any case, they are not meaningful unless the error occurred in main memory.

CSR — This is the address of the CSR which indicated an error.

(CSR) — This is the contents of that CSR. A second entry is also printed for an MS11M or MS11L with the extended address function selected.

K Range — This tells how finely the CSR contents can resolve the location of the error. For all current memory controllers (MF11-LP, MF11-UP, MF11-WP, MF11S-K, MM11-DP, MM11-YP, and MS11-JP), this value is 1, indicating that the CSR locates the error to within 1K words. For older semiconductor memory controllers (MS11-AP, MS11-BP, and MS11-CP) that do not provide the address of the error in the CSR, this value is the number of 1K-word blocks (up to 31.) controlled by the CSR.

Fail — When it discovers a CSR announcing an error, RSTS/E looks through all memory in the range of the error (see above) to find which words actually have errors. Thus, for a controller which resolves the error location to within 1K words, RSTS/E checks 1Ø24. words (or 512. if the memory is interleaved). This word tells how many words in the range had an error indication.

Base Adr — To get the physical address of the start of the range mentioned above, add two zero digits (six bits) to this value.

Fail #n — The next words in the error log tell which words in the given range actually have errors. The addresses are given as byte offsets from the base given above. Up to 5 such offsets may be logged.

Once a second, except on 11/7Ø's, RSTS/E checks all memory CSR's for errors that do not cause traps or aborts. The logs for these errors can be recognized by the fact that "MEM ERR" is zero. These errors include uncorrectable errors detected on NPR cycles, and correctable (single-bit) ECC errors. Uncorrectable errors are logged whenever they are detected, but only one correctable error will be logged for each CSR, for each time RSTS/E is started. When a correctable error is logged, only the CSR address and CSR contents will be meaningful.

RSTS/E V7.Ø
Error Control Package
ANALYS

MAKING /WIDE THE DEFAULT - ANALYS FEATURE PATCH

PROBLEM:

ANALYS for RSTS/E V7.Ø produces a wide, annotated memory dump if the user specifies "/WIDE" with the output file. Since the annotations can be extremely helpful in analyzing dumps produced after a system crash or those produced as a result of a "SNAP", DIGITAL strongly recommends that this switch always be specified. The switch was not made a default because many users do not have line printers or terminals which are more than 8Ø characters wide.

SOLUTION:

The patching procedure detailed below makes /WIDE the default in ANALYS. The new default may be overridden by including "/NARROW" with the output specification.

PROCEDURE:

1. For purposes of this discussion, we will assume that the program to be patched is located in your privileged account on the public disk structure. If this is not the case, replace all program references with suitable text according to the requirements of your installation.

   We assume the executable version of the program will be stored in the System Library Account ($). If this is not the case, replace references to $ in Step 3 below with the appropriate package or library account.

   The procedure below assumes that BASIC-PLUS is your system default run-time system.

   If you are using CSPCOM or BASIC-PLUS-2, refer to Section 8.4 of the RSTS/E V7.Ø Release Notes for the correct procedure to compile the program.

   If you are using BASIC-PLUS-2, this program should be compiled under the BASIC2 Run-Time System.

2. This patch is contained in a patch file appearing in patch kit version "A" or later. If you do not have this distribution you can produce a command file by specifying a file for [logfile=] in the procedure below.

   To apply the patch manually, perform the following RSTS/E system commands.

```
RUN $CPATCH<cr>
<CPATCH's header line>

File to patch - ANALYS.BAS=ANALYS.BAS<cr>
#[logfile=]KB:/CS:51797<cr>
*H/2!/V<cr>
2!              PROGRAM         : ANALYS.BAS
```

```
*H/1Ø9Ø<tab>/V<cr>
1Ø9Ø     OPEN S$ FOR INPUT AS FILE 3%, MODE 256%+8192% &
*3AV<cr>
        \ DEF.WIDTH$="" &
*G/"/I%/WIDE%V<cr>
        \ DEF.WIDTH$="/WIDE" &
*EX<cr>
Patch from KB:[P,PN]CPATCH.CMD complete
#^Z
File to patch - ^Z

Ready
```

3. Use this procedure if your system default Run-Time System is BASIC-PLUS; otherwise, refer to Section 8.4 of the RSTS/E V7.Ø Release Notes for the correct procedure to compile the program.

   To re-compile the program and re-enter it into the system library, type the following RSTS/E commands.

   ```
   OLD ANALYS<cr>

   Ready

   COMPILE $ANALYS<cr>

   Ready
   ```

4. The person making the changes to the program should now take whatever steps are necessary, according to installation guidelines, to save the new version of the program.

5. The source (.BAS) version of the program should now be removed from the public structure.

RSTS/E V7.Ø
Error Control Package
ANALYS

MODIFYING ANALYS FOR DECNET/E V1.Ø - ANALYS FEATURE PATCH

PROBLEM:

ANALYS, as released with RSTS/E V7.Ø, produces a NIP-like report if your system is configured with DECnet/E. The state information supplied with nodes and lines is compatible with those printed by NIP for DECnet/E V1.1. If you are using DECnet/E V1.Ø on your system, this state information may be misleading.

SOLUTION:

The patching procedure detailed below removes the V1.1 specific state information. Note that this patch should only be applied if you are using DEcnet/E V1.Ø.

PROCEDURE:

1. For purposes of this discussion, we will assume that the program to be patched is located in your privileged account on the public disk structure. If this is not the case, replace all program references with suitable text according to the requirements of your installation.

   We assume the executable version of the program will be stored in the System Library Account ($). If this is not the case, replace references to $ in Step 3 below with the appropriate package or library account.

   The procedure below assumes that BASIC-PLUS is your system default run-time system.

   If you are using CSPCOM or BASIC-PLUS-2, refer to Section 8.4 of the RSTS/E V7.Ø Release Notes for the correct procedure to compile the program.

   If you are using BASIC-PLUS-2, this program should be compiled under the BASIC2 Run-Time System.

2. This patch is contained in a patch file appearing in patch kit version "A" or later. If you do not have this distribution you can produce a command file by specifying a file for [logfile=] in the procedure below.

   To apply the patch manually, perform the following RSTS/E system commands.

```
RUN $CPATCH<cr>
<CPATCH's header line>

File to patch - ANALYS.BAS=ANALYS.BAS<cr>
#[logfile=]KB:/CS:11286<cr>
*H/2!/V<cr>
2!              PROGRAM         : ANALYS.BAS
*H/13050<tab>/V<cr>
13050           TEMP$ = "NSD" IF NF.NSD% &
*I/!/V<cr>
13050   !       TEMP$ = "NSD" IF NF.NSD% &
```

```
*AJC/!/V<cr>
        !       TEMP$ = "LSD" IF NF.LSD% &
*AJC/!/V<cr>
        !       TEMP$ = "SHD" IF ((NF.NSD%*2%) = NF.LSD%) AND (NF.NSD% <> Ø%) &
*H/1354Ø<tab>/V<cr>
1354Ø           TEMP$="Off" &
*4AJC/!/V<cr>
        !       TEMP$="Dis" IF S.UP%=512% AND DSR.1%=Ø% &
*AJC/!/V<cr>
        !       TEMP$="DSR" IF S.UP%=512% AND DSR.1%=256% &
*AJC/!/V<cr>
        !       TEMP$="Int" IF S.UP%=512% AND DSR.1%=768% &
*EX<cr>
Patch from KB:[P,PN]CPATCH.CMD complete
#^Z
File to patch - ^Z

Ready
```

3. Use this procedure if your system default Run-Time System is BASIC-PLUS; otherwise, refer to Section 8.4 of the RSTS/E V7.Ø Release Notes for the correct procedure to compile the program.

   To re-compile the program and re-enter it into the system library, type the following RSTS/E commands.

```
OLD ANALYS<cr>

Ready

COMPILE $ANALYS<cr>

Ready
```

4. The person making the changes to the program should now take whatever steps are necessary, according to installation guidelines, to save the new version of the program.

5. The source (.BAS) version of the program should now be removed from the public structure.

MANUAL SHUTDOWN OF THE LINE-PRINTER SPOOLING PACKAGE

On occasion it may become necessary to shutdown the SPOOL program without shutting down the entire system. There are several ways to perform the shutdown.

At any time, an operator may issue the END command to the spooler. This command conditions the spooler to shut itself down (kill itself) the next time it is ready to get a new job. Thus, if the spooler is printing a job, it will complete the job's output before shutting down. If the spooler is waiting for a job, it will shut down immediately. If the operator doesn't want the job currently printing to complete, the ABORT command may be used to terminate the job.

A second method of shutting down the spooler is through an OFFLINE command. This command causes immediate termination of any job being printed with an ABORT message on the listing. It also causes the END processing, and the spooler will kill itself after the current job.

If the operator wishes to terminate a printing job for shutdown or any other reason, the line printer must be online. Under no circumstances will the spooler consider a job completed until all the data which the spooler has tried to print (including termination messages) has been actually printed. If a job is ABORTed by the operator, the spooler will clear out, at the first opportunity, whatever data has already been buffered; the line printer must be online to allow this. If it is desired that no further output be done, it is necessary to use the UTILTY program to KILL the spooler job. In this case, no cleanup can be performed. This process may be necessary in case of certain hardware failures (for example, if the line printer interface will not raise the READY flag). It should only be used in a crisis. The job which was being printed should be killed from the queue file by a QUE/K command from the operator.

For example, to manually shut down the LPØSPL spooler in logical end mode, the following command is used:

```
PLE/INT LPØSPL:END
```

At the completion of the current job, the spooler will kill itself. To manually shut down the BAØSPL job in immediate mode, the command:

```
PLE/INT BAØSPL:OFFLINE
```

is used. BAØSPL will ABORT the current job, clear it from QUEUE.SYS, and kill itself.

WILDCARD FILE LOOKUP OPERATIONS OF THE SPOOLING PACKAGE

It is of importance that one understand the general operation and timing of the file lookup procedure in the spooling package and the restrictions which result.

Assume that a job that was queued with wild card file specifications has progressed to the head of the queue. It will remain there until QUEMAN receives a request for a new job from the spooler.

QUEMAN actually begins the file lookup when the new job request is received. QUEMAN searches the file structure, from the beginning, looking for the first occurrence of a file which matches the file specification. It then sends a file packet, including the file's name, to the spooler. The spooler will then request the next message from QUEMAN.

QUEMAN again starts its search at the beginning, but, this time, looks for the second file matching the file specification. Obviously, in order to find the second file, it must once again encounter the first match. QUEMAN now will either send a file packet, if it finds a match, or an end job packet if it does not. This process is repeated until no match is found.

The potential for problems arises for two reasons: first, the amount of time actually required for the entire lookup process may be minutes. Second, the lookup is sequential, and all future lookups are dependent upon the stability and linear integrity of the preceding lookups.

The consequence of this is that during the file lookup phase there must not be any changes (creations or deletions) to files that match the requested file specification. If there are, it is possible that QUEMAN's relative file numbering scheme will be thrown off, yielding such errors as duplicate files being processed or files being skipped entirely.

Because of this, the following restriction must be enforced when using wild card file specifications:

1. Do not queue files that will not be in a stable existence for the duration of the entire lookup process. If it is necessary to queue a job that depends upon future events use a sufficiently long AFTER switch to ensure that the necessary environment is stable for the lookup.

2. For any job which has been requeued, (i.e., a job which had begun to print and was put back into the queue before completion) those files which match the file specification must not be changed.

NOTES ON THE USE OF THE SERIAL LA18Ø (LA18ØS) TERMINAL UNDER RSTS/E V7.Ø

The first part of this article describes the behavior of the Serial LA18Ø terminal (called the LA18ØS) when its power is turned off, either at system start-up time or at some later time during timesharing, and makes recommendations to prevent loss of data.

If any terminal (including an LA18ØS) is powered off when the system is booted, the device appears to the system to be operational. If a user tries to print a file on the terminal, the system will send characters to the device at the usual speed and assume that they are being printed properly. Obviously, since the device is powered off, the characters are not being printed, but the device can not return any indication of that fact to the system. The entire file will be sent as if it were printing.

If a spooler is started on such a KB: line, any files queued to that spooler will be sent, just as if the device were printing properly. The queue will gradually empty, and any files queued with /DELETE will be deleted.

Since there is no method by which the software can accurately differentiate between a terminal which is powered off and one which is printing properly, it is highly recommended that the person starting up a RSTS/E system check to ensure that all terminal devices which are ordinarily spooled are powered on and properly online before system start-up is performed.

A second class of problems involves the behavior of the LA18ØS when it is powered off or powered on after it has been ASSIGNed or OPENed.

During its power down and power up sequence, the LA18ØS terminal may send spurious characters to the host processor. The ASCII code of these characters appears to be random, but they occasionally take on the value of control characters (for example, CTRL/C). Since the program printing on the terminal may not expect input of any kind from the terminal, these spurious characters may result in unexpected errors: a CTRL/C may cause the program to print READY; a CTRL/O will cause all further output to be discarded without printing; etc. There is no way for the software to protect against all possible spurious inputs. RSTS/E is able to protect against all spurious input except CTRL/S (XON) and CTRL/Q (XOFF) via the MODE 32% option on the OPEN statement. However, spurious XON/XOFF characters can still cause loss of synchronization between the host system and the terminal. The only way to prevent this problem is to ensure that the LA18ØS is never powered off or powered on while it is ASSIGNed or OPENed.

A final problem concerns loss of data when the LA18ØS is powered off. The LA18ØS contains an internal buffer which is used to hold characters which have been received from the host processor but which have not yet been printed. When the LA18ØS goes OFFLINE (either because the ONLINE/OFFLINE switch is placed in the OFFLINE position, or because some exception condition, such as a paper jam, has been detected) this buffer may still contain some unprinted data. If the terminal is placed back ONLINE, printing will resume with no loss of data. If, however, the terminal is turned off, the buffer is cleared and any data in the buffer is lost. The only protection against this loss is to ensure that the LA18ØS is not powered off while it is printing.

In summary, DIGITAL recommends that any LA18ØS terminal to be used under RSTS/E timesharing be powered on before system start-up and that it remain on whenever it is in use. If the terminal is powered off, loss of data and unexpected errors may result.

Another kind of problem has been encountered when spooling to an LA18ØS. This problem concerns the manner in which the terminal device handles printed lines which are longer than the width of the terminal.

When a line is printed on the LA18ØS, the device prints characters until it gets to the physical right margin. At that point, the logic of the terminal inhibits further printing until a carriage return character moves the print head back from the margin. When a CR is encountered in such a situation, the LA18ØS also generates a line-feed, causing spacing to a new line. Since most print lines are terminated by a CR-LF sequence, however, the second LF will, again, cause spacing to a new line, leaving one line blank. Thus, printing a line which is longer than the width of the LA18ØS (i.e., longer than 132 characters) will cause a blank line on the printed output.

When performing normal print operations under RSTS/E, the user will not ordinarily encounter this problem. The RSTS/E Terminal Service ordinarily keeps track of the characters printed on the terminal and explicitly issues a CR-LF sequence to the terminal when the print head reaches the right margin (as defined by the WIDTH parameter in a TTYSET command). Thus, lines printed to that terminal which are longer than the width of the terminal result in printing of all characters, but on multiple lines. The only programs which will have trouble are those which need to keep track of vertical position on the page (for example, the line printer spooler) and, therefore, invoke the special MODE (Mode 4%) to inhibit the automatic CR-LF by the Terminal Service. Files printed by such programs will have blank lines after each line which was longer than the terminal width. There is no practical software remedy for this problem.

SPOOLING JOBS FROM A BASIC-PLUS PROGRAM

There is now a supported way for a BASIC-PLUS program to request that jobs be queued to any valid spooling device (e.g., LP:, BA:, RJ:, etc.) without the overhead of CHAINing to QUE. Such a request may now be included in a BASIC-PLUS program by the use of the Spooling SYS call, described in detail in article Seq 19.6.5 N.

Note that a MACRO program may execute a similar request by using the UU.SPL UUO subfunction, which is described in article Seq 19.24.1 N.

LINE PRINTER SPOOLER RESTRICTIONS

PROBLEM:

The line printer spooler for RSTS/E V7.Ø (as well as VØ6C) does not support files over 32767 blocks or CC:PRN files. The spooler supports a subset of VFC (Variable with Fixed Control) format files (i.e., header size of 2) and most CC:FTN files (i.e., does not support the "$" format character).

RSTS/E V7.Ø
Spooling and Operator Services Package
QUE

DISABLE OPERATOR QUEUEING TO OTHER USERS ACCOUNT - QUE FEATURE PATCH

PROBLEM:

Operators as defined in OPSER's table of operators can QUE files/jobs to be printed or run under BATCH for another user. Some installations may consider this too great a security risk.

SOLUTION:

The following patch disables the above mentioned feature. It does not affect the operators ability to modify or kill other users jobs once they are in the queue, as that poses no apparent threat to system security. Nor does it affect a privileged user's ability to que files/jobs to another account. Refer also to Seq 14.6.3 F, which supplies a similar patch for QUMRUN.

PROCEDURE:

1. For purposes of this discussion, we will assume that the program to be patched is located in your privileged account on the public disk structure. If this is not the case, replace all program references with suitable text according to the requirements of your installation.

   We assume the executable version of the program will be stored in the System Library Account ($). If this is not the case, replace references to $ in step 3 below with the appropriate package or library account.

   The procedure below assumes that BASIC-PLUS is your system default run-time system.

   If you are using CSPCOM or BASIC-PLUS-2, refer to Section 8.4 of the RSTS/E V7.Ø Release Notes for the correct procedure to compile the program.

   If you are using BASIC-PLUS-2, this program should be compiled under the BASIC2 Run-Time System.

2. This patch is contained in a patch file appearing in patch kit version "A" or later. If you do not have this distribution you can produce a command file by specifying a file for [logfile=] in the procedure below.

   To apply the patch manually, perform the following RSTS/E system commands.

```
RUN $CPATCH<cr>
<CPATCH's header line>

File to patch - QUE.BAS=QUE.BAS<cr>
#[logfile=]KB:/CS:11874<cr>
*H/2!/V<cr>
2!              PROGRAM         : QUE
```

```
*H/2Ø15<tab>/V<cr>
2Ø15     IF NOT PRIV% AND Y1%(Ø%,1%)<>SWAP%(P8%)+P9% THEN &
*G/NOT PRIV%/-9C/P8%<>1%/V<cr>
2Ø15     IF P8%<>1% AND Y1%(Ø%,1%)<>SWAP%(P8%)+P9% THEN &
*EX<cr>
Patch from KB:[P,PN]CPATCH.CMD complete
#^Z
File to patch - ^Z

Ready
```

3. Use this procedure if your system default Run-Time System is BASIC-PLUS; otherwise, refer to Section 8.4 of the RSTS/E V7.Ø Release Notes for the correct procedure to compile the program.

   To re-compile the program and re-enter it into the system library, type the following RSTS/E commands.

   ```
   OLD QUE<cr>

   Ready

   COMPILE $QUE<232><cr>

   Ready
   ```

4. The person making the changes to the program should now take whatever steps are necessary, according to installation guidelines, to save the new version of the program.

5. The source (.BAS) version of the program may now be removed from the public structure.

RSTS/E V7.Ø
Spooling and Operator Services Package
QUMRUN

Seq 14.6.1 F

DISABLING 'JOB WITH DIFFERENT FORM NAME WAITING' MESSAGE - QUMRUN FEATURE PATCH

PROBLEM:

QUMRUN will print a message on the OPERATOR SERVICES CONSOLE whenever one of the LP: spoolers has no JOBs which it may print but there are eligible JOBs waiting in the spooler's queue with a form name which is different from the spooler's. It may be desirable to disable this feature.

SOLUTION:

The following patch will disable this feature.

PROCEDURE:

1. For purposes of this discussion, we will assume that the program to be patched is located in your privileged account on the public disk structure. If this is not the case, replace all program references with suitable text according to the requirements of your installation.

   We assume the executable version of the program will be stored in the System Library Account ($). If this is not the case, replace references to $ in step 3 below with the appropriate package or library account.

   The procedure below assumes that BASIC-PLUS is your system default run-time system.

   If you are using CSPCOM or BASIC-PLUS-2, refer to Section 8.4 of the RSTS/E V7.Ø Release Notes for the correct procedure to compile the program.

   If you are using BASIC-PLUS-2, this program should be compiled under the BASIC2 Run-Time System.

2. This patch is contained in a patch file appearing in patch kit version "A" or later. If you do not have this distribution you can produce a command file by specifying a file for [logfile=] in the procedure below.

   To apply the patch manually, perform the following RSTS/E system commands.

```
RUN $CPATCH<cr>
<CPATCH's header line>

File to patch - QUMRUN.BAS=QUMRUN.BAS<cr>
#[logfile=]KB:/CS:57315<cr>
*H/2!/V<cr>
2!              PROGRAM         : QUMRUN.BAS
```

```
*H/1380<tab>/V<cr>
1380     DFORMS% = Ø% &
*G/IF Z4% = 2%/I/ AND Ø%/V<cr>
          \        IF Z4% = 2% AND Ø% THEN &
*EX<cr>
Patch from KB:[P,PN]CPATCH.CMD complete
#^Z
File to patch - ^Z

Ready
```

3. Use this procedure if your system default Run-Time System is BASIC-PLUS; otherwise, refer to Section 8.4 of the RSTS/E V7.Ø Release Notes for the correct procedure to compile the program.

   To re-compile the program and re-enter it into the system library, type the following RSTS/E commands.

   ```
   OLD QUMRUN<cr>

   Ready

   COMPILE $QUMRUN<cr>

   Ready
   ```

4. The person making the changes to the program should now take whatever steps are necessary, according to installation guidelines, to save the new version of the program.

5. The source (.BAS) version of the program may now be removed from the public structure.

RSTS/E V7.Ø
Spooling and Operator Services Package
QUMRUN

RESTRICTING THE ASSIGNMENT OF BA: - QUMRUN FEATURE PATCH

PROBLEM:

Some installations may find it desirable to limit the number of BATCH streams which can be used for jobs that are queued to BA:.

SOLUTION:

The patching procedure detailed below will cause QUMRUN to avoid assigning jobs to BA6: and BA7: unless the user has specified BA6: or BA7:.

PROCEDURE:

1. For purposes of this discussion, we will assume that the program to be patched is located in your privileged account on the public disk structure. If this is not the case, replace all program references with suitable text according to the requirements of your installation.

   We assume the executable version of the program will be stored in the System Library Account ($). If this is not the case, replace references to $ in step 3 below with the appropriate package or library account.

   The procedure below assumes that BASIC-PLUS is your system default run-time system.

   If you are using CSPCOM or BASIC-PLUS-2, refer to Section 8.4 of the RSTS/E V7.Ø Release Notes for the correct procedure to compile the program.

   If you are using BASIC-PLUS-2, this program should be compiled under the BASIC2 Run-Time System.

2. This patch is contained in a patch file appearing in patch kit version "A" or later. If you do not have this distribution you can produce a command file by specifying a file for [logfile=] in the procedure below.

   To apply the patch manually, perform the following RSTS/E system commands.

```
RUN $CPATCH<cr>
<CPATCH's header line>

File to patch - QUMRUN.BAS=QUMRUN.BAS<cr>
#[logfile=]KB:/CS:1Ø597<cr>
*H/2!/V<cr>
2!              PROGRAM         : QUMRUN.BAS
```

```
*H/15Ø7Ø<tab>/V<cr>
15Ø7Ø   DEF* FNN% &
*G/DFORMS2%/V<cr>
        \       DFORMS2%=(FNC%(LEFT(O$(Z3%,2%),2%),ZØ%(Z7%,9%)) AND &
*ØAI<cr>
<tab>\<tab>DFORMS1%=Ø% IF O$(Z3%,Ø%)="BA" &<cr>
<tab><tab><tab>IF (ZØ%(Z7%,1%) AND 255%)=255% IF O%(Z3%,1%)>5% &<cr>
<esc>*V<cr>
        \       DFORMS2%=(FNC%(LEFT(O$(Z3%,2%),2%),ZØ%(Z7%,9%)) AND &
*EX<cr>
Patch from KB:[P,PN]CPATCH.CMD complete
#^Z
File to patch - ^Z

Ready
```

3. Use this procedure if your system default Run-Time System is BASIC-PLUS; otherwise, refer to Section 8.4 of the RSTS/E V7.Ø Release Notes for the correct procedure to compile the program.

   To re-compile the program and re-enter it into the system library, type the following RSTS/E commands.

```
OLD QUMRUN<cr>

Ready

COMPILE $QUMRUN<cr>

Ready
```

4. The person making the changes to the program should now take whatever steps are necessary, according to installation guidelines, to save the new version of the program.

5. The source (.BAS) version of the program may now be removed from the public structure.

RSTS/E V7.Ø
Spooling and Operator Services Package
QUMRUN

DISABLE OPERATOR QUEUEING TO OTHER USERS ACCOUNT - QUMRUN FEATURE PATCH

PROBLEM:

Operators as defined in OPSER's table of operators can QUE files/jobs to be printed or run under BATCH for another user. Some installations may consider this too great a security risk.

SOLUTION:

The following patch disables the above mentioned feature. It does not affect the operators ability to modify or kill jobs once they are in the queue, as that poses no apparent threat to system security. Nor does it affect a privileged user's ability to que files/jobs to another account. Refer also to Seq 14.4.1 F, which supplies a similar patch for QUE.

PROCEDURE:

1. For purposes of this discussion, we will assume that the program to be patched is located in your privileged account on the public disk structure. If this is not the case, replace all program references with suitable text according to the requirements of your installation.

   We assume the executable version of the program will be stored in the System Library Account ($). If this is not the case, replace references to $ in step 3 below with the appropriate package or library account.

   The procedure below assumes that BASIC-PLUS is your system default run-time system.

   If you are using CSPCOM or BASIC-PLUS-2, refer to Section 8.4 of the RSTS/E V7.Ø Release Notes for the correct procedure to compile the program.

   If you are using BASIC-PLUS-2, this program should be compiled under the BASIC2 Run-Time System.

2. This patch is contained in a patch file appearing in patch kit version "A" or later. If you do not have this distribution you can produce a command file by specifying a file for [logfile=] in the procedure below.

   To apply the patch manually, perform the following RSTS/E system commands.

```
RUN $CPATCH<cr>
<CPATCH's header line>

File to patch - QUMRUN.BAS=QUMRUN.BAS<cr>
#[logfile=]KB:/CS:16561<cr>
*H/2!/V<cr>
2!              PROGRAM         : QUMRUN.BAS
```

```
*H/1ØØ7Ø<tab>/V<cr>
1ØØ7Ø   IF (CVT$%(MID(M$,16%,2%))=Ø%) OR ONE.SHOT% THEN &
*3AV<cr>
                        E%=-1% UNLESS VALID.OP% &
*G/UNLESS /I/(P% AND 255%)=1% OR (/G/VALID.OP%/I/ AND T%<>6%)/V<cr>
                        E%=-1% UNLESS (P% AND 255%)=1% OR (VALID.OP% AND T%<>6%) &
*EX<cr>
Patch from KB:[P,PN]CPATCH.CMD complete
#^Z
File to patch - ^Z

Ready
```

3. Use this procedure if your system default Run-Time System is BASIC-PLUS; otherwise, refer to Section 8.4 of the RSTS/E V7.Ø Release Notes for the correct procedure to compile the program.

   To re-compile the program and re-enter it into the system library, type the following RSTS/E commands.

   ```
   OLD QUMRUN<cr>

   Ready

   COMPILE $QUMRUN<cr>

   Ready
   ```

4. The person making the changes to the program should now take whatever steps are necessary, according to installation guidelines, to save the new version of the program.

5. The source (.BAS) version of the program may now be removed from the public structure.

RSTS/E V7.Ø
Spooling and Operator Services Package
SPLRUN

ELIMINATING EXTRA FORM FEEDS ON SPECIAL FORMS - SPLRUN FEATURE PATCH

PROBLEM:

At the beginning of each job, the spooler performs two top of form operations. The purpose of this action is to ensure that the new job really starts at the top of a new form, with at least one blank page between it and the last job. When using special paper forms, however, it is usually necessary to do a forms alignment. In this case, it may be annoying to have the extra top of form executed because it wastes a page of the form.

SOLUTION:

The following patch inhibits the extra top of form operation at the start of a new job. Note that it inhibits the action not only after a forms alignment process, but also at the beginning of any job. If this patch is used, it is essential that the paper be very carefully aligned in the printer. If the paper is misaligned, one job will begin on the same physical page as the end of the previous job.

PROCEDURE:

1. For purposes of this discussion, we will assume that the program to be patched is located in your privileged account on the public disk structure. If this is not the case, replace all program references with suitable text according to the requirements of your installation.

   We assume the executable version of the program will be stored in the System Library Account ($). If this is not the case, replace references to $ in step 3 below with the appropriate package or library account.

   The procedure below assumes that BASIC-PLUS is your system default run-time system.

   If you are using CSPCOM or BASIC-PLUS-2, refer to Section 8.4 of the RSTS/E V7.Ø Release Notes for the correct procedure to compile the program.

   If you are using BASIC-PLUS-2, this program should be compiled under the BASIC2 Run-Time System.

2. This patch is contained in a patch file appearing in patch kit version "A" or later. If you do not have this distribution you can produce a command file by specifying a file for [logfile=] in the procedure below.

   To apply the patch manually, perform the following RSTS/E system commands.

```
RUN $CPATCH<cr>
<CPATCH's header line>

File to patch - SPLRUN.BAS=SPLRUN.BAS<cr>
#[logfile=]KB:/CS:32856<cr>
*H/2!/V<cr>
2!              PROGRAM         : SPLRUN.BAS
```

```
*H/3200/V<cr>
3200    GOSUB 14500 IF PRTLNE%<>1% &
*AV<cr>
        \ NXTLNE%=2% \ RECLEN%=0% \ GOSUB 14800 &
*2KV<cr>
        \ GO SUB 21000 &
*EX<cr>
Patch from KB:[P,PN]CPATCH.CMD complete
#^Z
File to patch - ^Z

Ready
```

3. Use this procedure if your system default Run-Time System is BASIC-PLUS; otherwise, refer to Section 8.4 of the RSTS/E V7.Ø Release Notes for the correct procedure to compile the program.

   To re-compile the program and re-enter it into the system library, type the following RSTS/E commands.

   ```
   OLD SPLRUN<cr>

   Ready

   COMPILE $SPLRUN<cr>

   Ready
   ```

4. The person making the changes to the program should now take whatever steps are necessary, according to installation guidelines, to save the new version of the program.

5. The source (.BAS) version of the program may now be removed from the public structure.

RSTS/E V7.Ø
Spooling and Operator Services Package
BATCH

WIPE OUT INTERMEDIATE COMMAND FILE - BATCH FEATURE PATCH

PROBLEM:

BATDEC puts the information from the user's command file into an intermediate command file. This file is later OPENed for OUTPUT, thus deleting the file before recreating it. The disk blocks containing the information from the intermediate command file can then be grabbed by another user. This situation is potentially dangerous if the file contained sensitive information.

SOLUTION:

The following patch will cause the monitor to write nulls into the intermediate command file before deleting it, thus preventing further access to this information.

PROCEDURE:

1. For purposes of this discussion, we will assume that the program to be patched is located in your privileged account on the public disk structure. If this is not the case, replace all program references with suitable text according to the requirements of your installation.

   We assume the executable version of the program will be stored in the System Library Account ($). If this is not the case, replace references to $ in step 3 below with the appropriate package or library account.

   The procedure below assumes that BASIC-PLUS is your system default run-time system.

   If you are using CSPCOM or BASIC-PLUS-2, refer to Section 8.4 of the RSTS/E V7.Ø Release Notes for the correct procedure to compile the program.

   If you are using BASIC-PLUS-2, this program should be compiled under the BASIC2 Run-Time System.

2. This patch is contained in a patch file appearing in patch kit version "A" or later. If you do not have this distribution you can produce a command file by specifying a file for [logfile=] in the procedure below.

   To apply the patch manually, perform the following RSTS/E system commands.

```
RUN $CPATCH<cr>
<CPATCH's header line>

File to patch - BATCH.BAS=BATCH.BAS<cr>
#[logfile=]KB:/CS:40834<cr>
*H/2!/V<cr>
2!              PROGRAM         : BATCH .BAS
```

```
*H/6200<tab>/V<cr>
6200     Z0%(0%,35%),Z%=FNA% &
*G/.COM/I/<188>/V<cr>
         \ Z$=A$+RAD$(LOGID1%)+"J"+J$+".COM<188>/CL:-16" &
*EX<cr>
Patch from KB:[P,PN]CPATCH.CMD complete
#^Z
File to patch - ^Z

Ready
```

3. Use this procedure if your system default Run-Time System is BASIC-PLUS; otherwise, refer to Section 8.4 of the RSTS/E V7.Ø Release Notes for the correct procedure to compile the program.

   To re-compile the program and re-enter it into the system library, type the following RSTS/E commands.

   ```
   OLD BATCH<cr>

   Ready

   COMPILE $BATCH<cr>

   Ready
   ```

4. The person making the changes to the program should now take whatever steps are necessary, according to installation guidelines, to save the new version of the program.

5. The source (.BAS) version of the program may now be removed from the public structure.

RSTS/E V7.Ø
Spooling and Operator Services Package
BATRUN

$EOJ DOES NOT RESET ERROR CONDITION - BATRUN FEATURE PATCH

PROBLEM:

The $EOJ command does not reset the "running in error" flag in BATRUN. Thus, exceeding the ERROR level prevents succeeding $JOB commands from being processed.

SOLUTION:

The following patch will allow each $EOJ command to reset the ERROR flag so that processing can continue with the next $JOB command.

PROCEDURE:

1. For purposes of this discussion, we will assume that the program to be patched is located in your privileged account on the public disk structure. If this is not the case, replace all program references with suitable text according to the requirements of your installation.

   We assume the executable version of the program will be stored in the System Library Account ($). If this is not the case, replace references to $ in step 3 below with the appropriate package or library account.

   The procedure below assumes that BASIC-PLUS is your system default run-time system.

   If you are using CSPCOM or BASIC-PLUS-2, refer to Section 8.4 of the RSTS/E V7.Ø Release Notes for the correct procedure to compile the program.

   If you are using BASIC-PLUS-2, this program should be compiled under the BASIC2 Run-Time System.

2. This patch is contained in a patch file appearing in patch kit version "A" or later. If you do not have this distribution you can produce a command file by specifying a file for [logfile=] in the procedure below.

   To apply the patch manually, perform the following RSTS/E system commands.

```
RUN $CPATCH<cr>
<CPATCH's header line>

File to patch - BATRUN.BAS=BATRUN.BAS<cr>
#[logfile=]KB:/CS:53265<cr>
*H/2!/V<cr>
2!          PROGRAM         : BATRUN.BAS<cr>
*H/4ØØ<tab>/V<cr>
4ØØ       ! &
*G/!<tab>E8%/V<cr>
!         E8%               RUNNING IN LOCAL ERROR STATE FLAG. &
*ØAKV<cr>
!         E9%               RUNNING IN ERROR STATE FLAG. &
```

```
*H/11Ø3Ø<tab>/V<cr>
11Ø3Ø   I%=I%+2% IF CVT$%(MID(OUTPUT$,I%+1%,2%))=3328% &
*G/<tab>\ E9%,/4DG/-1%/-3C/Ø%/V<cr>
        \ E9%,Z0%(J9%,5Ø%)=FNEØ%("ERROR LEVEL EXCEEDED",Ø%) IF &
*H/13Ø1Ø<tab>/V<cr>
13Ø1Ø   E8%=Ø% &
*G/E8%=Ø%/-6C/E9%=Ø%/V<cr>
13Ø1Ø   E9%=Ø% &
*H/17131Ø<tab>/V<cr>
17131Ø   GOSUB 11ØØØ &
*G/FNW%=-1% IF E8%/-3C/E9%/V<cr>
        \ FNW%=-1% IF E9% &
*EX<cr>
Patch from KB:[P,PN]CPATCH.CMD complete
#^Z
File to patch - ^Z

Ready
```

3. Use this procedure if your system default Run-Time System is BASIC-PLUS; otherwise, refer to Section 8.4 of the RSTS/E V7.Ø Release Notes for the correct procedure to compile the program.

   To re-compile the program and re-enter it into the system library, type the following RSTS/E commands.

```
OLD BATRUN<cr>

Ready

COMPILE $BATRUN<cr>

Ready
```

4. The person making the changes to the program should now take whatever steps are necessary, according to installation guidelines, to save the new version of the program.

5. The source (.BAS) version of the program may now be removed from the public structure.

RSTS/E V7.Ø
System Management Guidelines
Package Notes

Seq 16.1.1 N

USING SAVE/RESTORE ON A TWO-DISK SYSTEM

This article describes the procedure for using SAVE/RESTORE to backup the disks on a system which has only two disk drives and no magnetic tape drives. On such a system, you can back up the system disk by shutting down the RSTS/E system and using the SAVRES option of INIT. It is impossible, however, to perform such a backup of the non-system disk, since the procedure would require that the system disk, the disk being backed up, and the SAVE volume all be mounted simultaneously, and hence would require three or more disk drives, or a magnetic tape drive.

The solution presented in this article is to place a copy of the INIT.SYS system initialization code on the non-system disk, allowing it to run as the "system" disk for off-line SAVE/RESTORE operations. This procedure has one drawback: the INIT.SYS initialization code requires about 45Ø blocks on the non-system disk (about 1Ø% of an RKØ5). If you have small disks, such as RKØ5's or RLØ1's, you may wish to make the dangerous choice of not backing up your non-system disk(s) and thereby risking loss of data from hardware or software failure, or you could consider expanding your hardware to include another disk drive or magnetic tape unit. For disks larger than RLØ1's, the overhead of having the INIT.SYS initialization code on each non-system disk is usually small enough to be tolerable, although you must decide based on your system's needs.

You should put the INIT.SYS initialization code ([Ø,1]INIT.SYS) on the non-system disk immediately after initializing the disk with DSKINT. This ensures that there is enough space for INIT.SYS. After generating the monitor, Run-Time System, and system library, perform the procedures outlined below. In the examples below, the notation "SYØ:" denotes the disk drive on which your normal system disk is mounted, and "xxn:" denotes the other disk drive, which is used for various purposes in the procedure.

1. Ensure that HOOK.SAV is on your system disk in account [1,2]. This will be the case if your SYSGEN target system was SY:. If this file does not already exist in [1,2], mount your distribution system disk (with the /RONLY switch) on xxn: and use the following commands to copy HOOK.SAV to your system disk (user responses are underlined):

   ```
   RUN $PIP.SAV<cr>
   *SYØ:$*.*=xxn:$HOOK.SAV<cr>
   *^Z

   Ready
   ```

2. If you have not yet initialized the non-system disk that you plan to use, shut down the system, mount the disk on xxn:, and use DSKINT to initialize the disk. You may then start timesharing.

3. Issue the commands given below:

```
RUN $PIP.SAV<cr>
*xxn:[Ø,1]*.*=[Ø,1]INIT.SYS<cr>
*^Z

Ready

RUN $HOOK.SAV<cr>
*xxn:[Ø,1]INIT.SYS<cr>
Hook complete
*^Z

Ready
```

4. Your non-system disk is now ready to use. You may dismount it and mount it as a public disk if you desire. When you wish to back up that disk, you must shut down your system, bootstrap the non-system disk, and mount a scratch disk in place of your normal system disk. You may now use the SAVRES option of INIT to SAVE or IMAGE-copy your non-system disk to the backup disk.

NOTE

If you ever patch [Ø,1]INIT.SYS on the system disk, be sure to patch that file on all disks which have [Ø,1]INIT.SYS.

USING BACKUP ON A TWO-DISK SYSTEM

This article describes the procedure for using the RSTS/E V7.Ø BACKUP package to back up the disks on a system which has only two disk drives and no magnetic tape drive. On such a system, you can back up the system disk normally by dismounting the non-system disk. The only requirement is that the BACKUP package exist entirely on the system disk. However, it is impossible to take a normal backup of the non-system disk. Such a procedure would require that the system disk, the disk being backed up, and the backup volume all be mounted simultaneously, and hence would require three or more disk drives.

The solution presented in this article is to place a minimal system on the other disk, allowing it to run as the system disk during the backup. You must follow this procedure for every RSTS/E file structured disk pack, public or private, which you ever mount along with the system disk and which you want to be able to back up. This procedure has one major drawback: the minimal system requires a large number of blocks on the non-system disk (approximately 14ØØ). If you have RKØ5 or RLØ1 disks, you should not use this procedure. Instead, use SAVE/RESTORE to preserve these disks. If you have RKØ6 disks, carefully consider the consequences of using a large part of your non-system disk capacity (almost 6% for RKØ6) for backup. If you do not wish to use such a large part of your non-system disk capacity for backup, you should use SAVE/RESTORE instead (see article Seq 16.1.1 N). The only disadvantage with SAVE/RESTORE is that you cannot save or restore individual files, but must copy an entire disk. For disks larger than RKØ6s, the overhead of having a small system on each nonsystem disk is usually small enough to be tolerable, although you must decide based upon your system's needs.

If you decide to build a small system on each non-system disk, you may use your standard monitor SIL, the SYSGEN.SIL provided by Digital, or another SIL generated especially for the purpose. The last choice allows you to build a small, special-purpose monitor for use only during backup and restore, thereby minimizing the disk space required. If you generate a special monitor, it should support at least 2 jobs, your two disks, one terminal (KBØ:), have nothing resident, no options, and a minimal number of small buffers. (Note that such a system may cause BACKUP to run slowly. If you can afford the disk space and wish to optimize the system for speed, more options and/or small buffers should be included.) Remember to install any applicable patches to the SIL.

You should place the minimal system for backup on the non-system disk immediately after initializing the disk with DSKINT. This placement ensures that contiguous space is available for the required files. After generating the monitor, Run-Time System, and system library (including the BACKUP package), perform the procedures outlined below. In the examples, the notation 'SYØ:' denotes the disk drive on which your normal system disk is mounted, and 'xxn:' denotes the other disk drive on the system, which is used for various purposes in the procedure. Note that the procedure below assumes that your default Run-Time System is BASIC-PLUS (e.g., the Backup Package, SHUTUP, and INIT are .BAC files). If these programs are in .TSK form on your system, be sure to substitute the proper extension where appropriate.

1. Ensure that the monitor SIL you want to use for your small backup system is on the system disk. Generate it or copy it if necessary.

2. If you have not yet initialized the non-system disk you plan to use, shut down the system, mount the disk on xxn:, and use DSKINT to initialize the disk. You may then start timesharing.

3. Issue the commands given below:

```
RUN $UTILTY<cr>
<UTILTY's header line>
#MOUNT xxn:packid/PRIVATE<cr>
#UNLOCK xxn:<cr>
#^Z

Ready

RUN $PIP.SAV<cr>
*xxn:[Ø,1]*.*/MO:16=[Ø,1]bbbbbb.RTS,[Ø,1]ERR.ERR<cr>
*xxn:[Ø,1]*.*=[Ø,1]INIT.SYS,[Ø,1]ssssss.SIL<cr>
        where ssssss is the SIL you wish to use on your backup disk, and bbbbbb
        is the Run-Time System under which SHUTUP, UTILTY, and the BACKUP
        package were compiled
*xxn:$*.*/W=SYØ:$BAC???.BAC,$BACKUP.PRM,$BACKUP.HLP,$SHUTUP.BAC,$UTILTY.BAC<cr>
        to copy SHUTUP, UTILTY, and the BACKUP package to the non-system disk
        (use an extension of .TSK if appropriate)
*xxn:[Ø,1]SWAP.SYS/MO:16/SI:128=NL:
        (if using BASIC-PLUS-2 or CSPCOM/RSX, use /SI:224 rather than /SI:128)
*^Z

Ready

RUN $HOOK.SAV<cr>
*xxn:[Ø,1]INIT.SYS<cr>
Hook complete
*^Z

Ready
```

4. Shut down your system, and use the BOOT option of the initialization code to bootstrap the disk xxn:. Use INSTALL to install the ssssss SIL, which you copied in step 3. Use DEFAULT to set defaults for the backup system. Specify 16K (28K for BASIC-PLUS-2 or RSX) as the SWAP MAX, and bbbbbb as the default Run-Time System.

Your non-system disk is now ready to use. You may dismount it and mount it as a public disk if you desire. When you wish to back up that disk, you must shut down your system and bootstrap the non-system disk. Use START to start timesharing from the small system on the non-system disk. Since INIT.BAC (or INIT.TSK) was not moved to the non-system disk, the messages:

```
?Can't find file or account
?Program lost-sorry

Ready
```

will be printed. You may now do your backup. (Alternatively, you may want to move INIT.BAC (or INIT.TSK) to your non-system disk at the time you moved the BACKUP package. In this case, you can include a control file to bring up a limited RSTS/E system and include the backup commands in the control file.)

Physically remove your normal system disk from SYØ: and mount a backup volume in its place. Then use the BACKUP package to preserve the files on the disk. Use a similar procedure to restore any files to the disk. If you lose the entire file structure of the disk, including the system and/or the BACKUP package, you can restore the disk using a system recovery medium as described in Release Note Article Seq 16.1.3 N.

NOTE

If you ever patch or replace on the system disk any of the BACKUP or system modules that you copied in the above procedure, you must do so while the non-system disk is not mounted as a public disk. After replacing the module(s), mount the non-system disk with the /PRIVATE switch and use PIP (with the /UP switch, update in place, if the output file is INIT.SYS, the monitor SIL, or the default run-time system) to change the file on the non-system disk. Repeat for each disk used on drive xxn: that contains a minimal system for backup purposes. Also, if any patches are installed to the RSTS/E monitor or the INIT.SYS code, be sure to install them on all disks which contain copies of a RSTS/E V7.Ø system.

RSTS/E V7.Ø
System Management Guidelines
Package Notes

CREATING A SYSTEM DISK RECOVERY MEDIUM

Regular use of the RSTS/E BACKUP package provides protection against loss of files if the system crashes, if disk errors occur, or if important files are accidently destroyed. However, the one situation from which BACKUP cannot recover is when the system disk or one or more of the monitor files is destroyed and the system cannot be started at all. One option in such an event is to use the RSTS/E distribution kit to regenerate the system, install all patches, build the complete system library, and then restore the remaining files on the system disk from the most recent backup. This is time consuming and error prone. Recovery is much simpler if you have a "recovery medium".

One means of creating a system disk recovery medium is SAVE/RESTORE. SAVE/RESTORE runs much more quickly than BACKUP for saving an entire disk and generates a bootable recovery medium. SAVE/RESTORE, however, cannot perform selective backups or back up the system disk while the system is running.

DIGITAL recommends that you use a procedure such as the one outlined below to regularly protect your system against the catastrophic loss of critical files that might result from hardware or software failure.

1. Use SAVE/RESTORE periodically (perhaps weekly) to back up all of your disks. This will provide you with a bootable medium that can be used to restore your entire disk(s) in the event of catastrophic failure.

2. Use BACKUP to back up those files which have changed since SAVE/RESTORE was last run. This will allow you to recover files that have changed recently without shutting down your system disk to back them up.

The remainder of this article tells you how to create a system recovery medium (disk or tape) if for some reason you choose not to use SAVE/RESTORE on your system. The resulting recovery medium will contain your fully patched monitor, BASIC-PLUS Run-Time System, and enough of the BACKUP package to restore the rest of the system disk file structure. The recovery medium is bootable in a manner similar to the distribution medium and any intermediate tape (or disk) which you may have created during SYSGEN. In order to recover from destruction of the system disk, you will need the recovery medium and a recent backup of your system disk. Note that this backup must include an up-to-date version of the file [1,2]ACCT.SYS so that the recovery can enter all accounts properly. If you ever change your monitor, or install any patches in any of the files placed on the recovery medium, you should rebuild that medium.

To build the recovery medium, you must have ADDed the RT11 Run-Time System to RSTS/E. You must also have the programs PIP.SAV and HOOK.SAV in account [1,2].

Follow the instructions in the following sections, depending on whether you wish to build a recovery disk or magnetic tape. In the examples, the file names RSTS.SIL and BASIC.RTS are used as the monitor SIL and default BASIC-PLUS Run-Time System. You should substitute the actual file names for your system.

Building a recovery disk

Using PIP or an editor, create the file RECOVR.CMD in [1,2] containing the following text:

```
SYØ:$*.*/RTS:BASIC=IN:$BAC???.*
SYØ:$*.BAC<232>/RE
SYØ:$BACENT.BAC<1Ø4>/RE
```

Mount a formatted and initialized disk on a free disk drive. (The text below uses xxn: as the device designator and 'packid' as the pack identification for the recovery pack.) If necessary, shut down your system and use DSKINT to format a pack, or use the program DSKINT.BAC to re-initialize an already formatted scratch pack. Using UTILTY or UMOUNT, logically mount xxn:. Follow the procedure below to build the recovery disk (user responses are underlined):

```
RUN $PIP.SAV<cr>
*xxn:[Ø,1]*.*/MO:16=[Ø,1]RT11.RTS,[Ø,1]BASIC.RTS,[Ø,1]ERR.ERR<cr>
*xxn:[Ø,1]=SY:[Ø,1]INIT.SYS,[Ø,1]RSTS.SIL<cr>
*xxn:$*.*=SY:$PIP.SAV,$UTILTY.SAV,$RECOVR.CMD,$BAC???.*<cr>
*^Z

Ready

RUN $HOOK.SAV<cr>
*xxn:[Ø,1]INIT.SYS<cr>
Hook complete
*^Z

Ready
```

Dismount the recovery disk and save it. Also dismount the distribution medium.

To recover from the destruction of the system disk or one or more of the system files, you need the recovery disk and a recent BACKUP. Mount the recovery disk on one disk drive and bootstrap it. The RSTS/E initialization code runs and prints a header and the OPTION: prompt. On another disk drive, mount the disk on which to restore the system disk. Using the DSKINT option, recreate your system disk with the same parameters that you used at system generation time. Then use the COPY option to move your monitor files to the new system disk. Use the /A switch to copy all relevant files (see the RSTS/E V7.Ø System Generation Manual). As part of the COPY option, INIT bootstraps that disk. If necessary, use the HARDWR option to change controller characteristics. Then INSTALL your monitor SIL and use REFRESH to create the required system files on your system disk. Use DEFAULT to change the system default run-time system to RT11. START the system on your new system disk and continue as follows:

```
?Can't find file or account
.ASSIGN xxn:IN
.MOUNT IN:packid/RO<cr>
.R IN:$PIP<cr>
*@IN:RECOVR.CMD<cr>
*^Z
.R IN:UTILTY<cr>
*DISMOUNT IN:<cr>
*SHUTUP<cr>

Option:
```

Dismount the recovery disk from drive xxn:. Use DEFAULT to change the system default run-time system back to BASIC. START the system. Your new system disk now has enough of the BACKUP package to restore the entire disk from a recent BACKUP. Do this now, specifying that no files be superseded. You must now use the STANDARD option of REACT to enter all accounts that BACKUP did not enter. Ignore the error messages generated when REACT tries to enter accounts that already exist. Your system disk is now ready to use.

Building a recovery magnetic tape

Using PIP or an editor, create the file RECOVR.CMD in [1,2] containing the following text:

```
SYØ:$*.*/BL/RTS:BASIC=IN:$*.*/RW:NO
SYØ:$*.BAC<232>/RE
SYØ:$BACENT.BAC<104>/RE
```

Mount a magnetic tape on a free drive (referred to below as xxn:). Follow the procedure below to build the recovery tape (user responses are underlined):

```
RUN $PIP.SAV<cr>
*xxn:/ZE<cr>
Really zero xxn:/density:ddd/parity:ppp? YES<cr>
*^Z

Ready

RUN $HOOK.SAV<cr>
*xxn:[Ø,1]INIT.SYS/D:8ØØ.,SY:[Ø,1]INIT.SYS<cr>
Hook complete
*^Z

Ready

RUN $PIP.SAV<cr>
*xxn:[Ø,1]*.*/MO:16=[Ø,1]RT11.RTS,[Ø,1]BASIC.RTS,[Ø,1]ERR.ERR<cr>
*xxn:[Ø,1]=SY:[Ø,1]RSTS.SIL
*xxn:$*.*=SY:$PIP.SAV,$UTILTY.SAV,$RECOVR.CMD,$BAC???.*<cr>
#^Z

Ready
```

Dismount the tape and save it.

To recover from the destruction of the system disk or one or more of the system files, you need the recovery tape and a recent BACKUP. Mount the recovery tape on a tape drive (referred to below as xxn:) and bootstrap it. The RSTS/E initialization code runs and prints a header and the OPTION: prompt. Mount the disk on which to restore the system disk. Using the DSKINT option, recreate your system disk with the same parameters that you used at system generation time. Then use the COPY option to move your monitor files to the new system disk. Use the /A switch to copy all files with the appropriate extensions (see the RSTS/E V7.Ø System Generation Manual). As part of the COPY option, INIT bootstraps that disk. If necessary, use the HARDWR option to change controller characteristics. Then INSTALL your monitor SIL and use REFRESH to create the required system files on your system disk. Use DEFAULT to change the default run-time system to RT11. START the system on your new system disk and continue as follows:

```
?Can't find file or account
.ASSIGN xxn:IN<cr>
.R IN:PIP<cr>
*$*.*=IN:$RECOVR.CMD<cr>
*@RECOVR.CMD<cr>
*RECOVR.CMD/DE<cr>
*^Z

Ready
```

Dismount the recovery tape from drive xxn:. Your new system disk now has enough of the BACKUP package to restore the entire disk from a recent BACKUP. Do this now, specifying that no files be superseded. You must now use the STANDARD option of REACT to enter all accounts that BACKUP did not enter. Ignore the error messages generated when REACT tries to enter accounts that already exist. Your system disk is now ready to use.

Notes on multiple disk systems

If your system disk is small, or if the public disk structure consists of more than one disk, some of the BACKUP or other programs may reside on a non-system public disk. In this case, you should copy to the recovery medium only those .BAC files that exist on the system disk. During the recovery procedure, after starting the system on your new system disk, you must physically mount and write-enable the other disks in your system's public structure. Type the command

```
MOUNT xxn:packid
```

for each non-system public disk, replacing xxn: with the disk name and unit, and packid with the pack ID for that disk. This procedure ensures that all required .BAC programs are available for the recovery.

Take the following precautions when creating a recovery disk:

1. If you can mount the recovery disk on an unused drive or in place of a private disk, do so. This leaves the full public disk structure mounted.

2. If you cannot mount the recovery disk in addition to the full public structure, you must dismount a non-system public disk. This disk must not contain any of the files required for the creation of the recovery disk or the actual recovery procedure. This disk can be dismounted while creating the recovery disk, and need not be mounted until a recovery is complete. If necessary, use PIP to copy files from this disk to another disk on the public structure:

   ```
   RUN $PIP.SAV<cr>
   *out:*.*=in:file.ext<cr>
   *in:file.ext/DE<cr>
   *^Z
   ```

   In these commands, in: is the name and unit number of the disk to be cleared of files, out: is the name and unit number of any other public disk, and file.ext is a file to be moved. Repeat these commands to PIP to copy each file on in: necessary for building the recovery disk or for the recovery procedure itself.

RSTS/E V7.Ø
Automated Patching Facility Package
Package Notes

NOTES ON ATPK

INTRODUCTION

ATPK (pronounced "at-pee-kay") is included in RSTS/E V7.Ø for use as a part of the BUILD procedure. It can also be used, however, as a general purpose indirect command file processor. ATPK can control a job by use of a pseudo keyboard (PK) and run programs from a script previously defined in a command file. As an indirect command file processor, ATPK runs at the user's keyboard and reports the job dialogue there as it happens. ATPK can also be used as a simple batch processor, running detached and reporting the job dialogue in a log file.

STARTING UP ATPK

ATPK may be invoked in one of the following ways:

```
RUN $ATPK
*<startup command>
```

or, if the CCL is installed,

```
ATPK <startup command>

        or

ATPK
*<startup command>
```

ATPK will accept either of the CCL commands "ATPK" or "@" if they have been installed by the system manager.

<startup command> is of the form:

```
[<log file> =] <command file> [/<switches>]
```

Valid switches are:

/DET	Run ATPK detached (available only to privileged users). Note that if a log file is not specified, output will be to the keyboard whether or not ATPK detaches. At the end of a successful detached run, ATPK will kill itself.
/DEL	Delete command file on successful completion (used by BUILD).
/RTS:<rts>	Start the controlled job under the keyboard monitor <rts>. If <rts> is not a keyboard monitor RTS, no error will be reported and the new job will start up under the system default RTS. If this switch is not specified, the job will start under the user's current job default RTS.
/LOG	If the user is privileged, the controlled job and any jobs started with the $LOGIN command will be created regardless of whether logins have been disabled.

/CHA[IN]: <program> [;<line>] [=\<core common>\]

On successful completion, chain to <program> at line number <line>, if specified, with <core common> loaded in core common, if specified. Any character not contained in <core common> may be used in place of the backslashes to delimit the <core common> argument. This switch allows BUILD to patch itself or ATPK and then chain back to itself for the rest of the build.

The dialogue of the controlled job will normally be printed on the terminal. If a log file is specified, the dialogue will also be printed there. If both a log file and the /DET switch are specified, the dialogue will be printed only in the log file.

The default extensions for the command and log files are, respectively, .CMD and .LOG.

PROCESSING THE COMMAND FILE

ATPK usually sends each line of the control file "as is" to the PK when the job at the PK enters a program or monitor keyboard wait stall. Four characters in the command file are handled specially, however:

$ as first character on line, indicates an ATPK command.
! as first character on line, indicates a comment.
^ indicates a control character.
_ indicates that the next character is not a special character. (i.e., an underscore character quotes the following character.)

ATPK commands:

Any command file line beginning with a dollar sign ($) will be considered an ATPK command. Valid ATPK commands are:

$DISABLE LOG	Turn dialogue reporting to the log file (and/or keyboard) off.
$ENABLE LOG	Resume printing the controlled job's dialogue in the log file (and/or at the keyboard).
$ALLOW NO ERRORS	Abort if any (fatal or warning) errors are encountered by the controlled job.
$ALLOW WARNING ERRORS	Abort if a fatal error is encountered by the controlled job, but allow warning errors.
$ALLOW FATAL ERRORS	Do not abort if any errors are encountered by the controlled job (default setting).

$WAIT	Accept a line of input from the keyboard and send it as input to the controlled job. If a program is running at the controlled job after this line has been processed, continue accepting command lines from the keyboard until the program has stopped running and the controlled job is in a keyboard monitor wait (i.e., SYSTAT would show the job in a "^C" state). After this command has been processed, input will resume from the command file.
$@<command file>	Switch input command stream to <command file>. ATPK will abort if the command file is not found. No nesting of these indirect command files is allowed; ATPK will abort if it finds the "$@" command in the <command file>.
$DETACH	Causes ATPK to detach. This command may be used after the $WAIT command to cause ATPK to detach after user input at the start of the controlled job.
$LOGIN [KB[n]:] [(p,pn)]	Log in a keyboard under the user's account or the account specified by (p,pn). This command, which is only available to privileged users, causes ATPK to create a job running LOGIN with instructions to attach to the specified keyboard and log in under the specified account. If (p,pn) is not specified, the account that ATPK is running under will be used. If n is not specified (e.g., $LOGIN KB:), LOGIN will be instructed to attach to the keyboard from which ATPK was started. This may be useful to log the user back in following a command that started up ATPK with the /DET switch. If the keyboard specification is missing altogether, a new job will be started at the pseudo keyboard under ATPK's control. The current job at the pseudo keyboard, if any, will first be killed in this case. If LOGIN fails to attach or log in at the pseudo keyboard under ATPK, an error message will be printed and ATPK will abort. If LOGIN fails at any other keyboard, no error will be returned to ATPK.

COMMENTS:

Any line starting with an exclamation point (!) will be recorded in the log and not sent to the controlled job.

CONTROL CHARACTERS:

Any character preceded by an uparrow (^) will be converted to a control character before being sent to the controlled job. If a command line consists only of an uparrow followed by a single character, the single control character will be sent to the controlled job with no extra line terminator.

QUOTED CHARACTERS:

Any "$" or "!" that is meant to be the first character of a line to be sent to the controlled job, or any "^" or "_" in the command file that is not to be interpreted as a special character by ATPK must be preceded with the underscore character.

Every underscore character (except those preceded by another underscore character) will be removed from the command line before it is sent to the controlled job.

USING ATPK FOR SILENT SYSTEM STARTUP

If it is desired, most of system startup may be controlled under ATPK. INIT.BAC (or INIT.TSK) need only run long enough to invoke ATPK. At the end of system startup, the ATPK log can be queued to the line printer. This section suggests guidelines for writing ATPK startup files and gives a sample set of these control files.

When designing the ATPK startup command files, most of the functions of INIT.BAC will have to be performed with UTILTY. The recommended procedure for bringing a non-system disk online after a crash is to clean the disk with ONLCLN.SAV and then MOUNT and UNLOCK the disk with UTILTY.

When starting up the spooling package under ATPK, two points in particular should be noted. First, the $LOGIN command should be used to re-log in the pseudo keyboard after a spooling program detaches. Secondly, when the spooling programs are invoked, they generally prompt for input and then sleep while awaiting a response. Since ATPK will not respond unless the controlled program enters an input stall (as opposed to a sleep), ATPK will not detect when the spooler needs the command line. This problem may be circumvented by including the startup command(s) on the same command file line as the command that invokes the spooling program. If the commands are separated with an uparrow-M, ATPK will insert a carriage return (the equivalent of a CTRL/M) between the commands, causing them to be sent to the pseudo keyboard at the same time, but as separate lines.

The way ERRCPY is normally started up by INIT is as follows: INIT runs as job 1, attaches to KBØ:, forces the commands to KBØ: to run ERRINT, and then exits. The system then processes the type-ahead on KBØ: and runs ERRINT, as job 1. To avoid using KBØ: in this way for the silent system startup, two chain entries have been added to ERRINT so that ATPK, running detached as job 1, can simply chain to ERRINT for normal or crash system startup. Chaining to ERRINT at line 31Ø2Ø can be used for normal startup; crash dump output will not be used. For a crash recovery, chaining to ERRINT at line 31Ø3Ø will cause crash dump output to be used.

Care must be taken when starting up certain programs by forcing to other terminals under ATPK, so that timing problems do not develop. For example, under INIT, VT5ØPY could be started up to run detached and report to a terminal by logging in that terminal, forcing the command to run VT5ØPY, forcing the VT5ØPY command line, and then forcing commands to modify the displayed options. Since INIT runs at a low priority, it would be prevented from running while VT5ØPY was running, and would force the next line only when VT5ØPY was ready. If the same technique were used; using ATPK's $LOGIN command to log in the keyboard and then forcing the commands with UTILTY, all the commands would be typed at the terminal before VT5ØPY had the chance to detach. As it detaches, VT5ØPY briefly closes the keyboard. The typed-ahead commands would then invoke LOGIN at the keyboard and prevent VT5ØPY from gaining access to it. VT5ØPY may be started up under ATPK using commands such as the following:

```
RUN $VT5ØPY
6Ø/24/DCA/TAB/ECHO/DET/KB5:
$LOGIN
RUN $UTILTY
FORCE KB5: F1
FORCE KB5: P
FORCE KB5: -O
FORCE KB5: J-S
EXIT
```

Because of the way the $LOGIN command works, VT5ØPY will be given enough time to open the keyboard (KB5: in the above example) before the other commands are forced.

The following is a sample set of control files that could be used for silent system startup.

$START.CTL:

```
FORCE KBØ: RUN $ATPK
FORCE KBØ: $STRTUP.LOG=$START.CMD/DET/LOG/CHA:$ERRINT;31Ø2Ø
END
```

$CRASH.CTL:

```
FORCE KBØ: RUN $ATPK
FORCE KBØ: $STRTUP.LOG=$CRASH.CMD/DET/LOG/CHA:$ERRINT;31Ø3Ø
END
```

$START.CMD:

```
$@[1,2]RTS.CMD
$@[1,2]MOUNT.CMD
$@[1,2]TTY.CMD
$@[1,2]SPOOL.CMD
$@[1,2]CCL.CMD
RUN $UTILTY
REMOVE LOGICAL LB
ADD LOGICAL SYØ:[1,1]LB
LOGINS
SEND ALL RSTS/E IS NOW ON THE AIR...
EXIT
RUN $QUE
LPØ:/DELETE/PRI:2ØØ=$STRTUP.LOG
EXIT
```

$CRASH.CMD:

```
$@[1,2]RTS.CMD
$@[1,2]ANALYS.CMD
$@[1,2]CLEAN.CMD
$@[1,2]MOUNT.CMD
$@[1,2]TTY.CMD
$@[1,2]SPOOL.CMD
RUN $QUE
LPØ:/PRI:2ØØ=$ANALYS.DMP
EXIT
$@[1,2]CCL.CMD
RUN $UTILTY
REMOVE LOGICAL LB
ADD LOGICAL SYØ:[1,1]LB
LOGINS
SEND ALL RSTS/E IS NOW ON THE AIR...
EXIT
RUN $QUE
LPØ:/DELETE/PRI:2ØØ=$STRTUP.LOG
EXIT
```

$RTS.CMD:

```
RUN $UTILTY
SEND KBØ:Adding Run-Time Systems
ADD RT11
ADD RSX
ADD RMS11
ADD BP2COM
ADD BASIC2
ADD BASIC
ADD BAS4F
ADD TECO
EXIT
```

$ANALYS.CMD:

```
RUN $UTILTY
SEND KBØ:Running ANALYS
EXIT
RUN $ANALYS
[Ø,1]CRASH.SYS
[1,2]ANALYS.DMP/WIDE
[1,2]ERRCRS.FIL
```

$CLEAN.CMD:

```
RUN $UTILTY
SEND KBØ:Cleaning Disks
EXIT
RUN $ONLCLN
DB1:
^C
RUN $ONLCLN
DB2:
^C
```

$MOUNT.CMD:

```
RUN $UTILTY
SEND KBØ:Mounting Disks
MOUNT DB1:PACK1
UNLOCK DB1:
MOUNT DB2:PACK2
UNLOCK DB2:
EXIT
```

$TTY.CMD:

```
RUN $UTILTY
SEND KBØ:Setting Terminal Characteristics
EXIT
RUN $TTYSET
KB16:;VT52;SPEED 24ØØ
KB17:;VT1ØØ;SPEED 48ØØ
EXIT
```

$SPOOL.CMD:

```
RUN $UTILTY
SEND KBØ:Starting Spoolers
EXIT
RUN $OPSER ^MLOG OPSER.LOG;ALL ^MCHA KBØ: ^MDETACH
$LOGIN
RUN $QUEMAN ^MDETACH/PRIORITY:Ø
$LOGIN
RUN $SPOOL ^MLPØ:/HEAD:2
$LOGIN
RUN $BATCH ^MBAØ:/ERROR:FATAL/NOQUEUE/NODELETE
$LOGIN
```

$CCL.CMD:

```
RUN $UTILTY
SEND KBØ:Adding CCL's.
CCL ATT-ACH=$LOGIN.*;PRIV 3ØØØØ
CCL ATP-K=$ATPK.*;PRIV 3ØØØØ
CCL BCK-=$RMSBCK.TSK;Ø
CCL BPC-REF=$BPCREF.*;3ØØØØ
CCL BYE-=$LOGOUT.*;PRIV Ø
CCL CNV-=$RMSCNV.TSK;Ø
CCL CRE-ATE=$EDIT.*;3ØØØØ
CCL DEF-=$RMSDEF.TSK;Ø
CCL DFN-=$RMSDFN.TSK;Ø
CCL DES-=$RMSDES.TSK;Ø
CCL DIS-MOUNT=$UMOUNT.*;PRIV 3ØØØØ
CCL DI-RECTORY=$DIRECT.*;PRIV 3ØØØØ
CCL DSP-=$RMSDSP.TSK;Ø
CCL EDT-=$EDT.TSK;Ø
CCL ED-IT=$EDIT.*;3ØØØØ
CCL FIT-=$FIT.*;PRIV 3ØØØØ
CCL HELLO-=$LOGIN.*;PRIV Ø
CCL HELP-=$HELP.*;3ØØØØ
CCL IFL-=$RMSIFL.TSK;Ø
CCL LBR-=$LBR.TSK;Ø
CCL LIBR-=$LIBR.SAV;82Ø8
CCL LIN-K=$LINK.SAV;82Ø8
CCL LOG-IN=$LOGIN.*;PRIV Ø
CCL MACR-O=$MACRO.SAV;8216
CCL MAC-=$MAC.TSK;Ø
CCL MAK-E=$TECO.TEC;Ø
CCL MOU-NT=$UMOUNT.*;PRIV 3ØØØØ
CCL MU-NG=$TECO.TEC;Ø
CCL PAT-=$PAT.TSK;Ø
CCL PIP-=$PIP.SAV;82Ø8
CCL PL-EASE=$PLEASE.*;PRIV 3ØØØØ
CCL QU-EUE=$QUE.*;PRIV 3ØØØØ
CCL RST-=$RMSRST.TSK;PRIV Ø
CCL SE-T=$TTYSET.*;PRIV 3ØØØØ
CCL SRT-=$SORT.TSK;Ø
CCL SU-BMIT=$QUE.*;PRIV 3ØØØØ
CCL SW-ITCH=$SWITCH.*;PRIV 3ØØØØ
CCL SY-STAT=$SYSTAT.*;PRIV 3ØØØØ
CCL TE-CO=$TECO.TEC;Ø
CCL TKB-=$TKB.TSK;Ø
CCL TY-PE=$TYPE.TEC;8
CCL UT-ILTY=$UTILTY.*;3ØØØØ
EXIT
```

OTHER FEATURES

By specifying a log file and KB: as your input file, you can make a copy of interactive dialogue to queue to the line printer, etc. Note that ATPK will not be completely transparent in such a session. Any line typed at the keyboard will not be sent to the PK until the next time the controlled job enters a keyboard input wait state.

When inputting from the terminal, CTRL/Z's get passed on through to the controlled job. A CTRL/C at any time will cause the controlled job to be killed, the log file closed, and ATPK to exit. A CTRL/C may be sent to the job by typing uparrow-C.

INSTALLING ATPK AS A CCL COMMAND

The following sequence will install the CCL commands "ATPK" and "@" to run the ATPK program. Either or both of the commands may be installed.

```
RUN $UTILTY<cr>
<UTILTY's header line>
#CCL ATP-K=[1,2]ATPK.BAC;PRIV 3ØØØØ<cr>
#CCL @-=[1,2]ATPK.BAC;PRIV 3ØØØØ<cr>
#^Z
```

(This assumes that ATPK is located on SY:[1,2].)

ALLOWING ACCESS TO ATPK BY NON-PRIVILEGED USERS

ATPK is installed on the system with a protection code of <124>. If non-privileged users are to be allowed to use ATPK, it must be renamed to have a protection code of <232> after the system installation is complete and after any patches have been applied to ATPK and it is re-compiled.

SUPPORT POLICY FOR ATPK

ATPK is supported only as a part of the BUILD procedure. DIGITAL makes no commitment, expressed or implied, to support ATPK as an indirect command file processor, or to supply an indirect command file processor in future releases of RSTS/E that is compatible with the present version of ATPK.

If you experience problems when using ATPK as an indirect command file processor, please submit an FYI-type Software Performance Report (SPR). While DIGITAL makes no commitment to fix problems reported with ATPK as an indirect command file processor, we would like to know about any problems that you encounter.

RSTS/E V7.Ø
Automated Patching Facility Package
Package Notes

ADDITIONAL FILES APPEARING ON PATCH KITS

There are several files on the RSTS/E patch kit which are not associated with the patching of any RSTS/E programs or product set modules.

EDUCATIONAL SERVICES' FILES ON PATCH KIT

There are five files located in account [1ØØ,1ØØ] on the RSTS/E V7.Ø Autopatch disk which are provided for students of the self-paced RSTS/E V7.Ø USER Course. The files, BOOK.LST, MAIL.DAT, and EMPFIL.DAT, are RMS files to be used in practice with the SORT-11 utility program. The files FILSPC.BAS and FILSPC.B2S are programs to be used as an aid in creating SORT-11 specification files.

NOTE

> These files appear only on the disk distribution Autopatch kits. Those installations with tape distribution that receive the Educational Services' RSTS/E V7.Ø USER Course will have these files included on the tape distribution for that package.

MAINTENANCE FILES

The file CSPLIB.OLB resides in account [1,2Ø] on the patch kit. This file is for maintenance purposes only and should be ignored.

RSTS/E V7.Ø
Automated Patching Facility Package
ATPK

ALLOWING ATPK TO DETACH FOR NON-PRIVILEGED USERS - ATPK FEATURE PATCH

At some installations it may be desirable to allow non-privileged users to run ATPK detached as a "mini batch processor", even though running a detached job is usually a privileged operation under RSTS/E. This optional patch will allow ATPK to detach for any user if the /DET switch is specified at the end of ATPK's startup command, or if the $DETACH command is encountered in the command file.

PROCEDURE:

1. For purposes of this discussion, we will assume that the program to be patched is located in your privileged account on the public disk structure. If this is not the case, replace all program references with suitable text according to the requirements of your installation.

   We assume the executable version of the program will be stored in the System Library Account ($). If this is not the case, replace references to $ in step 3 below with the appropriate package or library account.

   The procedure below assumes that BASIC-PLUS is your system default run-time system.

   If you are using CSPCOM or BASIC-PLUS-2, refer to Section 8.4 of the RSTS/E V7.Ø Release Notes for the correct procedure to compile the program.

   If you are using BASIC-PLUS-2, this program must be task built against the BP2COM Run-Time System.

2. This patch is contained in a patch file appearing in patch kit version "A" or later. If you do not have this distribution you can produce a command file by specifying a file for [logfile=] in the procedure below.

   To apply the patch manually, perform the following RSTS/E system commands.

```
RUN $CPATCH<cr>
<CPATCH's header line>

File to patch - ATPK.BAS=ATPK.BAS<cr>
#[logfile=]KB:/CS:62461<cr>
*H/2!/V<cr>
2!              PROGRAM         : ATPK.BAS
*H/3000<tab>/V<cr>
3000    ! &
*G/IF DETACHED% /V<cr>
        \ RETURN IF DETACHED% OR NOT PRIV% &
*I/!/V<cr>
        \ RETURN IF DETACHED% !OR NOT PRIV% &
*EX<cr>
Patch from KB:[P,PN]CPATCH.CMD complete
#^Z
File to patch - ^Z

Ready
```

3. Use this procedure if your system default Run-Time System is BASIC-PLUS; otherwise, refer to Section 8.4 of the RSTS/E V7.Ø Release Notes for the correct procedure to compile the program.

   To re-compile the program and re-enter it into the system library, type the following RSTS/E commands.

   ```
   OLD ATPK<cr>

   Ready

   COMPILE SYØ:$ATPK<232><cr>

   Ready
   ```

4. The person making the changes to the program should now take whatever steps are necessary, according to installation guidelines, to save the new version of the program.

5. The source (.BAS) version of the program may now be removed from the public structure.

RSTS/E V7.Ø
Automated Patching Facility Package
ATPK

LIMITING THE USE OF THE $LOGIN COMMAND - ATPK FEATURE PATCH

The $LOGIN command of ATPK provides an easy way for a privileged user to log in under any account at any keyboard. While the $LOGIN command does not compromise normal RSTS/E security (it is available only to privileged users), it may be desirable to disable this feature at installations that have implemented their own security measures.

This optional patch will cause ATPK to ignore any keyboard or account number specification used with the $LOGIN command. The $LOGIN command may still be used to log a privileged user back in under his or her own account at the pseudo keyboard after the job controlled by ATPK has detached or terminated.

PROCEDURE:

1. For purposes of this discussion, we will assume that the program to be patched is located in your privileged account on the public disk structure. If this is not the case, replace all program references with suitable text according to the requirements of your installation.

   We assume the executable version of the program will be stored in the System Library Account ($). If this is not the case, replace references to $ in step 3 below with the appropriate package or library account.

   The procedure below assumes that BASIC-PLUS is your system default run-time system.

   If you are using CSPCOM or BASIC-PLUS-2, refer to Section 8.4 of the RSTS/E V7.Ø Release Notes for the correct procedure to compile the program.

   If you are using BASIC-PLUS-2, this program must be task built against the BP2COM Run-Time System.

2. This patch is contained in a patch file appearing in patch kit version "A" or later. If you do not have this distribution you can produce a command file by specifying a file for [logfile=] in the procedure below.

   To apply the patch manually, perform the following RSTS/E system commands.

```
RUN $CPATCH<cr>
<CPATCH's header line>

File to patch - ATPK.BAS=ATPK.BAS<cr>
#[logfile=]KB:/CS:28893<cr>
*H/2!/V<cr>
2!              PROGRAM         : ATPK.BAS
```

```
*H/2900<tab>/V<cr>
2900     ! &
*G/!\/V<cr>
         !\ T0%=0% &
*-2JDV<cr>
         \ T0%=0% &
*EX<cr>
Patch from KB:[P,PN]CPATCH.CMD complete
#^Z
File to patch - ^Z

Ready
```

3. Use this procedure if your system default Run-Time System is BASIC-PLUS; otherwise, refer to Section 8.4 of the RSTS/E V7.Ø Release Notes for the correct procedure to compile the program.

   To re-compile the program and re-enter it into the system library, type the following RSTS/E commands.

   ```
   OLD ATPK<cr>

   Ready

   COMPILE SY0:$ATPK<cr>

   Ready
   ```

4. The person making the changes to the program should now take whatever steps are necessary, according to installation guidelines, to save the new version of the program.

5. The source (.BAS) version of the program may now be removed from the public structure.

BUILDING THE SORT UTILITY PACKAGE

Section 6.7.3 of the RSTS/E V7.Ø System Generation Manual states that a question will be asked ("Did you SYSGEN in RSX Directives and Resident Libraries?") to determine which of two versions of the SORT program is to be installed. The description of the question indicates that if both of these items have been configured into your system, a more efficient SORT program (utilizing the RMS Resident Library) can be installed. Since publication of the manual, it has been determined that building SORT against this library offers no speed improvement, due to the nature of the SORT ODL file. For this reason, the question has been removed from the SORT BUILD dialogue.

TS11 DEVICE DESIGNATION

The RSTS/E V7.Ø System Generation Manual refers to the new magnetic tape subsystem as TSØ4. As mentioned earlier in these Release Notes, the device is actually named TS11. In addition, the manual states that the maximum number of units supported is 8. The actual maximum number of supported units is 4.

RSTS/E V7.Ø
Documentation
RSTS/E V7.Ø System Manager's Guide

Seq 19.3.1 N

UNDOCUMENTED UTILTY FLAG COMMAND SWITCH - /NOCTG

The following text should be added to Section 7.1.4.8, FLAG Command and Options, of the RSTS/E V7.Ø System Manager's Guide.

The FLAG command accepts a /NOCTG switch, which flags a contiguous file as non-contiguous, allowing it to be extended. The format is as follows:

```
FLAG filename/NOCTG
```

Please note that access to a non-contiguous file may be slower than that to a contiguous file, particularly with random access. Note that this switch returns a "PROTECTION VIOLATION" error if you do not have write access to the file.

UNDOCUMENTED MESSAGE RECEIVER STATUS FLAG

The bit definitions for the access control field in the DECLARE RECEIVER SYS call have been extended to allow a user to indicate that a job should handle one and only one link. Because of this change, new flags have been added to the list of items that can appear in the message receiver status column.

For ANALYS, this flag is "oneshot". For the display programs (VT5DPY and VT5ØPY), screen size limitations necessitate three abbreviations:

Flag	Meaning
Nt	Network senders are allowed for this receiver
1S	The receiver can handle one and only one link
N1	Both of the above

RSTS/E V7.Ø
Documentation
RSTS/E System User's Guide

Seq 19.4.1 N

UNDOCUMENTED BASIC-PLUS FEATURE - THE EXIT KEYBOARD COMMAND

The following feature was added to BASIC-PLUS after the documentation was completed. It will be included in the next release of the RSTS/E System User's Guide.

The EXIT Keyboard command allows you to return to your private default run-time system. The EXIT command accepts no arguments and, when executed, returns you to your private default run-time system. Note that if you are currently under the BASIC-PLUS Run-Time System and it is also your private default, an EXIT command clears the current program from memory and sets up NONAME.

CHAINING BETWEEN BASIC-PLUS PROGRAMS WITH DIFFERENT RUN-TIME SYSTEMS

The RSTS/E System User's Guide describes how a job can select a private default Run-Time System (RTS) which is different from the system default RTS. It states that when a program terminates execution, it returns to the job's default RTS. However, when chaining between programs, it is possible to exit to an RTS other than the job's default if the last program is a source program (no .BAC file exists). If the last program chained to is a source program, it is OLDed and RUN under the RTS which chained to it, even if that RTS is not the job's default. When this program terminates, it exits to the keyboard monitor of the RTS which did the OLD and RUN, regardless of the job's default. For example, suppose a job's default RTS is BASIC, and a program (PROGA), which was compiled under another BASIC-PLUS RTS called BAS2F, is run. If PROGA chains to a program (PROGB) which exists only in source form, the source will be OLDed under BAS2F, not BASIC. When PROGB executes an END statement, it will exit to the BAS2F keyboard monitor.

Note that in the above case, the job's default RTS does not change to BAS2F. It still remains BASIC. Running any compiled program, or typing the "EXIT" command, will cause a return to the BASIC keyboard monitor when the program terminates.

RSTS/E V7.Ø
Documentation
RSTS/E System User's Guide

Seq 19.4.3 N

UNDOCUMENTED PIP.SAV SWITCH

The following section should be added to the PIP.SAV description in the RSTS/E System User's Guide.

16.2.2.13 Blocksize Switch (/BSIZE:n)

The /BSIZE switch is used for output files to magnetic tape to specify a blocksize other than 512 bytes. The /BSIZE switch is only applicable to the output device. Magnetic tapes with blocksizes other than 512 bytes which are read by PIP are handled automatically. This switch is also only applicable to magnetic tape drives. If specified for an output device other than a magnetic tape drive, it will be ignored. The blocksize specification (:n) must be an even integer between 18 and 4Ø96.

There are three restrictions associated with the use of the /BSIZE:n switch.

1. If you are writing an ANSI format magnetic tape intended for interchange with a non-DIGITAL operating system, the blocksize must be between 18 and 2Ø48 bytes.

2. If you intend to interchange data with a DIGITAL RT-11 system, the blocksize must be 512 bytes.

3. The use of the /BSIZE:n switch may increase the amount of buffer space needed by PIP. It is important, therefore, to invoke PIP with a larger buffersize when reading and writing blocksizes larger than 512 bytes. This may be accomplished with either the /SIZE:n switch to the CCL command, or the SIZE n keyboard monitor command while running under the RT11 Run-Time System.

RSTS/E V7.Ø
Documentation
RSTS/E System User's Guide

Seq 19.4.4 N

UNDOCUMENTED SYSTAT MESSAGE RECEIVER STATUS FLAG

The bit definitions for the access control field in the DECLARE RECEIVER SYS call have been extended to allow a user to indicate that a job should handle one and only one link. Because of this change, a new flag ("oneshot") has been added to the list of items that can appear in the message receiver status column.

RSTS/E V7.Ø
Documentation
RSTS/E System User's Guide

Seq 19.4.5 N

UNDOCUMENTED ENHANCEMENT TO THE QUE/AFTER SWITCH

Table 2Ø-2 in the RSTS/E System User's Guide contains a description of QUE Job Output Options. The description of the /AFTER: switch indicates that the user supplied arguments must be in the form "dd-mmm-yy:hh:mm". For example, /AFTER:1Ø-Sep-79:15:2Ø to cause a job to be started after 3:3Ø PM on September 1Ø, 1979. The table should also indicate that the /AFTER: switch allows both relative times and AM/PM time formats.

The /AFTER option arguments that specify the date and time may take one or more of the following forms:

calendar date	in the form :dd-mmm-yy
numeric date	in the form :yy.mm.dd
relative date	in the form :+n D[AYS], where n is the number of days from the current date.
military time	24-hour time in the form :hh:mm
AM/PM time	in the form :hh:mm:AM/PM
relative time	in the form :+n H[OURS], where n is an integer in the range of Ø to 24.

Examples of acceptable /AFTER switch arguments are:

/AF:+1 HOUR	means start the job after 1 hour
/AF:+1 DAY:+Ø HOURS	means start the job this time tomorrow
/AF:+1 DAY	means start the job tomorrow
/AF:+1 DAY:7:ØØ PM	means start the job tomorrow at 7:ØØ PM

Note that the date and time specifications used with the /AFTER option are subject to the following rules:

1. The date specifications (calendar, numeric, and relative), when used without a time specification, cause the job to begin at midnight of the specified date.

2. Only one specification of each type is allowed. For instance, a relative date cannot be used with a calendar date nor can a relative time be used with military time.

3. When a relative time of Ø is used with a relative date specification, it causes the job to begin at the current time on the date specified. For example, if the following option is used a 3:ØØ PM on September 12:

   ```
   /AFTER:+2 DAYS:+Ø HOURS
   ```

   the job is initiated at 3:ØØ PM on September 15.

OBTAINING HELP: THE HELP PROGRAM

A HELP facility was added to RSTS/E V7.Ø after the documentation was completed. A complete description of the facility will be included in the next release of the RSTS/E System User's Guide.

The HELP facility consists of files that contain descriptions of other system programs and system resource commands. HELP is an interactive program that prompts you for topics and subtopics. To obtain information on system programs or commands, you type the desired topic and optional subtopic(s) in response to the HELP prompt. Your response causes HELP to display the contents of the appropriate file.

To invoke the HELP program, type:

```
HELP<cr>
```

When successfully invoked, HELP prints a short description of itself, a list of topics, and a prompt for the desired topic. For example:

```
HELP<cr>

Help can be obtained on a particular topic by typing:
        HELP  topic  subtopic  subsubtopic  ...
A topic can have the following format:
   1) an alphanumeric string (e.g., a command name, option, etc.)
   2) same preceded by a "/" (=> interpreted as a switch)
   3) the match-all symbol "*"
Examples:
    HELP DIRECTORY /S
    HELP SET LC
Abbreviations result in all matches being displayed.

Additional information is available on:

/OUTPUT         /PROMPT
ASSIGN          ATTACH          BASIC           BYE             COPY
DEASSIGN        DISMOUNT        DIRECTORY       EXIT            FILENAMES
FIT             HELP            HELLO           KEYBOARD        LOGIN
MOUNT           PIP             QUE             REASSIGN        RT11
RSTS            RSX             RUN             SET             SWITCH
SYSTAT          TECO            TYPE            VTEDIT

Topic?
```

In response to the Topic? prompt, type one of the following:

1. The name of the system program or resource command you wish information on. If you abbreviate the name, information on all topics that match that abbreviation are displayed. In addition to the program or command, you can specify one or more of the subtopics (separated by spaces) associated with that program or command.

2. An asterisk wildcard (*), which causes HELP to print information on all of the available topics.

By default, HELP displays the requested information on your terminal. To override the default, you can use the /OUTPUT: switch. The /OUTPUT causes HELP to output the requested information to a user file. The switch has one of the following formats:

```
HELP/OUTPUT:filespec topic [subtopic [...]]
HELP/OUTPUT:filespec * [...]
```

where filespec can be a complete file specification as described in Chapter 11 of the RSTS/E System User's Guide. A filename is required; HELP defaults the extension to .LST. For example:

```
HELP/OUTPUT:INFO RSTS MAGTAPE FUNCTION<cr>
```

causes HELP to output the help text for "RSTS MAGTAPE FUNCTION" to a file named INFO.LST in the current account on the public structure.

If you specify a HELP topic for which no file exists, HELP prints:

```
SORRY, NO INFORMATION AVAILABLE ON topic
```

followed by information on valid topics (i.e., as if you had just typed HELP followed by the RETURN key).

Many of the topics described by HELP are further explained in subtopics. For example:

```
Topic? SYS<cr>


SYSTAT

  The SYSTAT program provides current system information.  It may be
  run with the RUN command or the CCL command, "SYS[TAT]".  The
  command string is of the form:
        [output file] [/options]
  Only logged-in users may specify an output file.  If none is
  specified, the output will be to the keyboard.  If no options are
  specified, the status of jobs, devices, disks, buffers, run-time
  systems, resident libraries and message receivers will be reported.
     Examples:
         SYSTAT
         SYS/A

  Additional information is available on:

  /A              /B              /D              /F
  /j              /Kn             /L              /M
  /N              /P              /proj,prog      /proj,*
  /R              /S              /U              /Ø,Ø
  /-
```

```
JOB             WHO             WHERE           WHAT
SIZE            STATE           SWAPPING        RUN-TIME
RTS

SYSTAT Subtopic?
```

In this example, the initial description of the DIRECTORY program (abbreviated to DIR) makes not of additional information available on various switches. To cause HELP to print the additional information, type the desired subtopic(s) in response to the HELP prompt. For example:

```
SYSTAT Subtopic? WHO<cr>

SYSTAT

  WHO

    The "Who" column in the job status report gives the account under
    which the job is running.  This column will contain one of the
    following:

        nn,mm     the job is running under account [nn,mm].
        [OPR]     the job is running under a system operator account.
        [SELF]    the job is running under your account.
        **,**     the job is not logged in.

 SYSTAT Subtopic?
```

The program continues to prompt for subtopics until you type CTRL/Z, which causes HELP to return to the Topic? prompt, or CTRL/C, which terminates the program.

A topic and (optionally) one or more subtopics can be specified with the originial CCL command HELP. In this case, the HELP program will print the requested information and exit without prompting for another topic or subtopic. For example:

```
HELP SYS/B<cr>

SYSTAT

  /B

    Report only busy device status.  Uses the following abbreviations:

      AS      device is explicitly assigned to a job.
      INIT    device is open on a channel.
      DOS     magnetic tape is assigned with DOS labeling format.
      ANSI    magnetic tape is assigned with ANSI labeling format.

Ready
```

If you wish to obtain help and then be prompted for another topic or subtopic, you may use the /PROMPT switch (which may be abbreviated to /PR) with the CCL command HELP. For example:

HELP/PR KEY<cr>

KEYBOARD

Special significance is attached to several characters when they are typed at a terminal keyboard. Several of these characters are "control characters". Control characters can be typed by holding down the CTRL key (as you might hold down the shift key) while striking a character. For example, "type CTRL/C" means "hold down the CTRL key while typing the letter C".

Additional information is available on:

CTRL/C	CTRL/O	CTRL/Q	CTRL/S
CTRL/U	CTRL/Z	ESCape	ALT MODE
RETURN	DELete	RUBOUT	TAB
CTRL/I	CTRL/L		

KEYBOARD Subtopic?

If the /PROMPT switch is used all by itself with no topic, the initial HELP message is suppressed and the HELP program prompts with the Topic? question. For example:

HELP/PROMPT<cr>

Topic?

The HELP program can also be invoked with the RUN command. This has the same effect as typing HELP/PROMPT. For example:

RUN $HELP<cr>

Topic?

RSTS/E V7.Ø
Documentation
RSTS/E System User's Guide

LISTING MEMORY ALLOCATION - THE /C SWITCH OF SYSTAT

The following feature was added to SYSTAT after the documentation was completed. It will be included in the next release of the RSTS/E System User's Guide.

The /C ("Core") switch of SYSTAT will give the privileged user a table showing memory allocations that have been made on the current system. The table is not included by default in any SYSTAT listing; the /C switch must be specified to invoke it. This feature should be particularly useful to the system manager who needs to determine the addresses at which run-time systems and/or resident libraries are to be added and/or loaded.

The memory allocation table lists the following items:

1. System memory allocation as set up by INIT.SYS. For a further discussion of these types of memory allocation, see the RSTS/E V7.Ø System Generation Manual's discussion of memory allocation changes under the DEFAULT option of INIT. This type of entry includes:

    a. Monitor (the resident portion).

    b. XBUF (the extended buffer pool, if present).

    c. Locked out memory (if any).

    d. Non-existent memory (if any exists before the end of physical memory).

2. Resident libraries.

3. Run-time systems that were added at specific addresses, or were loaded with the STAY attribute.

The items displayed are divided into the categories "Permanent" and "Temporary". Permanent items include all the items determined at INIT.SYS time as described above, and run-time systems and resident libraries that are permanently loaded in memory (such as with the /STAY switch of UTILTY). The system default run-time system is always listed as a permanent item. Any memory that is not permanently allocated in one of these ways is labeled as "(User)" in the permanent column.

Temporary items include all run-time systems and resident libraries that are added at a specified address but not permanently loaded. Since resident libraries must be added at specific addresses, all added resident libraries that are not permanently loaded will be listed as temporary items.

For each section of memory described by SYSTAT's memory allocation table, the starting and ending addresses of that portion of memory are given, along with the length of that segment of memory. Run-time systems and resident libraries are listed by their name followed with "RTS" or "LIB", respectively. The terminal output as the result of invoking SYSTAT with the CCL command SYS/C from a privileged account might look like the following:

```
Memory allocation table:
 Start    End  Length  Permanent    Temporary
   ØK -   37K (  38K) MONITOR
  38K -   53K (  16K) BASIC  RTS
  54K -   61K (   8K) ** XBUF **
  62K -   84K (  23K)   (User)     RMSRES LIB
  85K -  12ØK (  36K)   (User)
 121K -  123K (   3K)   (User)     RSX    RTS
 124K -  *** END ***
```

Other run-time systems may be added and may or may not be in memory at the time the table is listed. The table shows only those items that have been assigned to specific addresses and does not reflect any dynamic state of memory.

If run-time systems are added with specified addresses in such a way that they overlap (this is not recommended), the table will show each overlapping section. For example, if the RT11 Run-Time System (size 4K) were added to the system shown above at address 12Ø, the following entries would be shown (again, this is NOT recommended):

```
 12ØK -  12ØK (   1K)   (User)     RT11   RTS
 121K -  123K (   3K)   (User)     RT11   RTS, RSX    RTS
```

In general, the table will divide memory into as many sections as is necessary to show the beginning and end of each item that has been allocated to a particular range of memory.

NOTE

This example shows run-time systems added at specified addresses. This is not an indication that they should always be added with specific addresses; whether the performance of a system would be improved by giving specific addresses to any or all run-time systems is determined by that particular system configuration and usage.

ADDITION TO CTRL/T JOB STATES

In addition to those already documented in the RSTS/E System User's Guide, the following mnemonics can be returned in the "job states" report generated by CTRL/T (FP(fff)):

UUO	process a UUO directive
WIN	window turn for disk files
EXT	extend a disk file
BYE	logout a user

RSTS/E V7.Ø
Documentation
BASIC-PLUS Language Manual

Seq 19.5.1 N

CORRECTION TO RECORDSIZE DESCRIPTION

Sections 1Ø.3 and 1Ø.4.3 of the BASIC-PLUS Language Manual describe the use of the RECORDSIZE option to specify the size of internal buffers for virtual arrays. However, the description of the legal RECORDSIZE arguments is misleading.

The sentence in Section 1Ø.3 which reads:

> If specified, the RECORDSIZE must be an even multiple of 512.

should read:

> If specified, the RECORDSIZE argument must be one of the following powers of 2:
>
> 512, 1Ø24, 2Ø48, 4Ø96, 8192, 16384

RSTS/E V7.Ø
Documentation
RSTS/E Programming Manual

UNDOCUMENTED CHANGE TO THE EXIT AND CLEAR PROGRAM SYS CALL

SYS call 9 (Exit and Clear program) has been extended to allow you to specify that you wish to use another run-time system without making the new run-time system your private default run-time system. This call, referred to as the "temporary switch" call, is equivalent to the old BASIC-PLUS keyboard command "NEW <rts>.RTS", which no longer exists. (This form of the SYS call has NOT yet been implemented in BASIC-PLUS-2).

This feature was added to BASIC-PLUS after the documentation was completed. It will be included in the next release of the RSTS/E Programming Manual. The following information should be added to the description of SYS call 9:

SYS Call Extension for Exit and Clear Program

Data Passed:

Byte(s)	Meaning
1	CHR$(9%), the exit and set up NONAME code.
2-3	The first 3 characters of the run-time system name, in Radix-5Ø format, to which control is to pass. If bytes 2-5 are zero, your private default run-time system is assumed.
4-5	The last 3 characters of the run-time system name, in Radix-5Ø format, to which control is to pass.
6	If not specified, the named run-time system is established as the job's private default run-time system. Otherwise, CHR$(N%), where N% is: 255% establish the run-time system as the job's private default run-time system. Ø% enter the specified run-time system but without establishing it as the job's default.

Discussion: If byte 6 is specified with a value of Ø, it causes a temporary switch to the run-time system named in bytes 2-5.

RSTS/E V7.Ø
Documentation
RSTS/E Programming Manual

UNDOCUMENTED CCL SYS CALL

The CCL SYS call (SYS call 14) has been added to allow BASIC-PLUS programs to execute CCL commands as though they had been typed directly to a keyboard monitor. Note that the successful execution of a CCL command will terminate execution of the current program. This SYS call can be disabled with the optional feature patch described in the article, Seq 4.1.7 F.

(SYS call 13 was reserved because of its similarity to FIP call 13, which zeroes the user's account if no device name is specified.)

This feature was added to BASIC-PLUS after the documentation was completed. It will be included in the next release of the RSTS/E Programming Manual. A description of the SYS call follows.

Execute CCL Command - Not Privileged (FØ=14)

Data Passed:

Byte(s)	Meaning
1	CHR$(14%), the execute CCL command code.
2-128	The string to be executed.

Data Returned: The target string is equivalent to the passed string.

Possible Errors:

	Meaning	ERR Value
?LINE TOO LONG	The string that was passed is too long to be executed as a CCL command. Note that the Monitor expands CCL abbreviations to their full command.	47
?ILLEGAL NUMBER	A non-numeric value was used as an argument in one of the CCL switches. For example, a /SIZE:A switch specification can cause this error.	52
?ILLEGAL SWITCH USAGE	An illegal switch was specified for the CCL command. For example, requesting a size that is larger than the system's SWAP MAX can cause this error.	67

Discussion: This call causes the Monitor to scan the string in bytes 2-128 to determine if it is a valid CCL command. If the string is valid, the command is executed as though it had been typed directly to a keyboard monitor. If the string is not valid, the monitor returns control to the caller and execution continues at the next program statement.

Note that additional errors can be detected after the current program is removed and execution of the CCL is attempted. For information on this condition, refer to the RSTS/E System Directives Manual.

RSTS/E V7.Ø
Documentation
RSTS/E Programming Manual

UNDOCUMENTED DATE AND TIME CONVERSION SYS CALL

A date and time conversion SYS call (SYS call 2Ø) has been added to allow any user to override the system date and time defaults. This feature was added to BASIC-PLUS after the documentation was completed. It will be included in the next release of the RSTS/E Programming Manual. A description of the SYS call follows.

Date and Time Conversion - Not Privileged (FØ=2Ø)

Data Passed:

Byte(s)	Meaning
1	CHR$(6%), the SYS call to FIP.
2	CHR$(2Ø%), the date and time conversion code.
3-4	CHR$(N%)+CHR$(SWAP%(N%)), where N% is the date to be converted or Ø for the current date.
5-6	CHR$(N%)+CHR$(SWAP%(N%)), where N% is: Ø use the system default format. <Ø use alphabetic date format. >Ø use ISO date format.
7-16	Not used.
17-18	CHR$(N%)+CHR$(SWAP%(N%)), where N% is the time to be converted or Ø for the current time.
19-2Ø	CHR$(N%)+CHR$(SWAP%(N%)), where N% is: Ø use the system default format. <Ø use AM/PM time format. >Ø use 24-hour time format.
21-3Ø	Not used.

Data Returned:

Byte(s)	Meaning
1	The current job number times two.
2	Not used.
3-6	Same as data passed.

7-16	The date string, padded to the right with zeroes.
17-2Ø	Same as data passed.
21-3Ø	The time string, padded to the right with zeroes.

Possible Errors: No errors are possible; however, if bytes 3-4 or 17-18 contain illegal date or time values, random output is returned.

Discussion: This call can be used in programs that need to override the system date and time defaults.

RSTS/E V7.Ø
Documentation
RSTS/E Programming Manual

UNDOCUMENTED SNAP SHOT DUMP SYS CALL

A snap shot dump SYS call (SYS call -27) has been added to allow a privileged user to take an on-line "snap shot dump". This feature was added to BASIC-PLUS after the documentation was completed. It will be included in the next release of the RSTS/E Programming Manual. A description of the SYS call follows.

Snap Shot Dump - Privileged (FØ=-27)

Data Passed:

Byte(s)	Meaning
1	CHR$(6%), the SYS call to FIP.
2	CHR$(-27%), the snap shot dump code.
3-3Ø	Not used.

Data Returned: No meaningful data is returned.

Possible Errors:

Meaning	ERR Value
?CAN'T FIND FILE OR ACCOUNT The call attempted to write data to the file CRASH.SYS but crash dump was not enabled. To enable crash dump, the system manager must answer yes to the CRASH DUMP question during system generation.	5
?PROTECTION VIOLATION A non-privileged user attempted to execute this call.	1Ø
VARIOUS DEVICE DEPENDENT ERRORS This call may return device dependent errors such as DEVICE HUNG or DISK PACK IS NOT MOUNTED.	

Discussion: The execution of this call writes the current monitor image executing in memory to the crash dump file [Ø,1]CRASH.SYS. The contents of the CRASH.SYS file can then be analyzed with the ANALYS program as described in the RSTS/E V7.Ø System Manager's Guide.

RSTS/E V7.Ø
Documentation
RSTS/E Programming Manual

UNDOCUMENTED SPOOLING SYS CALL

The spooling SYS call (SYS call -28) has been added to RSTS/E V7.Ø to allow a user program to spool one or more files to any spooler without CHAINing to QUE. The SYS function is not privileged, but will enforce restrictions due to file protection codes.

This feature was added to BASIC-PLUS after the documentation was completed. It will be included in the next release of the RSTS/E Programming Manual. A description of the SYS call follows.

Spooling - Not Privileged (FØ=-28)

Data Passed:

Byte(s)	Meaning
1	CHR$(6%) the SYS call to FIP.
2	CHR$(-28%), the spool request code.
3-4	Not used.
5-6	The project-programmer number of the file to spool. If bytes 5-6 are zero, the current user account is used.
7-1Ø	The filename (which can include wildcards), in Radix-5Ø format, of the file to spool.
11-12	The file extension (which can include wildcards), in Radix-5Ø format, of the file to spool.
13-14	The two character ASCII device name to spool the file to. If bytes 13-14 are zero, LP is used.
15	The unit number of the spooling device specified in bytes 13-14.
16	The unit number flag of the spooling device specified in bytes 13 and 14 (-1 if byte 15 contains an actual unit number, Ø if bytes 13-14 contain a generic device name).
17-18	Reserved, must be zero.
19-2Ø	The flag word passed to QUEMAN (these values can be combined), as follows: 1% = File is spooled with FORTRAN carriage control, equivalent to QUE /TYP:FTN option. 2% = Restart, equivalent to the QUE /RE option. 4% = Delete the file after spooling; equivalent to the QUE /DE option.

8% = Binary file; equivalent to the QUE /BI option.

16% = End, equivalent to the QUE /END option.

32% = No header, equivalent to the QUE /NH option.

21-22 Not used.

23-24 The device name where the file to be spooled can be found. The device must be a disk. If bytes 23-24 are zero, SY: (the public structure) is used.

25 The unit number of the device containing the file to be spooled. This byte is ignored if byte 26 is zero.

26 The unit number flag of the device containing the file to be spooled. If non-zero, this indicates that the unit number specified in byte 25 is real.

27-3Ø Not used.

Data Returned: No meaningful data is returned.

Possible Errors:

Meaning	ERR Value
?NO ROOM FOR USER ON DEVICE The number of messages pending for the queue is at its declared maximum. This may be a transient condition, retry the operation.	4
?CAN'T FIND FILE OR ACCOUNT The account specified in bytes 5-6 does not exist on the device specified, the filename or extension specified in bytes 7-12 cannot be found, or QUEMAN is not installed as a message receiver.	5
?NOT A VALID DEVICE An attempt was made to spool a file to a spooling device that had a unit number greater than 7, or an attempt was made to spool a file from a non-disk or invalid device.	6
?PROTECTION VIOLATION An attempt was made to queue a file to which the user did not have read access, queue a compiled file, or queue a file which has the privileged <128> bit set and the user is not privileged.	1Ø
?DEVICE HUNG OR WRITE LOCKED This error is caused by a hardware	14

condition.

?DISK PACK IS NOT MOUNTED 21
The specified disk device is not mounted; logically mount the disk with UTILTY or INIT.

?DISK PACK IS LOCKED OUT 22
The disk is in a locked state. Execute the call under a privileged account to override this condition.

?DEVICE NOT FILE STRUCTURED 3Ø
The device specified in the call is not a file-structured device.

?NO BUFFER SPACE AVAILABLE 32
System buffers are not currently available to store this message. This may be a transient condition, retry the operation.

Discussion: This call allows you to spool one or more files without chaining to QUE. When this call is executed, the Monitor performs the following checks:

1. Ensures that the specified filename is legally formatted.

2. Ensures that the specified device (the device containing the file to be spooled) is a mounted RSTS/E file structured disk and that the caller has appropriate privilege to access it.

3. If there are no wildcards specified in bytes 7-12, ensures that the specified file exists and that the caller has appropriate privilege to access it.

4. Performs all appropriate send/receive buffer quota checks and ensures that QUEMAN is available (not hibernating).

If any of these conditions are not met, the call is aborted and an error is returned (see Possible Errors). If all the conditions are met, the monitor formats a message and sends it to QUEMAN. The monitor places no restriction on the device you can spool to; this call may be used to any spooler running under the control of QUEMAN.

ERRORS RETURNED BY THE BROADCAST AND FORCE TO A TERMINAL SYS CALLS

Sections 7.2.5.4 and 7.2.5.5 of the RSTS/E Programming Manual describe the broadcast and force to a terminal SYS calls. The sections describing "Possible Errors" when using these calls are incorrect and should read as follows:

Possible Errors:

Meaning	ERR Value
?Protection Violation The job is not privileged or byte 3 contains an illegal KB: number.	1Ø
?Illegal byte count for I/O An attempt was made to broadcast a zero-length string.	31

CHANGE TO THE DECLARE RECEIVER SYS CALL

If a network program which handles incoming CONNECTs kills itself on receiving a DISCONNECT, there is a window between the time the DISCONNECT is received and the job removes itself as a receiver. During this period, the job is an eligible receiver for another CONNECT. If NSP queues a CONNECT INITIATE in this time window and the job does a REMOVE receiver or kills itself, the CONNECT will be rejected.

In order to alleviate this race condition, a new bit (value=8) has been defined in the access control field of the DECLARE RECEIVER SYS call. If this bit is set, NSP sets the receiver's link max to Ø after queueing one CONNECT INITIATE message. This prevents the possibility that NSP will queue a CONNECT INITIATE after its last logical link is disconnected and before it issues a REMOVE RECEIVER. This guarantees that a job will receive one and only one link.

RSTS/E V7.Ø
Documentation
RSTS/E Programming Manual

CORRECTION TO RETURN JOB STATUS SYS CALL DESCRIPTION

Section 7.2.25 of the RSTS/E Programming Manual describes the Return Job Status SYS call. The section describing bytes 5-6 of the data returned portion (S% is Ø%) is incorrect. Specifically, the sentence which reads "byte 5 contains the controlling job's job number times two..." should be "byte 5 contains the controlling job's job number times two plus one...".

RSTS/E V7.Ø
Documentation
RSTS/E Programming Manual

Seq 19.6.9 N

BASIC-PLUS SYS CALL DOES NOT ALLOW USER TO DEASSIGN ALL USER LOGICALS

Section 7.2.4.5 of the RSTS/E Programming Manual is incorrect. Its title, which currently reads:

Deassign All Devices and All User Logicals

should be changed to read:

Deassign All Devices.

Also, the sentence under the example heading, which reads:

> The following statement deassigns all devices currently assigned to the job and all user logicals currently assigned.

should be changed to read:

> The following statement deassigns all devices currently assigned to the job.

RSTS/E V7.Ø
Documentation
RSTS/E Programming Manual

Seq 19.6.1Ø N

CHANGE TO THE CREATE A JOB SYS CALL

Section 7.2.23 of the RSTS/E Programming Manual describes the Create a Job SYS call (SYS call 24). This call has been extended to allow you to specify that the job should be created even if logins are disabled (i.e., "spawn regardless"). The following change should be made to the section describing bytes 3 and 4 of the data passed:

Data Passed:

Byte(s)	Meaning
3	Not used, must be zero.
4	CHR$(N%), where N% is:
	Ø create the job only if logins are enabled.
	128 create the job even if logins are disabled.

The section describing "Possible Errors" when using this call should be changed to read:

Possible Errors:

Meaning	ERR Value
?NO ROOM FOR USER ON DEVICE The new job can not be created. Probable causes are: a. further logins are disabled and byte 4 was specified as zero. b. the system's job or swap slots are (currently) full.	4

ACCESSING TU58 DECTAPE II

The TU58 DECtape II is a block structured, mass storage, random access device. The device contains 512 blocks (each of which is 512 bytes long) and, as with the TU56 DECtape, specific blocks on the tape can be accessed. However, unlike the TU56, only non-file structured processing is allowed.

To initiate processing on the TU58, the following two forms of the OPEN statement are allowed:

```
200  OPEN "DD1:" AS FILE 4%

200  OPEN "DD1:" FOR INPUT AS FILE 4%
```

the OPEN FOR OUTPUT statement is not allowed.

After opening the TU58 DECtape II, BASIC-PLUS GET and PUT statements can be used to read and write specific physical blocks on the tape. For example:

```
300  GET #4%, RECORD 6%, COUNT N%

500  PUT #4%, RECORD R%, COUNT N%
```

where COUNT N% is greater than zero. The value for RECORD is interpreted as the block number and is in the range Ø to 511. Note that if the COUNT value in a PUT statement is not a multiple of 512 bytes, the remainder of the last block written is padded with NULs. Also, unlike the TU56, TU58 DECtape II does not allow the program to read or write backwards (the RECORD value must be positive).

RESCOM AND RESLIB INPUT OPTION SPECIFICATION CORRECTION

Sections 3.2.4.3 and 7.3.1 of the RSTS/E Task Builder Reference Manual Update state that the RESCOM and RESLIB options accept filename, extension, and account specifications but do not accept device and unit number specifications. This is incorrect. RESLIB and RESCOM accept filename, account, and device and unit number specifications. That is, extensions are not allowed; device and unit numbers are allowed.

PAR OPTION

Because of the manner in which the Task Builder stores the library name, the filename in a task, symbol table, and PAR option specification must be the same.

UU.SPL - FILE SPOOL UUO SUBFUNCTION

The UU.SPL .UUO subfunction provides a means for a RSTS/E user to spool a file to a program on-line to the queue manager (QUEMAN), without having to chain to QUE. To utilize this directive, the FIRQB must contain the following information:

Data Passed -- File spool directive

FIRQB

Offset Octal	Mnemonic		Offset Octal	Mnemonic
1		//	Ø	
3	FQFUN	UU.SPL = -28. //////////////////////	2	
5		//	4	
7		project number | programmer number	6	FQPPN
11		file name (2 words in RAD5Ø format)	1Ø	FQNAM1
13			12	
15		file extension (1 word in RAD5Ø format)	14	FQEXT
17		Device name to spool to (2 bytes ASCII)	16	
21		<>Ø,device no. real | spool device unit no.	2Ø	
23		must be zero	22	FQMODE
25		flag word passed to the queue manager	24	
27		//	26	
31		Device name for file (2 bytes ASCII)	3Ø	FQDEV
33		<>Ø,device no. real | device unit number	32	FQDEVN
35		//	34	
37		//	36	

Data returned:

Other than a possible error in byte Ø of the FIRQB, no data is returned by the UU.SPL .UUO subfunction.

The flag word passed to QUEMAN in bytes 24 and 25 of the FIRQB has the following meaning:

1 = file is spooled with FORTRAN carriage control.

2 = restart, equivalent to the QUE /RE option.

4 = delete the file after spooling, equivalent to the QUE /DE option.

8 = binary file, equivalent to the QUE /BI option.

16 = end, equivalent to the QUE /END option.

32 = no header, equivalent to the QUE /NH option.

UU.CNV - DATE AND TIME CONVERSION UUO SUBFUNCTION

The date and time .UUO subfunction allows any user to override the system date and time defaults. To utilize this directive, the FIRQB must contain the following information:

Data Passed -- Date and Time Conversion

FIRQB

Offset Octal	Mnemonic		Offset Octal
1		//////////	Ø
3	FQFUN	UU.CNV = 2Ø. //////////	2
5		internal form date	4
7		date flag word	6
11		//////////	1Ø
13		//////////	12
15		//////////	14
17		//////////	16
21		//////////	2Ø
23		internal form time	22
25		time flag word	24
27		//////////	26
31		//////////	3Ø
33		//////////	32
35		//////////	34
37		//////////	36

FIRQB+4 See the .DATE directive for a discussion of this format.

FIRQB+6 This word is used to indicate the type of date format desired:

=Ø Use system default format

<Ø Use alphabetic month format

>Ø Use ISO format

FIRQB+22 See the .DATE directive for a discussion of this format.

FIRQB+24 This word is used to indicate the type of time format desired:

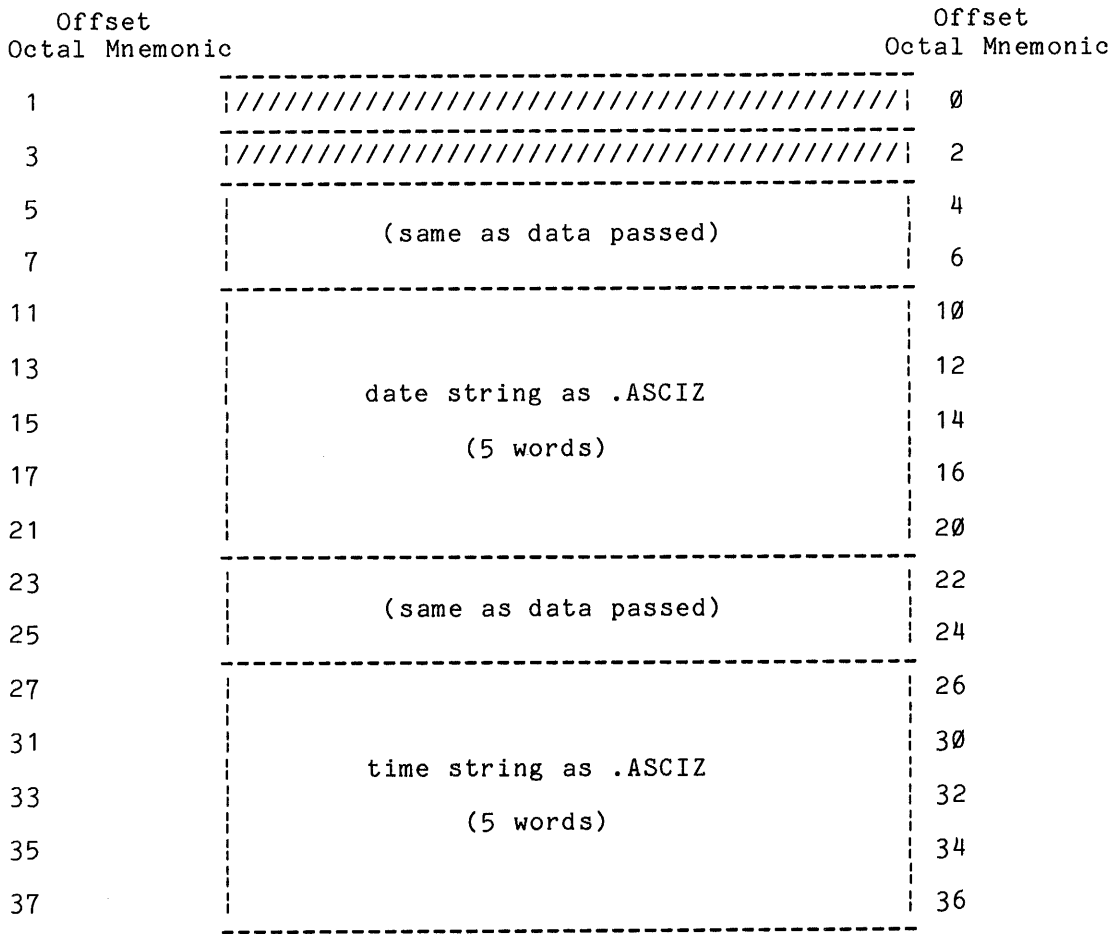
=Ø Use system default format

<Ø Use AM/PM format

>Ø Use 24-hour format

Data Returned -- Date and Time Conversion

FIRQB



FIRQB+1Ø "DD-MMM-YY" or "YY.MM.DD"

FIRQB+26 "HH:MM XM" or "HH:MM "

No errors are possible with the UU.CNV subfunction. However, if bytes 3-4 or 17-18 contain illegal date or time values, random output is returned.

RSTS/E V7.Ø
Documentation
RSTS/E System Directives Manual

Seq 19.24.3 N

CORRECTION TO GTSK$ DIRECTIVE

The description of the GTSK$ (Get Task Parameters) directive in the RSTS/E System Directives Manual is incorrect. The description of buf+34 (the system in which the program/job/task is running) should read:

Ø = RSX-11D
1 = RSX-11M
2 = RSX-11S
3 = IAS
4 = RSTS/E
5 = VAX/VMS
6 = RSX-11M+ (TRAX)
7 = RT-11

RSTS/E V7.Ø
Documentation
RSTS/E System Directives Manual

Seq 19.24.4 N

TYPOGRAPHICAL ERROR IN SECTION 5.1Ø

Section 5.1Ø of the RSTS/E System Directives Manual contains a typographical error. The paragraph tha reads:

> Some older Run-Time Systems based on RSX... do not eliminate this directive and will return IE.SDP error code.

should be changed to read:

> Some older Run-Time Systems based on RSX... do not emulate this directive and will return IE.SDP error code.

RSTS/E V7.Ø
RSX Emulator and Utilities Package
Package Notes

USING RTSODT FOR DEBUGGING RUN-TIME SYSTEMS

A version of ODT is provided on the RSTS/E V7.Ø distribution kit as an unsupported feature which can be linked with user-written run-time systems.

To use RTSODT with run-time systems that are assembled and task-built under the RSX Run-Time System you may specify "RTSODT" in the ODL file. For example:

```
; ODTBLD.ODL
;
;       *** RSXODT.RTS OVERLAY DESCRIPTION ***
;
; RSXODT VERSION V7.Ø
;

        .NAME   RSXODT
        .ROOT   RSXODT-LB:SYSLIB/DL-RSXEMU-RSXPMD-RSXRUN-RSXPLA-RSXKBM-ODT
;
; RSXRTS MUST BE THE FIRST MODULE LINKED IN ANY RSX BASED RTS
; RSXODT MUST BE THE LAST MODULE LINKED IN IF YOU WANT ODT
;
RSXEMU: .FCTR   LB:SYSLIB/LB:RSXRTS:RSXIO:RSXAST:RSXSST:RSXDIR
RSXPMD: .FCTR   LB:SYSLIB/LB:RSXPMD
RSXRUN: .FCTR   LB:SYSLIB/LB:RSXRUN
RSXPLA: .FCTR   LB:SYSLIB/LB:RSXPLA
RSXKBM: .FCTR   LB:SYSLIB/LB:RSXKBM:RSXAT:RSXHLP:RSXMCR
ODT:    .FCTR   LB:SYSLIB/LB:RTSODT
        .END
```

Refer to article Seq 22.1.3 N for a description of the use of RTSODT with run-time systems that are assembled and linked under the RT11 Run-Time System.

RSTS/E V7.Ø
RSX Emulator and Utilities Package
RSX.RTS

MAKING TASK IMAGES NON-SWAPPABLE - RSX.RTS FEATURE PATCH

Sometimes a situation exists where it would be desirable to lock a job in memory so that it would not be swapped out between run bursts. The RSTS/E Programming Manual describes the BASIC-PLUS SYS call which locks a job in memory. This article describes how task images may be made non-swappable through the use of a Task Builder switch.

CAUTION

> Locking jobs in memory may cause fragmentation of user space and is normally not recommended. Use of the capability described in this article may cause severe system performance degradation.

The Task Builder (TKB) switch /CP indicates a "checkpointable" or swappable task or job. This is the default for all .TSK files built by the V7.Ø TKB. By specifying /-CP for the task output file, e.g.,

```
TKB TASK/-CP,MAP=OBJ
```

the resulting task image will be marked as non-swappable. When the task is run, if the patch described below is installed in the Run-Time System, the job will be locked for the duration of its execution. The job will be unlocked when the task terminates and "Ready" (or an equivalent prompt) is printed.

The following patch applies only to the RSX.RTS distributed with V7.Ø, and does not apply to any other Run-Time System. Thus, the only task images which can be locked in memory are those built to run under the RSX Run-Time System. This excludes all task images built using the HISEG option.

PROCEDURE:

1. This is a feature patch to the RSX Run-Time System only.

2. The patch described in Step 4 below can be installed using the PATCH option of INIT.SYS:

```
Option: PATCH
File to patch? RSX.RTS                    (RSX Run-Time System name)
```

3. This patch can be installed manually using ONLPAT, the on-line patching program:

```
RUN $ONLPAT
Command File Name? <cr>                   (RETURN for manual patch installation)
File to patch? [Ø,1]RSX.RTS               (RSX Run-Time System name)
File found in account [Ø,1]
```

   This patch is contained in a patch file appearing in patch kit version "A" or later.

4. The patch is as follows:

```
Base address? ..NSW
Offset address? Ø
 Base   Offset  Old     New?
??????  ØØØØØØ  ØØØ24Ø  ? .SET
??????  ØØØØØ2  Ø137ØØ  ? ^C          (up-arrow/C to exit;CTRL/C for INIT)
```

5. If the above patch was installed using ONLPAT, it will take effect the next time the run-time system is reloaded. If the run-time system has been ADDed, or if it is your system default run-time system, execute the following commands:

```
RUN $UTILTY<cr>
<UTILTY's header line>
#UNLOAD RSX<cr>                        (RSX Run-Time System name)
#EXIT<cr>

Ready
```

NOTE: The UNLOAD command will not remove the run-time system, but simply instructs the monitor to reload it the next time a job requests it.

If, in addition to being previously ADDed, the run-time system had been ADDed with the /STAY switch, also reload it as follows:

```
RUN $UTILTY<cr>
<UTILTY's header line>
#LOAD RSX/STAY/ADDR:nnn<cr>            (RSX Run-Time System name)
#EXIT<cr>

Ready
```

RSTS/E V7.Ø
RSX Emulator and Utilities Package
RSX.RTS

DISABLING POST-MORTEM DUMPS FOR CTRL/C ABORTS - RSX.RTS FEATURE PATCH

The Post-Mortem Dump capability of the V7.Ø RSX Run-Time System is enabled for task images built with the Task Builder /PM switch. (See the RSTS/E Task Builder Manual.) When such a task is executed, a post-mortem dump file is created if the task is aborted for any reason. Thus, the error condition which caused the task to abort may be analyzed through examination of the saved job image.

Strictly speaking, typing CTRL/C to terminate the execution of a task is to abort it. For example, the program might be in an infinite loop, and, unless CTRL/C were typed, the program would continue to execute indefinitely. In such a case, a post-mortem dump caused by typing CTRL/C may reveal what the program was doing over and over again. However, it may not always be desirable to generate a dump at the typing of CTRL/C, while at the same time a dump is desired for other kinds of aborts. This article describes a feature patch to the RSX Run-Time System to disable the generation of post-mortem dumps when CTRL/C is typed to abort any task.

PROCEDURE:

1. This is a feature patch to the RSX Run-Time System.

2. The patch described in Step 4 below can be installed using the PATCH option of INIT.SYS:

```
Option: PATCH
File to patch? RSX.RTS                    (RSX Run-Time System name)
```

3. This patch can be installed manually using ONLPAT, the on-line patching program:

```
RUN $ONLPAT
Command File Name? <cr>                   (RETURN for manual patch installation)
File to patch? [Ø,1]RSX.RTS               (RSX Run-Time System name)
File found in account [Ø,1]
```

   This patch is contained in a patch file appearing in patch kit version "A" or later.

4. The patch is as follows:

```
Base address? ..CCPM
Offset address? Ø
 Base   Offset  Old     New?
??????  ØØØØØØ  ØØØ4Ø4  ? 24Ø
??????  ØØØØØ2  123727  ? ^C              (up-arrow/C to exit;CTRL/C for INIT)
```

5. If the above patch was installed using ONLPAT, it will take effect the next time the run-time system is reloaded. If the run-time system has been ADDed, or if it is your system default run-time system, execute the following commands:

```
RUN $UTILTY<cr>
<UTILTY's header line>
#UNLOAD RSX<cr>                          (RSX Run-Time System name)
#EXIT<cr>

Ready
```

NOTE: The UNLOAD command will not remove the run-time system, but simply instructs the monitor to reload it the next time a job requests it.

If, in addition to being previously ADDed, the run-time system had been ADDed with the /STAY switch, also reload it as follows:

```
RUN $UTILTY<cr>
<UTILTY's header line>
#LOAD RSX/STAY/ADDR:nnn<cr>              (RSX Run-Time System name)
#EXIT<cr>

Ready
```

RSTS/E V7.Ø
RSX Emulator and Utilities Package
CRF.TSK

USING THE RSX CRF UTILITY

An RSX CRF.TSK utility is included on the RSTS distribution kit as an unsupported product. It is not, therefore, copied from the distribution by any of the build procedures. This program will create cross reference listings from the binary cross reference (.CRF) files optionally produced by the RSX task builder (TKB) and and RSX MACRO assembler (MAC). The cross reference file created by MAC lists page and line number references to data items and routines whereas the cross reference file created by TKB lists tha names of the object modules which reference these items. The following are examples of these two types of output:

MACRO Assembler output:

```
CRF        CREATED BY  MACRO  ON 15-JUL-79 AT Ø6:Ø2     PAGE 1
SYMBOL CROSS REFERENCE                                   CREF   Ø7.Ø65
SYMBOL  VALUE           REFERENCES
CR      = ØØØØ15        #4-18Ø
FF      = ØØØØ14        #4-18Ø
HT      = ØØØØ11        #4-18Ø
LF      = ØØØØ12        #4-18Ø
L$$IST  = ******         1-1        1-2        2-53       3-1        4-182
RSTS    = ØØØØØ1        #4-39
R$$11M  = ØØØØØØ        #4-38
SPA     = ØØØØ4Ø        #4-18Ø
VT      = ØØØØ13        #4-18Ø
V1145   = ******         4-157
$$$VER  = Ø34Ø66         #3-2       #5-1
```

Task Builder output:

```
CRF        CREATED BY  TKB    ON 15-JUL-79 AT Ø6:23     PAGE 1

GLOBAL CROSS REFERENCE                                   CREF   Ø7.Ø65

SYMBOL  VALUE      REFERENCES...

A.BTTN  ØØØØØ2    # DIRSYM
A.DFUI  ØØØ1Ø2      CRFIN    # FCSGBL
A.LULU  ØØØØØ2    # DIRSYM
A.LUNA  ØØØØØ4    # DIRSYM
A.LUNU  ØØØØØ6    # DIRSYM
A.TRBA  ØØØØØ2    # DIRSYM
BADDIR  ØØØØØ1    # ERR       FIP
BADNAM  ØØØØØ2    # ERR       FIP
```

More details on the CRF output can be found in the RSX-11 Utilities Procedures Manual.

To request that MAC or TKB produce a binary cross reference file, the /CR switch must be specified in the command line to these programs. For MAC the switch is included with the listing file specification and for TKB it is included with the map file specification. The binary file created has the same name as the listing or map file and its extension is always .CRF.

CRF.TSK is run using either the RUN command or the CRF CCL if installed. When run using the RUN command, CRF will prompt with "CRF>". Typing the CTRL/Z combination to this prompt will cause an exit to your default RTS. The command line is a single RSTS/E file specification of the file into which the cross reference listing is to be placed. The default extension for this file is .LST. If the file specified already exists, CRF appends the listing to it, otherwise a new file is created. The CRF output is variable length records and, therefore, if the file specified for output already exists, it must also be a variable length file. The listing file produced by MAC and the map file created by TKB are both variable length files making them prime candidates.

CRF also accepts as input an indirect command file (preceded by "@") which can contain a list of file specifications. Each specification in the file will be processed as if they were separately entered to the CRF> prompt. The default extension for the command file is .CMD. CRF only allows one level of command indirection.

CRF requires the binary cross reference file to have the same name as the output file with an extension of .CRF. CRF always deletes the binary .CRF file when it is done. Therefore, if it is desirable to keep the binary file, it must be copied to a non-.CRF file prior to running CRF.TSK.

THE RT-11 V3 EMULATOR FOR RSTS/E V7.Ø

The RT-11 emulator is designed to allow many standard RT-11 CUSPs (e.g., MACRO and LINK) to run under RSTS/E. This is done by intercepting calls to the RT-11 monitor (EMTs) and translating them to the corresponding calls to the RSTS/E executive. The environment provided is similar to an RT-11 single job system. The commands and EMTs supported by the emulator are described below.

The "Prefix EMT" under the RT-11 Emulator

In order to execute RSTS/E directives directly from programs running under the RT-11 Emulator, every RSTS/E directive must be preceded by the special "prefix EMT" (EMT 377). the prefix EMT is described in the RSTS/E System Directives Manual (Section 2.5.1, "PF.EMT"), and directs the RSTS/E executive to interpret the word which immediately follows as a RSTS/E directive rather than as an RT-11 directive to be processed by the RT-11 Emulator.

RT Keyboard Monitor KMON commands

When the emulator is used as a keyboard monitor, the user controls execution with commands to the RT-11 Keyboard Monitor. Available commands are listed below and may be used whenever the RT-11 "." prompt has been printed. The part of the command name listed in brackets is optional. Arguments must be separated from the command name by a blank.

SI[ZE] [n]

Sets current memory size to nK words of memory. About 1K of this is used by the emulator for read/write scratch space; the rest is available to the program being run. If the 'n' argument is omitted, the current allocation is printed on the keyboard.

RU program

Runs a program. If no PPN is specified, the program is assumed to be in the current user's account. The program is loaded into memory via the standard scatterload technique (see the RT-11 documentation set), and may be preceded by 'GET' commands.

GE[T] program

Loads a program into memory (as with 'RU'), but does not start it.

R program

Loads a program into memory (without scatterloading it) and starts it. If no PPN is specified, the program is assumed to be in the current library account (see the 'LIB' command).

RE[ENTER]

Re-enter a program at its reentry address. This command is used to restart a CUSP if it has been stopped with CTRL/C or CTRL/Z.

ST[ART]

Start a program at its start address. It must be loaded in memory using the GET command.

CL[OSE]

Closes all currently open files (see the RT-11 System User's Guide).

DA[TE]

Prints today's date.

TI[ME]

Prints the current time of day.

E, D, and B

Examine, deposit, and base commands work as described in the RT-11 System User's Guide.

VE[RSION]

Prints the current RT-11 emulator version header.

RSTS/E Standard Commands

RUN program

Runs a program. If no PPN is specified, the program is assumed to be in the current user's account. The program is invoked using the RSTS/E .RUN function (refer to the RSTS/E System Directives Manual).

AS[SIGN] arg

Similar to the BASIC-PLUS ASSIGN command. 'AS phy:log' assigns a logical name for a physical device. 'AS phy:' assigns the named device to your job. 'AS [p,pn]' associates the named PPN with the character '@'.

DEA[SSIGN] [arg]

Similar to the BASIC-PLUS DEASSIGN command. 'DEA phy:' deassigns the named device from your job. 'DEA log' deassigns the logical name. 'DEA @' deassigns the PPN assigned with 'AS [p,pn]'. 'DEA' with no argument removes all logical assignments and all physical device assignments.

REA[SSIGN] [arg]

Similar to the BASIC-PLUS REASSIGN command.

HELLO

Similar to the BASIC-PLUS HELLO command.

BYE

Similar to the BASIC-PLUS BYE command.

EXIT

Exits from the emulator and returns to the user's default run-time system.

RT11.RTS Specific Commands

RN program

Runs a program. If no PPN is specified, the program is assumed to be in the current user's account. The program is invoked using the RSTS/E .RUN function (refer to the RSTS/E System Directives Manual).

PPN ppn

Set the search PPN. Whenever the emulator attempts to open a file for input for which no explicit PPN has been given, it first attempts to find the file in the current user's account. If this fails, it attempts to find the file in the account specified in the most recent 'PPN' command. If this fails, the current library account (as set by the 'LIB' command) is tried. Only if all these fail does the emulator report an error. The search PPN is initially disabled. It is set with the 'PPN' command specifying an explicit PPN. It can be disabled by typing 'PPN' with no argument.

LIB[RARY] ppn

In a manner similar to the 'PPN' command, the LIB command sets up the user's library account. This account is used as a last resort in file lookups, and as the default account for the 'R' command. The library PPN is initially [1,2]. It may be changed or disabled.

IN[ITIALIZE]

Initialize user area, reset channels, clear low core.

MO[NITOR]

Exits from the emulator and returns to the user's default run-time system.

ER[ROR]

Prints the most recent RSTS/E error message. For example, any failure on file lookup will give a message similar to '?Fil not fnd?'. Typing CTRL/C to get to the KMON and then typing ERR will print the full RSTS/E error message for the error encountered.

RT11 Directives supported under RSTS/E

Since all RSTS/E I/O is synchronous, all RT-11 I/O operations will be implicitly synchronous. However, completion routines are supported, as are all other modes of I/O. Since there is no access to the I/O page, nor is there any ability to change the processor priority, user devices and user device drivers are not supported. The following monitor calls may not be used: .DEVICE, .PROTECT, .SYNCH, .INTEN .

All RSTS/E device handlers are permanently resident; thus, the .FETCH EMT is reduced to simply a check on the legality of the device name. .QSET is ignored.

Some non-file structured devices allow non-standard information to be passed via the block number. For example, writing to block 1 of a keyboard causes binary mode output. The emulator normally clears the block number in an I/O request to such a device. If a program actually requires the special function, it may set a special block number in the first word of the read/write area (i.e., referenced as @54) immediately before a read or a write operation. This block number will be used for the actual call to the RSTS/E executive.

Since a single CTRL/C to RSTS/E causes an interrupt, the character CTRL/Z may be used to cause a program to exit. Whenever a program does a .TTYIN for a CTRL/Z (which may be typed ahead), it will be forced to exit. When CTRL/Z is typed ahead as input to the teletype handler, it assumes its normal function of end-of-file, and does not cause program termination. (Immediate termination, of course, can be effected by typing a single CTRL/C.)

Monitor calls

Except as noted previously, the following RT-11 monitor calls work exactly as described in the RT-11 Advanced Programmer's Guide:

.DATE	Return date in RØ
.CHAIN	Chain to another program
.CLOSE	Close a channel
.CSIGEN	Command string interpreter, general mode
.CSISPC	Command string interpreter, special mode
.DELETE	Delete a file
.DSTATUS	Return device status
.ENTER	Open a file for output
.EXIT	Return to the monitor
.FETCH	Get device
.GTIM	Get time of day
.GTJB	Get job parameters
.GVAL	Get offset value from monitor
.HRESET	Hard reset
.LOCK	(Ignored)
.UNLOCK	(Ignored)
.LOOKUP	Open a file for input
.PRINT	Print a string on the terminal
.PURGE	Reset a channel
.RCTRLO	Reset CTRL-O mode on the terminal
.READ	Read to memory
.READC	Read to memory
.READW	Read to memory
.RELEASE	Release device
.RENAME	Rename a file
.REOPEN	Reopen a file
.SAVESTATUS	Preserve file information and close channel
.SETTOP	Request high core limit
.SFPA	Set floating point exception trap address
.SRESET	Reset channels
.TRPSET	Set intercept for traps to 4 and 1Ø
.TTYIN	Get character from terminal
.TTINR	Get character from terminal
.TTYOUT	Print character on terminal
.TTOUTR	Print character on terminal
.TWAIT	Sleep
.WAIT	Wait for channel ready
.WRITE	Output
.WRITC	Output
.WRITW	Output

SPECIAL RT-11 EMTS

The following EMT codes are implemented in the RSTS/E RT-11 emulator to provide easy access to common RSTS/E functions.

.SETFQB EMT 36Ø

After executing a .CSISPC or .CSIGEN call to RT-11, the .SETFQB EMT can be used to return RSTS/E specific information about the filenames that were scanned.

To use the call, load RØ with a pointer to the file description location in which RT-11 returned the CSI information and execute EMT 36Ø. The FIRQB and flags returned in the XRB will be restored to the state they were in after the particular file specification was scanned. A detailed description of this format is in the RSTS/E System Directives Manual under the .FSS call.

EXAMPLE:

```
        .MCALL  .CSISPC

.SETFQB =       EMT+36Ø                 ; SET UP FIRQB CALL TO RT-11

        .CSISPC #OUTSPC,#DEFEXT,#CSTRING; SCAN THE COMMAND STRING
        BCS     ERROR                   ; AN ERROR OCCURRED, ABORT PROCESSING
        MOV     #OUTSPC,RØ              ; SET UP TO GET INFO FOR 1ST FILE SPEC
        .SETFQB                         ; SET UP THE FIRQB AND XRB FLAGS
        BIT     #1,XRB+1Ø               ; WAS THERE A FILE NAME?
        BEQ     NOFILE                  ; NO, GO PROCESS FILE NAME ERROR
        TST     FIRQB+FQPPN             ; WAS THERE A PPN?
        BEQ     NOPPN                   ; NO, GO PROCESS NO PPN ERROR
```

The above example scans the command string at CSTRING (which must be an ASCIZ string), and checks for errors in the first file name by using the information returned by EMT 36Ø.

.DATTIM EMT 361

This call is used to return a DATE$ or TIME$ string in the current monitor format (alphabetic or numeric, AM/PM or 24-hour) without altering the FIRQB. To execute the function, load RØ with the address of the area in which the string is to be placed (9 bytes for date, 8 bytes for time); to convert a date, put the number ((year-197Ø)*1ØØ + day) in location XRB; to convert a time, clear location XRB and place the time specification (minutes until midnight, 1 to 144Ø.) in location XRB+2. After the EMT, RØ will point to the byte following the date or time string returned.

EXAMPLE:

```
        .MCALL  .EXIT,.PRINT

.PRIV   =       EMT+377                 ; PREFIX EMT TO RSTS/E
.DATTIM =       EMT+361                 ; DATE/TIME EMT TO RT-11
XRB     =       442                     ; LOCATION OF XRB
.DATE   =       EMT+34                  ; RSTS/E .DATE EMT

.CSECT

START:  .PRIV,  .DATE                   ; GET DATE INFORMATION FROM RSTS/E
        MOV     XRB+2,-(SP)             ; SAVE THE TIME SPECIFICATION
        MOV     #DATE$,RØ               ; SET THE ADDRESS FOR THE DATE STRING
        .DATTIM                         ; GET THE DATE STRING
        CLRB    (RØ)+                   ; CLEAR THE LAST BYTE FOR ASCIZ STRING
        .PRINT  #DATMSG                 ; PRINT THE DATE MESSAGE
        CLR     XRB                     ; CLEAR THE DATE LOCATION
        MOV     (SP)+,XRB+2             ;   AND SET THE TIME SPECIFICATION
        MOV     #TIME$,RØ               ; SET THE ADDRESS FOR THE TIME STRING
        .DATTIM                         ; GET THE TIME STRING
        CLRB    (RØ)+                   ; MAKE THE STRING ASCIZ
        .PRINT  #TIMMSG                 ; PRINT THE TIME MESSAGE
        .EXIT                           ; END THE PROGRAM

DATMSG: .ASCII  /THE DATE IS NOW /      ; THIS IS THE DATE MESSAGE
DATE$:  .BLKW   12                      ; THIS IS WHERE TO PUT THE DATE STRING
TIMMSG: .ASCII  /THE TIME IS NOW /      ; THIS IS THE TIME MESSAGE
TIME$:  .BLKW   12                      ; THIS IS WHERE TO PUT THE TIME STRING

.END    START
```

The above program, when assembled, will print the following on the user's terminal

```
THE DATE IS NOW Ø1-APR-8Ø
THE TIME IS NOW 12:55 PM
```

if it is run at 12:55 PM on Ø1-Apr-8Ø, and the system default date string is alphabetic and the time string default is AM/PM.

.SETCC EMT 362

This call is used to set a CTRL/C interrupt routine address. To use it, put the address of the CTRL/C processing routine into RØ and execute EMT 362. To disable CTRL/C trapping, pass a value of Ø in RØ.

When you type a CTRL/C, the program is interrupted and the CTRL/C routine is entered. Before entering the CTRL/C routine, the CTRL/C trapping is disabled. Therefore, if you type two CTRL/C's, you can abort the running program. You must re-execute this EMT to re-enable trapping, if desired. Note that the usual rules for CTRL/C apply, and the program must reset CTRL/O when the trap routine is entered.

EXAMPLE:

```
        .MCALL  .REGDEF,.EXIT,.RCTRLO,.PRINT
        .REGDEF

.SETCC  =       EMT+362                 ; SET CTRL/C TRAP TO RT-11

.CSECT

START:  MOV     #CTRLC,RØ               ; SET ADDRESS OF THE CTRL/C ROUTINE
        .SETCC                          ; SET THE INTERRUPT VECTOR
WAITER: BR      WAITER                  ; LOOP TO WAIT FOR A CTRL/C

CTRLC:  .RCTRLO                         ; RESET THE CTRL/O ISSUED BY CTRL/C
        .PRINT  #MESAGE                 ; PRINT THE MESSAGE
        .EXIT                           ; EXIT THE PROGRAM

MESAGE: .ASCIZ  /YOU TYPED A CTRL/C!/

.END    START
```

The above program will loop until a CTRL/C is typed, and then will print "YOU TYPED A CTRL/C!" and exit. If you wish to return control to the location where the program was interrupted, the CTRL/C processing routine should use the RTI instruction.

.DORUN EMT 363

This EMT is used to chain to another program which may run under the control of a different run-time system. This function is required, because RT-11 does not keep the user's logical assignments in the same locations as other run-time systems do.

To execute this call, load RØ with the address of a file specification block in RT-11 format (four words of radix-5Ø containing device, filename, and extension). The fifth word of the block should contain the line number (or other parameter) which will be passed to the program. Errors are returned as if from .RUN if the run can not be accomplished.

EXAMPLE:

```
.SETFQB =       EMT+36Ø                 ; SET UP FIRQB CALL TO RT-11
.DORUN  =       EMT+363                 ; TRY TO RUN A FILE CALL TO RT-11

        .CSISPC #OUTSPC,#DEFEXT,#Ø      ; READ AND SCAN THE COMMAND STRING
        BCS     ERROR                   ; THE COMMAND STRING WAS IN ERROR
        MOV     #OUTSPC,RØ              ; GET THE FILE BLOCK FOR THE 1ST FILE
        CLR     4*2(RØ)                 ; ENTER THE PROGRAM AT LINE Ø
        .DORUN                          ; TRY TO RUN THE FILE
ERROR:  .PRINT  ERRMES                  ; PRINT AN ERROR MESSAGE
        .EXIT

ERRMES: .ASCIZ  /?CAN'T RUN THAT FILE/
```

The above program segment uses the CSI to scan a file specification, ensures that the program is entered with no parameter, and then attempts the run command with .DORUN. If an error occurs in either the .CSISPC or the .DORUN, a message is printed.

```
.PRTERR         EMT     364
```

This call is used to print the RSTS/E error text corresponding to an error code on the user's terminal. The error code is passed in RØ. The error text is printed without a carriage-return/line-feed combination.

EXAMPLE:

```
        .MCALL  .EXIT,.PRINT,.REGDEF
        .REGDEF

.PRTERR =       EMT+364                 ; PRINT RSTS/E ERROR MSG CALL

.CSECT

START:  MOV     #256.,R1                ; THE NUMBER OF ERRORS TO PRINT
        CLR     RØ                      ; START WITH ERROR ZERO
1Ø$:    .PRTERR                         ; PRINT ERROR TEXT
        .PRINT  #CRLF                   ; FOLLOWED BY A CR/LF
        INC     RØ                      ; INCREMENT RØ FOR NEXT ERROR MESSAGE
        SOB     R1,1Ø$                  ; LOOP UNTIL ALL ARE PRINTED
        .EXIT                           ; EXIT THE PROGRAM

CRLF:   .BYTE   Ø

.END    START
```

The above program will print all the errors in the RSTS/E error text file.

.DOFSS EMT 365

This call is used to perform a RSTS/E file string scan. The address of the file string to scan is passed in RØ. The string to scan must be in ASCIZ format. The results are returned in the FIRQB and the XRB exactly as they are for the RSTS/E .FSS function. The reason for this function is the same as the reason for the .DORUN function - since RT-11 does not use the standard location for the user's logical assignments, the call must be made through the emulator to work properly.

EXAMPLE:

```
        .MCALL  .REGDEF,.PRINT,.EXIT
        .REGDEF

.DOFSS  =       EMT+365                 ; DO FILE STRING SCAN CALL TO RT-11
.PERR   =       EMT+364                 ; PRINT RSTS/E ERROR MSG CALL

.CSECT

START:  MOV     #FILE$,RØ               ; SET THE ADDRESS OF THE FILE STRING
        .DOFSS                          ; SCAN THE NAME
        TSTB    @#4Ø2                   ; TEST THE FIRQB LOCATION FOR AN ERROR
        BNE     PRNTER                  ; AN ERROR OCCURRED, PRINT THE MESSAGE
        .DORUN                          ; TRY TO DO A FILE RUN
                                        ; IF WE RETURN THERE WAS AN ERROR
PRNTER: MOVB    @#4Ø2,RØ                ; GET THE ERROR CODE FROM THE FIRQB
        .PERR                           ; PRINT THE ERROR CODE
        .PRINT  #FILERR                 ; FOLLOWED BY THE FILE NAME
        .EXIT                           ; AND EXIT THE PROGRAM

FILERR: .ASCII  " - "                   ; SEPARATOR FOR THE ERROR MESSAGE
FILE$:  .ASCIZ  "SY:[1,2]DIRECT.BAC"    ; THE FILE NAME TO RUN

.END    START
```

The above program will attempt to run the DIRECT program in account [1,2], if an error occurs during the file string scan, or the run command the error will be printed followed by the file name. For instance, if the file could not be found the following would be printed:

```
?Can't find file or account - SY:[1,2]DIRECT.BAC
```

.GETCOR EMT 366

This call is used to dynamically expand the core allocation of the user program. The new size desired (in K words) is passed in RØ.

This function should be used instead of the RSTS/E .CORE EMT because RT-11 keeps certain information about the user's job in high core. This call dynamically moves the information for the user and expands his job image to the desired size.

The information that RT-11 keeps at the top of the user's job image is pointed to by location 54. All programs written for the RT-11 emulator should take care that they not write above the location pointed to by location 54.

If the request for more memory cannot be honored due to the system or run-time system swap max parameter, the carry bit is set on return and FIRQB+Ø contains the proper error number.

EXAMPLE:

```
;+
; ALOCAT - ALLOCATE MORE STORAGE FOR THE PROGRAM
;
;       RØ = K SIZE TO ALLOCATE
;
;       JSR     PC,ALOCAT
;-

        .MCALL  .EXIT,.PRINT,.REGDEF
        .REGDEF

.GETCOR =       EMT+366                 ;ALLOCATE MEMORY CALL TO RT-11

.CSECT
.GLOBL  ALOCAT

ALOCAT: .GETCOR                         ;TRY TO ALLOCATE MORE CORE
        BCS     NOMEM                   ;COULDN'T GET IT
        RTS     PC                      ;RETURN TO THE CALLING ROUTINE

NOMEM:  .PRINT  #MEMERR                 ;PRINT AN ERROR MESSAGE
        .EXIT                           ; AND EXIT THE USER'S PROGRAM

MEMERR: .ASCIZ  /?CANNOT ALLOCATE MORE MEMORY - FATAL ERROR/

.END
```

The above subroutine can be called by a program to allocate more memory, if there is an error the program will stop execution and print:

?CANNOT ALLOCATE MORE MEMORY - FATAL ERROR

.DOCCL EMT 367

This call is used to perform a RSTS/E .CCL. Since the .CCL will cause a new program to run, the logical name table must be restored to the standard location. RT-11 will exchange the logical name table, attempt the .CCL, and, if the .CCL fails, will reset the logical name table. For this call, RØ points to an ASCIZ string which is the possible CCL command.

EXAMPLE:

```
.MCALL  .PRINT

.DOCCL  =       EMT!367
.PRTERR =       EMT!364
FIRQB   =       4Ø2

;+
; TRYCCL - TRY A CCL COMMAND
;
;       RØ -> AN ASCIZ STRING, HOPEFULLY A CCL COMMAND
;
;       CALL    TRYCCL
;
;       RØ = ERROR CODE WHICH OCCURRED, CONTROL DOES NOT RETURN
;               THE STRING IS A VALID CCL COMMAND
;-

TRYCCL: .DOCCL                          ;TRY TO EXECUTE A CCL, CONTROL WON'T RETURN
        MOVB    FIRQB,RØ                ;GET THE ERROR CODE
        BEQ     1Ø$                     ;NO ERROR OCCURRED, PRINT DEFAULT MESSAGE
        .PRTERR                         ;PRINT AN ERROR MESSAGE
        .PRINT  #CRLF                   ;PRINT A CARRET
        RTS     PC                      ; RETURN

1Ø$:    .PRINT  #CCLERR                 ;PRINT THE DEFAULT ERROR TEXT
        RTS     PC

CCLERR: .ASCII  /?NOT A CCL COMMAND/
CRLF:   .BYTE   Ø
```

The above routine will execute the CCL command pointed to by RØ or print an error message and return if the string is not a CCL command.

RSTS/E V7.Ø
RT-11 Emulator and Utilities Package
Package Notes

USE OF UNDERSCORE IN RT11 EMULATOR UTILITIES

RSTS/E V7.Ø allows the use of the underscore ("_") character before a device name in a file string to indicate that the device name is not to undergo logical translation. Utilities provided with the RT11 emulator will allow the use of the underscore character; however, due to the nature of the emulator, it is still possible for the device name to undergo translation before a file is opened.

In the RT11 emulator, file specifications given to a utility are scanned using the RT11 CSI (command string interpreter). At this time, the device designation may or may not undergo logical translation (depending on whether or not the underscore was used). The CSI returns RT11 file description blocks to the utilities.

When the utility program then requests the emulator to open a file the file open code does an additional logical translation on the specified device name. This is an unavoidable consequence of allowing logical device names to be specified in RT11 file description blocks.

RSTS/E V7.Ø
RT-11 Emulator and Utilities Package
Package Notes

USING RTSODT FOR DEBUGGING RUN-TIME SYSTEMS

A version of ODT is provided on the RSTS/E V7.Ø distribution kit as an unsupported feature which can be linked with user-written run-time systems.

To use RTSODT with run-time systems that are assembled and linked under the RT11 Run-Time System, execute the following commands:

```
RUN $LBR.TSK<cr>
LBR>RTSODT.OBJ=LB:SYSLIB/EX:RTSODT<cr>
LBR>^Z

Ready

RUN $PIP.SAV<cr>
*RTSODT.OBJ=RTSODT.OBJ/RMS:FB<cr>
*^Z

Ready
```

You may now include RTSODT.OBJ in the list of input files during the LINK. For example:

```
RUN $LINK<cr>
*RT11OD/Z,RT11OD/W,RT11OD=RT11/X/H:#177776/U:#4ØØØ/C<cr>
*#ERR.STB,#RTSODT<cr>
*PATCH<cr>
*^Z

Ready
```

Refer to article Seq 21.1.1 N for a description of the use of RTSODT with run-time systems that are assembled and linked under the RSX Run-Time System.

RSTS/E V7.Ø
RT-11 Emulator and Utilities Package
HOOK.SAV

NOTES ON HOOK.SAV

HOOK.SAV is a program which will write a bootstrap program on RSTS/E file-structured disks and magnetic tapes. Though it is intended only for use during SYSGEN, it can be used to make any RSTS/E file-structured disk or tape bootable.

Creating a bootable RSTS/E File-structured Disk

Mount the disk to be "HOOKed":

```
RUN $UTILTY<cr>
<UTILTY's header line>
#MOUNT xxn:packid/PRIVATE<cr>
#^Z

Ready
```

Copy INIT.SYS from your system disk to the output device:

```
RUN $PIP.SAV<cr>
*xxn:[Ø,1]*.*=SYØ:[Ø,1]INIT.SYS<cr>
*^Z

Ready
```

Then, run HOOK.SAV:

```
RUN $HOOK.SAV<cr>
*xxn:[Ø,1]INIT.SYS<cr>
Hook complete
*^Z

Ready
```

Creating a bootable RSTS/E DOS-format Magnetic Tape

Mount a scratch magnetic tape on a tape drive. (Note that HOOK.SAV will zero the tape before using it.) Then, run HOOK.SAV:

```
RUN $HOOK.SAV<cr>
*xxn:[Ø,1]INIT.SYS[/D[:n.]],SYØ:[Ø,1]INIT.SYS<cr>
Hook complete
*^Z

Ready
```

NOTES

1. Magnetic tape density: Specify "/D:n" after the "output" dev:filnam, as follows:

/D:8ØØ.	Set 8ØØ BPI, odd parity
/D	Same as /D:8ØØ.
/D:16ØØ.	Set 16ØØ BPI, phase encoded
/D:16ØØ	Same as /D:16ØØ.

   Note:

   - o /D:8ØØ (no dot) is an illegal specification
   - o If /D is not specified, the drive's current settings are used.
   - o HOOK will assign the tape drive to you, in DOS format, at the density that you specify (or do not specify).

2. Normally, HOOK finds the bootstraps for disks and magnetic tapes in SYØ:[Ø,1]INIT.SYS. If, as may happen on a development machine, [Ø,1]INIT.SYS does not have the most recent version of the appropriate bootstrap, you can specify an alternate file in which to find the bootstraps. Hence, a full HOOK command line looks like

   <outdev:filename> , <file for mt-only> , <bootstraps>

   The bootstrap file must meet the following criteria:

   - o It must reside on a mounted disk
   - o It must be a SAV-format SIL with a symbol table
   - o It must have the global symbols "BOOTS" and "MBOOTS" defined for the start of the disk and magnetic tape boots, respectively.
   - o The bootstraps must be in the standard RSTS/E format, 4ØØ(8) bytes for each disk bootstrap, and 1ØØØ(8) bytes for each magnetic tape bootstrap.
   - o In short, the bootstrap file should look like INIT.SYS.

3. HOOK writes magnetic tapes with DOS labels. If the tape is written at 8ØØ BPI, the name of the first file on the tape will be [Ø,1]MTBOOT.SYS. If the tape is written at 16ØØ BPI, the name of the first file will be [Ø,1]MSBOOT.SYS. Note that the difference is important; the 8ØØ BPI bootstrap is significantly different from the 16ØØ BPI bootstrap.

4. If no filename is specified with the device specification for magnetic tape, the default name INIT.SYS is used, with the current PPN. In other words, if you are logged in under [123,1] and specify

   MTØ:/D:8ØØ.,[Ø,1]FOO.FOO

the files on the tape will be "[Ø,1]MTBOOT.SYS" and "[123,1]INIT.SYS".

If you have any reason to wonder if a magnetic tape is bootable, check the directory, which should have 2 files with legal RSTS/E filenames. The first file should be 1 block long. If the directory looks unusual, it is not a bootable RSTS/E tape.

5. The following might be useful information.

Standard command lines:

DKØ:[Ø,1]INIT.SYS
Load and execute [Ø,1]INIT.SYS when DKØ: is booted.

MM1:[Ø,1]INIT.SYS/D:16ØØ,DRØ:[Ø,1]INIT.SYS
Load and execute MM1:[Ø,1]INIT.SYS when MM1: is booted; copy the file to be executed from DR1:[Ø,1]INIT.SYS; set the density to 16ØØ BPI.

Non-standard command lines:

DB3:[Ø,1]INIT.SYS,,[4,4]NEWINI.SYS
Load and execute [Ø,1]INIT.SYS when DB3: is booted; get the bootstrap from [4,4]NEWINI.SYS.

MMØ:,[Ø,1]INIT.SYS <not recommended>
Load and execute a file called INIT.SYS that was copied from [Ø,1]INIT.SYS.

6. Possible Errors:

?Can't open disk NFS
Someone else is using the disk to be HOOKed, or you are not privileged.

?Can't open [Ø,1]SYØ:INIT.SYS
You must be privileged to run HOOK.SAV

?Directory error
The output disk is not a valid RSTS/E file-structured disk.

?File high limit too large
The file to be loaded by the bootstrap program is too large.

?Error reading boot block
An I/O error occurred while reading the disk to be HOOKed; it may be off-line.

?Error writing boot block
An I/O error occurred while writing the bootstrap program; the disk may be write-locked.

?Error reading pack ID
An I/O error occurred while reading the disk to be HOOKed.

?Error reading INIT.SYS
An I/O error occurred while reading INIT.SYS.

?Explicit unit number required
The device name specified must include an explicit unit number.

?Ill cmd?
HOOK.SAV could not interpret the specified command line.

?Illegal density
The density that you specified cannot be used on this tape drive.

?Illegal overlay number in INIT
You are using an obsolete version of INIT.SYS.

?INIT.SYS is not a SAV format SIL
You are using an obsolete version of INIT.SYS.

?Not a bootable device
HOOK.SAV cannot write a bootstrap for that device.

?Null file can't be hooked
Self explanatory.

?No BOOT in STB
You are using an obsolete version of INIT.SYS.

Second input file required for magtape hook
You must specify a file to be copied to the magnetic tape.

?UFD open failure
An I/O error occurred.

RSTS/E V7.Ø
RT-11 Emulator and Utilities Package
PIP.SAV

CREATING NEW FILES WITH PIP - PIP.SAV FEATURE PATCH

PROBLEM:

When transferring a disk file to a new disk file, PIP transfers the creation date and time, and last access date to the newly created file. This may be undesirable in some environments where files are backed up by creation or access dates.

SOLUTION:

This problem may be solved on a per-command basis by using the /NEWFILE option. However, if it is deemed that this operation is desired as the default, the following PIP patch may be applied.

If this patch is applied /NEWFILE will be the default operation. To transfer a file and retain all accounting information the /RETAIN switch may be used on a per-command basis to override the new default.

PROCEDURE:

1. This is a feature patch to PIP.SAV.

2. The patch described in Step 4 below can NOT be installed using the PATCH option of INIT.SYS.

3. This patch can be installed manually using ONLPAT, the on-line patching program:

```
RUN $ONLPAT
Command File Name? <cr>                 (RETURN for manual patch installation)
File to patch? [1,2]PIP.SAV
File found in account [1,2]
```

   The patch is also contained in a patch file appearing in patch kit version "A" or later.

4. The patch is as follows:

```
Base address? ..NEWF
Offset address? Ø
 Base   Offset  Old     New?
??????  ØØØØØØ  ØØ1Ø37  ? (Q&377)+BNE
??????  ØØØØØ2  ØØ5767  ? ^C            (up-arrow/C to exit)
Patch complete
1 patch installed

Command File Name? ^Z                   (CTRL/Z to exit)
```

RSTS/E V7.Ø
RT-11 Emulator and Utilities Package
PIP.SAV

TRANSFERRING FILES BY DATE OF LAST ACCESS - PIP.SAV FEATURE PATCH

PROBLEM:

PIP allows file operations selectively by creation date through the use of a number of switches (i.e., /TODAY, /SINCE, /UNTIL, etc.). In some instances it would be desirable to have such switches key off of the last access date.

SOLUTION:

Two new switches have been added to PIP to allow specifying either creation date (/CREATION) or date of last access (/DLA). The default operation is to key off of the creation date. The following patch will set the default operation to key all date selections off of the date of last access.

PROCEDURE:

1. This is a feature patch to PIP.SAV.

2. The patch described in Step 4 below CANNOT be installed using the INIT.SYS PATCH option.

3. This patch can be installed manually using ONLPAT, the on-line patching program:

```
RUN $ONLPAT
Command File Name? <cr>                 (RETURN for manual patch installation)
File to patch? [1,2]PIP.SAV
File found in account [1,2]
```

   The patch is also contained in a patch file appearing in patch kit version "A" or later.

4. The patch is as follows:

```
Base address? ..DLAC
Offset address? Ø
 Base   Offset  Old     New?
??????  ØØØØØØ  ØØ1ØØ2  ? (Q&377)+BR
??????  ØØØØØ2  Ø167ØØ  ? ^C            (up-arrow/C to exit)
Patch complete
1 patch installed

Command File Name? ^Z                   (CTRL/Z to exit)
```

RSTS/E V7.Ø
RT-11 Emulator and Utilities Package
PIP.SAV

REQUIRING VOLUME I.D. SPECIFICATIONS WITH ANSI MAGNETIC TAPES
- PIP.SAV FEATURE PATCH

PROBLEM:

When using PIP.SAV to write multi-volume ANSI format magnetic tapes, a volume ID may be specified when mounting a tape. The default action for PIP is to check the volume I.D. against that of the tape being mounted, and if the two do not match to reject the mounted tape. In some environments it might be desirable to force the specification of the volume I.D. to insure maximum tape security.

SOLUTION:

The following patch will reject the mounting of an ANSI format magnetic tape if no volume I.D. is specified.

PROCEDURE:

1. This is a feature patch to PIP.SAV.

2. The patch described in Step 4 below can NOT be installed using the PATCH option of INIT.SYS.

3. This patch can be installed manually using ONLPAT, the on-line patching program:

```
RUN $ONLPAT
Command File Name? <cr>                 (RETURN for manual patch installation)
File to patch? [1,2]PIP.SAV
File found in account [1,2]
```

   The patch is also contained in a patch file appearing in patch kit version "A" or later.

4. The patch is as follows:

```
Base address? ..VIDS
Offset address? Ø
 Base   Offset  Old     New?
??????  ØØØØØØ  ØØØ4Ø3  ? NOP
??????  ØØØØØ2  Ø327Ø4  ? ^C             (up-arrow/C to exit)
Patch complete
1 patch installed

Command File Name? ^Z                   (CTRL/Z to exit)
```

RSTS/E V7.Ø
RT-11 Emulator and Utilities Package
PIP.SAV

TRANSFERRING FILES BETWEEN PUBLIC DISKS IN THE SAME ACCOUNT

A number of users have asked why it is that PIP.SAV skips some files when transferring files from one public disk to another in the same account. For example, on a system with two RPØ4 disk drives in the public structure, a user might attempt to re-organize the files in his account by transferring all the files residing on DB1: to DBØ:. (This seems to be done quite frequently; we doubt that the procedure has any discernible benefit.) The discussion which follows will refer to the following command line, where both DBØ: and DB1: are in the public disk structure:

```
PIP DBØ:*.*=DB1:*.*
```

After the first file has been opened on DB1: and the output file has been created on DBØ: (PIP.SAV opens the first input file before it opens the output file), RSTS/E determines that a file with the same name (the input file) already exists in the public disk structure on DB1:. Since the file on DB1: is currently open, a flag is set to indicate that the file on DB1: should be deleted as soon as it is closed. When that file has been transferred, PIP.SAV closes the input and output files, causing the input file to be deleted.

After processing the first file, PIP.SAV looks for the second file on DB1: that matches the wildcard specification. However, since the first file has been deleted, the file returned in response to a request for the second file is the file that was originally the third file - hence, in this example, every second file is skipped.

Note that transferring files from one public disk to another within the same account can be an extremely hazardous procedure since an exceptional condition such as an I/O error or CONTROL/C may cause both the old and the new copies of the file to be deleted. For this reason, we strongly discourage this operation. If you find that it is necessary to transfer files between public disks in the same account, there are two safe ways to transfer the files:

1. Mount either the source or the destination disk as a private disk. This will prevent RSTS/E from deleting the source file when the destination file is opened. After you have confirmed that the operation has succeeded, delete the source files, dismount the private disk and remount it as a public disk.

2. Assign a temporary name to each file to transfer, and then delete the old files after the transfer operation has succeeded. For instance, to move all *.ODL files from DB1: to DBØ:, you might use the following commands:

```
PIP DB1:*.XDL=DB1:*.ODL/RE      [Rename all the .ODL files on DB1:]
PIP DBØ:*.ODL=DB1:*.XDL         [Transfer the files to DBØ:]
PIP DB1:*.XDL/DE                [Delete the files from DB1:]
```

RSTS/E V7.Ø
RT-11 Emulator and Utilities Package
PIP.SAV

CREATING MAGNETIC TAPE FILES WITH BLOCKSIZES LARGER THAN 512 BYTES

When creating a file on magnetic tape it is often desirable to use a blocksize larger than 512 bytes. To accomodate this, a new switch has been included in PIP.SAV for RSTS/E V7.Ø.

The new switch, /BSIZE:n, allows the specification of a blocksize on the output magnetic tape file. The blocksize specification must be an even integer between 18 and 4Ø96.

It is important here to note a couple of restrictions imposed by standards used by many systems. If you are writing an ANSI format magnetic tape which is intended for interchange with an operating system other than RSTS/E, the blocksize must be between 18 and 2Ø48. If you intend to interchange data with an RT11 system the blocksize must be 512.

It is also important to note that the /BSIZE:n switch is only applicable to the output device. Magnetic tapes written with blocksizes other than 512 bytes are handled automatically by PIP, thus there is no need to specify a /BSIZE:n switch for an input device.

The /BSIZE:n option is only meaningful for output magnetic tape devices. If it is used and the output device is not magnetic tape, the switch will be ignored. Since the /BSIZE:n may increase the amount of buffersize needed by PIP it is important to invoke PIP with a larger buffersize when reading and writing blocksizes larger than 512 bytes (this may be accomplished with either the /SIZE:n switch to the CCL command, or the SIZE n command in the RT11 keyboard monitor).

RSTS/E V7.Ø
RT-11 Emulator and Utilities Package
PIP.SAV

USING PIP TO STORE/TRANSFER FILES TO ANSI LABELLED MAGNETIC TAPES

PIP.SAV in V7.Ø is able to store or transfer files and retain the file attribute information. This is possible because PIP.SAV can now write magnetic tapes that conform to the American National Standards Institute (ANSI) magnetic tape label and file structure standard X3.27-1978. This format of tape has more information stored about each file than DOS format (the other tape format supported by RSTS/E) and it is used by PIP.SAV to store attributes and other information about the file. Thus, with RSTS/E V7.Ø PIP.SAV, you are able to save and restore RMS sequential, relative, and indexed files with no loss of data format information.

Saving files for RSTS to RSTS transfer

PIP.SAV allows you to use ANSI labelling format for copying files to magnetic tape with no loss of attributes or data format. Here are the steps that are needed to put an attributed file on to a brand new magnetic tape.

1. a) Mount the magnetic tape on a tape drive.

   b) Set the labelling format for the drive to ANSI. For example,

      ```
      ASSIGN MTØ:.ANSI
      ```

      If the system manager has set the system magnetic tape labelling default to ANSI this step is not neccessary.

   c) Use PIP.SAV to zero the volume and write a volume ID. You may also set the density and parity at this time. For example,

      ```
      PIP MTØ:LAUREN/ZERO/DEN:8ØØ/PARITY:ODD
      ```

2. a) Copy the file or files to the tape using the /BL option. This procedure is applicable for files with and without attributes. For example,

      ```
      PIP MTØ:FORMS.DAT=FORMS.DAT/BL
      ```

      The tape file will be created in U (undefined) format and all attribute information will be retained in the file header labels on the tape. Note that THIS METHOD CAN ONLY BE USED FOR RSTS TO RSTS TRANSFERS, since the file format (U) is recognized only by PIP.SAV; it is NOT a format defined by ANSI standard X3.27-1978.

Saving files for interchange with other operating systems

If the files written to tape are intended to be transfered to another operating system step 2 of the above procedure can not be used. The procedure above writes tape format U, which is not a format defined in the ANSI standard. For interchange purposes you may only use record formats F (fixed length records) and D (variable length records). The following procedures can be used for the specific cases noted:

1. a) To transfer a non-attributed stream ASCII file to ANSI D format use the /RMS:FA option. For example:

```
PIP MMØ:REPORT.DAT/RMS:FA=REPORT.DAT
```

   b) To transfer FCS or RMS attributed files no switch is needed. However, only sequential access files may be copied. Relative access, and indexed files must be converted to sequential files using RMSCNV before they can be transferred to an ANSI format tape. Files with attributes will be mapped into their appropriate ANSI record formats.

   It is important to note at this time that ANSI standard X3.27-1978 defines that only 7-bit and 8-bit ASCII codes can be used for information interchange with this standard. Therefore, binary data may not be transferable to other operating systems.

Copying files from tape to disk

1. a) Mount the tape and ASSIGN the drive to ANSI labelling format, or use the UMOUNT program to assign and mount the tape. For example,

```
MOUNT MTØ:LAUREN/ANSI/DEN:8ØØ/PARITY:ODD
```

   b) Copy the tape to disk with PIP. PIP will read the attributes of the file from the tape and restore the file to its former format. For example,

```
PIP FORMS.DAT=MTØ:FORMS.DAT
```

   If the file was transfered to the magnetic tape using the /RMS:FA switch (to convert from stream ASCII to ANSI D format) the output disk file will be an FCS variable length, implied carraige control file.

   If the file had a file cluster size other that the pack cluster size you will want to restore the disk file with that file cluster size. To do this use the /CL switch. For example,

```
PIP FORMS.DAT/CL:16=MTØ:FORMS.DAT
```

   c) If you had used RMSCNV to convert the file from relative or indexed format to sequential format, then you must convert the file back to its previous format.

RSTS/E V7.Ø
RT-11 Emulator and Utilities Package
PIP.SAV

SETTING /LOG AS A DEFAULT PIP.SAV OPTION - PIP.SAV FEATURE PATCH

PROBLEM:

In some environments it is desirable to have the /LOG option with PIP.SAV set for all transactions.

NOTE

Certain control files which use PIP.SAV, such as the DX/RSTS V3.Ø build procedures, will not run with this optional feature patch installed.

SOLUTION:

The following PIP.SAV feature patch will set the /LOG option for every PIP.SAV command which allows the use of /LOG. Note that the /NOLOG switch may be used to suppress the logging for a particular operation.

PROCEDURE:

1. This is a feature patch to PIP.SAV.

2. The patch described in Step 4 below can NOT be installed using the PATCH option of INIT.SYS.

3. This patch can be installed manually using ONLPAT, the on-line patching program:

```
RUN $ONLPAT
Command File Name? <cr>                    (RETURN for manual patch installation)
File to patch? [1,2]PIP.SAV
File found in account [1,2]
```

   The patch is also contained in a patch file appearing in patch kit version "A" or later.

4. The patch is as follows:

```
Base address? ..SCNF
Offset address? Ø
 Base   Offset  Old     New?
??????  ØØØØØØ  ØØØØØØ  ? 1ØØØØØ
??????  ØØØØØ2  ØØ4437  ? ^C              (up-arrow/C to exit)
Patch complete
1 patch installed

Command File Name? ^Z                      (CTRL/Z to exit)
```

RSTS/E V7.Ø
RT-11 Emulator and Utilities Package
SAVRES.SAV

CHANGE SAVRES DEFAULT DENSITY TO 16ØØ BPI - SAVRES.SAV FEATURE PATCH

PROBLEM:

SAVRES, as released, defaults to the lowest density (of 8ØØ BPI and 16ØØ BPI) allowed by a given tape drive (normally 8ØØ BPI) for all tape I/O. It may be desirable at some installations to have SAVRES default to the highest density allowed by a given tape drive for more compact storage of data. This is especially true if all drives on the system support 16ØØ BPI and the problem described below is not applicable.

CAUTION

If the tape drive being used has a TMØ2 formatter, the hardware bootstrap of a 16ØØ BPI tape will not be possible. (You can determine the formatter type by using the HARDWR LIST suboption of INIT.) Therefore, if it is desirable for SAVRES to create tapes which are bootable on such a drive it is recommended that this patch not be installed. Normally, TU16 and TU45 drives use a TMØ2 formatter, TE16 and TU77 drives do not.

SOLUTION:

The following feature patch will cause SAVRES to default to the highest density allowed by a given tape drive for all tape I/O. This default can be overridden by attaching the /DENSITY:8ØØ switch to the device specification. Refer also to article Seq 1.1.1 F, which supplies a similar patch for the SAVRES option of INIT.

PROCEDURE:

1. This is a feature patch to SAVRES.SAV.

2. The patch described in Step 4 below can NOT be installed using the PATCH option of INIT.SYS.

3. This patch can be installed manually using ONLPAT, the on-line patching program:

```
RUN $ONLPAT
Command File Name? <cr>                    (RETURN for manual patch installation)
File to patch? [1,2]SAVRES.SAV
File found in account [1,2]
```

   The patch is also contained in a patch file appearing in patch kit version "A" or later.

4. The patch is as follows:

```
Base address? DIACTL
Offset address? 31Ø
 Base   Offset  Old     New?
??????  ØØØ31Ø  Ø3ØØ7Ø  ? "16
??????  ØØØ312  177460  ? "ØØ
??????  ØØØ314  Ø33Ø61  ? Ø34377
??????  ØØØ316  Ø3ØØ6Ø  ? <lf>          (no change; verify only)
??????  ØØØ32Ø  ØØØ377  ? <lf>          (no change; verify only)
??????  ØØØ322  ØØ144Ø  ? 16ØØ.
??????  ØØØ324  ØØ31ØØ  ? 8ØØ.
??????  ØØØ326  ØØØØØØ  ? ^Z            (CTRL/Z for new offset)
Offset address? ^Z                      (CTRL/Z for new base)
Base address? SAVCTL
Offset address? 42
 Base   Offset  Old     New?
??????  ØØØ42  Ø3ØØ7Ø  ? "16
??????  ØØØ44  177460  ? "ØØ
??????  ØØØ46  Ø33Ø61  ? Ø34377
??????  ØØØ5Ø  Ø3ØØ6Ø  ? <lf>          (no change; verify only)
??????  ØØØ52  ØØØ377  ? <lf>          (no change; verify only)
??????  ØØØ54  ØØ144Ø  ? 16ØØ.
??????  ØØØ56  ØØ31ØØ  ? 8ØØ.
??????  ØØØ6Ø  ØØØØØØ  ? ^Z            (CTRL/Z for new offset)
Offset address? ^Z                      (CTRL/Z for new base)
Base address? RESCTL
Offset address? 3Ø
 Base   Offset  Old     New?
??????  ØØØ3Ø  Ø3ØØ7Ø  ? "16
??????  ØØØ32  177460  ? "ØØ
??????  ØØØ34  Ø33Ø61  ? Ø34377
??????  ØØØ36  Ø3ØØ6Ø  ? <lf>          (no change; verify only)
??????  ØØØ4Ø  ØØØ377  ? <lf>          (no change; verify only)
??????  ØØØ42  ØØ144Ø  ? 16ØØ.
??????  ØØØ44  ØØ31ØØ  ? 8ØØ.
??????  ØØØ46  ØØØØØØ  ? ^C            (up-arrow/C to exit)
Patch complete
1 patch installed

Command File Name? ^Z                   (CTRL/Z to exit)
```

RSTS/E V7.Ø
RT-11 Emulator and Utilities Package
SAVRES.SAV

MAKING /NOERROR THE DEFAULT - SAVRES.SAV FEATURE PATCH

PROBLEM:

The /NOERROR switch indicates that SAVRES should abort under the following conditions:

1. A contiguous file is made non-contiguous
2. A placed file is 'unplaced'
3. A bad comparison occurs
4. An unexpected bad block is encountered on the input RSTS/E disk in a SAVE or IMAGE operation

Some installations, however, may always want SAVRES to abort under these circumstances.

SOLUTION:

The following feature patch will make /NOERROR, rather than /ERROR, the default in SAVRES.SAV. Note that the user may override the default for a particular operation by specifying /ERROR. Refer also to article Seq 1.1.2 F, which supplies a similar patch for the SAVRES option of INIT.

PROCEDURE:

1. This is a feature patch to SAVRES.SAV.

2. The patch described in Step 4 below can NOT be installed using the PATCH option of INIT.SYS.

3. This patch can be installed manually using ONLPAT, the on-line patching program:

```
RUN $ONLPAT
Command File Name? <cr>                 (RETURN for manual patch installation)
File to patch? [1,2]SAVRES.SAV
File found in account [1,2]
```

   The patch is also contained in a patch file appearing in patch kit version "A" or later.

4. The patch is as follows:

```
Base address? ..NOER
Offset address? 2
 Base   Offset  Old     New?
??????  ØØØØØ2  ØØØØ1Ø  ? 4
??????  ØØØØØ4  ??????  ? <lf>          (no change; verify only)
??????  ØØØØØ6  ??????  ? <lf>          (no change; verify only)
??????  ØØØ1Ø  ??????  ? <lf>          (no change; verify only)
??????  ØØØ12  ØØØØØ4  ? 1Ø
??????  ØØØ14  ??????  ? ^C            (up-arrow/C to exit)
Patch complete
1 patch installed

Command File Name? ^Z                   (CTRL/Z to exit)
```

RSTS/E V7.Ø
RT-11 Emulator and Utilities Package
SAVRES.SAV

MAKING /NOSTATS THE DEFAULT - SAVRES.SAV FEATURE PATCH

PROBLEM:

SAVRES always prints a summary report after completing a transfer unless the /NOSTATS (no statistics) switch is specified. Certain installations may prefer that this report NOT be printed unless specifically requested.

SOLUTION:

The following feature patch will make /NOSTATS, rather than /STATS, the default in SAVRES.SAV. Note that the user may override this default for a particular operation by specifying /STATS. Refer also to article Seq 1.1.3 F, which supplies a similar patch for the SAVRES option of INIT.

PROCEDURE:

1. This is a feature patch to SAVRES.SAV.

2. The patch described in Step 4 below can NOT be installed using the PATCH option of INIT.SYS.

3. This patch can be installed manually using ONLPAT, the on-line patching program:

```
RUN $ONLPAT
Command File Name? <cr>                 (RETURN for manual patch installation)
File to patch? [1,2]SAVRES.SAV
File found in account [1,2]
```

   The patch is also contained in a patch file appearing in patch kit version "A" or later.

4. The patch is as follows:

```
Base address? ..NOST
Offset address? 2
 Base   Offset  Old     New?
??????  ØØØØØ2  ØØØØ4Ø  ? 2Ø
??????  ØØØØØ4  ??????  ? <lf>          (no change; verify only)
??????  ØØØØØ6  ??????  ? <lf>          (no change; verify only)
??????  ØØØØ1Ø  ??????  ? <lf>          (no change; verify only)
??????  ØØØØ12  ØØØØ2Ø  ? 4Ø
??????  ØØØØ14  ??????  ? ^C            (up-arrow/C to exit)
Patch complete
1 patch installed

Command File Name? ^Z                   (CTRL/Z to exit)
```

INSTALLING DMS-5ØØ VØ2 ON RSTS/E V7.Ø

Prior to the installation of DMS-5ØØ VØ2 on RSTS/E V7.Ø, you must be running under the BASIC-PLUS Run-Time System. If you are not, you must first switch run-time systems. For example,

```
RUN $SWITCH<cr>
Run-Time System to switch to? BASIC<cr>

Ready
```

Various error messages may appear during the running of the demonstration program as part of the installation procedure. These problems will be corrected by the auto-patch kit.

Note that optional patch file PA24Ø1.CMD, included in Patch Kit "A" of the RSTS/E V7.Ø distribution kit, will repeat just the demonstration portion of the installation procedure.

NOTE

DMS-5ØØ VØ2 does not support large files (greater than 65535 blocks) on RSTS/E V7.Ø.

RSTS/E 278Ø Package V3.Ø
for RSTS/E V7.Ø
RJ278Ø

FAILURE TO CLEAR 'JOB ACTIVE' ON ATTACH - MANDATORY RJ278Ø PATCH

NOTE

This patch appeared in an earlier issue of the RSTS/E VØ6B-Ø2 Software Dispatch (September, 1977) as a patch to the RSTS/E VØ6B RJ278Ø.BAS program. Since the same program is used under the V7.Ø system, this patch is republished for the RSTS/E V7.Ø Release Notes. Because of this, the edit level remains at "VØ6B-Ø3". If you have applied the earlier VØ6B patch, do not apply this patch again.

PROBLEM:

Once RJ278Ø has been put into SPOOL mode, an ATTACH should cause any job being sent by QUEMAN to be REQUEUEd, and all variables in the RJ278Ø program itself concerning that job to be cleared. Processing should then resume, taking input from the terminal to which the program is ATTACHed. When SPOOL mode is re-entered, the same job should be sent by QUEMAN, and processed by the RJ278Ø program. Instead, the RJ278Ø program becomes confused when it receives a new SPOOL command, and the NEWJOB sent by QUEMAN causes an error message and the job is not processed.

SOLUTION:

The following patch causes reset of all variables upon ATTACH.

PROCEDURE:

1. For purposes of the discussion, we will assume that the required program to be patched is located on the public disk structure. In the patching procedure, we will refer to the program by its most simple name format: <program name>.BAS. The person performing the patching operation should, if necessary, replace all program references with suitable text according to the requirements of the installation.

   Also, for purposes of presenting the patching procedure, we assume that the patching operation will be performed in a privileged account.

2. The command file for this patch appears in patch kit version "A" or later. If you do not have this distribution you can produce a command file by specifying a file for [logfile=] in the procedure below.

   To apply the patch manually, perform the following RSTS/E system commands.

```
RUN $CPATCH<cr>
<CPATCH's header line>

File to patch - RJ278Ø.BAS=RJ278Ø.BAS<cr>
#[logfile=]<cr>
*H/6!/V<cr>
6!              EDIT            :       Ø3
*G/Ø3/I/A/V<cr>
6!              EDIT            :       Ø3A
```

```
*H/7!/V<cr>
7!              EDIT DATE       :       Ø2-JUN-77
*G/Ø2-JUN-77/-9C/Ø6-JUL-77/V<cr>
7!              EDIT DATE       :       Ø6-JUL-77
*H/21!/V<cr>
21!     VER/ED          EDIT DATE       REASON
*G/REASON/I<cr>
<lf>
  !<tab>6B-Ø3A<tab><tab>6-JUL-77<tab>FIX ATTACH/DETACH PROBLEM WITH<lf>
  !<tab><tab><tab><tab><tab>QUEMAN COMMUNICATION.<esc>*V<cr>
  !                                             QUEMAN COMMUNICATION.
*H/1Ø1Ø<tab>/V<cr>
1Ø1Ø    I$="VØ6B-Ø3"
*G/Ø3/I/A/V<cr>
1Ø1Ø    I$="VØ6B-Ø3A"
*H/1492<tab>/V<cr>
1492    M$="ATTACH"
*G/DETACH%=/V<cr>
        \ DETACH%=((E3% AND 1%)<>Ø%)
*ØAI<cr>
<tab>\ INJOB%,ABTJOB%=Ø%<lf>
<esc>*V<cr>
        \ DETACH%=((E3% AND 1%)<>Ø%)
*EX<cr>
Patch from KB:[P,PN]CPATCH.CMD complete
#^Z
File to patch - ^Z

Ready
```

3. To re-compile the program and re-enter it into the system library, type the following RSTS/E commands.

```
OLD RJ278Ø<cr>

Ready

COMPILE $RJ278Ø<cr>

Ready
```

4. The person making the changes to the program should now take whatever steps are necessary, according to installation guidelines, to save the new version of the program.

5. The source (.BAS) version of the program should now be removed from the public structure.

RSTS/E 278Ø Package V3.Ø
for RSTS/E V7.Ø
RJ278Ø

LOCAL BINARY TRANSMIT FAILURE - MANDATORY RJ278Ø PATCH

NOTE

RJ278Ø.BAS as released with RSTS/E VØ6B is used on all VØ6B and later versions of RSTS/E. For this reason, the edit level of the program remains at "VØ6B-Ø3".

PROBLEM:

If the default transmit command is 278Ø or GEN, it should be possible to send, as part of a multiple file transfer, a file or files in binary mode. This is done by using the "/B" switch to override the transmit command for those files which should be sent untranslated. Currently, the mode of the first file of a multiple file transfer controls the mode for the complete transfer.

SOLUTION:

The patching procedure detailed below will correct this problem.

NOTE

The mandatory 278Ø Device Driver patch described in article Seq 5.1.9 M, published in these notes, must be installed along with this patch.

PROCEDURE:

1. For purposes of the discussion, we will assume that the required program to be patched is located on the public disk structure. In the patching procedure, we will refer to the program by its most simple name format: <program name>.BAS. The person performing the patching operation should, if necessary, replace all program references with suitable text according to the requirements of the installation.

   Also, for purposes of presenting the patching procedure, we assume that the patching operation will be performed in a privileged account.

2. The command file for this patch appears in patch kit version "A" or later. If you do not have this distribution you can produce a command file by specifying a file for [logfile=] in the procedure below.

   To apply the patch manually, perform the following RSTS/E system commands.

```
RUN $CPATCH<cr>
<CPATCH's header line>

File to patch - RJ278Ø.BAS=RJ278Ø.BAS<cr>
#[logfile=]<cr>
*H/6!/V<cr>
6!                EDIT          :       Ø3A
*G/Ø3A/I/B/V<cr>
6!                EDIT          :       Ø3AB
```

```
*H/7!/V<cr>
7!              EDIT DATE       :       Ø6-JUL-77
*G/Ø6-JUL-77/-9C/23-MAR-79/V<cr>
7!              EDIT DATE       :       23-MAR-79
*H/21!/G/REASON/V<cr>
21!     VER/ED          EDIT DATE       REASON
*I<cr>
<lf>
  !<tab>6B-Ø3AB<tab><tab>23-MAR-79<tab>FIX LOCAL BINARY PROBLEM.<esc>*V<cr>
  !     6B-Ø3AB         23-MAR-79       FIX LOCAL BINARY PROBLEM.
*H/1Ø1Ø<tab>/V<cr>
1Ø1Ø    I$="V Ø6B-Ø3A"
*G/Ø3A/I/B/V<cr>
1Ø1Ø    I$="V Ø6B-Ø3AB"
*H/4Ø1Ø<tab>/V<cr>
4Ø1Ø    FIELD #1%, 134% AS B1$
*1ØAV<cr>
                ! FIELD THE RJ: BUFFER.
*I<cr>
<tab>\ PUTBIN%=NEWBIN%<lf>
<esc>*V<cr>
                ! FIELD THE RJ: BUFFER.
*H/413Ø<tab>/V<cr>
413Ø    F$=LEFT(CHR$(27%)+F$,MAXLEN%) IF X.TEST% AND 128%
*8AV<cr>
        \  RETURN
*I<cr>
<tab>\ PUTBIN%=NEWBIN%<lf>
<esc>*V<cr>
        \  RETURN
*H/423Ø<tab>/V<cr>
423Ø    E%,E2%=Ø%
*3AV<cr>
        \ PUTBIN%=(R% OR (B1% AND B2%)) AND 1%
*G/PUTBIN/-6C/NEWBIN/V<cr>
        \ NEWBIN%=(R% OR (B1% AND B2%)) AND 1%
*EX<cr>
Patch from KB:[P,PN]CPATCH.CMD complete
#^Z
File to patch - ^Z

Ready
```

3. To re-compile the program and re-enter it into the system library, type the following RSTS/E commands.

```
OLD RJ278Ø<cr>

Ready

COMPILE $RJ278Ø<cr>

Ready
```

4. The person making the changes to the program should now take whatever steps are necessary, according to installation guidelines, to save the new version of the program.

5. The source (.BAS) version of the program should now be removed from the public structure.

RSTS/E V7.Ø
DECnet/E Utilities V1.Ø
Package Notes

INSTALLING DECNET/E V1.Ø ON RSTS/E V7.Ø

Prior to the installation of DECnet/E V1.Ø on RSTS/E V7.Ø, you must be running under the BASIC-PLUS Run-Time System. If you are not, you must first switch run-time systems. For example,

```
RUN $SWITCH<cr>
Run-Time System to switch to? BASIC<cr>

Ready
```

Prior to the installation of DECnet/E V1.Ø on RSTS/E V7.Ø, LB: must be assigned to SY:[1,1].

DECnet/E Utilities V1.Ø
for RSTS/E V7.Ø
NCP

NCP ALLOWS INVALID NODE ADDRESS IN DEFINE NODE COMMAND - MANDATORY NCP PATCH

PROBLEM:

NCP allows a node address of 1 to be specified with the define node command. This is in violation of the NSP protocol. The valid range for node addresses is 2 to 24Ø.

SOLUTION:

The patching procecure detailed below corrects the above problem.

NOTE

When applying this patch, and if you are communicating with RSX family or IAS nodes, please insure that that node has installed the optional patch Seq 1Ø.2Ø.7.3 0 as published in the June 1978 issue of the RSX-11M Software Dispatch and the RSX-11D/IAS Software Dispatch.

PROCEDURE:

1. For purposes of the discussion, we will assume that the required program to be patched is located on the public disk structure. In the patching procedure, we will refer to the program by its most simple name format: <program name>.BAS. The person performing the patching operation should, if necessary, replace all program references with suitable text according to the requirements of the installation.

   Also, for purposes of presenting the patching procedure, we assume that the patching operation will be performed in a privileged account.

2. The command file for this patch appears in patch kit version "A" or later. If you do not have this distribution you can produce a command file by specifying a file for [logfile=] in the procedure below.

   To apply the patch manually, perform the following RSTS/E system commands.

```
RUN $CPATCH<cr>
<CPATCH's header line>

File to patch - NCP.BAS=NCP.BAS<cr>
#[logfile=]<cr>
*H/6!/V<cr>
6!              EDIT            : Ø3
*G/Ø3/I/A/V<cr>
6!              EDIT            : Ø3A
*H/7!/V<cr>
7!              EDIT DATE       : 3Ø-JAN-78
*G/3Ø-JAN-78/-9C/1Ø-MAY-78/V<cr>
7!              EDIT DATE       : 1Ø-MAY-78
```

```
*H/2Ø/V<cr>
2Ø      !
*G/REASON/V<cr>
21!     VER/ED          EDIT DATE       REASON
*I<cr>
<lf>
  !<tab>6C-Ø3A<tab><tab>1Ø-MAY-78<tab>SET NODE ADDRESS RANGE TO 2-24Ø<esc>*V<cr>
  !     6C-Ø3A          1Ø-MAY-78       SET NODE ADDRESS RANGE TO 2-24Ø
*H/1Ø1Ø<tab>/V<cr>
1Ø1Ø    I$="V06C-Ø3"
*G/Ø3/I/A/V<cr>
1Ø1Ø    I$="V06C-Ø3A"
*H/753Ø<tab>/V<cr>
753Ø    GOTO 91ØØ IF FNGPRM%(TEXT2.MSG$,NUM1$(NODE.ADDR%))
*3AV<cr>
        \ Z0%=FNINT.VAL%(Z$,1%,24Ø%)
*G/1%/-2C/2%/V<cr>
        \ Z0%=FNINT.VAL%(Z$,2%,24Ø%)
*EX<cr>
Patch from KB:[P,PN]CPATCH.CMD complete
#^Z
File to patch - ^Z

Ready
```

3. To re-compile the program and re-enter it into the system library, type the following RSTS/E commands.

```
OLD NCP<cr>

Ready

COMPILE $NCP<cr>

Ready
```

4. The person making the changes to the program should now take whatever steps are necessary, according to installation guidelines, to save the new version of the program.

5. The source (.BAS) version of the program may now be removed from the public structure.

NCU DOES NOT CORRECTLY HANDLE DETECTION OF ANOTHER NCU - MANDATORY NCU PATCH

PROBLEM:

If another copy of NCU is running, NCU goes into an infinite loop when extending the buffer file used for loop test.

SOLUTION:

The patching procedure detailed below corrects the above problem.

PROCEDURE:

1. For purposes of the discussion, we will assume that the required program to be patched is located on the public disk structure. In the patching procedure, we will refer to the program by its most simple name format: <program name>.BAS. The person performing the patching operation should, if necessary, replace all program references with suitable text according to the requirements of the installation.

   Also, for purposes of presenting the patching procedure, we assume that the patching operation will be performed in a privileged account.

2. The command file for this patch appears in patch kit version "A" or later. If you do not have this distribution you can produce a command file by specifying a file for [logfile=] in the procedure below.

   To apply the patch manually, perform the following RSTS/E system commands.

```
RUN $CPATCH<cr>
<CPATCH's header line>

File to patch - NCU.BAS=NCU.BAS<cr>
#[logfile=]<cr>
*H/6!/V<cr>
6!              EDIT            : Ø3
*G/Ø3/I/A/V<cr>
6!              EDIT            : Ø3A
*H/7!/V<cr>
7!              EDIT DATE       : 3Ø-JAN-78
*G/3Ø-JAN-78/-9C/Ø4-APR-78/V<cr>
7!              EDIT DATE       : Ø4-APR-78
*H/2Ø/V<cr>
2Ø      !
*G/REASON/V<cr>
21!     VER/ED          EDIT DATE       REASON
*I<cr>
<lf>
 !<tab>6C-Ø3A<tab><tab>Ø4-APR-78<tab>KILL BUFFER FILE SO TWO CAN NOT RUN.<esc>*V<cr>
 !      6C-Ø3A          Ø4-APR-78       KILL BUFFER FILE SO TWO CAN NOT RUN.
*H/1Ø1Ø<tab>/V<cr>
1Ø1Ø    I$="V6C-Ø3"
*G/Ø3/I/A/V<cr>
1Ø1Ø    I$="V6C-Ø3A"
```

```
*H/17ØØ/V<cr>
17ØØ    OPEN    "$NCUBUF.SYS" AS FILE 7%
*AV<cr>
        \ LOOP.DATA$(8%) = NULL$
*I<cr>
<tab>\ KILL "$NCUBUF.SYS"<lf>
<esc>*V<cr>
        \ LOOP.DATA$(8%) = NULL$
*EX<cr>
Patch from KB:[P,PN]CPATCH.CMD complete
#^Z
File to patch - ^Z

Ready
```

3. To re-compile the program and re-enter it into the system library, type the following RSTS/E commands.

```
OLD NCU<cr>

Ready

COMPILE $NCU<cr>

Ready
```

4. The person making the changes to the program should now take whatever steps are necessary, according to installation guidelines, to save the new version of the program.

5. The source (.BAS) version of the program may now be removed from the public structure.

DECnet/E Utilities V1.Ø
for RSTS/E V7.Ø
NCU

NCU DOES NOT CORRECTLY SET ITS RUN BURST AND PRIORITY - MANDATORY NCU PATCH

PROBLEM:

NCU does not correctly set its RUN BURST and PRIORITY at startup.

SOLUTION:

The patching procedure detailed below corrects the above problem.

PROCEDURE:

1. For purposes of the discussion, we will assume that the required program to be patched is located on the public disk structure. In the patching procedure, we will refer to the program by its most simple name format: <program name>.BAS. The person performing the patching operation should, if necessary, replace all program references with suitable text according to the requirements of the installation.

   Also, for purposes of presenting the patching procedure, we assume that the patching operation will be performed in a privileged account.

2. The command file for this patch appears in patch kit version "A" or later. If you do not have this distribution you can produce a command file by specifying a file for [logfile=] in the procedure below.

   To apply the patch manually, perform the following RSTS/E system commands.

```
RUN $CPATCH<cr>
<CPATCH's header line>

File to patch - NCU.BAS=NCU.BAS<cr>
#[logfile=]<cr>
*H/6!/V<cr>
6!              EDIT            : Ø3A
*G/Ø3A/I/B/V<cr>
6!              EDIT            : Ø3AB
*H/7!/V<cr>
7!              EDIT DATE       : Ø4-APR-78
*G/Ø4-APR-78/-9C/Ø8-AUG-78/V<cr>
7!              EDIT DATE       : Ø8-AUG-78
*H/2Ø/V<cr>
2Ø      !
*G/REASON/V<cr>
21!     VER/ED          EDIT DATE       REASON
*I<cr>
<lf>
  !<tab>6C-Ø3AB<tab><tab>Ø8-AUG-78<tab>FIX CALL TO SET PRIORITY AND<lf>
  !<tab><tab><tab><tab><tab>ALLOW RUN-BURST TO BE NON MULTIPLE OF 6<esc>*V<cr>
  !                                     ALLOW RUN-BURST TO BE NON MULTIPLE OF 6
```

```
*H/1Ø1Ø<tab>/V<cr>
1Ø1Ø     I$="VØ6C-Ø3A"
*G/Ø3A/I/B/V<cr>
1Ø1Ø     I$="VØ6C-Ø3AB"
*H/2Ø4Ø<tab>/V<cr>
2Ø4Ø     GOTO 2Ø9Ø UNLESS FNR%("RUNBURST",3%)
*AV<cr>
         \ Z%=FNINT.VAL%(CØ$,6%,255%)
*G/Z%/-2C/RUNB%/V<cr>
         \ RUNB%=FNINT.VAL%(CØ$,6%,255%)
*G/6%/-2C/1%/V<cr>
         \ RUNB%=FNINT.VAL%(CØ$,1%,255%)
*AV<cr>
         \ GOTO 2ØØØ IF Z%=999%
*G/Z%/-2C/RUNB%/V<cr>
         \ GOTO 2ØØØ IF RUNB%=999%
*AV<cr>
         \ RUNB%=Ø%
*K<cr>
*V<cr>
         \ RUNB%=RUNB%+6% UNTIL RUNB% >= Z%
*K<cr>
*H/2Ø6Ø<tab>/V<cr>
2Ø6Ø     Z$=SYS(CHR.6$+CHR$(-13%)
*G/-13%)/I/+CHR.M1$/V<cr>
2Ø6Ø     Z$=SYS(CHR.6$+CHR$(-13%)+CHR.M1$
*EX<cr>
Patch from KB:[P,PN]CPATCH.CMD complete
#^Z
File to patch - ^Z

Ready
```

3. To re-compile the program and re-enter it into the system library, type the following RSTS/E commands.

```
OLD NCU<cr>

Ready

COMPILE $NCU<cr>

Ready
```

4. The person making the changes to the program should now take whatever steps are necessary, according to installation guidelines, to save the new version of the program.

5. The source (.BAS) version of the program may now be removed from the public structure.

DECnet/E Utilities V1.Ø
for RSTS/E V7.Ø
TLK

TLK DOES NOT CHECK TO SEE IF LSN JOB NUMBER EXISTS - MANDATORY TLK PATCH

PROBLEM:

If the receiving terminal in a TLK dialogue is busy, LSN outputs a message indicating the TLK command line which should be typed to continue the dialogue. This command line is in the form "/LSNjobnumber source-node:: source-terminal:". TLK outputs an error message of '?COMMAND SYNTAX ERROR - ODD ADDRESS TRAP' if you specify a job number for which there is no active job.

SOLUTION:

The patching procedure detailed below corrects the above problem.

PROCEDURE:

1. For purposes of the discussion, we will assume that the required program to be patched is located on the public disk structure. In the patching procedure, we will refer to the program by its most simple name format: <program name>.BAS. The person performing the patching operation should, if necessary, replace all program references with suitable text according to the requirements of the installation.

   Also, for purposes of presenting the patching procedure, we assume that the patching operation will be performed in a privileged account.

2. The command file for this patch appears in patch kit version "A" or later. If you do not have this distribution you can produce a command file by specifying a file for [logfile=] in the procedure below.

   To apply the patch manually, perform the following RSTS/E system commands.

```
RUN $CPATCH<cr>
<CPATCH's header line>

File to patch - TLK.BAS=TLK.BAS<cr>
#[logfile=]<cr>
*H/6!/V<cr>
6!              EDIT            : Ø3
*G/Ø3/I/A/V<cr>
6!              EDIT            : Ø3A
*H/7!/V<cr>
7!              EDIT DATE       : 3Ø-JAN-78
*G/3Ø-JAN-78/-9C/1Ø-MAY-78/V<cr>
7!              EDIT DATE       : 1Ø-MAY-78
*H/2Ø/V<cr>
2Ø      !
*G/REASON/V<cr>
21!     VER/ED          EDIT DATE       REASON
*I<cr>
<lf>
  !<tab>6C-Ø3A<tab><tab>1Ø-MAY-78<tab>MAKE SURE JOB IS ALIVE ON ATTACH.<esc>*V<cr>
  !     6C-Ø3A          1Ø-MAY-78       MAKE SURE JOB IS ALIVE ON ATTACH.
```

```
*H/1Ø1Ø<tab>/V<cr>
1Ø1Ø     I$="V06C-Ø3"
*G/Ø3/I/A/V<cr>
1Ø1Ø     I$="V06C-Ø3A"
*H/1ØØ2Ø<tab>/V<cr>
1ØØ2Ø    E$="Command Syntax error - "
*14AV<cr>
         \ RIB%=PEEK(PEEK(PEEK(JOBTBL%+2%*LSN.JOB%)+8%)+18%)
*KV<cr>
         \ GOTO 1Ø22Ø IF RIB%=Ø%
*I<cr>
<tab>\ JDB% = PEEK(JOBTBL%+2%*LSN.JOB%)<lf>
<tab>\ GOTO 1Ø2ØØ UNLESS JDB% <> Ø%<lf>
<tab>\ RIB% = PEEK(PEEK(JDB%+8%)+18%)<lf>
<esc>*V<cr>
         \ GOTO 1Ø22Ø IF RIB%=Ø%
*17AV<cr>
                  ! BY FIRST INSURING THAT IT'S A LEGAL JOB # (<=MAXCNT).
*G/)/I/ AND ACTIVE/V<cr>
                  ! BY FIRST INSURING THAT IT'S A LEGAL JOB # (<=MAXCNT) AND ACTIVE.
*EX<cr>
Patch from KB:[P,PN]CPATCH.CMD complete
#^Z
File to patch - ^Z

Ready
```

3. To re-compile the program and re-enter it into the system library, type the following RSTS/E commands.

```
OLD TLK<cr>

Ready

COMPILE $TLK<232><cr>

Ready
```

4. The person making the changes to the program should now take whatever steps are necessary, according to installation guidelines, to save the new version of the program.

5. The source (.BAS) version of the program may now be removed from the public structure.

NODE NAMES OF LESS THAN 6 CHARACTERS NOT ACCEPTED - MANDATORY NCROOT PATCH

PROBLEM:

NETCPY does not pad node names to six characters with nulls. This causes NETCPY to give an INVALID NODE SPECIFICATION error on node names less than six characters long.

SOLUTION:

The patching procedure detailed below corrects the above problem.

PROCEDURE:

1. For purposes of the discussion, we will assume that the required program to be patched is located on the public disk structure. In the patching procedure, we will refer to the program by its most simple name format: <program name>.BAS. The person performing the patching operation should, if necessary, replace all program references with suitable text according to the requirements of the installation.

   Also, for purposes of presenting the patching procedure, we assume that the patching operation will be performed in a privileged account.

2. The command file for this patch appears on the patch kit version "A" or later. If you do not have this distribution you can produce a command file by specifying a file for [logfile=] in the procedure below.

   To apply the indicated corrections manually, perform the following RSTS/E system commands.

```
RUN $CPATCH<cr>
<CPATCH's header line>

File to patch - NCROOT.BAS=NCROOT.BAS<cr>
#[logfile=]<cr>
*H/6!/V<cr>
6!                EDIT            : Ø3
*G/Ø3/I/A/V<cr>
6!                EDIT            : Ø3A
*H/7!/V<cr>
7!                EDIT DATE       : 3Ø-JAN-78
*G/3Ø-JAN-78/-9C/16-FEB-78/V<cr>
7!                EDIT DATE       : 16-FEB-78
*H/2Ø/V<cr>
2Ø      !
*G/REASON/V<cr>
21!     VER/ED          EDIT DATE       REASON
*I<cr>
<lf>
  !<tab>6C-Ø3A<tab><tab>16-FEB-78<tab>PAD NODENAME TO 6 CHAR FOR CI.<esc>*V<cr>
  !     6C-Ø3A          16-FEB-78       PAD NODENAME TO 6 CHAR FOR CI.
```

```
*H/1Ø1Ø<tab>/V<cr>
1Ø1Ø     I$="V06C-Ø3"
*G/Ø3/I/A/V<cr>
1Ø1Ø     I$="V06C-Ø3A"
*H/1Ø8Ø<tab>/V<cr>
1Ø8Ø     OØ$=LEFT(A$,I%-1%)
*5AV<cr>
         \ IØ$=RIGHT(A$,I%+1%)
*I<cr>
<tab>\ TO.NODE.NAME$ = FNPAD$(TO.NODE.NAME$,6%)<lf>
<esc>*V<cr>
         \ IØ$=RIGHT(A$,I%+1%)
*4AV<cr>
         \ IØ$ = RIGHT(IØ$,I2%+2%) IF I2%
*I<cr>
<tab>\ FROM.NODE.NAME$ = FNPAD$(FROM.NODE.NAME$,6%)<lf>
<esc>*V<cr>
         \ IØ$ = RIGHT(IØ$,I2%+2%) IF I2%
*EX<cr>
Patch from KB:[P,PN]CPATCH.CMD complete
#^Z
File to patch - ^Z

Ready
```

3. To re-compile the program and re-enter it into the system library, type the following RSTS/E commands.

```
OLD NCROOT<cr>

Ready

APPEND $NETFNC.BAS<cr>

Ready

COMPILE $NETCPY<232><cr>

Ready
```

4. The person making the changes to the program should now take whatever steps are necessary, according to installation guidelines, to save the new version of the program.

5. The source (.BAS) version of the program may now be removed from the public structure.

DECnet/E Utilities V1.Ø
for RSTS/E V7.Ø
NCROOT

NETCPY DOES NOT ALLOW COPIES BETWEEN MM: AND MT: - MANDATORY NCROOT PATCH

PROBLEM:

NETCPY gives a MUST HAVE SAME TYPE DEVICES error on a copy between an MM: and an MT:.

SOLUTION:

The patching procedure detailed below corrects the above problem.

PROCEDURE:

1. For purposes of the discussion, we will assume that the required program to be patched is located on the public disk structure. In the patching procedure, we will refer to the program by its most simple name format: <program name>.BAS. The person performing the patching operation should, if necessary, replace all program references with suitable text according to the requirements of the installation.

   Also, for purposes of presenting the patching procedure, we assume that the patching operation will be performed in a privileged account.

2. The command file for this patch appears in patch kit version "A" or later. If you do not have this distribution you can produce a command file by specifying a file for [logfile=] in the procedure below.

   To apply the patch manually, perform the following RSTS/E system commands.

```
RUN $CPATCH<cr>
<CPATCH's header line>

File to patch - NCROOT.BAS=NCROOT.BAS<cr>
#[logfile=]<cr>
*H/6!/V<cr>
6!              EDIT            : Ø3A
*G/Ø3A/I/B/V<cr>
6!              EDIT            : Ø3AB
*H/7!/V<cr>
7!              EDIT DATE       : 16-FEB-78
*G/16-FEB-78/-9C/14-APR-78/V<cr>
7!              EDIT DATE       : 14-APR-78
*H/2Ø/V<cr>
2Ø      !
*G/REASON/V<cr>
21!     VER/ED          EDIT DATE       REASON
*I<cr>
<lf>
  !<tab>6C-Ø3AB<tab><tab>14-APR-78<tab>ALLOW MM: TO MT: COPY<esc>*V<cr>
  !     6C-Ø3AB         14-APR-78       ALLOW MM: TO MT: COPY
```

```
*H/1Ø1Ø<tab>/V<cr>
1Ø1Ø     I$="VØ6C-Ø3A"
*G/Ø3A/I/B/V<cr>
1Ø1Ø     I$="VØ6C-Ø3AB"
*H/11ØØ<tab>/V<cr>
11ØØ     E1%=3Ø%
*6AV<cr>
         \ IF LEFT(IØ$,2%)<>LEFT(OØ$,2%) THEN
*G/IF /I/(/V<cr>
         \ IF (LEFT(IØ$,2%)<>LEFT(OØ$,2%) THEN
*G/OØ$,2%)/I/)/V<cr>
         \ IF (LEFT(IØ$,2%)<>LEFT(OØ$,2%)) THEN
*I<cr>
<lf>
<tab><tab>AND ((LEFT(IØ$,2%)<>"MM" AND LEFT(IØ$,2%)<>"MT") OR<lf>
<tab><tab><tab>(LEFT(OØ$,2%)<>"MM" AND LEFT(OØ$,2%)<>"MT"))<esc>*V<cr>
                              (LEFT(OØ$,2%)<>"MM" AND LEFT(OØ$,2%)<>"MT")) THEN
*EX<cr>
Patch from KB:[P,PN]CPATCH.CMD complete
#^Z
File to patch - ^Z

Ready
```

3. To re-compile the program and re-enter it into the system library, type the following RSTS/E commands.

```
OLD NCROOT<cr>

Ready

APPEND $NETFNC.BAS<cr>

Ready

COMPILE $NETCPY<232><cr>

Ready
```

4. The person making the changes to the program should now take whatever steps are necessary, according to installation guidelines, to save the new version of the program.

5. The source (.BAS) version of the program may now be removed from the public structure.

DECnet/E Utilities V1.Ø
for RSTS/E V7.Ø
NFT

CREATE NFT/FAL OBJECT LIBRARY FOR PATCHING - MANDATORY NFT/FAL PATCH

PROBLEM:

The NETSUB library (including NFT and FAL) and the overlay description files, NFT.ODL and FAL.ODL, must be available for patching. After patching the OBJs the library file must be retained so that succeeding patches can be applied without having to reapply the earlier patches. These necessary files were not saved at the time of DECnet/E installation.

SOLUTION:

The patching procedure detailed below corrects the above problem.

PROCEDURE:

1. For purpose of this discussion, we will assume that the DECnet/E distribution kit is available and assigned with a logical name of I:.

   Also, for purposes of presenting the patching procedure, we assume that the patching operation will be performed in a privileged account.

2. The command file for this patch appears in patch kit version "A" or later.

   To apply the patch manually, perform the following RSTS/E system commands.

```
RUN $PIP.SAV<cr>
*SY:=I:$NETSUB.OLB<cr>
*SY:=I:$FAL.OBJ<cr>
*SY:=I:$FAL.ODL<cr>
*SY:=I:$NFT.OBJ<cr>
*SY:=I:$NFT.ODL<cr>
*^Z

Ready

RUN $LBR.TSK<cr>
LBR>NETSUB.OLB/IN=NFT.OBJ/-EP<cr>
LBR>NETSUB.OLB/IN=FAL.OBJ/-EP<cr>
LBR>^Z

Ready
```

```
RUN $CPATCH<cr>
<CPATCH's header line>

File to patch - NFT.ODL=NFT.ODL<cr>
#NFT.LOG=KB:<cr>
*H/.ROOT/V<cr>
.ROOT    NET:NFT-NFTØ-NFT1-RMSROT-RMSALL
*G?NET:?I?NETSUB/LB:?V<cr>
.ROOT    NET:NETSUB/LB:NFT-NFTØ-NFT1-RMSROT-RMSALL
*EX<cr>
Patch from KB:[P,PN]CPATCH.CMD complete
#^Z
File to patch - FAL.ODL=FAL.ODL<cr>
#NFT.LOG=KB:<cr>
*H/.ROOT/V<cr>
.ROOT    NET:FAL-FALØ-FAL1-RMSROT-RMSALL
*G?NET:?I?NETSUB/LB:?<cr>V<cr>
.ROOT    NET:NETSUB/LB:FAL-FALØ-FAL1-RMSROT-RMSALL
*EX<cr>
Patch from KB:[P,PN]CPATCH.CMD
#^Z
File to patch - ^Z

Ready

RUN $PIP.SAV<cr>
*LB:=NETSUB.OLB,NFT.ODL,FAL.ODL<cr>
*NETSUB.OLB,FAL.OBJ,FAL.ODL,NFT.OBJ,NFT.ODL/DE<cr>
*^Z

Ready
```

DECnet/E Utilities V1.Ø
for RSTS/E V7.Ø
NFT

RSTS/E-RT11 DIFFERENT BLOCK NUMBER FOR FIRST BLOCK OF FILE - MANDATORY NFT PATCH

PROBLEM:

NFT on RSTS/E and RT11 differ in the number of the first block in a file. RSTS/E uses 1 while RT11 uses Ø for the block number of the first block of a file. This causes the first block to get lost or to be copied twice when transferring files between RT11 and RSTS/E in block mode.

SOLUTION:

The patching procedure detailed below corrects the above problem.

NOTE

The mandatory FAL program patch described in article Seq 31.11.1M must be installed along with this patch.

PROCEDURE:

1. For purposes of this discussion, we will assume that the program to be patched is located in your privileged account on the public disk structure. If this is not the case, replace all program references with suitable text according to the requirements of your installation.

2. The command file for this patch appears in patch kit version "A" or later.

   To apply the patch manually, perform the following RSTS/E system commands.

```
RUN $PIP.SAV<cr>
*NFTØØ2.MAC=KB:<cr>
<tab>.TITLE NFT<tab>NETWORK FILE TRANSFER UTILITY<cr>
<tab>.IDENT<tab>"Ø6C.Ø59"<cr>
<tab>.PSECT<tab>NFT<cr>
$$BASE = .<cr>
.=<tab>$$BASE+5366<cr>
<tab>NOP<cr>
<tab>NOP<cr>
<tab>NOP<cr>
<tab>NOP<cr>
<tab>NOP<cr>
.=<tab>$$BASE+7672<cr>
<tab>.WORD Ø<cr>
.=<tab>$$BASE<cr>
.END<cr>
^Z
*^Z

Ready
```

```
RUN $MAC.TSK<cr>
MAC>NFTØØ2=NFTØØ2<cr>
MAC>^Z

Ready

RUN $LBR.TSK<cr>
LBR>NFT.OBJ=LB:NETSUB.OLB/EX:NFT<cr>
LBR>^Z

Ready

RUN $PAT.TSK<cr>
PAT>NFTNEW.OBJ=NFT.OBJ/CS:137767,NFTØØ2.OBJ/CS:Ø1ØØØ5<cr>
PAT>^Z

Ready

RUN $LBR.TSK<cr>
LBR>LB:NETSUB.OLB/RP=NFTNEW.OBJ/-EP<cr>

MODULE "NFT    " REPLACED
LBR>^Z

Ready

RUN $UTILTY<cr>
<UTILTY's header line>
#REMOVE LOGICAL NET<cr>                    <NOTE: this may generate an error>
#ADD LOGICAL LB: NET<cr>
#REMOVE LOGICAL RMS<cr>                    <NOTE: this may generate an error>
#ADD LOGICAL LB: RMS<cr>
#^Z

Ready

RUN $TKB.TSK<cr>
TKB>NFT/PI,NFT=NET:NFT.ODL/MP<cr>
ENTER OPTIONS:
TKB>STACK=1Ø24<cr>
TKB>HISEG=RMS11<cr>
TKB>//<cr>

Ready

RUN $PIP.SAV<cr>
*SY:$NFT.TSK<232>=NFT.TSK<cr>
*NFTØØ2.OBJ,NFTØØ2.MAC,NFT.OBJ,NFTNEW.OBJ/DE<cr>
*^Z

Ready
```

DECnet/E Utilities V1.Ø
for RSTS/E V7.Ø
NFT

INCORRECT DEFAULT FOR MAXIMUM RECORD SIZE ATTRIBUTE - MANDATORY NFT/FAL PATCH

PROBLEM:

NFT and FAL on RSTS/E use a default maximum record size (MRS) of 512 bytes. If a file with a maximum record size of Ø (no limit on record size) is copied using NFT, the copy of the file will have a maximum record size of 512 instead of Ø.

SOLUTION:

The patching procedure detailed below corrects the above problem.

PROCEDURE:

1. For purposes of this discussion, we will assume that the program to be patched is located in your privileged account on the public disk structure. If this is not the case, replace all program references with suitable text according to the requirements of your installation.

2. The command file for this patch appears in patch kit version "A" or later.

   To apply the patch manually, perform the following RSTS/E system commands.

```
RUN $PIP.SAV<cr>
*RMSØØ3.MAC=KB:<cr>
<tab>.TITLE<tab>RMSFAR<tab>RMS FILE ACCESS ROUTINES<cr>
<tab>.IDENT<tab>"Ø6C.ØØ5"<cr>
<tab>.PSECT<tab>RMSFAR<cr>
$$BASE = .<cr>
. = $$BASE+66<cr>
<tab>.WORD<tab>Ø<cr>
. = $$BASE<cr>
<tab>.END<cr>
^Z
*^Z

Ready

RUN $MAC.TSK<cr>
MAC>RMSØØ3=RMSØØ3<cr>
MAC>^Z

Ready

RUN $LBR.TSK<cr>
LBR>RMSFAR.OBJ=LB:NETSUB.OLB/EX:RMSFAR<cr>
LBR>^Z

Ready
```

```
RUN $PAT.TSK<cr>
PAT>RMSNEW.OBJ=RMSFAR.OBJ/CS:124336,RMSØØ3.OBJ/CS:ØØ611Ø<cr>
PAT>^Z

Ready

RUN $LBR.TSK<cr>
LBR>LB:NETSUB.OLB/RP=RMSNEW.OBJ<cr>

MODULE "RMSFAR" REPLACED
LBR>^Z

Ready

RUN $UTILTY<cr>
<UTILTY's header line>
#REMOVE LOGICAL NET<cr>                    <NOTE: this may generate an error>
#ADD LOGICAL LB: NET<cr>
#REMOVE LOGICAL RMS<cr>                    <NOTE: this may generate an error>
#ADD LOGICAL LB: RMS<cr>
#^Z

Ready

RUN $TKB.TSK<cr>
TKB>FAL/PI,FAL=NET:FAL.ODL/MP<cr>
ENTER OPTIONS:
TKB>STACK=1Ø24<cr>
TKB>HISEG=RMS11<cr>
TKB>//<cr>

Ready

RUN $TKB.TSK<cr>
TKB>NFT/PI,NFT=NET:NFT.ODL/MP<cr>
ENTER OPTIONS:
TKB>STACK=1Ø24<cr>
TKB>HISEG=RMS11<cr>
TKB>//<cr>

Ready

RUN $PIP.SAV<cr>
*SY:$FAL.TSK<232>=FAL.TSK<cr>
*SY:$NFT.TSK<232>=NFT.TSK<cr>
*RMSØØ3.MAC,RMSØØ3.OBJ,RMSFAR.OBJ,RMSNEW.OBJ/DE<cr>
*^Z

Ready
```

DECnet/E Utilities V1.Ø
for RSTS/E V7.Ø
NFT

RESOLUTION OF BLOCK NUMBER PROBLEM BETWEEN RSTS/E-RT11 - MANDATORY NFT PATCH

PROBLEM:

As described in article Seq 31.1Ø.2 M, published in these Release Notes, NFT on RSTS/E and RT11 differ in the number of the first block in a file. This situation has changed since publication of that article, and RSTS/E and RT11 now number the first block of a file in the same manner.

SOLUTION:

The patching procedure detailed below resolves the above problem.

NOTE

The mandatory FAL program patch described in article Seq 31.11.3 M must be installed along with this patch.

PROCEDURE:

1. For purposes of the discussion, we will assume that the required files to be patched are located on the public disk structure. The person performing the patching operation should, if necessary, replace all program references with suitable text according to the requirements of the installation.

   Also, for purposes of presenting the patching procedure, we assume that the patching operation will be performed in a privileged account.

2. The command file for this patch appears in patch kit version "A" or later.

   To apply the patch manually, perform the following RSTS/E system commands.

```
RUN $PIP.SAV<cr>
*NFTØØ4.MAC=KB:<cr>
<tab>.TITLE<tab>NFT<tab>NETWORK FILE TRANSFER UTILITY<cr>
<tab>.IDENT<tab>"Ø6C.Ø59"<cr>
<tab>.PSECT<tab>NFT<cr>
$$BASE = .<cr>
.=<tab>$$BASE+5366<cr>
<tab>ADD<tab>#1,46(RØ)<cr>
<tab>ADC<tab>5Ø(RØ)<cr>
.=<tab>$$BASE+7672<cr>
<tab>.WORD<tab>1<cr>
.=<tab>$$BASE<cr>
.END<cr>
^Z
*^Z

Ready
```

```
RUN $MAC.TSK<cr>
MAC>NFTØØ4=NFTØØ4<cr>
MAC>^Z

Ready

RUN $LBR.TSK<cr>
LBR>NFT.OBJ=LB:NETSUB.OLB/EX:NFT<cr>
LBR>^Z

Ready

RUN $PAT.TSK<cr>
PAT>NFTNEW.OBJ=NFT.OBJ/CS:143610,NFTØØ4.OBJ/CS:ØØ7405<cr>
PAT>^Z

Ready

RUN $LBR.TSK<cr>
LBR>LB:NETSUB.OLB/RP=NFTNEW.OBJ/-EP<cr>

MODULE "NFT    " REPLACED
LBR>^Z

Ready

RUN $UTILTY<cr>
<UTILTY's header line>
#REMOVE LOGICAL NET<cr>                    <NOTE: this may generate an error>
#ADD LOGICAL LB: NET<cr>
#REMOVE LOGICAL RMS<cr>                    <NOTE: this may generate an error>
#ADD LOGICAL LB: RMS<cr>
#^Z

Ready

RUN $TKB.TSK<cr>
TKB>NFT/PI,NFT=NET:NFT.ODL/MP<cr>
ENTER OPTIONS:
TKB>STACK=1Ø24<cr>
TKB>HISEG=RMS11<cr>
TKB>//<cr>

Ready

RUN $PIP.SAV<cr>
*SY:$NFT.TSK<232>=NFT.TSK<cr>
*NFTØØ4.OBJ,NFTØØ4.MAC,NFT.OBJ,NFTNEW.OBJ/DE<cr>
*^Z

Ready
```

DECnet/E Utilities V1.Ø
for RSTS/E V7.Ø
FAL

RSTS/E-RT11 DIFFERENT BLOCK NUMBER FOR FIRST BLOCK OF FILE - MANDATORY FAL PATCH

PROBLEM:

FAL on RSTS/E and RT11 differ in the number of the first block in a file. RSTS/E uses 1 while RT11 uses Ø for the block number of the first block of a file. This causes the first block to get lost or to be copied twice when transferring files between RT11 and RSTS/E in block mode.

SOLUTION:

The patching procedure detailed below corrects the above problem.

NOTE

The mandatory NFT program patch described in article Seq 31.1Ø.2M must be installed along with this patch.

PROCEDURE:

1. For purposes of this discussion, we will assume that the program to be patched is located in your privileged account on the public disk structure. If this is not the case, replace all program references with suitable text according to the requirements of your installation.

2. The command file for this patch appears in patch kit version "A" or later.

   To apply the patch manually, perform the following RSTS/E system commands.

```
RUN $PIP.SAV<cr>
*FALØØ1.MAC=KB:<cr>
<tab>.TITLE FAL<tab>FILE ACCESS LISTENER<cr>
<tab>.IDENT<tab>"Ø6C.41"<cr>
<tab>.PSECT<tab>FAL<cr>
$$BASE = .<cr>
.=<tab>$$BASE+1412<cr>
<tab>NOP<cr>
<tab>NOP<cr>
<tab>NOP<cr>
<tab>NOP<cr>
<tab>NOP<cr>
.=<tab>$$BASE+16Ø6<cr>
<tab>NOP<cr>
<tab>NOP<cr>
<tab>NOP<cr>
<tab>NOP<cr>
<tab>NOP<cr>
.=<tab>$$BASE<cr>
.END<cr>
^Z
*^Z

Ready
```

```
RUN $MAC.TSK<cr>
MAC>FALØØ1=FALØØ1<cr>
MAC>^Z

Ready

RUN $LBR.TSK<cr>
LBR>FAL.OBJ=LB:NETSUB.OLB/EX:FAL<cr>
LBR>^Z

Ready

RUN $PAT.TSK<cr>
PAT>FALNEW.OBJ=FAL.OBJ/CS:145264,FALØØ1.OBJ/CS:Ø1ØØ33<cr>
PAT>^Z

Ready

RUN $LBR.TSK<cr>
LBR>LB:NETSUB.OLB/RP=FALNEW.OBJ/-EP<cr>

MODULE "FAL    " REPLACED
LBR>^Z

Ready

RUN $UTILTY<cr>
<UTILTY's header line>
#REMOVE LOGICAL NET<cr>                  <NOTE: this may generate an error>
#ADD LOGICAL LB: NET<cr>
#REMOVE LOGICAL RMS<cr>                  <NOTE: this may generate an error>
#ADD LOGICAL LB: RMS<cr>
#^Z

Ready

RUN $TKB.TSK<cr>
TKB>FAL/PI,FAL=NET:FAL.ODL/MP<cr>
ENTER OPTIONS:
TKB>STACK=1Ø24<cr>
TKB>HISEG=RMS11<cr>
TKB>//<cr>

Ready

RUN $PIP.SAV<cr>
*SY:$FAL.TSK<232>=FAL.TSK<cr>
*FALØØ1.OBJ,FALØØ1.MAC,FALNEW.OBJ,FAL.OBJ/DE<cr>
*^Z

Ready
```

DECnet/E Utilities V1.Ø
for RSTS/E V7.Ø
FAL

SET ONE SHOT BIT IN FAL'S DECLARE RECEIVER - MANDATORY FAL PATCH

PROBLEM:

FAL is an eligible receiver for a CONNECT after a disconnect is queued for it. FAL after receiving the disconnect does a REMOVE RECEIVER. If a CONNECT is queued to FAL when it does the REMOVE RECEIVER the CONNECT will be rejected.

SOLUTION:

The patching procedure detailed below corrects the above problem.

NOTE

The mandatory DECnet/E NSP patch described in article Seq 6.1.1 M must be installed along with this patch.

PROCEDURE:

1. For purposes of this discussion, we will assume that the program to be patched is located in your privileged account on the public disk structure. If this is not the case, replace all program references with suitable text according to the requirements of your installation.

2. The command file for this patch appears in patch kit version "A" or later.

   To apply the patch manually, perform the following RSTS/E system commands.

```
RUN $PIP.SAV<cr>
*FALØØ2.MAC=KB:<cr>
<tab>.TITLE FAL<tab>FILE ACCESS LISTENER<cr>
<tab>.IDENT<tab>"Ø6C.41"<cr>
<tab>.PSECT<tab>IMPURE<cr>
$$IMPB = .<cr>
.=<tab>$$IMPB+1565<cr>
<tab>.BYTE<tab>8.+4.<cr>
<tab>.PSECT<tab>FAL<cr>
$$BASE = .<cr>
.=<tab>$$BASE+3224<cr>
<tab>JMP<tab>1Ø$<cr>
.=<tab>$$BASE+3466<cr>
1Ø$:<cr>
.=<tab>$$BASE<cr>
.END<cr>
^Z
*^Z
```

   Ready

```
RUN $MAC.TSK<cr>
MAC>FALØØ2=FALØØ2<cr>
MAC>^Z

Ready

RUN $LBR.TSK<cr>
LBR>FAL.OBJ=LB:NETSUB.OLB/EX:FAL<cr>
LBR>^Z

Ready

RUN $PAT.TSK<cr>
PAT>FALNEW.OBJ=FAL.OBJ/CS:151371,FALØØ2.OBJ/CS:ØØ7235<cr>
PAT>^Z

Ready

RUN $LBR.TSK<cr>
LBR>LB:NETSUB.OLB/RP=FALNEW.OBJ/-EP<cr>

MODULE "FAL    " REPLACED
LBR>^Z

Ready

RUN $UTILTY<cr>
<UTILTY's header line>
#REMOVE LOGICAL NET<cr>                  <NOTE: this may generate an error>
#ADD LOGICAL LB: NET<cr>
#REMOVE LOGICAL RMS<cr>                  <NOTE: this may generate an error>
#ADD LOGICAL LB: RMS<cr>
#^Z

Ready

RUN $TKB.TSK<cr>
TKB>FAL/PI,FAL=NET:FAL.ODL/MP<cr>
ENTER OPTIONS:
TKB>STACK=1Ø24<cr>
TKB>HISEG=RMS11<cr>
TKB>//<cr>

Ready

RUN $PIP.SAV<cr>
*SY:$FAL.TSK<232>=FAL.TSK<cr>
*FALØØ2.OBJ,FALØØ2.MAC,FALNEW.OBJ,FAL.OBJ/DE<cr>
*^Z

Ready
```

DECnet/E Utilities V1.Ø
for RSTS/E V7.Ø
FAL

RESOLUTION OF BLOCK NUMBER PROBLEM BETWEEN RSTS/E-RT11 - MANDATORY FAL PATCH

PROBLEM:

As described in article Seq 31.11.1 M, published in these Release Notes, FAL on RSTS/E and RT11 differ in the number of the first block in a file. This situation has changed since publication of that article, and RSTS/E and RT11 now number the first block of a file in the same manner.

SOLUTION:

The patching procedure detailed below resolves the above problem.

NOTE

The mandatory NFT program patch described in article Seq 31.1Ø.4 M must be installed along with this patch.

PROCEDURE:

1. For purposes of the discussion, we will assume that the required files to be patched are located on the public disk structure. The person performing the patching operation should, if necessary, replace all program references with suitable text according to the requirements of the installation.

   Also, for purposes of presenting the patching procedure, we assume that the patching operation will be performed in a privileged account.

2. The command file for this patch appears in patch kit version "A" or later.

   To apply the patch manually, perform the following RSTS/E system commands.

```
RUN $PIP.SAV<cr>
*FALØØ3.MAC=KB:<cr>
<tab>.TITLE<tab>FAL<tab>FILE ACCESS LISTENER<cr>
<tab>.IDENT<tab>"Ø6C.41"<cr>
<tab>.PSECT<tab>FAL<cr>
$$BASE = .<cr>
.=<tab>$$BASE+1412<cr>
<tab>ADD<tab>#1,46(R1)<cr>
<tab>ADC<tab>5Ø(R1)<cr>
.=<tab>$$BASE+16Ø6<cr>
<tab>ADD<tab>#1,46(RØ)<cr>
<tab>ADC<tab>5Ø(RØ)<cr>
.=<tab>$$BASE<cr>
.END<cr>
^Z
*^Z

Ready
```

```
RUN $MAC.TSK<cr>
MAC>FALØØ3=FALØØ3<cr>
MAC>^Z

Ready

RUN $LBR.TSK<cr>
LBR>FAL.OBJ=LB:NETSUB.OLB/EX:FAL<cr>
LBR>^Z

Ready

RUN $PAT.TSK<cr>
PAT>FALNEW.OBJ=FAL.OBJ/CS:154140,FALØØ3.OBJ/CS:ØØ7Ø33<cr>
PAT>^Z

Ready

RUN $LBR.TSK<cr>
LBR>LB:NETSUB.OLB/RP=FALNEW.OBJ/-EP<cr>

MODULE "FAL    " REPLACED
LBR>^Z

Ready

RUN $UTILTY<cr>
<UTILTY's header line>
#REMOVE LOGICAL NET<cr>                    <NOTE: this may generate an error>
#ADD LOGICAL LB: NET<cr>
#REMOVE LOGICAL RMS<cr>                    <NOTE: this may generate an error>
#ADD LOGICAL LB: RMS<cr>
#^Z

Ready

RUN $TKB.TSK<cr>
TKB>FAL/PI,FAL=NET:FAL.ODL/MP<cr>
ENTER OPTIONS:
TKB>STACK=1Ø24<cr>
TKB>HISEG=RMS11<cr>
TKB>//<cr>

Ready

RUN $PIP.SAV<cr>
*SY:$FAL.TSK<232>=FAL.TSK<cr>
*FALØØ3.OBJ,FALØØ3.MAC,FAL.OBJ,FALNEW.OBJ/DE<cr>
*^Z

Ready
```

RPG II V8.Ø2
for RSTS/E V7.Ø
Package Notes

INSTALLING RPG II V8.Ø2 ON RSTS/E V7.Ø

To install RPG II V8.Ø2 on RSTS/E V7.Ø, you must use the RPGGEN.CTL and RPGNB2.CTL command files on Patch Kit "A" of the RSTS/E V7.Ø distribution kit instead of those on the RPG distribution kit.

Various error messages will appear during the compilation of RPGDMP. These problems are corrected by a patch file in the patch kit.

WISE V1
for RSTS/E V7.Ø
Package Notes

INSTALLING WISE V1 ON RSTS/E V7.Ø

To install WISE V1 on RSTS/E V7.Ø, you must use the INWISE.BAS program on Patch Kit "A" of the RSTS/E V7.Ø distribution kit instead of the one on the WISE distribution kit. The PIP command line syntax in the version on the WISE kit is not compatible with PIP.SAV.

DIBOL-11/DECFORM V4D
for RSTS/E V7.Ø
Build Procedures

INSTALLING DIBOL V4D ON RSTS/E V7.Ø

Prior to the installation of DIBOL V4D on RSTS/E V7.Ø, you must be running under the BASIC-PLUS Run-Time System. If you are not, you must first switch run-time systems. For example,

```
RUN $SWITCH<cr>
Run-Time System to switch to? BASIC<cr>

Ready
```

Various error messages may appear during the task-build of the demonstration program as part of the installation procedure. These messages may be ignored; the demonstration program will execute correctly.

DIBOL-11/DECFORM V4C
for RSTS/E V7.Ø
Build Procedures

INSTALLING DIBOL V4C ON RSTS/E V7.Ø

Prior to the installation of DIBOL V4C on RSTS/E V7.Ø, you must be running under the BASIC-PLUS Run-Time System. If you are not, you must first switch run-time systems. For example,

```
RUN $SWITCH<cr>
Run-Time System to switch to? BASIC<cr>

Ready
```

DATATRIEVE V1.1
for RSTS/E V7.Ø
Package Notes

INSTALLING DATATRIEVE V1.1 ON RSTS/E V7.Ø

Prior to the installation of Datatrieve V1.1 on RSTS/E V7.Ø, LB: must be assigned to SY:[1,1]. In order to install Datatrieve V1.1, you must be logged into a privileged account other than [1,2].

COBOL-11 V4.Ø
for RSTS/E V7.Ø
Package Notes

INSTALLING COBOL V4.Ø ON RSTS/E V7.Ø

Prior to the installation of COBOL V4.Ø on RSTS/E V7.Ø, LB: must be assigned to SY:[1,1]. When building COBOL V4.Ø, use the INSCBL.CMD command file on Patch Kit "A" of the RSTS/E V7.Ø distribution kit instead of the one on the COBOL distribution kit.

The RMS-11 distribution contains ODL files that, in the past, were installed from the COBOL kit. If the command file from the COBOL kit is used, it will replace the new ODLs with obsolete ones. The COBOL Merge Utility uses these files to build overlay descriptor references to RMS I/O routines. The RMS-11 V1.8 distribution kit contains the correct ODLs.

The RSTS/E V7.Ø patch kit contains a new copy of CBLMRG.OLB and an object module to be included in COBOVR.OLB. These will automatically be included when the patches to COBOL V4.Ø are applied.

DX/RSTS V3.Ø
for RSTS/E V7.Ø
Package Notes

INSTALLING DX/RSTS V3.Ø ON RSTS/E V7.Ø

1. DX/RSTS V3.Ø will not build if PIP.SAV has been patched to make "/LOG" the default. (See article Seq 22.13.7 F.) This patch must be removed to accomplish the build.

2. DX/RSTS V3.Ø DXLPT program will not run when invoked with the "RUN" command. Modify the CCL file as described on page 5 of the DX/RSTS Installation Guide.

3. When building DX/RSTS V3.Ø from magnetic tape distribution media, do not issue the "MOUNT" command called for in paragraph 2, page 3 of the DX/RSTS Installation Guide. This command applies only when building from disk distribution media.

USER ASSIGNED COMPONENTS

Many users generate patch command and patch files for their own software or generate installation specific patches for Digital supplied software. For those users who wish to maintain these files in a format consistant with the Digital supplied patches, the following Component.subcomponent numbers have been reserved:

9Ø.Ø User Assigned Components - For Development Use
91.Ø - 99.Ø User Assigned Components - For User Use

Component 9Ø will be used by the RSTS/E development group for notes such as this and any information we feel may be of use to the users of this facility. Components 91 through 99 will not be assigned by the development group and will be permanently "Reserved."

9.Ø Software Product Descriptions (SPDs) and Option Bulletins

9.1 Software Product Descriptions (SPDs)

Periodically, new or revised Software Product Description (SPD) bulletins will appear in the RSTS/E Software Dispatch for various software products marketed by DIGITAL. This section of the notebook is provided as a convenient place to file these SPDs for future reference.

Every SPD has a unique number, a sample of which is shown below:

SPD 14.52.Ø

This number appears on the first page of the SPD bulletin, at the far right hand side of the PRODUCT NAME: line.

Note that the last part of the SPD number is the revision level, with Ø being assigned for original issues of SPDs.

9.2 Option Bulletins

Periodically, new or revised Option Bulletins will appear in the RSTS/E Software Dispatch for various products marketed by DIGITAL. This section of the release notes is provided as a convenient place to file these bulletins. There is no filing scheme for option bulletins.

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 MK1-2/ H3
DIGITAL EQUIPMENT CORPORATION
CONTINENTAL BOULEVARD
MERRIMACK N.H. 03054

— — — Do Not Tear - Fold Here and Tape

Cut Along Dotted Line