

RSTS/E UPDATE

OVR

Overview of RSTS/E V7.0

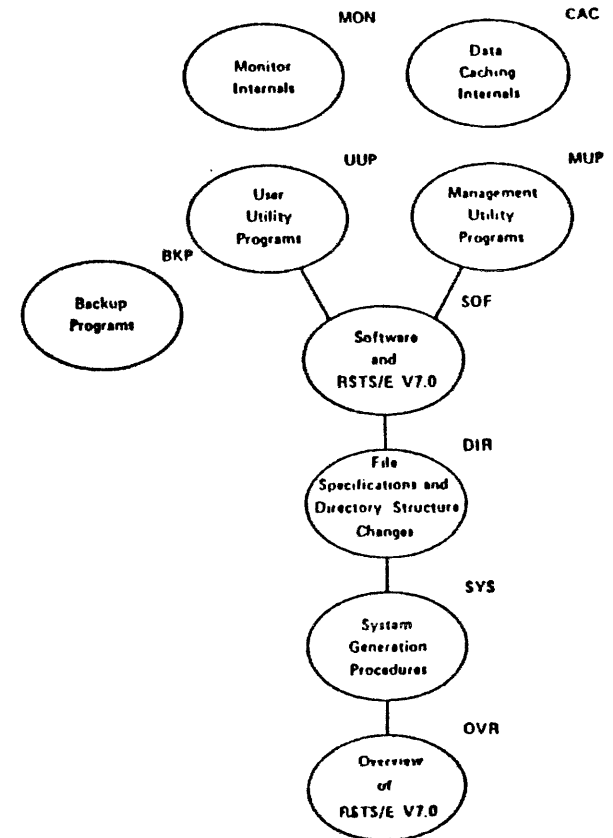
Overview of RSTS/E V7.0

CONTENTS

Introduction	OVR-1
New Hardware Support	OVR-3
Disks.....	OVR-3
Terminals.....	OVR-5
Additional VT100 Documentation.....	OVR-12
Definition of Each VT100 Set-up Feature.....	OVR-12
ANSI/VT52 Mode.....	OVR-12
Answerback Message.....	OVR-12
Auto Repeat.....	OVR-12
Auto XON/XOFF.....	OVR-13
Bits Per Character.....	OVR-14
Characters Per Line.....	OVR-14
Cursor.....	OVR-14
Interlace.....	OVR-15
Keyclick Tone.....	OVR-15
LINE/LOCAL.....	OVR-15
Margin Bell.....	OVR-15
New Line.....	OVR-16
Parity.....	OVR-16
Parity Sense.....	OVR-16
Power.....	OVR-17
Receive Speed.....	OVR-17
Screen Background.....	OVR-17
Screen Brightness.....	OVR-17
Scroll.....	OVR-18
Tabs.....	OVR-19
Transmit Speed.....	OVR-18
Wraparound.....	OVR-19
Magnetic Tapes.....	OVR-20
Obsolete Hardware.....	OVR-22
New Operating System Features.....	OVR-23
Large Files.....	OVR-23
Data Caching.....	OVR-23
RSX Directives in Monitor.....	OVR-24
Resident Libraries.....	OVR-24

User Logical Assignments with PPN's.....	OVR-25
Placed Files.....	OVR-25
Tentative Files.....	OVR-25
Conditionally Contiguous Files.....	OVR-25
No File Supersede.....	OVR-26
Zero a Disk File.....	OVR-26
Make a Contiguous File Non-Contiguous.....	OVR-26
Library Segmentation.....	OVR-27
ANSI Magnetic Tape Support.....	OVR-27
BASIC-PLUS-2 Only Systems.....	OVR-27
COBOL Only Systems.....	OVR-28
Utility Programs.....	OVR-29
New Utilities.....	OVR-29
Discontinued Utilities.....	OVR-30
Bundled Software.....	OVR-31
Unbundled Software.....	OVR-32
Category C Software.....	OVR-32
Documentation.....	OVR-33

COURSE MAP



Overview of RSTS/E V7.0

INTRODUCTION

RSTS/E V7.0 is a major feature release of the operating system. Performance improvements, enhanced functionality, and new hardware support have been included while maintaining upward compatibility with previous versions of the RSTS/E operating system.

- o Have you been concerned with long task build times?
- o Have you been concerned by disk I/O throughput time?
- o Have you been concerned with the 65K block limitation for files?
- o Have you been concerned with COBOL performance?
- o Have you been concerned with the limitations of BACKUP?

RSTS/E V7.0 addresses all of the above and more.

This module covers the major features of RSTS/E V7.0 at a conceptual level. The modules that follow address these new features in more detail.

DIGITAL

RSTS/E UPDATE
OVERVIEW OF RSTS/E V7.0

DIGITAL

RSTS/E UPDATE
OVERVIEW OF RSTS/E V7.0

Overview of RSTS/E V7.0

This page is for notes.

NEW HARDWARE SUPPORT

Disks

1. RL02
 - o Up to 4 RL01/RL02 disks on a system
2. RX02 - floppy disk
 - o Double or single density user selectable
 - o Single density is RX01 compatible
 - o Uses RX01-K diskettes
 - o DMA access device
 - o Supported as non-file structured device only
 - o Up to 8 RX01/RX02 diskettes on a system
 - o Improved diagnostic capability

Table OVR-1. Disk Characteristics

DISK NAME	RL02	RX02 DOUBLE DENSITY	RX02 SINGLE DENSITY
DISK TYPE	CARTRIDGE	DOUBLE DENSITY FLOPPY	SINGLE DENSITY FLOPPY
No. of Surfaces/disk	2	1	1
No. of Blocks/disk	20460	1782	1782
No. of Words/disk	5.2M	256K	128K
Device Clustersize	1	N/A	N/A
Max. No. of Drives	4	8	8
Access Time	100 Milli- seconds	Average 262 Milliseconds	Average 483 Milliseconds
Transfer Time	3.9 words/mi- crosecond	Peak 62kb/ second	Peak 31 kb/ second

Terminals

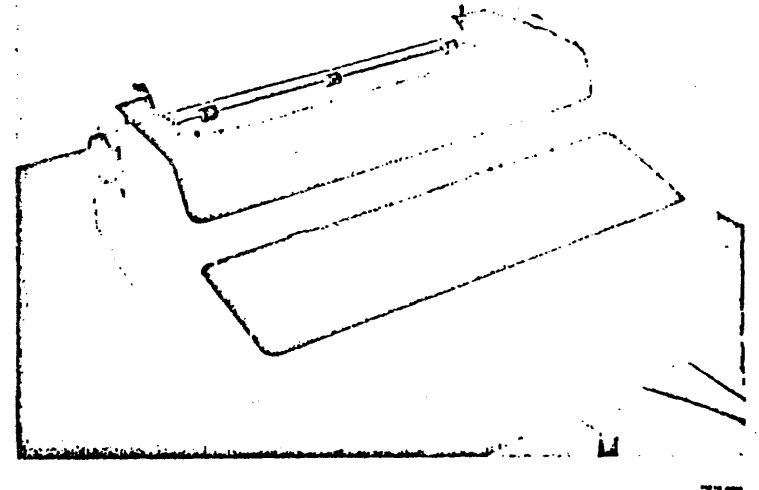


Figure OVR-1. LA34-DECwriter IV

- o LA36 equivalent
- o Up to 300 baud
- o Up to 220 columns (depending on print size)
- o Set up mode allows user selectable terminal functions

DIGITAL

RSTS/E UPDATE
OVERVIEW OF RSTS/E V7.0

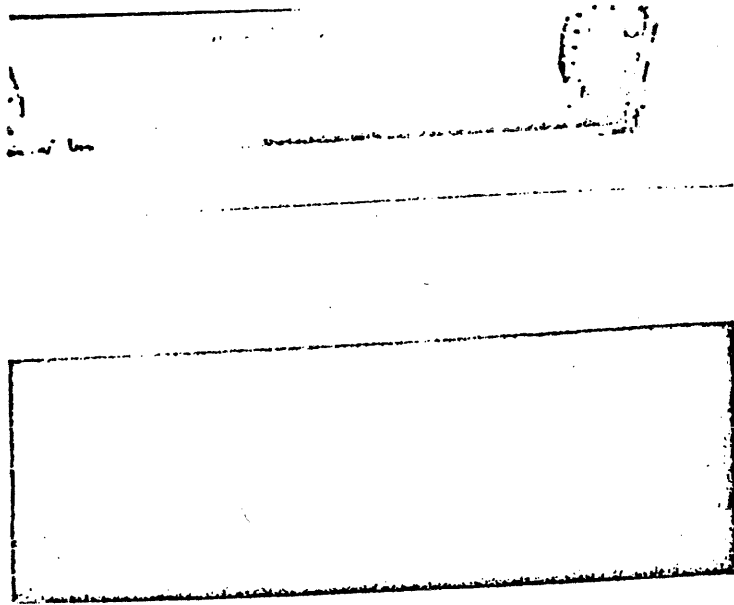


Figure OVR-4. LA120 - DECwriter III Keyboard

OVR-8

COMPANY CONFIDENTIAL

DIGITAL

RSTS/E UPDATE
OVERVIEW OF RSTS/E V7.0

DECWRITER III LA120 OPERATOR REFERENCE CARD

SET UP

KEY **FUNCTION COMMENTS**

Ctrl) and SET UP Load LA120 in set up mode. SET UP light flashes.

To load set up mode press **SET UP**.

SET UP Place LA120 in set up mode using **SET UP** in hold down. SET UP light flashes.

To load set up mode release **SET UP**.

NOTE

- LA120 must be in set up mode to set the following features.
- Do not use **SHIFT** unless specified.

FORMS

KEY **FUNCTION COMMENTS**

SHIFT Display current line number.

Releasing **SHIFT** returns display to current column number.

1 Set horizontal tab at current column.

SHIFT and **1** Set vertical tab at current line.

2 Clear horizontal tab at current column.

SHIFT and **2** Clear vertical tab at current line.

3 Clear other control tabs.

SHIFT and **3** Clear all vertical tabs.

4 or **SHIFT** and **4** Extend top of form (TOP).

5 Set minimum column number right margin.

SHIFT and **5** Set minimum line number right margin.

6 Set maximum column number right margin.

SHIFT and **6** Set maximum line number right margin.

7 Clear left and right margins.

SHIFT and **7** Clear top and bottom margins.

8 Form length.

NOTE Changing form length clears top and bottom margins and establishes TOP.

Display

0-9 Lines per form

100 } Lines per form

FORMS CONT.

KEY **FUNCTION COMMENTS**

0 Minimum column number right margin.

NOTE Changing column number sets minimum and maximum margins.

DISPLAY	PRICE
5	100.00
6	80.00
7	60.00
8	40.00
10	20.00
12	12.00
13	10.00
16	8.00

9 Vertical position of margin.

DISPLAY	PRICE
2	24.00
3	24.00
4	24.00
5	24.00
6	24.00
8	24.00
12	24.00

OPERATOR CONTROL

KEY **FUNCTION COMMENTS**

0 Set printing.

0 Low volume.

1 High volume.

0 Set feed.

0 0 = ON

1 0 = OFF

0 Auto format.

0 0 = ON

1 0 = OFF

COMMUNICATION

KEY **FUNCTION COMMENTS**

0 Auto printout.

0 0 = ON

1 0 = OFF

0 0 = 0-10

0 0 = 0-10

1 0 = 0-10

0 0 = 0-10000

1 0 = 0-10000

2 0 = 0-10

NOTE Changing form length clears top and bottom margins and establishes TOP.

Auto format

0 0 = ON

1 0 = OFF

0 0 = 0-10

0 0 = ON

1 0 = OFF

Figure OVR-5. LA120 Operator Reference Card (page 1 of 2)

OVR-9

COMPANY CONFIDENTIAL

DIGITAL

RSTS/E UPDATE
OVERVIEW OF RSTS/E V7.0

DIGITAL

RSTS/E UPDATE
OVERVIEW OF RSTS/E V7.0

COMMUNICATION ICONS

KEY FUNCTION/COMMENTS

[A] Auto wrap line at right margin
0 = Off
1 = On

[B] Auto line feed (return) feed
0 = Off
1 = On

[C] Mute
1 = FOR No Mute
2 = FOR Mute
3 = FOR Supervisory
4 = FOR EOF
5 = FOR ETE

[D] Forward and printer character set
1 = United States
2 = Great Britain

[E] Alternate character set
0 = Set A not enabled
1 = Set B not enabled

[F] Parity and data bits

DISPLAY	DATA BITS	PARITY	BAUD
1	7	IGNORE	SPACE
2	7	IGNORE	SPACE
3	8	IGNORE	ODD
4	7	IGNORE	EVEN
5	7	ODD	ODD
6	7	EVEN	EVEN
7	7	NONE	NONE
8	8	NONE	NONE
9	8	ODD	ODD
10	8	EVEN	EVEN

[G] HDL hand coding mode
0 = Enabled
1 = Receive

[H] Secondary channel

DISPLAY	MODS	REV CH
1	Receipt	Yes

* Set 14 Key Mute
0 = On or Off
1 = On or Off
2 = On or Off

[I] Printer "purge" character
1 = None
2 = Line Feed (LF)
3 = Return (CR)

[J] NON-ASCII
0 = Disabled
1 = Enabled

[K] Alternate keypad mode
0 = Disabled
1 = Enabled

COMMUNICATION ICONS

KEY FUNCTION/COMMENTS

[L] Baud rate
Selects received and transmit baud rates and parity of data bits

BAUD RATE DISPLAY	STOP BITS
50	1
75	2
110	2
134	1
150	1
200	1
300	1
600	1
1200	1
1800	1
2400	1
4800	1
7200	1
9600	1

[M] and **[N]** Selects stop baud rates
[M] selects receive baud rate, **[N]** and **[O]** transmit baud rates of those in receive baud rate

RECEIVE BAUD RATE	TRANSMIT BAUD RATE	TRANSMIT STOP BITS
600	75	2
	150	1
	300	1
1200	75	2
	150	1
	300	1
2400	300	1
	600	1
4800	300	1
	600	1
9600	300	1
	600	1

STORE RECALL AND STATUS

KEY FUNCTION/COMMENTS

[P] Range Select factory set up parameters

[Q] Print status message

[R] Recall set up parameters

[S] and **[T]** Store set up parameters

BELL TEST

KEY FUNCTION/COMMENTS

[U] Indicate printing test test

[V] and **[W]** Indicate non-printing test test
NOTE: Type any character in set up mode to stop test test

Copyright © 1979 by Digital Equipment Corporation 1st printing

PRINTED IN U.S.A. BY 9138 90 70 000 03 02

Figure OVR-5. LA120 Operator Reference Card (page 2 of 2)

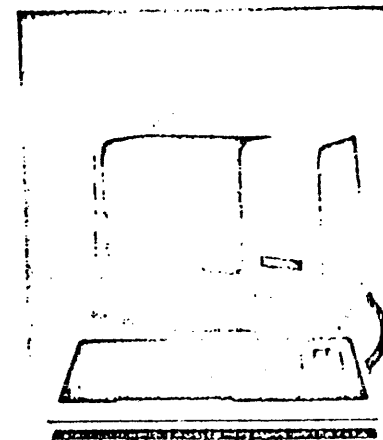


Figure OVR-6. VT100

- o VT52 replacement
- o Supported in VT52 mode
- o Up to 9600 baud
- o Screen
 - 132 columns x 14 lines
 - 80 columns x 24 lines
- o Set up mode allows user selectable terminal functions
- o Advanced video option includes:
 - 132 columns x 24 lines
 - Ability to combine attributes
 - Full terminal capability

ADDITIONAL VT100 DOCUMENTATION

Definition of Each VT100 Set-Up Feature

This section describes each SET-UP feature in detail (in alphabetical order) and states how each feature affects the terminal.

Note

Unless otherwise stated, entering SET-UP mode and changing features does not result in the loss of data displayed on the screen.

ANSI/VT52 Mode

The VT100 terminal is compatible with two different programming standards - American National Standards Institute (ANSI) and VT52. In ANSI mode, the VT100 will generate and respond to coded sequences per ANSI standard X3.41 - 1974. In VT52 mode, the VT100 terminal is compatible with previous DIGITAL software using the VT52 video terminal.

Answerback Message

The VT100 answerback feature provides the terminal with the capability to identify itself to the host. Answerback is a question and answer sequence where the host computer asks the terminal to identify itself. The terminal sends its unique identifier, or answerback message, to verify the identity of the terminal. The entire answerback sequence takes place automatically without affecting the screen or requiring operator action.

Auto Repeat

The auto repeat feature allows a key to be automatically repeated at the rate of about 30 characters per second when the key is held down for more than one-half

second. The auto repeat feature affects all keyboard keys except the following:

SET-UP
ESC
NO SCROLL
TAB
RETURN
ENTER
CTRL and any key

Auto XON/XOFF

The VT100 is capable of automatically generating synchronizing codes XON (DC1) and XOFF (DC3). The XOFF control sequence is used to stop the transmission of data from the computer to the terminal; the XON sequence is used to resume transmission. With the feature enabled, the VT100 will automatically generate the XOFF code when one of the following events occur:

1. The internal buffer is nearly full
2. The NO SCROLL key is pressed
3. The terminal is placed in SET-UP mode

When either the buffer empties, the NO SCROLL key is pressed again, or the terminal is taken out of SET-UP mode, the VT100 will automatically transmit the XON code to resume transmission from the computer to the terminal.

If the host computer software does not support the XON/XOFF codes, data sent during buffer full conditions, or when the terminal is in SET-UP mode, will be lost.

Note

The VT100 will always stop transmission when an XOFF (DC3) code is received and will resume transmission when an XON (DC1) code is received, regardless of the AUTO XON/XOFF feature setting.

Bits Per Character

This feature allows the terminal to transmit and receive 7 or 8-bit characters. In most cases, the terminal will be set for 7-bit operation (normal ASCII standard characters). In those cases where the communication protocol requires 8-bit characters, bit 8 is set to a space (or 0) for characters transmitted and is ignored on all characters received.

Characters Per Line

The VT100 is capable of displaying either 80 or 132 characters per line. In the 80 character per line mode, the screen is 80 characters wide by 24 lines high. In the 132 character per line mode, the screen is 132 characters wide by 14 lines high. In the 132 character per line mode, the displayed lines are physically the same width as in the 80 character per line mode, but the characters are more compact.

Note

When changing from an 80 to 132 character per line mode or vice-versa, the current contents of the screen are lost.

Cursor

The VT100 offers a choice of two cursors to indicate the "active positions" or where the next character will be placed on the screen. The cursor may be displayed as either a blinking underline () or a blinking block (■).

Interlace

Interlace changes the scanning rate of the screen to allow for alternating scans of odd and even rows in the character. This feature is used for high resolution memory options. When in the alphanumeric mode, the interlace feature should be turned off since the use of the smooth scroll feature with the interlace feature may cause the display to appear jittery and difficult to read when the screen is scrolling.

Keyclick Tone

The keyclick is a tone which is generated every time a key is pressed. The keyclick may be turned on or off to suit the operator's needs. However, research and experience has shown that an operator is more accurate when there is an audible feedback from the keyboard.

Like the bell tone, the keyclick volume is not adjustable.

LINE/LOCAL

The LINE/LOCAL feature allows the operator to easily place the terminal in either an ON-LINE or LOCAL (off-line) condition. When the terminal is on-line (the keyboard ON-LINE indicator is ON) all characters typed on the keyboard are sent directly to the computer and messages from the computer are displayed on the screen. In the LOCAL condition (the keyboard LOCAL indicator is ON), the terminal is electrically disconnected from the computer, messages are not sent to or received from the computer, and characters typed on the keyboard are echoed on the screen directly.

Margin Bell

The margin bell feature is much the same as the bell in a typewriter. When the cursor is eight characters from the end of the current line, the VT100 sounds a tone to alert the operator.

New Line

The new line feature enables the RETURN key on the terminal to function like the RETURN key on an electric typewriter. When the new line feature is enabled, pressing the RETURN key generates the carriage return (CR) and line feed (LF) codes. When a line feed code is received, the code is interpreted as a carriage return and line feed.

When the new line feature is disabled, the RETURN key generates only the CR code; an LF code causes the terminal to perform a line feed only.

Note

If double line feeds occur consistently, turn this feature off since the computer is already performing this function automatically.

Parity

When enabled, parity checks for correct data transmission. If a transmission error occurs, the VT100 can detect it and indicate its presence by placing a checkerboard character (☒) on the screen in place of the character with the error. The parity sense feature determines if the parity is even or odd.

Parity Sense

The parity sense feature defines which of the two methods of parity checking, odd or even, is being used by the VT100. If the parity feature is on, the terminal's parity sense must be matched to the parity the computer is sending. If the parity sense features do not match, every character sent to the computer will be rejected even though the character was received correctly by the VT100. If a parity incompatibility occurs, the checkerboard character will be shown on the screen in place of the received character.

Note

If the parity feature is turned off, the parity sense selection will be totally disregarded.

Power

During the initial installation, the terminal display must be synchronized to the power line frequency. In the U.S. this is set to 60 hertz.

Receive Speed

The receive speed determines how fast the characters sent by the computer can be accepted by the terminal. The VT100 is capable of receiving at any one of the following preselected speeds: 50, 75, 110, 134, 150, 200, 300, 600, 1200, 1800, 2000, 2400, 3600, 4800, and 9600 baud.

The receive speed is independent of the transmit speed; the terminal may receive data at one speed and transmit data at a totally different speed.

Screen Background

The screen background feature of the VT100 allows the operator to determine the background of the screen. In the normal screen mode, the display contains light characters on a dark background; in the reverse screen mode, the display contains dark characters on a light background.

Screen Brightness

Unlike most video terminals, the VT100 does not contain switches or knobs to adjust screen brightness. Instead, the VT100 electronically controls the screen brightness. This feature eliminates the high failure rate of mechanical knobs and allows the operator to select the desired level of brightness for maximum comfort under varied lighting conditions. This setting may be saved like any other feature in the terminal.

Scroll

Scrolling is the upward or downward movement of existing lines on the screen to make room for new lines at the bottom or top of the screen. It can be performed in two ways: jump scroll or smooth scroll. In jump scroll mode, new lines appear on the screen as fast as the computer sends them to the terminal. At the higher baud rates, the data is very difficult to read due to the rapid upward movement of the lines. In smooth scroll mode, a limit is placed on the speed at which new lines of data may be sent to the terminal. The upward movement of lines occurs at a smooth steady rate allowing the data to be read as it appears on the screen.

Note

Smooth scroll mode allows a maximum of six lines of data per second to be added to the screen. The Auto XON/XOFF feature must be enabled and supported by the host computer to ensure that data is not lost when smooth scroll mode is enabled.

Tabs

Just like a typewriter, the VT100 can jump or tab to preselected points on a line. These tab stops may be individually changed, or totally cleared and then reset.

Transmit Speed

Transmit speed determines how fast the characters will be sent from the terminal to the computer. The VT100 is capable of transmitting at any one of the following preselected transmit speeds: 50, 75, 110, 134, 150, 200, 300, 600, 1200, 1800, 2000, 2400, 3600, 4800, and 9600 baud.

Transmit speed is independent of receive speed; the terminal may transmit data at one speed and receive data at a different speed.

Wraparound

The wraparound feature removes the maximum limit on the number of characters that may be entered for one line. When this feature is enabled, the 81st or 133rd character (depending upon the line size selected) typed on a line is automatically placed in the first character position of the next line without a carriage return (CR) and a line feed (LF) code is generated. If the wraparound feature was not enabled, the 81st or 133rd character and all following characters would be overwritten into the last character position of the current line until a CR and an LF code are generated.

Note

Since wraparound is a VT100 terminal feature dealing with physical lines, the CR and LF codes must still be generated to end the logical line.

⌘
3

(shifted)

The VT100 contains character sets for the U.S. and the United Kingdom. The difference between the two character sets is one character, the ⌘ or ⌘ symbol. When the U.S. character set is selected, the uppercase 3 key on the main keyboard displays the ⌘ character. The ⌘ character is displayed when the U.K. character set is selected.

Magnetic Tapes

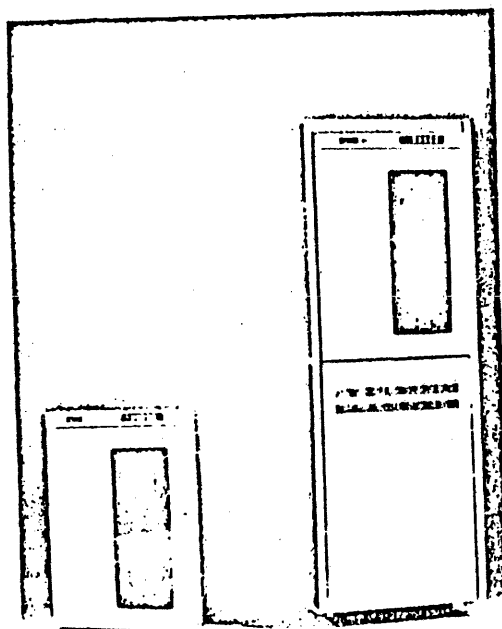


Figure OVR-7. TS04

- o 9-track magtape tape
- o 800 or 1500 BPI preset at factory
- o Up to 8 units on a system

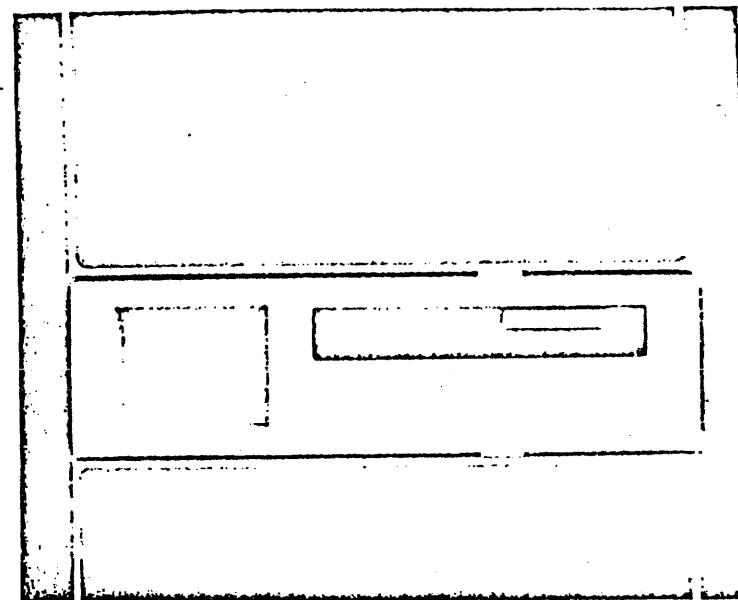


Figure OVR-8. TUS8 - DECTape II

- o Low cost tape cartridge
- o Random access to 512 blocks of 512 bytes each
- o 10 second average access time
- o Up to 2 drives per controller
- o Up to 4 controllers/8 units on a system
- o Uses a DL11A terminal interface

KMC11

- o A microprocessor used by the 3271 package
- o Up to 16 allowed on a system

OBSOLETE HARDWARE

- o RC11 - fixed head disk
- o LK11 - hardware which forced device interrupts

NEW OPERATING SYSTEM FEATURES

Large files

1. System generation option
2. Data files limited only by the size of a disk
3. Resource cost
 - o 2K words in monitor
 - o One additional small buffer for each file opened, no additional buffers for multiple opens of the same file.
4. Benefits include:
 - o Extension of non-contiguous files opened for update/write-shared access
 - o Access to non-file structured disk by block rather than device cluster number
 - o Some performance improvement

Data Caching

1. System generation option
2. Data, as well as directories, can be cached in memory
3. User selectable random or sequential data caching (privileged operation)
4. Resource cost
 - o Minimum 2K words of XBUF
 - o Heavier CPU usage
5. Benefits
 - o Overall I/O throughput improvement on disk bound systems

RSX Directives In Monitor

1. System generation option
2. User jobs do not have the RSX Run-Time System as part of the user virtual space.
3. Resource cost
 - o 1K words in the monitor
4. Benefits
 - o 31K word user space when using the RSX Run-Time System
 - o 24K word user space when using RMS resident library

Resident Libraries

1. System generation option
2. RMS routines memory resident and shareable by multiple users.
3. Recommended for systems with a minimum of 124K words and applications that use RMS-11K.
4. Resource cost for RMS Resident Library
 - o Up to 24K words of physical memory
 - o 8K words in user virtual space
 - o RSX directives in the monitor are required
5. Benefits for RMS Resident Library
 - o Reduced task build time
 - o Smaller task images on disk
 - o Faster execution speed - reduced overlaying
 - o Overall performance improvements on systems with RMS-11 applications

User Logical Assignments with PPN's

1. Logical assignments that associate a device and PPN with the logical assignment
2. Benefits
 - o Less programming restrictions
3. Limitations
 - o Only three logical assignments available if a PPN is included

Placed Files

1. First block of a file can be positioned at or near a specified physical disk location.
2. Benefits
 - o Optimal placement of files to minimize head movement on disks

Tentative Files

1. Files can be created without becoming permanent until closed.
2. Benefits
 - o Less program code
 - o Data security in case of a program abort or a system crash

Conditionally Contiguous Files

1. Files are created contiguous or non-contiguous, based on available disk space.

2. Benefits

- o Less error handling overhead

No File Supersede

1. Files opened for output return an error if a file with the same name already exists

2. Benefits

- o Less program code
- o Greater file protection

Zero a Disk File

1. Null characters are written over the specified file if it is explicitly or implicitly deleted.
2. Privileged operation that can be specified at open time

3. Benefits

- o Security for files containing sensitive information

4. Limitations

- o This operation on large files can be very time consuming.

Make a Contiguous File Non-Contiguous

1. Contiguous files can be made non-contiguous while a file is being accessed by multiple users.

2. Benefits

- o Ability to extend a contiguous file while application programs are executing

Library Segmentation

1. Utility programs and their associated files can be stored in accounts other than (1,2) .

2. Benefits

- o Faster access to utility programs
- o Reduced overhead on the system library account

ANSI Magnetic Tape Support

1. Multi-volume magnetic tape support through PIP.SAV only.

2. Benefits

- o Transportability of RMS-11 Sequential files between RSTS/E and VAX or RSX-11M
- o Allows selective backup of large files to tape

3. Limitations

- o RMS-11 Relative and Indexed files transportable only to other RSTS/E systems

BASIC-PLUS-2 Only Systems

1. BP2COM can be installed as the system default run-time system.

2. Benefits

- o Utility programs as task images
- o Less memory used for the system default run-time system

3. Limitations

- o No Immediate mode

COBOL Only Systems

1. RSX is the system default run-time system
2. Benefits
 - o No need for BASIC-PLUS or BASIC-PLUS-2
 - o No need for RMS-11 Run-Time System if using the RMS resident library
 - o Utility programs as task images

UTILITY PROGRAMS

All BASIC-PLUS utility programs can be compiled and executed under BASIC-PLUS-2 without translation.

New Utilities

1. SAVRES.SAV
 - o Fast volume save/restore
 - o Image copy of disks
 - o Executes stand alone or during time-sharing
 - o Handles bad blocks
 - o On-line for unmounted disks
 - o Off-line for system disk
2. ONLCLN.SAV
 - o On-line disk clean utility
 - o More thorough disk clean than the SYS call-clean
3. MAKSIL.TSK
 - o Converts task images into resident library format
 - o Converts task images or .SAV images into run-time system format
4. FIT
 - o Supersedes PIPEXT.BAS
 - o File transfer program for DECTape II, floppy disk, RT11 disks and DOS disks
5. DEVTST
 - o Supersedes UETP
 - o Device testing package
6. RMSIFL
 - o Performs fast loading of indexed files

DIGITAL

RSTS/E UPDATE
OVERVIEW OF RSTS/E V7.0

Discontinued Utilities

- o UETP - superseded by DEVTST
- o SYSCAT.BAS - superseded by DIRECT
- o PRIOR.BAS - superseded by UTILTY
- o PIPML.BAS - superseded by PIP.SAV
- o FLX.BAS - superseded by PIP.SAV
- o PIPEXT.BAS - superseded by PIP.SAV
- o ROLLIN - superseded by SAVRES

DIGITAL

RSTS/E UPDATE
OVERVIEW OF RSTS/E V7.0

BUNDLED SOFTWARE

- o RMS-11K V1.0
- o SORT-11 V2.0
- o KMC11 driver
- o II driver (IBM Interconnect)

UNBUNDLED SOFTWARE

- o BASIC-PLUS-2 V1.6
- o COBOL V4
- o DATATRIEVE V1.1
- o DIBOL/DEFORM V4C and V4D
- o RPG II V8.02
- o FORTRAN IV V2.1
- o FORTRAN IV PLUS V2.5
 - Category C
 - No RMS Support
- o KMC/3271 V1.0
- o DECNET/E V1.1
 - Virtual terminal support
- o WISE
- o DECAL

CATEGORY C SOFTWARE

- o APL-11
- o FORTRAN IV PLUS

DOCUMENTATION

1. Updated/reorganized manuals

- o BASIC-PLUS Language Manual
- o RSTS/E System User's Guide
- o RSTS/E Programming Manual
- o RSTS/E V7.0 System Manager's Guide
- o RSTS/E V7.0 System Generation Manual
- o RSTS/E Task Builder Manual
- o RSTS/E V7.0 Documentation Directory

2. RSTS/E Programmer's Utilities Manual

- o RSX based utilities manual
 - LBR
 - PAT
 - MAKSIL
 - MAC

3. RSTS/E System Directives Manual

- o A MACRO programmer's manual
 - General monitor directives
 - RSX emulator directives
 - Run-Time Systems

4. RMS-11 User's Guide

- o All RMS-11 documentation consolidated into one manual.

5. RMS-11 MACRO-11 Reference Manual

- o RMS-11 implementation in MACRO-11

DIGITAL

RSTS/E UPDATE
OVERVIEW OF RSTS/E V7.0

This page is for notes.

RSTS/E UPDATE

SYS

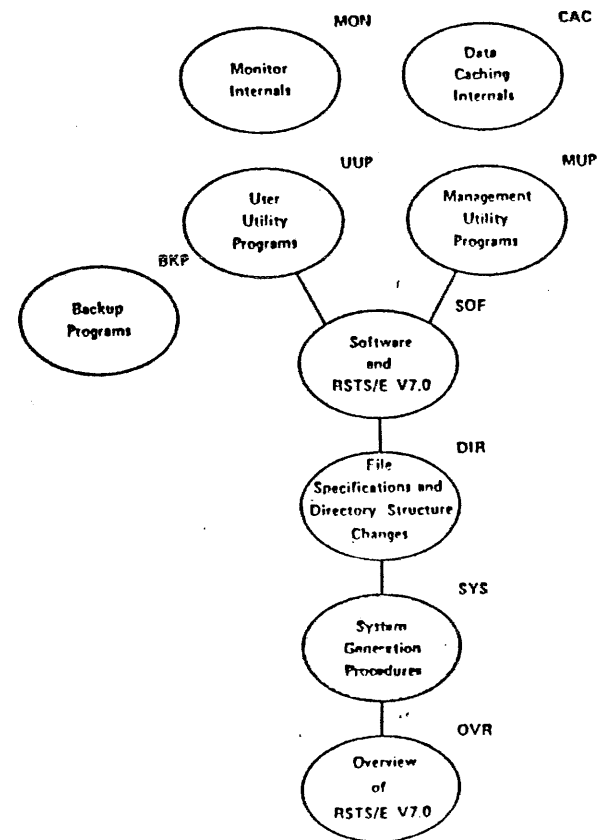
System Generation Procedures

System Generation Procedures

CONTENTS

Introduction.....	SYS-1
Distribution Media.....	SYS-2
General System Generation Procedures flowchart...	SYS-5
Enhanced INIT Options.....	SYS-9
DSKINT Option.....	SYS-9
HARDWR Option.....	SYS-10
REFRESH Option.....	SYS-12
DEFAULT Option.....	SYS-15
New and Enhanced SYSGEN.SAV Questions.....	SYS-16
Distribution Questions.....	SYS-16
RSX Questions.....	SYS-17
Terminal Service Questions.....	SYS-19
Disk Questions.....	SYS-20
Peripheral Device Questions.....	SYS-21
Optional Monitor Feature Questions.....	SYS-24
Programs Relating to Library Building.....	SYS-26
MINCOP Program.....	SYS-26
PATCPY Program.....	SYS-28
BUILD Program.....	SYS-32
ATPK Program.....	SYS-37
CSPCOM (CUSP.Compiler) Program.....	SYS-44
Autopatching Package.....	SYS-45
Example System Generation.....	SYS-46

COURSE MAP



System Generation Procedures**INTRODUCTION**

Before RSTS/E can be made available for general time sharing:

- o The monitor must be generated,
- o The default run-time system, system library, and optional software must be built, and
- o Command and control files must be created.

These procedures are normally done at system generation time.

With RSTS/E V7.0, the system generation procedures have been enhanced to allow support of:

- o New hardware
- o New INIT features
- o New monitor features
- o New library building features

This module covers the changes and enhancements to system generation for RSTS/E V7.0.

DIGITAL

RSTS/E UPDATE
System Generation Procedures

DISTRIBUTION MEDIA

Table SYS-1. Magtape Distribution Kits

DEC Order No.	BPI	Contents
7-Track	800	
AP-2772G-BC AP-2725G-BC AP-C726G-BC AP-C883B-BC		System Generation System Library I System Library II RSX Run-Time System and Library
AP-D047C-BC AP-H455A-BC		RMS Run-Time System and Library RSTS/E V7.0 Patch Kit "A"
9-Track	800	
AP-2773G-BC AP-2753G-BC AP-C725G-BC AP-C881B-BC		System Generation System Library I System Library II RSX Run-Time System and Library
AP-5226D-BC AP-H466A-BC		RMS Run-Time System and Library RSTS/E V7.0 Patch Kit "A"
9-Track	1600	(2400 ft. reel)
AB-H751G-BC		System Generation Libraries, RSX Run-Time System and Library
AB-H752G-BC AB-H753A-BC		RMS Run-Time System and Library RSTS/E V7.0 Patch Kit "A"

SYS-2

COMPANY CONFIDENTIAL

DIGITAL

RSTS/E UPDATE
System Generation Procedures

Table SYS-2. Disk Distribution Kits

DEC Order No.	Pack ID	Contents
RK05		
AN-2771G-BC AN-2751G-BC AN-5444G-BC AN-C885B-BC	SYSGNG SYSL1G SYSL2G RSXLBB	System Generation System Library I System Library II RSX Run-Time System and Library
AN-5227D-BC	RMSKIT	RMS Run-Time System and Library
AN-H467A-BC	PATCHA	RSTS/E V7.0 Patch Kit "A"
RK06		
AM-2774G-BC	SYSGNG	System Generation, RSX Run-Time System and Library
AM-5445G-BC AM-5228C-BC	SYSLIB RMSKIT	System Libraries RMS Run-Time System and Library
AM-H468A-BC	PATCHA	RSTS/E V7.0 Patch Kit "A"
RK07		
AY-D526G-BC	SYSGNG	System Generation, System Libraries, RSX Run-Time System and Library
AY-D946C-BC	RMSKIT	RMS Run-Time System and Library
AY-H469A-BC	PATCHA	RSTS/E V7.0 Patch Kit "A"

SYS-2

COMPANY CONFIDENTIAL

Table SYS-2. Disk Distribution Kits (Cont.)

DEC Order No.	Pack ID	Contents
RL01		
AX-D527G-BC	SYSGNG	System Generation, RSX Run-Time System and Library
AX-D528G-BC AX-D945C-BC	SYSLIB RMSKIT	System Libraries RMS-Run-Time System and Library
AX-H470A-BC	PATCHA	RSTS/E V7.0 Patch Kit "A"

GENERAL SYSTEM GENERATION PROCEDURES FLOW CHART

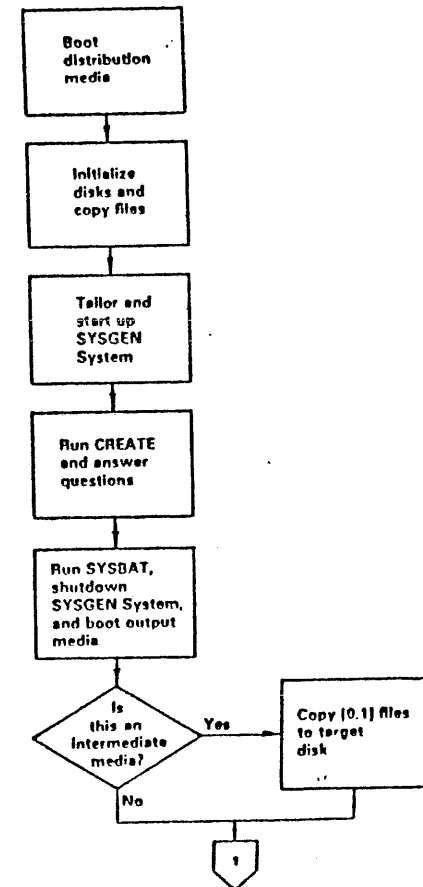


Figure SYS-1. Sygen Flowchart (Page 1 of 4)

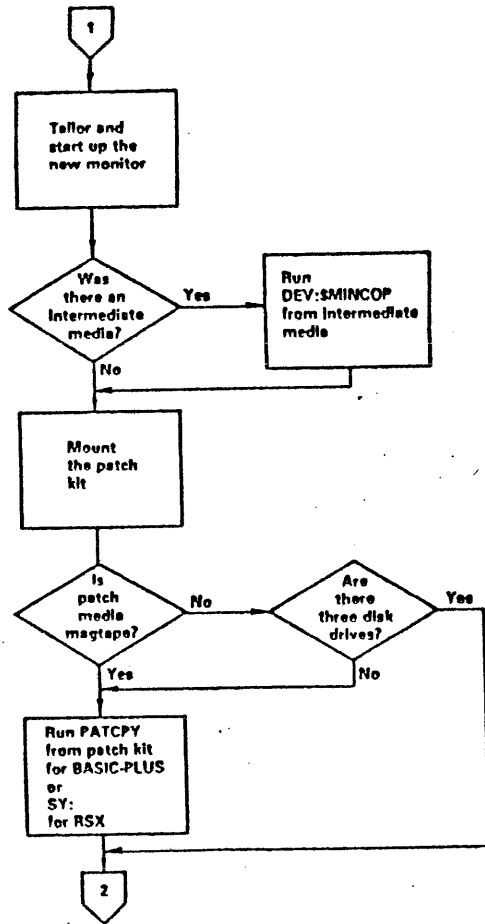


Figure SYS-1. Sysgen Flowchart (Page 2 of 4)

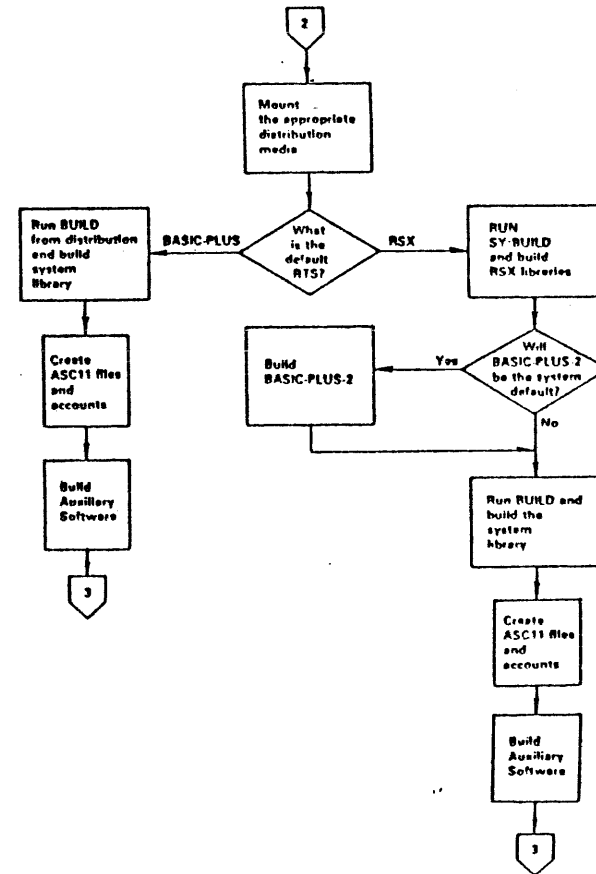


Figure SYS-1. Sysgen Flowchart (Page 2 of 4)

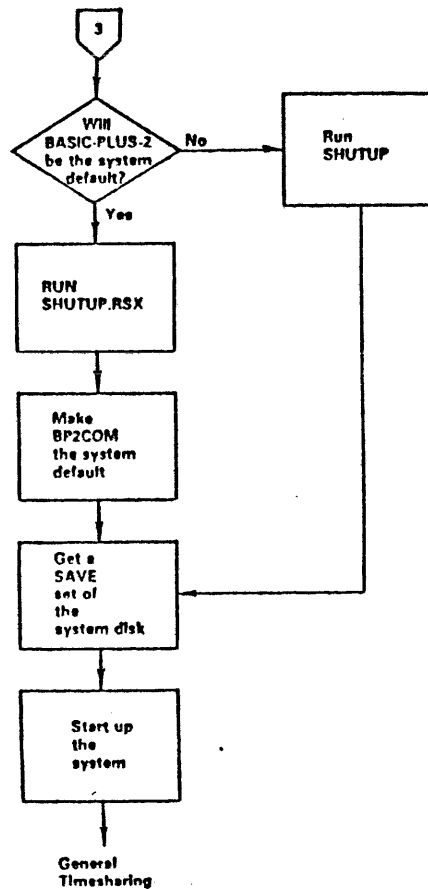


Figure SYS-1. Sysgen Flowchart (Page 4 of 4)

ENHANCED INIT OPTIONS

DSKINT Option

Option: DSKINT07-Feb-79?
02:07 PM?Disk? DM0:This disk pack appears to be a RSTS/E formatted
disk with the following characteristics:Pack ID : SYSPAK
Pack Cluster Size : 8
Pack is currently : Private,
Update access date on writes.

Pack ID? SYSPAK
 Pack cluster size? 8
 SATT.SYS base?
 Type a block number to locate the file SATT.SYS at a
 specific place on the disk, or type <LF> to let RSTS find a
 location.
 SATT.SYS base?
 MFD password? V70
 MFD cluster size? 16
 Pre-extend directories (no) ? Y
 PUB, PRI, or SYS? SYS
 Library password? LIBR
 Library UFD cluster size? 16
 Library account base?
 Type a block number to locate the library account at a
 specific place on the disk, or type <LF> to let RSTS find a
 location.
 Library account base?
 Date last modified (no)? Y
 New files first (no)?
 Use previous bad block info (no)? Y
 Format (no)? Y
 Patterns? 2
 Proceed (Y or N)? Y

DIGITAL

RSTS/E UPDATE
System Generation Procedures

HARDWR Option

Option: HARDWR

HARDWR suboption? LIST

Name	Address	Vector	Comments
TT:	177560	050	
RL:	174400	150	Units: 0(RL01) 2(RL02) 3(RL02)
RM:	177440	210	Units: 0(RK06) 1(RK06) 2(RK07)
RB:	175700	254	Units: 0(RP04 DP) 1(RP04 DP)
RR:	176300	150	Units:
TU:	172440	224	Units: 0(TU16 @TM03 #0) 1(TU16 @TM03 #0)
KW11L	177546	100	
KW11P	172540	104	
SR	177570		
DR	177570		

Hertz = 60.

Other: Cache

HARDWR suboption? DISABLE

Controller to disable? RR0

HARDWR suboption? UNITS

Disk to set units for? DB
 Unit? 0
 Type of this unit?
 Unit? 1
 Type of this unit?
 Unit?

HARDWR suboption? TU58

DL FOR DD0 (?)?
DL FOR DD2 (?)? 2

HARDWR suboption? LIST

3 changes pending.

DIGITAL

RSTS/E UPDATE
System Generation Procedures

Name	Address	Vector	Comments
TT:	177560	050	
RL:	174400	150	Units: 0(RL01) 2(RL02) 3(RL02)
RM:	177440	210	Units: 0(RK06) 1(RK06) 2(RK07)
RB:	176700	254	Units: 0(RP04 DP) 1(RP04 DP)
RR:	176300	150	Units:
{RR:	System		Disabled)
TU:	172440	224	Units: 0(TU16 @TM03 #0) 1(TU16 @TM03 #0)
KW11L	177546	100	
KW11P	172540	104	
SR	177570		
DR	177570		

Hertz = 60.

Other: Cache

HARDWR suboption? EXIT

3 changes being made.

Rebooting . . .

DIGITAL

RSTS/E UPDATE
System Generation Procedures

REFRESH Option

Option: REFRESH

07-Feb-79?
02:27 PM?

DISK? DM0

Clean? Y

Disk is being cleaned - wait ...

REFRESH suboption? LIST

File Name	Required?	File Flags	Status	Current Size	Minimum Size	Start LBN
System files:						
SWAP .SYS	YES		CRE		64	
SWAP0 .SYS	NO		OK			
SWAP1 .SYS	NO		OK			
SWAP3 .SYS	NO		OK			
OVR .SYS	NO		OK		54	
ERR .SYS	NO		OK		16	
BUFF .SYS	NO		OK			
CRASH .SYS	NO		OK		25	

Others:

BAD0 .SYS		NOD			40	
SATT .SYS		NOD CTG			1	13545
INIT .SYS		NOD CTG			417	
ERR .ERR			CTG		16	649
SYSGEN.SIL		NOD CTG			201	
RT11 .RTS			CTG		20	897

DIGITAL

RSTS/E UPDATE
System Generation Procedures

REFRESH suboption? CHANGE

SWAP.SYS changes? YES

Size? 20*31

Base? 13545

SWAP0.SYS changes?

SWAP1.SYS changes?

SWAP3.SYS changes?

OVR.SYS changes?

ERR.SYS changes?

BUFF.SYS changes?

CRASH.SYS changes? Y

Size? 80

Base?

DIGITAL

RSTS/E UPDATE
System Generation Procedures

REFRESH suboption? LIST

File Name	Required?	File Flags	Status	Current Size	Minimum Size	Start LBN
System files:						
SWAP .SYS	YES	NOD CTG	OK	2400	54	13665
SWAP0 .SYS	NO		OK			
SWAP1 .SYS	NO		OK			
SWAP3 .SYS	NO		OK			
OVR .SYS	NO		OK		54	
ERR .SYS	NO		OK		16	
BUFF .SYS	NO		OK			
CRASH .SYS	NO	NOD CTG	OK	80	25	921
Others:						
BADB .SYS		NOD		40		
SATT .SYS		NOD CTG		1		13545
INIT .SYS		NOD CTG		417		
ERR .ERR		CTG		16		649
SYSGEN.SIL		NOD CTG		201		
RT11 .RTS		CTG		20		897

REFRESH suboption?

DIGITAL

RSTS/E UPDATE
System Generation Procedures

DEFAULT Option

Option: DEFAULT

No defaults are currently set in SYSGEN.SIL

You currently have: JOB MAX = 2, SWAP MAX = 16K.

JOB MAX or SWAP MAX changes? Y

New JOB MAX? 2

New SWAP MAX? 28

You currently have: JOB MAX = 2, SWAP MAX = 28K.

JOB MAX or SWAP MAX changes?

Run Time System? RT11

Error message file? ERR

Installation name? SYSGEN

Memory allocation table:

```

0K: 00000000 - 00127777 ( 22K) : EXEC
22K: 00130000 - 00147777 ( 4K) : RTS (RT11)
26K: 00150000 - 00757777 ( 98K) : USER
124K: 00750000 - End : NXM

```

Table suboption?

You currently have crash dump disabled.

Crash dump? Y

Magtape labelling default (none)? DOS

Preferred clock (P 100)?

Date format (ALPHABETIC)?

Time format (AM/PM)?

Power fail delay (1)? 50

DIGITAL

RSTS/E UPDATE
System Generation Procedures

NEW AND ENHANCED SYSGEN.SAV QUESTIONS

Distribution Questions

The RSTS/E system is distributed on magtape, RK05 (DK), RL01 (DL), and RK06 and RK07 (DM) cartridge disks, or may already have been transferred to the system disk (SY). For magtape, a distinction must be made between the TU10, TE10, and TS03 drives, which use the device name "MT"; the TU16, TE16, and TU45 drives, which use the name "MM"; and the TS04, which uses the name "MS". Enter the type of distribution medium for this system generation (MT, MM, MS, DK, DL, DM, or SY).

Distribution medium? **???**

DIGITAL

RSTS/E UPDATE
System Generation Procedures

RSX Questions

You have the option of installing RSX or BASIC-PLUS-2 as the system default Run-Time System. Will you want RSX as the system default Run-Time System (Yes or No)?

RSX as default RTS? **NO**

The system generation process can automatically patch the generated RSX RTS. You must have an RSX patch file on some RSTS/E file structured medium. Would you like to automatically patch the generated RSX RTS (Y or N)?

RSX patching? **???**

The RSX RTS patch file can exist on the current system disk (SY); another disk (DF, DK, DL, DM, DP, DR or DB); TU10, TE10, or TS03 magtape (MT); TU16, TE16, or TU45 magtape (MM), TS04 magtape (MS), or DECTape (DT). Enter the name of the medium which has your RSX RTS patch file.

Patch file medium?

DIGITAL

RSTS/E UPDATE
System Generation Procedures

The disk on which your patch file exists has a logical name called the pack identification. This was assigned to the disk when it was initialized. Please specify that pack ID.

Pack ID?

Enter the file name of the RSX RTS patch file. You may enter an account number (for example, [10,23]) or \$ (for [1,2]) along with the file name.

Patch file name? #SRSXRTS.CMD #

DIGITAL

RSTS/E UPDATE
System Generation Procedures

Terminal Service Questions

The RSTS/E terminal service can be built to respond to CTRL/T by printing a one-line status report describing the job's status. This report includes the job name, RTS name, run state, job and RTS sizes, and total and incremental CPU times. The job is not disturbed when the report is printed. Would you like to have this feature in your monitor (YES or NO)?

One-line status report? Y

Disk Questions

The RL11 controller is used to control any combination of up to 4 of the RL01 2.6 million word or RL02 5.2 million word top loading cartridge disk drives. Enter the number of RL01 and RL02 drives on this system (0 to 4).

RL01/RL02's?

Overlapped Seek?

##

The RK611 controller is used to control any combination of up to eight RK06 (6 million word) or RK07 (12 million word) top loading cartridge disk drives. Enter the total number of RK06 and RK07 drives on this system (0 to 8).

RK06/RK07's?

Overlapped seek?

The RM02/3 disk system consists of an RH11 controller and up to eight RM02 or RM03 moving-head disk pack drives. Type the total number of RM02 or RM03 drives in this configuration (0 to 8).

RM02/RM03's?

Overlapped seek?

Peripheral Device Questions

The TS04 magtape system consists of a TS11 controller and a TS04 drive. RSTS/E supports up to 8 TS04 systems. If this system has no TS04 magtape drives answer 0; otherwise, how many TS04's exist on this system (0 to 8)?

TS04's?

The TU58 DECTape - II system consists of two drives and a controller, connected through a DL11 interface. RSTS/E supports up to four TU58 systems for a maximum of eight drives. Please enter the number of TU58 drives attached to this system (0 to 8).

TU58's?

The RX11 or RX211 floppy disk system consists of a UNIBUS interface and an RX01 (single density) or RX02 (double density) floppy disk subsystem. The RX01 and RX02 each include a disk controller and two floppy disk drives. RSTS/E supports up to four RX11 or RX211 systems for a maximum of eight drives. Please enter the number of floppy disk drives attached to this system (0 to 8).

RX01/RX02's?

DIGITAL

RSTS/E UPDATE
System Generation Procedures

The KMC11 is an intelligent microprocessor controller which, when loaded with appropriate micro-codes, can be used to control various types of peripherals. How many KMC11 units do you have on this system (0 to 16)?

KMC11's?

The KMC11 can emulate the operation of various IBM protocols. This support permits the RSTS/E system to communicate with any IBM system which supports the protocol. Do you want the KMC IBM protocol support (YES or NO)?

KMC IBM protocol support? NO

Each KMC11 can control one DUP11 synchronous line interface to the 3271 host. How many simultaneous KMC/3271 links do you want?

Simultaneous links?

DIGITAL

RSTS/E UPDATE
System Generation Procedures

The LP driver can be made to take advantage of the extended buffer pool rather than always buffering character output from the small buffer pool. This will particularly benefit those systems which have an increased contention for system small buffers. Do you want the extended buffering for LP (YES or NO)?

Extended buffering for LP? YES

Optional Monitor Feature Questions

An optional feature of RSTS/E is the ability to read and write files which are longer than 65535 blocks. Including large file support also reduces file creation and access times and makes disk update mode significantly faster. Including large file support will make your monitor 2K larger, and will use more small buffers for each open file. Do you want your system to handle large files (YES or NO)?

Large files? YES

An extended version of FIP buffering is available under RSTS/E which allows jobs to specify, on a file by file basis, that user data files are to be cached in the extended buffer pool. This caching is in addition to the FIP directory buffering. Extended FIP buffering can also cache clusters of data, and do read ahead of user data. It requires you to allocate at least 2K words to the extended buffer pool. Do you want to include extended FIP buffering in your system (YES or NO)?

Extended data buffering? NO

At the cost of a physically larger monitor, the disk drivers and their subroutines can be built into a separate monitor "phase". If you want to maximize the size of your small buffer pool, you should answer "YES". Build monitor with a separate disk driver phase (YES or NO)?

Disk driver phase? YES

An optional feature of RSTS/E allows programs to reduce I/O for overlaying by attaching to sharable memory-resident code. If you have at least 124K words of memory, and intend to use the RMS-11 resident library, answer 'YES'.

Resident libraries? YES

The RSTS/E monitor can be made to take over some of the function of the RSX Run-Time System. Programs which normally run under RSX Run-Time System don't need any run-time system, so they can grow to 31K or use the address space for a resident library. This option makes the monitor 1K larger. Do you want to include RSX directives in your monitor (YES or NO)?

RSX directives? YES

PROGRAMS RELATING TO LIBRARY BUILDING

MINCOP Program

When an intermediate media is created during system generation, the MINCOP program must be run to copy the utility programs needed for building the system library.

RUN DEV:\$MINCOP

DEV: is the device specification of the intermediate disk or magtape. Do not specify an extension for the program in the RUN command.

<p>Note</p> <p>If the intermediate media is a disk, the MINCOP program will logically dismount the disk when the copying is completed.</p>
--

Programs Copied into SY:[1,2] by MINCOP

Table SYS-3.

Program	Prot	RTS
PIP.SAV	<232>	RT11
PIPSAV.TXT	<60>	RT11
ONLPAT.SAV	<104>	RT11
MACRO.SAV	<104>	RT11
CREF.SAV	<104>	RT11
LINK.SAV	<104>	RT11
LIBR.SAV	<104>	RT11
PATCH.SAV	<104>	RT11
PAT.SAV	<104>	RT11
UTILTY.SAV	<104>	RT11
SYSLIB.OBJ	<40>	RT11
COMMON.MAC	<40>	RT11
PATCPY.TSK*	<104>	RSX
LOGIN.TSK*	<104>	RSX
BUILD.TSK*	<104>	RSX
ATPK.TSK*	<104>	RSX
CPATCH.TSK*	<104>	RSX
AUTOED.TSK*	<104>	RSX

*Only copied if the system default run-time system is RSX.

PATCPY Program

The PATCPY program, executed prior to building the system library, transfers command and patch files from a patch kit to a disk. These files are used during the build procedure to automatically patch utility programs.

RUN DEV:\$PATCPY

The device from which PATCPY is run depends upon the system default run-time system. For BASIC-PLUS, PATCPY is run from the patch kit. For RSX, PATCPY is run from SY:

Example PATCPY Dialogue

RUN MM0:\$PATCPY

PATCPY X7.0-05 RSTS V7.0-05 TIMESHARING

Enter distribution device/PPN<MM0:[1,2]>:

Enter output device/PPN: SY:[200,200]

!Can't find file or account - SY:[200,200]

Attempt to create account SY:[200,200] <No>? Y

Account SY:[200,200] created with your password

Packages to patch? BUILD

Copying MM0: [1,2]BUILD.CMD to SY: [200,200]BUILD.CMD

Copying MM0: [1,2]PA1002.CMD to SY: [200,200]PA1002.CMD

Copying MM0: [1,2]PA1002.001 to SY: [200,200]PA1002.001

.

.

5 files copied

Copy operation complete

Table SYS-4. PATCPY

Response to Packages to patch?	Files copied
ALL	*.*
EXEC	EXEC.CMD MONITR.CMD PA03??.* INIT.CMD PA01??.* RJ2780.CMD PA05??.* DECNET.* DECNT?.* PA45??.* BASIC.CMD PA04??.*
INIT	INIT.CMD PA01??.*
SYSGEN	SYSGEN.CMD PA02??.*
MONITR	MONITR.CMD PA03??.*
BASIC	BASIC.CMD PA04??.*
RJ2780	?2780.CMD PA05??.* PA25??.*
DECNET	DECNET.* DECNT?.* NET9LD.* PA45??.* PA46??.*

Table SYS-4. PATCPY (Cont.)

Response to Packages to patch?	Files Copied
BUILD	BUILD.CMD PA10??.* PA11??.* PA12??.* PA17??.* PA2218.*
SPLER	SPLER.CMD PA14??.*
BIGPRG	BIGPRG.CMD PA10??.* PA11??.* PA2219.*
BACKUP	BACKUP.CMD PA13??.*
DEVTST	DEVTST.CMD PA18??.*
RT11	RT11.CMD PA22??.*
RSX	RSX.CMD PA21??.*
RMS	RMS.* PA29??.*
SORT	SORT.CMD PA30??.*

Table SYS-4. PATCPY (Cont.)

Response to Packages to patch	Files Copied
STANDARD	INIT.CMD SYSGEN.CMD MONITR.CMD BASIC.CMD RJ2780.* DECNET.* NETBLD.* BUILD.CMD SPLER.CMD BIGPRG.CMD BACKUP.CMD DEVTST.CMD UNSUPP.CMD RSX.CMD RT11.CMD RMS.* SORT.CMD PA0???.* PA1???.* PA21??.* PA22??.* PA31??.*
Other valid responses	BASIC2 FORTRAN DTR DMS500 DIBOLV4C DIBOLV4D RPG APL WISE DECAL DECNETV1.0 BASIC2V1.5

BUILD Program

The BUILD program has been re-written for RSTS/E V7.0. Enhancements include:

- o Autopatching as part of the BUILD procedure,
- o Packages (BACKUP, SPLER, DEVTST, and Error Logging) in accounts other than (1,2),
- o CUSPs build under BASIC-PLUS, RSX or BASIC-PLUS-2 with the same set of control files,
- o Specifying 'LB:' on other than SY:[1,1],
- o Dynamically creating START.CTL and CRASH.CTL files, and
- o Automatically 'ADDs' RT11.

RUN DEV:\$BUILD

The device from which BUILD is run depends upon the system default run-time system. For BASIC-PLUS execute BUILD from the distribution kit. For RSX, execute BUILD from SY:.

Example BUILD Dialogue**RUN-MM0: BUILD**

```
BUILD X7.0-05 RSTS V7.0-05 TIMESHARING
System Build <No>? YES
Source Input Device <MM0:/DEN:800/PAR:ODD>?
Library Output Device <SY:>?
Target System Device <SY0:>?
Library Account <[1,2]>?
```

*** Copying file MM0:[1,2]BUILD.CTL to BLD1.TMP ***

```
Locate logical 'LB:' on <SY:[1,1]>?
Function (Build/Patch, Patch, Build) <Build/Patch>?
Save patched sources <No>?
Run-Time System <BASIC>?
Locate ERROR Package on <SY:[1,2]>? SY:[1,10]
```

```
%Can't find file or account - SY:[1,10]
Attempt to create account SY:[1,10] <No>? Y
Account SY:[1,10] created with your password
Additional Control File is <None>?
```

.

.

Command Format for 'Source Input Device?' Prompt

[logfile=]dev:[/switches]

where

logfile

is the file specification of the file in which to log the build procedure.

dev:

is the actual source input device.

/switches

can be any one or more of the following:

/DEN[SITY]:800 OR /DEN[SITY]:1600 only valid for magtape
 /PAR[ITY]:ODD OR /PAR[ITY]:EVEN only valid for magtape
 /DOS only valid for magtape
 /ANSI only valid for magtape
 /DET[ACH]

The DETACH switch causes BUILD (actually ATPK) to detach. If a logfile is also specified, this can vastly speed up some BUILD/PATCH procedures, because it eliminates time spent waiting for the KB: output.

Note

DIGITAL recommends that the /DETACH switch is not used until after the standard CUSPs are built (BUILD.CTL).

Response to Run-Time System? question must be:

- o A keyboard monitor which is present in [0,1] or is the current system default.
- o A BASIC-PLUS, RSX or BASIC-PLUS-2 run-time system with an executable extension of .BAC or .TSK.

BUILD Control Files Supplied with Distribution

BUILD.CTL	Used in a system build. Utilitites needed on all systems.
BIGPRG.CTL	Large Utilities
SPLER.CTL	Operator Service and Spooling Utilities
BACKUP.CTL	BACKUP package
DEVTST.CTL	Device exercising package
RMSBLD.CTL	RMS-11 package
RSX.CTL	RSX package
TECO.CTL	TECO package

DIGITAL

RSTS/E UPDATE
System Generation Procedures

RSX.CTL Contents

RSX.RTS
SYSLIB.OLB
RSXMAC.SHL
CSPC04.OLB
TKB.TSK
MAC.TSK
LBR.TSK
PAT.TSK
EDT.TSK
CSPCOM.TSK
SHUTUP.RSX
EDTCOM.TXT
SLOTKB.TSK

DIGITAL

RSTS/E UPDATE
System Generation Procedures

ATPK Program

ATPK (pronounced "at-pee-kay") is included in RSTS/E V7.0 for use as a part of the BUILD procedure. It can also be used 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 "0" 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. 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, logins will be raised, if necessary, to start the controlled job (used by BUILD).

/CHA[IN]: <program> [;<line>] [=

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 only be printed 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:

SLOG OFF Turn dialogue reporting to the log file (and/or keyboard) off.

SLOG ON Resume printing the controlled job's dialogue in the log file (and/or at the keyboard).

SALLOW NO ERRORS Abort if any (fatal or warning) errors are encountered by the controlled job.

SALLOW WARNING ERRORS Abort if a fatal error is encountered by the controlled job, but allow warning errors.

SALLOW 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 (that is, 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 number n under 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 (for example, \$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 on 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 System Startup

If it is desired, most of system startup may be controlled under ATPK. For example, the file START.CTL to be processed by the INIT utility could contain only the following:

DIGITAL

RSTS/E UPDATE
System Generation Procedures

```
DETACH
LOGIN KB0: [1,2]
FORCE KB0: RUN $ATPK
FORCE KB0: $START.LOG=$START.CMD/DET/LOG
END
```

When designing the ATPK startup command file, most of the functions of INIT will have to be performed with UTILITY. The recommended procedure for bringing a non-system disk on-line after a crash is to clean the disk with ONLCLN.SAV and then MOUNT and UNLOCK the disk with UTILITY.

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 two commands, causing them to be sent to the pseudo-keyboard at the same time, but as two lines. As an example, the following sequence in the ATPK command file (START.CMD in the above example) will start up OPSER, QUEMAN, one line printer spooler and one batch processor.

```
RUN $OPSER ^MLOG OPSER.LOG;ALL ^MCHA KB0: ^MDETACH
$LOGIN
RUN $QUEMAN ^MDETACH
$LOGIN
RUN $SPOOL ^MLP0:
$LOGIN
RUN $BATCH ^MBA0:/NOQUEUE/NODELETE/ERROR:PATAL
$LOGIN
```

DIGITAL

RSTS/E UPDATE
System Generation Procedures

When using ATPK to start up programs such as VT50PY, care must be taken to avoid sequencing problems. If VT50PY is invoked and directed to detach and display information on another terminal, immediate attempts to force commands to that terminal that would select the desired VT50PY features to be displayed may be forced to the terminal before VT50PY has a chance to open the terminal. In this case, the forced characters may invoke LOGIN at that terminal resulting in a failure by VT50PY to gain access to the terminal. This problem may be avoided by having ATPK execute some other command(s) (such as an immediate mode SLEEP statement, if running under the BASIC-PLUS Run-Time System) after starting the display program and before forcing to the display program's terminal.

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. The echo of everything you type is typed back at you again after you end each line, and nothing you type will be sent to the PK until the next time your controlled job enters a KB input wait.

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 "0" to run the ATPK program. Either or both of the commands may be installed.

```
RUN $UTILITY
!CCL AT-PK=[1,2]ATPK.*;PRIV 30000
!CCL 0=[1,2]ATPK.*;PRIV 30000
!Z
```

(This assumes that ATPK and UTILITY are both located on SY:[1,2].)

CSPCOM (CUSP. Compiler) Program

The CUSP COMPILER

A new feature of RSTS/E V7.0, CSPCOM (the CUSP Compiler) generates object modules from the BASIC-PLUS utility programs that can be linked against the RSX Run-Time System, eliminating 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 will be included in the RSTS/E kit as a maintenance tool only. The use of 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.

Autopatching Package

AUTOPATCH

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

- o Autopatching of the CUSPs when the library is built (via BUILD), rather than the old sequence of OLD-COMPILE-PATCH-COMPILE,
- o Checksum verification of source-level patches, and
- o Comments in all phases, except within the text of a patch.

EXAMPLE SYSTEM GENERATION

RUN SYSGEN
0000-Mar-7900

Parameters of RSTS/E system generation.

Questions come in long and short forms.
If you are familiar with these, answer
"S" for short; otherwise, answer "L" for
long form.

Form ?	09 0
Save system ?	0Y 0
Distribution medium ?	0**0 0Y
Output medium ?	0Y0
Delete files ?	0N00
LP for SYSGEN ?	0Y 0 NO
Generate monitor ?	0Y 0 Y/L
Monitor name ?	0RST00 BUS
Monitor pathing ?	0**0 NO
Generate BASIC-PLUS ?	0Y 0
BASIC-PLUS RTS name ?	0BASIC0 BUSBAS
BASIC-PLUS pathing ?	0**0 NO
RSE as default RTS ?	0N00

Now you must specify the hardware con-
figuration on which this RSTS/E system
will run.

KL11/LC11/BL11A/BL11D's ?	0030
DL11C/BL11D's ?	0000
DC11's ?	0000
DL11E's ?	0010
DJ11's ?	0000
DM11's ?	0010
DM11 unit 00 lines enabled ?	0100
Dataset support for DM11's ?	0Y 0
DZ11's ?	0000
Pseudo keyboards ?	0000

Figure SYS-2. System Generation (Page 1 of 7)

2741 support ?	0N00
Multi-terminal service ?	0Y 0
Echo control ?	0Y 0
One-line status report ?	0Y 0
RF/RB11's ?	0N00
RS03/RS04's ?	0010
RK05's ?	0000 2
Overlapped seek ?	0Y 0 M
RL01's ?	0000
RK06/RK07's ?	0000
RP02/RP03's ?	0000
RN02/RN03's ?	0000
RP04/RP05/RP06's ?	0020
Overlapped seek ?	0Y 0
TU1A/TU16/TU45's ?	0000 2
TU10/TU10/TU02's ?	0000
TB04's ?	0000
DEClapes ?	0000
TU5B's ?	0000
Printers ?	0010
RK01/RK02's ?	0000
CR11/CH11 card reader ?	0N00
CB11 card reader ?	0N00
P.T. reader ?	0N00
P.T. punch ?	0N00
DHC11's ?	0010
DECnet network support ?	0Y 0 NO
KHC11's ?	0000
Extended buffering for LP ?	0Y 0
2700 support ?	0N00
Large files ?	0Y 0
Mainframe jobs ?	0100

Figure SYS-2. System Generation (Page 2 of 7)

```

Small buffers ?           01700
System wide lockouts ?   0100
Monitor statistics ?     0N00 YES
PIP buffering ?          0Y 0
Extended data buffering ? 0N00 YES
Disk driver phase ?     0Y 0
Resident libraries ?    0Y 0
ASX directives ?        0Y 0
Resident disk handling ? 0Y 0
Resident send/receive ? 0N00
Resident same SYS calls ? 0N00
Resident file delete/rename ? 0N00
Res. login/attach/attribute ? 0N00
Resident catalog/lookup ? 0N00

```

The following questions deal with the BASIC-PLUS run-time system

```

FPP ?                     0Y 0
Math precision ?         0020
Log functions ?          0Y 0
Trig functions ?        0Y 0
Print using ?           0Y 0
Matrices ?              0N00 YES
Strings arithmetic ?    0N00

```

The system generation dialog is finished. If you have any special requirements which require editing the generated file (CONFIG.MAC(system configuration file) or SYSTEM.CTL(batch control file) you save do it now. When ready type "R SYSBAT".

```

Ready
RUN SYSBAT

```

SYSTEM batch processing has started. If any problems develop during the batch process it may be aborted by typing "Control/C". To restart type "R SYSBAT".

Figure SYS-2. System Generation (Page 3 of 7)

```

^C
HELLO 1/2
PASCOPY01
Jobs 3 4 5 & 7 8 are detached under this account
Job number to attach to?
7 other users are logged in under this account

```

```

Ready

```

```

RUN UTILITY.SAV
$SWITCH RT11
^C

```

```

.SIZE 24

```

```

.ASSIGN BY1 IN

```

```

.R PIP.SAV
@TBL.OPJ,TTDINT.OPJ,TTDUR.OPJ/DEINQVARN
@TBL.LBI,TTDINT.LST,TTDUR.LBI/DEINQVARN
@RSTS.SAV,TER.SAV,DSK.SAV/DEINQVARN
@ENT.SAV,FIP.SAV,RSX.SAV,OUK.GAV/DEINQVARN
@RSTS.MAP,TER.MAP,DSK.MAP/DEINQVARN
@ERT.MAP,FIP.MAP,RSX.MAP,OUK.MAP/DEINQVARN
@RSTS.STB,TER.STB,DSK.STB/DEINQVARN
@ENT.STB,FIP.STB,RSX.STB,OUK.STB/DEINQVARN
@NSP.SAV,XMDUR.SAV,RJ2780.SAV/DEINQVARN
@NSP.MAP,XMDUR.MAP,RJ2780.MAP/DEINQVARN
@NSP.STB,XMDUR.STB,RJ2780.STB/DEINQVARN
^C

```

```

.R MACRO.SAV
@TTDUR,TTDUR/C=INICOMMON,KERNEL,DKICONFIG,INICHECK,KBDEF,TTDINT
ERRORS DETECTED! 0
^C

```

```

.R MACRO.SAV
@TTDINT,TTDINT/C=INICOMMON,KERNEL,DKICONFIG,INICHECK,KBDEF,TTDINT
ERRORS DETECTED! 0
^C

```

```

.R MACRO.SAV
@TBL,TBL/C=INICOMMON,KERNEL,DKICONFIG,INICHECK,TBL
ERRORS DETECTED! 0
^C

```

```

.R LINK.SAV
@RSTS/Z,RSTS/A/U-RSTS=TBL,@ERR.BTR/X/DIO/EI012500/U01000/I/C
@TTDINT/C
@INIRSTB
Extend section? MORBUF
Round section? MORBUF
Library search? SYPLRO
Library search? BDFEXT
Library search? LPDURX
Library search? LIDMIS
Library search?
^C

```

Figure SYS-2. System Generation (Page 4 of 7)


```

.JR LINK.SAV
DSSA/Z/DSS/A/B,DSS=INIDBR.DKIRSTS.BTB/X/B10117000/U101000/I/C
@INIRSTB
Round section? DSKPAT
Library search? DBSEEK
Library search?
@C

.JR LINK.SAV
DTER/Z/TER/A/M,TER=INITER.DKIRSTS.BTB/X/B10117000/U101000/I/C
@ITERM/C
@INIRSTB
Round section? TERPAT
Library search? TTSYST
Library search?
@C

.JR LINK.SAV
@XNDUR/Z/XNDUR/A/M,XNDUR=INIXNP.DKIRSTS.BTB/X/B10117000/U101000/I/C
@INIRSTB
Round section? XNDPAT
Library search? XNDUR
Library search?
@C

.JR LINK.SAV
@ENT/Z/ENT/A/M,ENT=INIENT.DKIRSTS.BTB/X/B10117000/U101000/I/C
@INIRSTB
Round section? ENTPAT
Library search? LIB
Library search?
@C

.JR LINK.SAV
@FIP/Z/FIP/A/M,FIP=INIFPLRG.DKIRSTS.BTB/X/B10117000/U101000/I/C
@INIRSTB
Round section? FIPPAT
Library search? DPL
Library search?
@C

.JR LINK.SAV
@RSX/Z/RSX/A/M,RSX=INIRBX.DKIENT.BTB/X/B10117000/U101000/I/C
@INIRSTB
Round section? RXPAT
@C

.JR LINK.SAV
@DVR/Z/DVR/A/M,DVR=INIDVRLRG.DKIFIP.BTB/X/B101000/C
@INIRSTB
@C

.JR SILUS.SAV
@SYO:(@,SUSUS.SIL-TTI-RSTS,TER/M,DSK/M,ENT/M,FIP/M/C
@RSX/M/C
@XNDUR/M/C
@DVR/M/C
@IN:DBI-DEFAULT
Directory of SIL SUS .SIL on 08-Mar-79 at 14147

Name Ident Load Size Transfer Total
RSTB 07.006 000000 072000 000001 15k
TER 07.005 120000 017000 000001 17k
BSK 07.005 120000 005000 000001 20k

```

Figure SYS-2. System Generation (Page 5 of 7)

```

@* 07.005 120000 010000 010001 20k
@* 07.005 120000 011000 000001 16k
@* 07.005 120000 007000 000001 30k
@* 07.005 120000 001000 000001 30k
@* 07.005 002000 004000 000001 43k
@* 07.005 140000 012500 142500 48k
DEFAULT 07.005 001000 002000 000001 48k

Resident monitor size is 30k (with 100 small buffers)
535 small buffers can be added to this system (716 total)
16 small buffers can be added without a size change (194 total)

```

```

@C
.JR PIP.SAV
@BASIC.SAV,BASIC.BTB/DEINOVARN
@C

.JR LINK.SAV
@BASIC/Z/SUSBAS/A/M,BASIC=INIRTS.DKIGERR.BTB/X/H10177776/U104000/C
@INIRTS2F/C
@INIRTS2F/C
@INIRTS2F/C
@INIRTS2F/C
@INIRTS2F/C
@INIRTS2F/C
@INIRTS2F/C
@INIRTS2F/C
Round section? PA
@C

.JR SILUS.SAV
@SUSBAS.RTS,TTI-BASIC
Directory of SIL SUSBAS.RTS on 08-Mar-79 at 14147

Name Ident Load Size Transfer Total
BASIC 07.005 104000 073776 000001 15k

```

```

@C
.DEASSIGN IN
.JR PIP.SAV
@SYO:(@,SUSUS.SIL-TTI-RSTS,TER/M,DSK/M,ENT/M,FIP/M/C
@RSX/M/C
@XNDUR/M/C
@DVR/M/C
@IN:DBI-DEFAULT
Directory of SIL SUS .SIL on 08-Mar-79 at 14147

```

```

.JR LDDOUT
Lddout Y
Saved all disk files! 12232 blocks in use
Job 11 User 1:2 logged off RBA at 08-Mar-79 14147
7 other users still logged in under this account
System RSTS V7.0-08 Moss FT (17Feb)
Run time was 5 minutes 37.6 seconds
Elapsed time was 10 minutes
Good afternoon

```

Figure SYS-2. System Generation (Page 6 of 7)

DIGITAL

RSTS/E UPDATE
System Generation Procedures

Batch Job completed.
Ready

Figure SYS-2. System Generation (Page 7 of 7)

RSTS/E V7.0 SWS UPDATE

DIR

File Specification and Directory Structure Changes

CONFIDENTIAL

REVISED UPDATE
File Specification and
Directory Structure Changes

File Specification and Directory Structure Changes

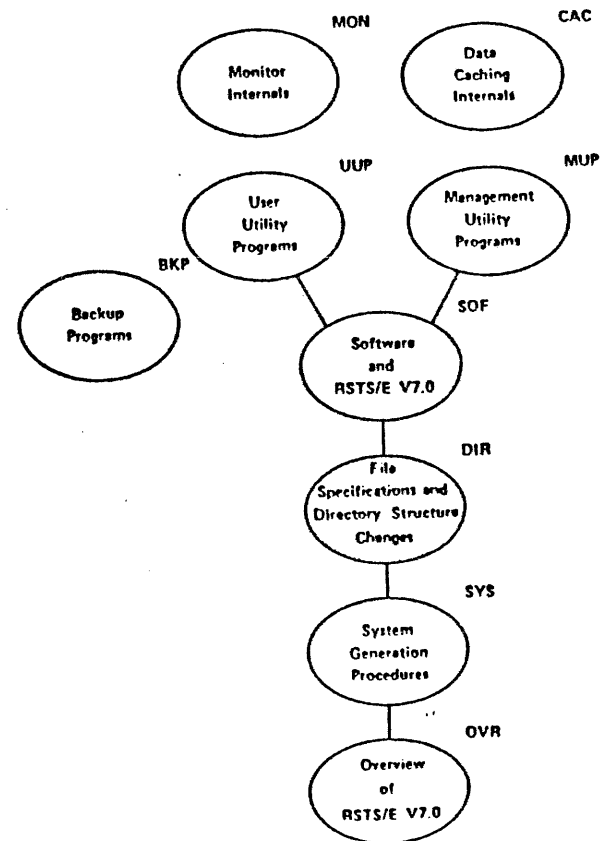
CONTENTS

Introduction.....	DIR-1
File Specification Changes.....	DIR-3
Device Designators.....	DIR-3
Protection Codes.....	DIR-4
File Specification Switches.....	DIR-4
Directory Structure Changes.....	DIR-6
MFD Changes.....	DIR-6
UFD Changes.....	DIR-8

CONFIDENTIAL

COMPANY CONFIDENTIAL

COURSE MAP



File Specification and Directory Structure Changes**INTRODUCTION**

The RSTS/E operating system supports a variety of high level languages: BASIC-PLUS, BASIC-PLUS-2, COBOL, DIBOL, RPGII, etc. Regardless of the language in which you code, the file specification is the way of telling the RSTS/E monitor exactly what actions you want taken when a file or device is opened. It is through the file specification that you can implement many of the new features of RSTS/E V7.0.

Furthermore, some of these new enhancements have required changes to or modifications of the current disk directory structure.

In this module, you will learn not only how to specify these new features using the RSTS/E file specification, but also the changes to the disk directory structure needed by RSTS/E to maintain your files.

DIGITAL

RSTS/E UPDATE
File Specification and
Directory Structure Changes

This page is for notes.

DIGITAL

RSTS/E UPDATE
File Specification and
Directory Structure Changes

File Specification and Directory Structure Changes

FILE SPECIFICATION CHANGES

Device Designators

Table DIR-1. Device Designators

Device Name	Device Designation	Valid Unit Numbers
RL02	DL	0-3
RX02	DX	0-7
TS04	MS	0-7
TU58	DD	0-7
KMC11	XK	0-15

The underscore character can precede the device designator. For example:

"_DL2:file.ext"

- Suppress logical device translation during file name string scan.

Protection Codes

Protection code - 128

- Settable by privileged users only.
- Write null characters over a compiled file or data file when implicitly or explicitly deleted by:
 - KILL a file
 - OPEN FOR OUTPUT
 - ZEROing an account

Note

On a large file this can be a time-consuming process.

File Specification Switches

1. /FILESIZE:n or /SIZE:n

- The only way to pre-extend RSTS/E V7.0 large files without explicitly writing actual or dummy records. For example:

"FILE.EXT/FILESIZE:200000"

Allocate 200,000 blocks for the file

2. /POSITION:n

- Place the first block of the file at or near device cluster n on the disk. For example:

"FILE.EXT/POSITION:25000"

Place the beginning of the file at or near device cluster 25000

Mode Value	Disk	Paper Tape Reader	Paper Tape Punch
2		Even parity checking	Punch even parity
2 + 1		Odd parity checking	Punch odd parity
32	Create a tentative file		
64	Create conditionally contiguous file		
128	File structured- No supersede		
	Non-file structured- NFS Block mode		
256	Random data cache the file		
256 + 2048	Sequential data cache the file.		
16384	Write access to a UFD #: <ol style="list-style-type: none"> 1. No one is logged into the account 2. The UFD is not open 3. No files in the UFD are opened 4. The account is not (0,1),(1,1) or (1,2) 		

Figure DIR-1. /MODE:n

DIGITAL

RSTS/E UPDATE
File Specification and
Directory Structure Changes

DIRECTORY STRUCTURE CHANGES

MFD Changes

1. MFD Label Entry (V6C change)

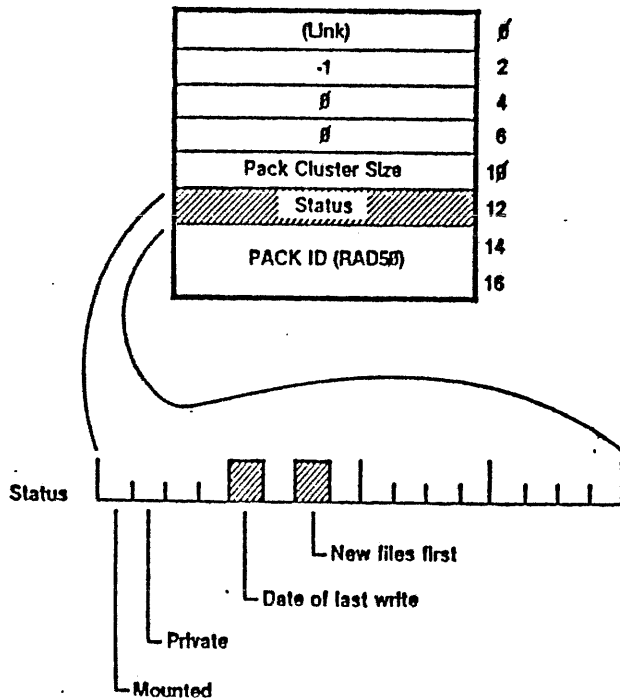


Figure DIR-2. MFD Label Entry (V6C change)

DIGITAL

RSTS/E UPDATE
File Specification and
Directory Structure Changes

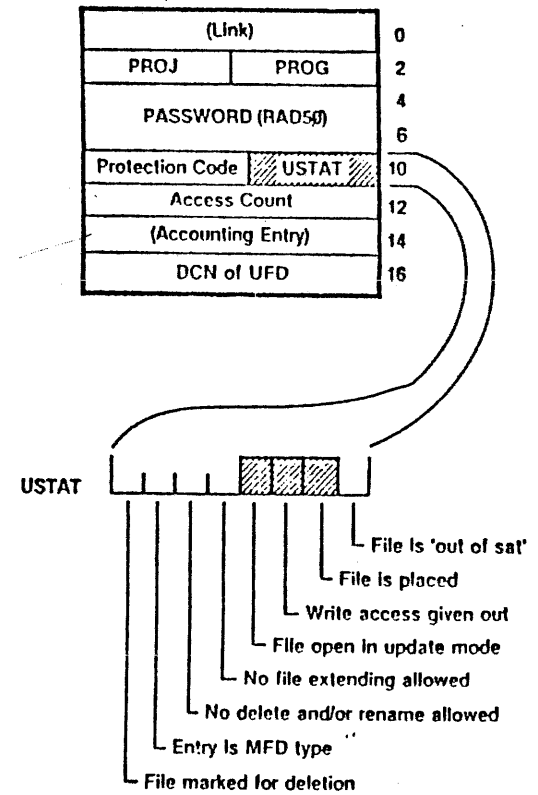


Figure DIR-3. MFD Name Entry

DIGITAL

RSTS/E UPDATE
File Specification and
Directory Structure Changes

UPD Changes

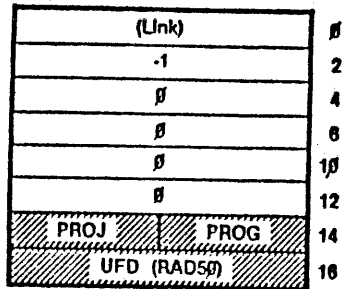


Figure DIR-4. UPD Label Entry (V6C change)

DIGITAL

RSTS/E UPDATE
File Specification and
Directory Structure Changes

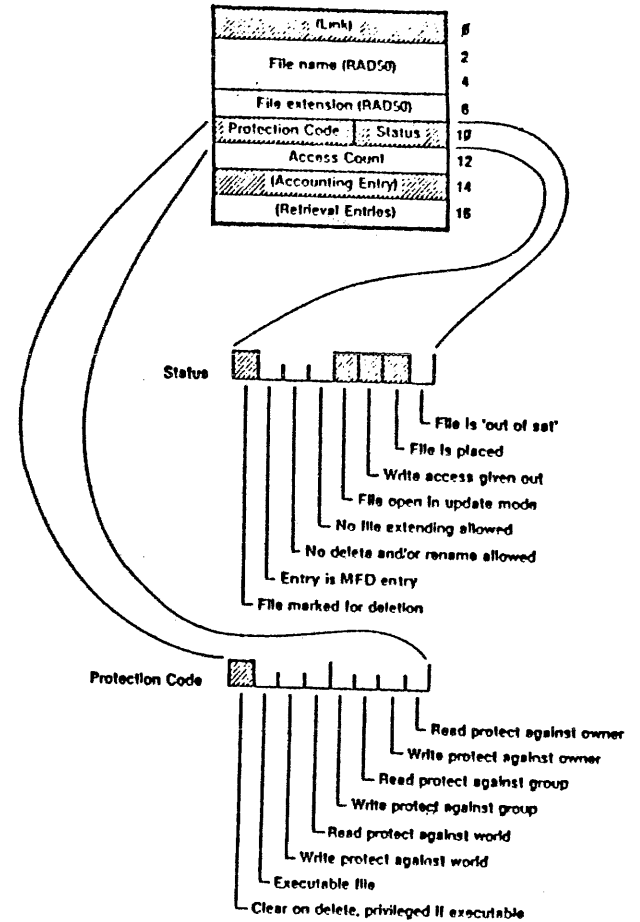


Figure DIR-5. UPD Name Entry

PROF/E UPDATE
File Specification and
Directory Structure Changes

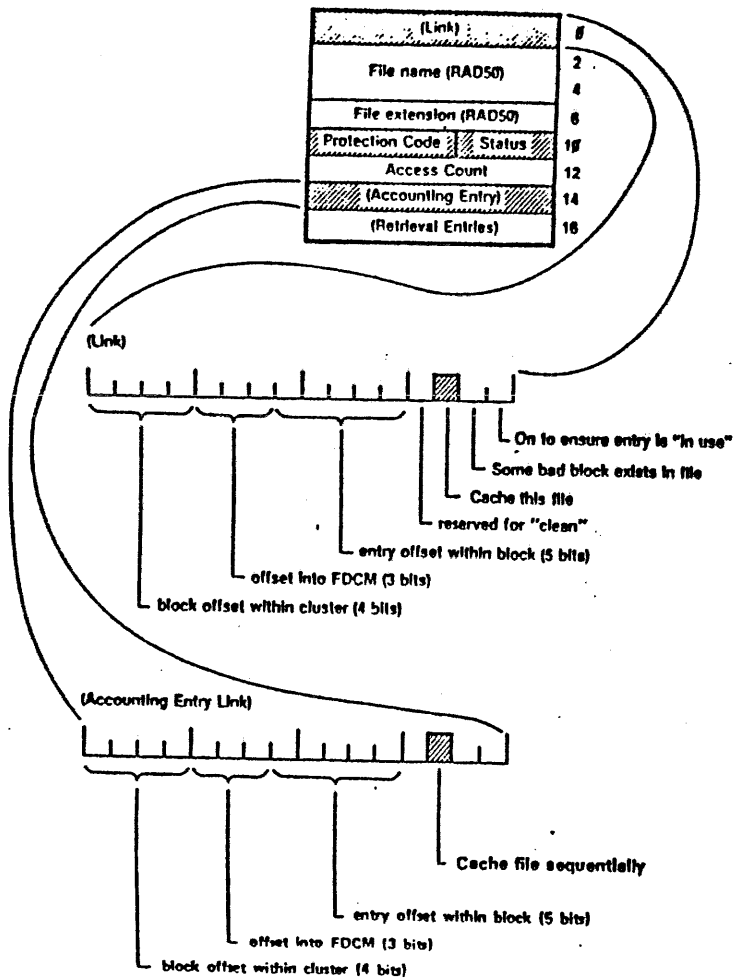


Figure DIR-6. UFD Name Entry (Cont.)

PROF/E UPDATE
File Specification and
Directory Structure Changes

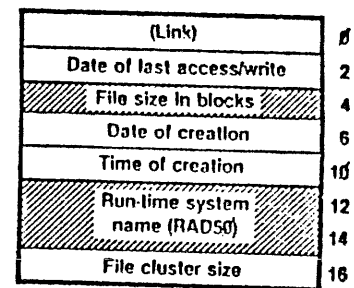


Figure DIR-7. UFD Accounting Entry on Small File Systems

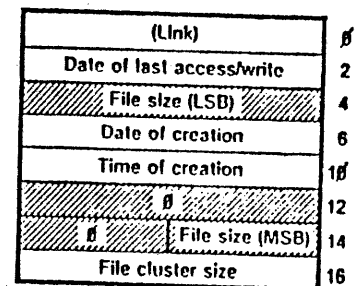


Figure DIR-8. UFD Accounting Entry on Large File Systems

Note
An executable file has run-time system name at offsets 12 and 14 on large file systems.

RSTS/E UPDATE
File Specification and
Directory Structure Changes

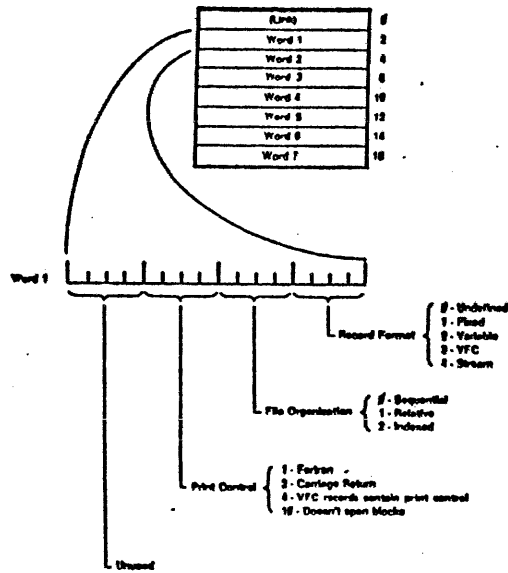


Figure DIR-9. UPD Attributes Entry #1 (V6C change)

- Word 2 Record Size (Actual size for fixed length records or maximum size for variable length records)
- Word 3 Highest virtual block number (MSB)
- Word 4 Highest virtual block number (LSB) (corresponds to the file size accounting entry)
- Word 5 EOF block number (MSB)
- Word 6 EOF block number (LSB) (block that is the logical end of file)
- Word 7 Offset to first usable byte in EOF block.

RSTS/E UPDATE
File Specification and
Directory Structure Changes

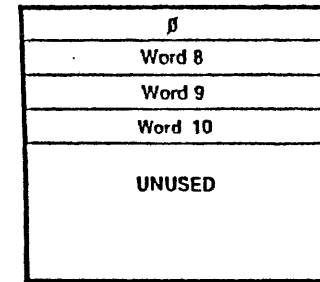


Figure DIR-10. UPD Attributes Entry #2 (V6C change)

- Word 8 Number of bytes in fixed control area (high byte) Bucketsize in blocks (low byte)
- Word 9 Maximum length of record actually read by RMS
- Word 10 Default extension quantity

DIGITAL

RTS 2 UPDATE
File Specification and
Directory Structure Changes

This page is for notes.

RSTS/E V7.0 SWS UPDATE

SOF

Software and RSTS/E V7.0

Software and RSTS/E V7.0

CONTENTS

Introduction.....	SOF-1
BASIC-PLUS and RSTS/E V7.0.....	SOF-3
General.....	SOF-3
Commands.....	SOF-3
Statements.....	SOF-4
Functions.....	SOF-6
System Function Calls.....	SOF-19
File Name String Scan SYS Call.....	SOF-19
Get Monitor Tables II SYS Call.....	SOF-20
Change System Date/Time SYS Call.....	SOF-21
Directory Lookup SYS Call.....	SOF-21
ASSIGN/REASSIGN SYS Call.....	SOF-22
DEASSIGN SYS Call.....	SOF-26
Create a Job SYS Call.....	SOF-28
Exit and Clear Program SYS Call.....	SOF-30
Wild Card PPN SYS Call.....	SOF-31
Return Job Status SYS Call.....	SOF-33
Execute CCL SYS Call.....	SOF-38
Data Caching SYS Calls.....	SOF-40
File Utility SYS Call.....	SOF-45
Resident Library SYS Calls.....	SOF-53
Implementing New RSTS/E Features Under BASIC-PLUS.....	SOF-63
RMS V1.8 and RSTS/E V7.0.....	SOF-80
General.....	SOF-80
Implementing New RSTS/E Features Under RMS V1.8.....	SOF-81
BASIC-PLUS-2 and RSTS/E V7.0.....	SOF-82
Commands.....	SOF-82
Statements.....	SOF-84
System Function Calls.....	SOF-84
Implementing New RSTS/E Features Under BASIC-PLUS-2 V1.60.....	SOF-85
Using the RMS Resident Library.....	SOF-86

DIGITAL

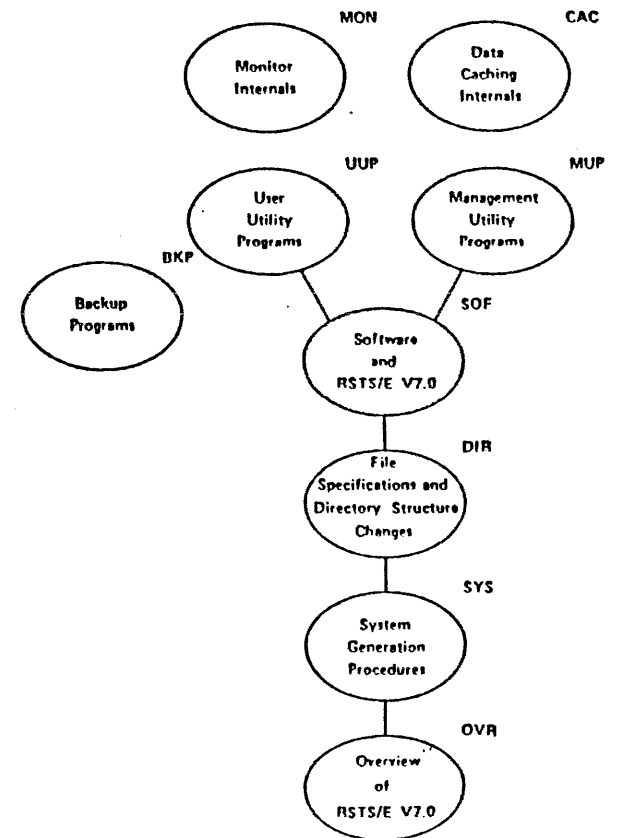
RSTS/E V7.0 SWS UPDATE
Software and RSTS/E V7.0

COBOL V4 and RSTS/E V7.0.....	SOF-89
General.....	SOF-89
Implementing New RSTS/E Features Under	
COBOL V4.....	SOF-92
Using the RMS Resident Library.....	SOF-93
MACRO and RSTS/E V7.0.....	SOF-100
General.....	SOF-100

DIGITAL

RSTS/E V7.0 SWS UPDATE
Software and RSTS/E V7.0

COURSE MAP



CONFIDENTIAL

RSTS/E V7.0 ONLY UPDATE
Software and RSTS/E V7.0

Software and RSTS/E V7.0

INTRODUCTION

The RSTS/E Operating System supports a variety of languages. With this release of RSTS/E, each of the supported languages is affected in different ways. New features are implemented, enhancements are included to improve performance, or additional support is supplied.

This module covers the following software as supported on RSTS/E V7.0.

- o BASIC-PLUS
- o RMS-11 V1.0
- o BASIC PLUS-2 V1.60
- o COBOL V4
- o MACRO-11

DIGITAL

RSTS/E V7.8 SAS UPDATE
Software and RSTS/E V7.8

This page is for notes.

DIGITAL

RSTS/E V7.8 SAS UPDATE
Software and RSTS/E V7.8

BASIC-PLUS and RSTS/E V7.8

General:

Virtual Arrays

- o I/O buffers larger than 512 bytes

Commands

COMPILE

- o Defaults to conditionally contiguous .BAC files

ASSIGN/REASSIGN

- o ASSIGN DEV:/MODE:1 Seizes the device (privileged)
- o ASSIGN DEV:[ppn]LIB Associates the logical name LIB with a device and account number

Note

Making a user logical PPN assignment reduces the number of user logical assignments to three.

EXIT

- o Returns to private default run-time system

DUMP

- o Dumps user space
- o Available as an optional patch

Statements

CLOSE -N

- o Device dependent close operations not performed
- o Pending I/O buffers not written
- o Close a tentative file without making it permanent
- o Example

```

1000  ON ERROR GOTO 19000
1010  OPEN "VIRT.DAT" AS FILE 10, RECORDSIZE 10240  &
      \DIM #10, P.ART$(30)=2560  &
      \PRINT "I/O BUFFERSIZE =",BUFSIZ(10)  &
      \PRINT
1020  FOR I# = 00 TO 20 &
      \P.ART$(I0) = STRING$(2560,I0+650) &
      \NEXT I0
32766  CLOSE -10
32767  END

```

Ready

RUNNH
I/O BUFFERSIZE = 1024

Ready

```

DIR VIRT.DAT /
Name .Ext  Size  Prot  Date      SY:[100,7]
VIRT .DAT  0    < 60> 28-Nov-78

```

Total of 0 blocks in 1 file in SY:[100,7]

Ready

Figure SOP-1. CLOSE with Negative Channel

CHAIN

- o Keyword LINE can be optionally included
- o Example

```

1000  CHAIN "CCPOS" LINE 900
32767  END
Ready

```

Figure SOP-2. CHAIN

Functions

CCPOS

- o Equivalent to POS(N)
- o Example

```
900 P.PRINTSTRINGS = "THIS IS THE FIRST LINE OF THE HEADER"  
1000 PRINT "...USING THE POS FUNCTION..."  
    \PRINT P.PRINTSTRINGS;POS(0)  
    \PRINT  
    \PRINT "...USING THE CCPOS FUNCTION..."  
    \PRINT P.PRINTSTRINGS;CCPOS(0)  
32767 END
```

Ready

```
RUNNH  
...USING THE POS FUNCTION...  
THIS IS THE FIRST LINE OF THE HEADER 36
```

```
...USING THE CCPOS FUNCTION...  
THIS IS THE FIRST LINE OF THE HEADER 36
```

Ready

Figure SOP-3. CCPOS

DEF*

- o Equivalent to DEF
- o Example

```
1300 DEF FNA(X) = X**2 + 3*X + 25  
2000 FOR I = 1 TO 4 STEP .5  
    \PRINT I,,FNA(I)  
    \NEXT I  
32767 END
```

Ready

```
RUNNH  
1 29  
1.5 31.75  
2 35  
2.5 38.75  
3 43  
3.5 47.75  
4 53
```

Ready

Figure SOP-4. DEF*

SPEC%

- o Performs special operations on disk, floppy disk, terminals and magnetic tape.
- o Disk
 - Explicitly locks up to 7 blocks of a file open in update mode
 - Releases explicit and implicit locks on a file block(s)

For disk devices, the SPEC% function allows you to explicitly lock a maximum of seven disk block ranges on a file that is open for update. A locked range of blocks is one that cannot be accessed by another user or from another channel. Thus, SPEC% extends the use of guarded update, which locks the last block or blocks read on a file.

SPEC% also allows you to release explicit or implicit locks. Note that when you close a file, all explicit and implicit locks are released for that file.

The SPEC% function for disk files has the following format:

VALUE%=SPEC%(FUNCTION%, BLOCK, CHANNEL%, HANDLER INDEX%)

where:

- VALUE% depends on the particular function code you specify in FUNCTION%. In most cases, VALUE% is equal to the BLOCK parameter.
- FUNCTION% is a function code that specifies the desired operation. During normal I/O operations, a block, or range of blocks is implicitly locked when you read the file with a BASIC-PLUS GET statement. The SPEC% function allows you to convert implicit locks to explicit locks and to release selected locked blocks. The code specified in FUNCTION% determines the use of SPEC%.

The codes are as follows:

FUNCTION%=0 release all locked blocks

FUNCTION%=1 release the current implicit lock

FUNCTION%=2 convert the current implicit lock to an explicit lock

FUNCTION%=3 release the explicitly locked block specified in the BLOCK parameter. If BLOCK is 0, all explicitly locked blocks are released. However, implicitly locked blocks remain locked.

FUNCTION%=4 convert an implicit lock to an explicit lock and release the implicit lock

BLOCK specifies the starting block number for releasing an explicit lock. Note that BLOCK must be a floating-point number.

CHANNEL% is the I/O channel on which the operation is to be performed.

HANDLER INDEX% The handler index of the I/O device open on CHANNEL%. The index for disk devices is 0%.

If you open a file with a Recordsize greater than 512, SPEC% allows you to lock more than one block when you GET a range of blocks read into the buffer. For example, if you open the file with RECORDSIZE 1024%, each GET operation reads (and implicitly locks) two blocks. If you explicitly lock blocks 2 and 3 as follows:

```
GET #1%, RECORD 2%
VALUE%=SPEC%(2%,0%,1,0%)
```

you can then read blocks 3 and 4 (GET RECORD 30) and cause implicit locks on these blocks. Note that if you attempt to lock a range of blocks that overlap an already explicitly locked range, the Monitor returns the ?DISK BLOCK IS INTERLOCKED error. Also, if a range of blocks is locked, an explicit release of those blocks must refer to the first block in the range.

The following errors are possible during a SPEC% operation:

Meaning	ERROR
?NO ROOM FOR USER ON DEVICE	4
There are too many locks pending on this device. You can lock a maximum of seven ranges of blocks on a file.	
?CAN'T FIND FILE OR ACCOUNT	5
You specified function code 3 in the FUNCTION% parameter and attempted to unlock a block that was not locked.	
?PROTECTION VIOLATION	10
You attempted to explicitly lock a block that had not been implicitly locked. An attempt to lock a block after a PUT or UNLOCK can cause this error.	
?DISK BLOCK IS INTERLOCKED	19
You attempted to explicitly lock a range of blocks that overlaps an already explicitly locked range of blocks.	

o Floppy Disk SPEC%

- Returns density of a mounted floppy
- Causes floppy driver to recompute density
- Reformats a floppy to density specified (RX02 only)

For floppy disk devices, the SPEC% function allows you to obtain the density (single or double) of the current floppy disk, to mount a new floppy disk and recompute the density, and to reformat an RX02 floppy disk for a desired density. Because the RX02 floppy disk drive supports single and double density floppy disks, the SPEC% function is especially useful for programmed floppy disk operations. For example, SPEC% allows you to mount a series of single and double density floppy disks without having to close and reopen the device for each mount. That is, the driver computes density once, during the initial open. If you insert a second floppy that is incompatible with the initially computed density, read or write operations will fail. SPEC% permits you to include an instruction in your program that causes the driver to recompute the density. Also, for RX02 floppy disk drives, SPEC% permits you to specify a density reformat operation.

The SPEC% function for floppy disks has the following format:

VALUE%=SPEC%(FUNCTION%,PARAMETER%,CHANNEL%,HANDLER INDEX%)

where:

VALUE% depends on the particular function code you specify in FUNCTION%.

FUNCTION% is a function code that specifies the desired operation. The codes are as follows:

FUNCTION%-0% returns the density of the currently mounted floppy disk in the form; **DENSITY%-VALUE%** AND 255%. If **DENSITY%-1%**, the floppy is single density; if **DENSITY%-2%**, the floppy is double density. Note that **PARAMETER%** must also be 0%.

FUNCTION%-1% causes the floppy disk driver to recompute density. If the diskette has been changed in the drive without closing and reopening the I/O channel, issue this code prior to any I/O operation on the floppy disk. This function also returns the computed density as described in **FUNCTION%-0%**. Note that **PARAMETER%** must be 0%.

FUNCTION%-2% reformats the current floppy disk to the density specified in **PARAMETER%**. **PARAMETER%** equals 1% for single density and 2% for double density. Note that this operation is allowed only on RX02 drives and that any data on the disk prior to the operation is lost.

PARAMETER% see the description of **FUNCTION%**.

CHANNEL% is the I/O channel on which the operation is to be performed.

HANDLER INDEX% the handler index on the I/O channel open on **CHANNEL%**. The index for floppy disks is 18%.

SPEC% can require as much as 20 seconds to reformat the density on an RX02 floppy disk and cannot be interrupted with CTRL/C. Note that if the operation is interrupted (by power failure or catastrophic error), the floppy disk is rendered unusable. That is, the floppy disk will contain both single and double density. To recover you must reformat the floppy disk.

The following errors are possible during a **SPEC%** operation:

Meaning	ERROR
?DEVICE HUNG OR WRITE LOCKED	14
A hardware error occurred. This can often be a transient condition. Retry the operation.	
?MISSING SPECIAL FEATURE	66
An attempt was made to reformat on an RX01 floppy disk drive. The use of SPEC% to reformat floppy disk density is allowed only on RX02 drives.	

SPEC% can be used in floppy disk programming operations to ensure that sector opens are correctly handled. That is, the conflict between 128-byte single density buffersizes and 256-byte double density buffersizes can be resolved by the following method:

To FIELD the buffer;

FIELD #channel number, 128% ***DENSITY%** AS BUFFER RX02S

To PUT the buffer;

PUT #channel number, COUNT 128% ***DENSITY%**

When **DENSITY%** is;

DENSITY%=SPEC%(0%,0%,CHAN%,18%) and 255%

o Magnetic tape SPEC%

- Provides equivalent functions as the MAGTAPE function.

For magnetic tape, the SPEC% function allows you to rewind the tape, skip records on the tape, and set tape density and parity.

The SPEC% function for magnetic tape has the following format:

VALUE%=SPEC%(FUNCTION%,PARAMETER,CHANNEL%,HANDLER INDEX%)

where

VALUE%	depends on the particular function code specified in FUNCTION%.
FUNCTION%	is the function code.
PARAMETER	depends on the particular function code specified in FUNCTION%.
CHANNEL%	is the I/O channel on which the operation is to be performed.
HANDLER	is the handler index on the I/O device open
INDEX%	on CHANNEL%. The index for magtape is 14%.

The SPEC% function performs various operations on magnetic tape as determined by the code you specify in FUNCTION%. These codes are as follows:

FUNCTION%=1	rewind the tape and take the unit off-line.
FUNCTION%=2	write an end of file marker.
FUNCTION%=3	rewind the tape but leave the unit on-line.

FUNCTION%=4 read the tape but skip specified records. You specify the number of records to be skipped in PARAMETER. Note that when FUNCTION% is 4, VALUE% returns the number of records that were read (i.e., not skipped).

FUNCTION%=5 backspace the tape over a specified number of records. You specify the number of records to be backspaced in PARAMETER. Note that when FUNCTION% is 5, VALUE% returns the number of records remaining (i.e., not backspaced).

FUNCTION%=6 set the tape's density and parity as specified in PARAMETER. The format for a PARAMETER specification, when FUNCTION% is 6, is E+D*4+P+S. The specifications E,D,P, and S have the following values:

E is phase encoded
 256 = phase encoded mode, refer to value of D
 0 = refer to values of D and P

D is density
 if E=0

0 = 200 bpi (7-track only)
 1 = 556 bpi (7-track only)
 2 = 800 bpi (7-track only)
 3 = 800 bpi (dump mode)

if E=256

0 = 1600 BPI
 1 = Reserved for future use
 2 = Reserved for future use
 3 = Reserved for future use

P is parity
 0 = odd parity
 1 = even parity

S is stay
0 = the OPEN MODE value is released on CLOSE
8192 = the OPEN MODE value is retained on CLOSE

FUNCTION%=7 returns the tape status in VALUE%. The status returned is encoded as follows:

Bit	Decimal Value	Meaning
0-2	0	last issued command was off-line
	1	last issued command was read
	2	last issued command was write
	3	last issued command was write EOF
	4	last issued command was rewind
	5	last issued command was skip record
6	0	last issued command was backspace record
	1	See bits 13-14
3	0	See bits 13-14
	1	See bits 13-14
4	0	unit is TU10/TE10/TS03
	1	unit is capable of 1600 BPI
5	1	unit is off-line
6	0	the last issued command was a read on a record that was larger than the I/O buffer (part of the record is lost)
	1	last command detected an EOF
7	1	tape is at its load point
8	1	tape is beyond the EOT marker
9	1	tape is write-locked
10	0	tape has odd parity
11	1	tape has even parity
12	0	9-track tape
13-14	1	7-track tape
	0	if bit 3 = 0
	0	0 = 200 bpi
	1	1 = 556 bpi
	2	2 = 800 bpi
3	3 = 800 bpi, dump mode	

if bit 3 = 1
0 = 1600 bpi
1 = reserved
2 = reserved
3 = reserved

15 1 last command caused an error

FUNCTION%=10 returns the file characteristics in VALUE%. The characteristics returned in VALUE% are encoded as follows:

Word=0 DOS format tape
Word<>0 ANSI format tape. If the word is non-zero, the bytes have the following meaning:

Bit	Decimal Value	Meaning
0-11	-	For fixed-length ANSI format, the value is the record length in bytes. For variable-length ANSI format, the value is the maximum record length in bytes.
	0	embedded carriage control
12-13	4096	FORTRAN carriage control
	8192	implied line feed/carriage return before the record
14-15	16384	fixed-length format
	32768	variable-length format

FUNCTION%=11 rewind the tape on CLOSE

Note that the operation and values described for FUNCTION% are covered in detail in the RSTS/E Programming Manual.

o Terminal SPEC%

For terminals, the SPEC% function allows you to cancel CTRL/O and set modes for tape, echo, and ODT.

The SPEC% function for terminals has the following format:

VALUE%=SPEC%(FUNCTION%,PARAMETER,CHANNEL%,HANDLER INDEX%)

where

VALUE%	depends on the particular function code specified in FUNCTION%.
FUNCTION%	is the function code. The SPEC% functions performs various operations on terminals as determined by the FUNCTION% code. These codes are as follows: FUNCTION%=0 Cancel CTRL/O FUNCTION%=1 set tape mode FUNCTION%=2 enable echo and clear tape mode FUNCTION%=3 disable echo FUNCTION%=4 set ODT mode FUNCTION%=7 cancel all type ahead
PARAMETER	specifies the terminal on which the operation is to take place. If PARAMETER is 0, the operation is performed on the currently open terminal. If you specify a keyboard number in PARAMETER, the operation is performed on that terminal. Note that the keyboard must be assigned to the calling job and must not be open.
CHANNEL%	specifies the I/O channel for the terminal in PARAMETER.
HANDLER INDEX%	the handler index of the I/O device open on CHANNEL%. The index for terminals is 2%.

System Function Calls

File Name String Scan SYS Call

- o Suppresses logical device translation
- o Example

ASSIGN MT:KB

Ready

LISTNH

1000 INPUT "FILE NAME?"; F% &

\V% = SYS(CHR\$(6%)&CHR\$(-10%)*"_"&F%) &

\DEVICES = MID(V%,23%,2%) &

\PRINT "DEVICE WITHOUT LOGICAL TRANSLATION IS "; DEVICES

1010 V% = SYS(CHR\$(6%)&CHR\$(-10%)*F%) &

\DEVICES = MID(V%,23%,2%) &

\PRINT "DEVICE WITH LOGICAL TRANSLATION IS "; DEVICES

32767 END

Ready

RUNNH

FILE NAME? KB:

DEVICE WITHOUT LOGICAL TRANSLATION IS KB

DEVICE WITH LOGICAL TRANSLATION IS MT

Ready

Figure SOF-5. PNC

Get Monitor Tables II SYS Call

- o Returns information on whether or not a system is configured for large files

o Example

```

1000 ON ERROR GOTO 19000
2000 R.RETURNS = SYS(CURS(60)+CHRS(-120))  IMONITOR TABLES II &
      \IF SWAP(CVTS(RIGHT(R.RETURNS,29)))  &
      THEN PRINT "LARGE FILE SYSTEM"      &
      ELSE PRINT "LARGE FILES NOT AVAILABLE" &
      \PRINT WHETHER THIS IS A LARGE FILE SYSTEM
2050 GOTO 32767
19000 PRINT ERR
      \RESUME 32767
32767 END

```

Ready

RUNNE
LARGE FILE SYSTEM

Ready

Figure SOP-6. Get Monitor Tables

Change System Date/Time SYS Call

- o Accepts zeroes for date/time

o Example

```

VS=SYS(CHRS(60)+CHRS(-140)+CHRS(00)  &
      +CHRS(SWAP(00))+CHRS(00)      &
      +CHRS(SWAP(00)))

```

RSTS/E V6C - ?Illegal mode mixing

RSTS/E V7.0 - No error returned
No change to date/time made

Directory Lookup Sys Call

- o Always returns the correct file size (large file systems only)

ASSIGN/REASSIGN Sys Call

- o Assigns/reassigns user logical names, user logical PPNs and user default protection codes
- o Seizes a device

Data Passed:

Byte(s)	Meaning
1	CHRS(6%), the SYS call to FIP
2	CHRS(10%), the assign/reassign and enter user logical code.
3-4	Not used
5-6	For user logical assignment, the project-programmer number to be assigned.
7-10	For assign device, bytes 7 through 10 must be 0. For reassign device, byte 7 is the job number to which the device is reassigned. Bytes 8 through 10 must be 0. For user logical assignment, bytes 7 through 10 contain the logical device name (in Radix-50 format) to be assigned.
11-12+	For assign/reassign device, either DOS or ANS (in Radix-50 format) to specify DOS or ANSI label format for the magnetic tape drive. For user logical assignment, not used.

Byte(s)	Meaning
13-16	Not used
17-18	For assign/reassign device, CVT%\$ (SWAP%(-32767%)) to assign a device that is currently assigned to another user. This usage is privileged and the target device must not be open. If this operation is not desired, bytes 17 and 18 are 0. For user logical assignment, not used.
19-20	Not used
21	For user logical assignment, CHRS(255%) to enable protection code assignment (see BYTE 22). For assign/reassign device, must be 0.
22	For user logical assignment, the protection code to be assigned. Byte 21 must be 255. For assign/reassign device, must be 0.
23-24+	Device name for assign/reassign and for user logical assignment.
25+	Unit number for assign/reassign and for user logical assignment.
26+	Unit number flag for assign/reassign and for user logical assignment.
27-30	Not used

Data Returned: No meaningful data is returned.

Possible Errors:

Meaning	ERR Value
?ACCOUNT OR DEVICE IN USE	3
The specified device is currently open or has an open file. This error can only occur during an assign call.	
?NOT A VALID DEVICE	6
The device name specified in bytes 23 and 24 is a logical device name for which a physical device is currently not assigned.	
?DEVICE NOT AVAILABLE	8
The device specified in bytes 23 through 26 exists on the system but the operation fails for one of the following reasons:	
1. The device is currently reserved by another job (see bytes 17 and 18).	
2. Ownership of the device requires privilege that the user does not have.	
3. The device or its controller is disabled.	
4. The device is a keyboard line for a pseudo keyboard use only.	
?PROTECTION VIOLATION	10
A program with temporary privilege assigned a device, dropped its privilege, and attempted to reassign the device.	
?ILLEGAL NUMBER	52
An attempt is made to transfer control to a non-existent job. This error can only occur during a reassign call.	

Discussion:

The user logical assignment call is used to enter logical device names, logical project-programmer numbers, and default output protection codes. To assign a user logical device name, bytes 7 through 10 must contain the logical device name and bytes 23 through 26 must contain a physical device name and unit number. To assign a user logical project-programmer number, specify the number in bytes 5 and 6. To assign a user default protection code, specify the code in bytes 21 and 22.

The assign/reassign call used bytes 17 and 18 to assign or reassign a device which is currently assigned. For the call to be successful, it must originate from a privileged account, the target device must not be open, and the current owner can not be performing a directory on that device.

The assign call reserves a physical device to a job* or transfers assignment of a currently owned device to another job. The actions are equivalent to the ASSIGN and REASSIGN Monitor commands. System logical names are assigned either by a SYS call 21 or by UTILTY program commands.

Example:

```

10 AS = SYS (CHR$(6))+CHR$(10)+STRING$(20,0)+
    "LP"+CHR$(1)+CHR$(255))
    I ASSIGN LP1: TO CURRENT JOB
20 INPUT "ASSIGN LP1: TO WHICH JOB"; X1
30 AS=SYS (CHR$(6))+CHR$(10)+STRING$(14,05)+
    CHR$(X1)+CHR$(0)+STRING+(14,0)+
    "LP"+CHR$(1)+CHR$(255))
    I REASSIGN LP1: TO JOB # X1.

```

*The system manager, through an initialization option, can designate that certain devices require privilege to be assigned.

Deassign SYS Call

- o Removes user logical assignments

Data Passed:

Byte(s)	Meaning
1	CHR\$(6%), the SYS call to FIP
2	CHR\$(11%), the deassign device and remove user logical code
3-4	Not used
5-6	For user logical removal, the project-programmer number to be removed. For device deassignment, must be 0.
7-10	For user logical removal, the logical device name (in Radix-50 format) to be removed. For device deassignment, must be 0.
11-20	Not used
21	For user logical removal, CHR\$(255%) to enable protection code removal. For device deassignment, must be 0.
22	For user logical removal, the protection code to be removed. Byte 21 must be 255. For device deassignment, must be 0.
23-24	For device deassignment, the device name to be deassigned. For user logical removal, must be 0.

- 26 For device deassignment, the unit number flag.
For user logical removal, must be 0.
- 27-30 Not used.

Data Returned: No meaningful data is returned

Possible Errors:

Meaning	ERR Value
?NOT A VALID DEVICE	6
The device or device type specified in bytes 23 through 26 is not configured on the system. This error can occur only on device deassignment calls.	

Discussion:

The user logical removal call is used to deassign logical device names, logical project-programmer numbers, and default output protection codes. To deassign a logical device name, specify the name in bytes 7 through 10. To deassign a project-programmer number, specify the number in bytes 5 and 6. To deassign a user logical protection code, specify the code in bytes 21 and 22. Note that if these bytes do not contain specific deassignments, the call deassigns all user logical device name, PPN, and protection code assignments.

The device deassignment call performs the same action as the DEASSIGN system command described in the RSTS/E System User's Guide. For example, the following statement deassigns line printer unit 1, which is assigned to the current job.

```
10 AS = SYS (CHR$(6%)+CHR$(11%)+STRING$(20%,0%)+
"LP"+CHR$(1%)+CHR$(255%))
IDEASSIGN LP1:
```

Create a Job SYS Call

- o Spawns a job as long as there is swap space available

Data Passed:

Byte(s)	Meaning
1	CHRS(6%), the SYS Call to FIP
2	CHRS(24%), the job creation code
3	CHRS(N%), where N% = 0% Spawn if LOGINS are enabled 127% Spawn regardless
	Note (N% AND 127%) must be 0
4	Not used, must be 0
5-6+	The project-programmer number of the program to be run.
7-10+	Program name of the program to be run in RADIX-50 format.
11-12+	Extension of the program to be run in RADIX-50 format.
13-22	Up to 10 bytes of information; placed into the created job's Core Common area.
23-26+	The device name and unit number of the program to be run.
27-28	The parameter word to be passed to the program to be run. The parameter word has exactly the same format and functions as the CCL command parameter word.
29-30	Not used

Data Returned:

Byte(s)	Meaning
3	The job number times 2 of the job just created.

Possible Errors:

Meaning	ERR Value
NO ROOM FOR USER ON DEVICE	4
The new job cannot be created. Probable causes are:	
a) Further logins are disabled	
b) The system's job or swap slots are (currently) full	

Discussion:

The job creating the new job must be running under control of the BASIC-PLUS Run-Time System. The job created can run under control of any available run-time system.

Exit and Clear Program SYS Call

- o Allows a temporary switch to the specified run-time system.

Data Passed:

Byte(s)	Meaning
1	CHRS(9%), the exit and set up NONAME code.
2-3	First 3 characters of the name of the run-time system, in RADIX-50 format, to which control is passed. If bytes 2-5 are not specified, or if they are 0, your private default run-time system is used.
4-5	Last 3 characters of the name of the run-time system, in RADIX-50 format, to which control is passed.
6	If unspecified, establishes the named run-time system as the job's private default run-time system. Otherwise, CHRS(N%), where N% is <ul style="list-style-type: none"> 255% to establish the run-time system as the job's private default run-time system. 0% to enter the specified run-time system without establishing it as the job's default run-time system.

Wildcard PPN Lookup SYS Call

- o New with V7.0

Wildcard PPN Lookup - Not Privileged (F=25)**Data Passed:**

Byte(s)	Meaning
1	CHRS(6%), the SYS call to PIP
2	CHRS(25%), the code for account number lookup on index
3-4	CHRS(N%)+CHRS(SWAP%(N%)), where N% is the index of the desired project-programmer number. If N% is 0, the call returns the project-programmer number of the first account on the disk that matches the wildcard specification. If N% is non-zero, the call returns the project-programmer number of the N+1 account on the specified disk that matches the wildcard specification.
5-6	The desired project-programmer number. A specification of 255 in either field represents a wildcard. A specification of 0 for N% in bytes 3-4 and a non-255 value in bytes 5-6 (no wildcard) verifies the existence of the specified account on the disk. If both fields are non-zero and neither field is 255, the call returns error 5 if the account does not exist.
7-22	Not used
23-24+	The device name, which must be a disk. If bytes 23 and 24 are both 0, SY0:(the system disk, not the entire public structure) is used.
25+	The device unit number

26+ The unit number flag. If byte 26 is 0, SY0:
(the system disk) is used.

Data Returned:

Byte(s)	Meaning
3-4	Internal code
5-6+	The project-programmer number located by the call
7-30	Same as passed

Possible Errors:

Meaning	ERR Value
?CAN'T FIND FILE OR ACCOUNT	5
The specified device in bytes 23-24+ is not a disk or no match exists for the specified index value in bytes 3-4(see also bytes 5-6).	
?DISK PACK IS LOCKED OUT	22
The disk is in the locked state. Execute the call under a privileged account to override this condition.	

Discussion:

This call allows you to specify a wildcard account number and increment an index value to determine a matching project-programmer number. The wildcard account specification is as returned from the filename string scan call.

Return Job Status SYS Call

o New with V7.0

Return Job Status - Privileged and Not Privileged (F0=26)

Data Passed:

Byte(s)	Meaning
1	CHR\$(6%), the SYS call to FIP
2	CHR\$(26%), the return job status code
3	CHR\$(J%), where J% is the number of the job for which status is desired. If J% is 0%, information on the caller's job is returned. If the caller is not privileged, J% is forced to 0%.
4	CHR\$(S%), where S% is 0% or 1%. The value of S% determines the information returned on the job (see Data Returned).
5-30	Not used

Data Returned:

If S% is 0%:

Byte(s)	Meaning
1	The calling job's job number times two.
2	Not used
3	Job number times two of the job for which data is being returned. If the caller is not privileged, this byte is equal to byte 1.

4	Keyboard number of the job's console. If the number is negative, the job is detached and the number is the complement of the keyboard number.
5-6	If the job is attached to a pseudo keyboard, byte 5 contains the controlling job's job number times two; otherwise byte 5 is 0. If the job is swapped out, byte 6 contains the job's swap slot location; otherwise byte 6 is 0.
7-8	The job's logged in CPU time (least significant word) for the current session in tenths of a second.
9-10	The job's current connect time in minutes.
11-12	The job's current KCTs (least significant word) for this session.
13-14	The job's accumulated device time for the current session in minutes.
15	The most significant byte of the job's KCT.
16	The most significant byte of the job's CPU time.
17-20	The job's name in two Radix-50 words.
21-22	The job's project-programmer number.
23-26	The name of the job's default run-time system in two Radix-50 words.
27-30	The name of the job's current run-time system in two Radix-50 words.

!! SA is 1:

Byte(s)	Meaning
1	The calling job's job number times two.
2	Not used
3	Job number times two of the job for which data is being returned. If the caller is not privileged, this byte is equal to byte 1.
4	Keyboard number of the job's console. If the number is negative, the job is detached and the number is the complement of the keyboard number.
5-6	The job's current flag word.
7	The job's current IOSTS byte.
8	The job's current information posting byte.
9-10	The job's current JBSTAT word.
11-12	The job's current JBWAIT word.
13	The size of the job's current user memory area in K words.
14	The job's current word from its memory control sub-block.
15-16	The job's current physical address in 32-word increments.
17	The job's priority.
18	The job's allotted run burst in tenths of a second.
19	The job's maximum allowable memory size in K words.

- 20 The value at offset 6 in the job's work block. This value is usually the channel number (times two) on which the job is performing an I/O operation.
- 21-22 If bytes 9 through 12 indicate that the job is in a keyboard wait state, bytes 21 and 22 contain the value at offset 12 (octal) in the job's work block. This value is the timeout parameter for input from the terminal. If the value is negative, it implies that the terminal is in a keyboard monitor (CTRL/C) input wait state.
- If bytes 9 through 12 indicate that the job is in an I/O stall for a non-keyboard device, bytes 21 and 22 contain the generic name (in ASCII) of the device for which the job is stalled.
- If bytes 9 through 12 indicate that the job is in a FIP wait state, byte 21 contains the byte value corresponding to the currently executing FIP function and byte 22 has no meaning.
- 23-24 The value at offset 16 (octal) in the job's work block. This value is usually an internal code that specifies whether the job is reading or writing on the current I/O channel. If the job is in an I/O wait state and this value is 2, the I/O operation is a read; if this value is 4, the I/O operation is a write.
- 25-26 A pointer to the beginning of the job's Job Data Block.
- 27-28 A pointer to the beginning of the job's second Job Data Block.
- 29-30 If the job is a receiver, these bytes contain a pointer to the job's receiver identification block; otherwise, these bytes are 0.

Possible Errors:

Meaning	ERR Value
?PROTECTION VIOLATION.	10
The calling job is not logged in.	
?ILLEGAL SYS() USAGE	18
The calling job number is less than zero or greater than JOBMAX.	

Execute CCL SYS Call

- o New with V7.0
- o Allows execution of CCL commands while a program is running

Data Passed:

Byte(s)	Meaning
1	CHRS(140), the execute CCL command Call
2-128	The string to be executed

Data Return: The target string is equivalent to the passed string.

Discussion:

This call scans the string that is passed to determine if it is a valid CCL command. If it is, the command is executed as though it had been typed directly to a keyboard monitor.

If the string passed is not a CCL command, execution of the program continues at the next statement.

Possible errors:

Meaning	ERR Value
?LINE TOO LONG	47
The string that was passed is too long to be executed as a CCL command.	
?ILLEGAL NUMBER	52
An illegal number was specified as an argument to one of the CCL switches, such as specifying "/SIZE:A"	

?ILLEGAL SWITCH USAGE

67

An illegal switch was specified for the CCL command, such as requesting a size that is larger than the System's SWAP MAX.

Note

Other errors will not be detected until the current program has been removed and execution of the invoked program is attempted. (See the discussion of .CCL in the RSTS/E System Directives Manual.)

Data Caching Sys Calls

- o Enabling and Disabling Disk Caching

Data Passed:

Byte(s)	Meaning
1	CHRS(6%), the SYS call to FIP
2	CHRS(19%), the enable and disable cache code
3	CHRS(N%), where N% is: <ul style="list-style-type: none"> 0 to enable directory and data caching. In addition to this byte, data caching requires a value setting in byte 11. Note that bytes 4 through 12 are used only if this byte equals 0. 128 to return the current caching parameters. A 128 value in this byte does not enable or disable data caching. 1 to disable all caching
4	CHRS(C%), where C% is the cache clustersize. If C% is 0, the current clustersize is used (see Discussion). Cache clustersize can be specified as 1,2,4, or 8 blocks. If C% is greater than 8, 8 is used.
5-6	CHRS(L%)+CHRS(SWAP%(L%)), where L% sets a limit on the total number of cache clusters that can be used. If L% is 0, the current limit is used (see Discussion). If L% is non-zero, it specifies an upper limit on the number of clusters in the cache. Note that if the amount of XBUF available to the cache is less than L%, the cache will not exceed XBUF.

7-8

CHRS(D%)+CHRS(SWAP%(D%)), where D% sets a limit on the total number of cache clusters allocated for directory caching. If D% is 0, the current limit is used (see Discussion). If D% is non-zero, it specifies an upper limit for the number of clusters in the cache that are available for directory caching. Note that the number of clusters allocated for directory caching during a particular operation can be less than D%.

9-10

CHRS(U%)+CHRS(SWAP%(U%)), where U% sets a limit on the total of cache clusters allocated to user data caching. If U% is non-zero, it specifies an upper limit for the number of clusters in the cache that are available for user data caching. Note that the number of clusters allocated for data caching during a particular operation can be less than U%.

11

CHRS(E%), where E% modifies the enabling/disabling of data caching as follows:

E%=0 use the current setting
 E%=1 enable data caching as specified in file OPEN MODE or directory flag
 E%=128 disable all data caching
 E%=64 cache all data transfers regardless of file OPEN MODE

12

CHRS(M%), where M% controls the cache's use of the small buffer pool as follows:

M%=0 use the current setting
 M%=1 use the small buffer pool
 M%=128 do not use small buffer pool

Data Returned:

Byte(s)	Meaning
1-2	Internal coding
3	Current cache setting and available options
0	Cache disabled, user data caching is not available
1	Cache enabled, user data caching is not available
128	Cache disabled, user data caching is available
129	Cache enabled, user data caching is available
4-12	Current settings of cache parameters, as described for passed data. Note that these bytes have meaning only if the system manager has installed user data caching during system generation.

Possible Errors:

Meaning	ERR Value
?ACCOUNT OR DEVICE IN USE	3
All of the clusters allotted to the cache are in use.	
?NO ROOM FOR USER ON DEVICE	4
An attempt was made to enable data caching without sufficient XBUF space allocated to the cache. The system manager must allocate at least 2K words of memory to XBUF for caching.	

?DEVICE NOT AVAILABLE

8

An attempt was made to change the cache clustersize (see Byte 4) while a cached file disk transfer was in progress. Retry the operation.

?PROTECTION VIOLATION

10

The current user is not privileged.

?MISSING SPECIAL FEATURE

Caching was not configured for the system or data caching was configured and use of the small buffer pool was requested.

Discussion:

Bytes 1, 2, and 3 of this call enable or disable the FIP buffering module that controls directory caching. The ENABLE CACHE and DISABLE CACHE commands of UTILITY use these bytes.

If user data caching was configured on the system during system generation, bytes 1 through 12 enable or disable user data caching and set the parameters of the cache. The system manager defines the total size of XBUF during system initialization and some portion of this space is, in turn, used by the cache. The disk caching SYS call defines the size of the directory portion or the data portion of the cache. The sizes defined in this call set upper limits, not fixed sizes. Thus, if a 40K word XBUF were defined at system initialization, the SYS call could define the directory and data portions of the cache as 25K each. That is, data can use the space in the cache up to a maximum of 25K, which leaves a minimum of 15K for the directory. The reverse is also true. In this manner, data and directory caching are guaranteed a minimum allocation and are allowed to overlap, which permits the cache to dynamically adjust to system and program requirements.

This SYS call is also used to limit the size of the total cache. Because XBUF is used by the cache and DECnet/E, limiting the cache guarantees that space is always available in XBUF for DECnet/E. Note that the amount of memory allocated to the cache is freed for other use when caching is not being performed.

Byte 4 of the call sets the cache clustersize. This parameter controls the number of contiguous blocks that are copied from the disk to the cache whenever file data or directory data are cached. The cache clustersize should be small enough to contain a reasonable number of clusters, but large enough to reduce the number of disk accesses. That is, you must anticipate data requests and ensure that the cache is equal to the file clustersize of the most often accessed file. If you specify a cache clustersize of 1, only random caching is allowed. Refer to the RSTS/E System Manager's Guide for cache clustersize guidelines.

Note that the parameters for cache clustersize and cluster allocation (Bytes 4 through 10) have default settings at system startup. The default settings are a cache clustersize of 4 and no limits on directory, data, or total cache size. The system manager can reset these defaults with an INIT option as described in the RSTS/E System Manager's Guide.

File Utility SYS Call

- o New with V7.0
- o Sets a file's directory data cache characteristics
- o Makes a contiguous file non-contiguous
- o Changes a file's placed bit setting
- o Obtains file's starting DCN
- o Changes a file's run-time system name
- o Changes a file's backup characteristics
- o Returns an unopened file's run-time system name and attributes

File Utility Functions - Privileged

Date Passed:

Byte(s)	Meaning
1	CHRS(6%), the SYS call to FIP.
2	CHRS(-26%), the file utility code.
3	CHRS(N%), where N% is the internal channel number (in the range of 1 to 12) on which the file is open. If N% is 0, specify the target file by project-programmer number, filename, and extension in bytes 5 through 12.
4	CHRS(F%), where F% specifies the file utility function. The function F% is one (or the sum) of the following codes:

- 1% set or reset the file's placed bit (can not be used with code 8%). See byte 15.
- 2% modify code 16% to return 0 as the device cluster number (DCN) if the file's placed bit is not set.
- 4% change the file's backup statistics.
- 8% change the file's run-time system name field.
- 16% return the file's retrieval information. That is, 16% causes the Monitor to map the virtual block number (VBN) of the file into the disk DCN. This code can be used to obtain an existing file's DCN in order to place a new file near it (see Discussion)
- 32% unset the file's contiguous bit. This code allows you to extend a contiguous file, however, the file is made non-contiguous.
- 64% enable/disable sequential mode caching if the file is cached (can not be used with code 8%). See byte 15.
- 128% enable/disable data caching on the file (can not be used with code 8%). See byte 15.
- 5-6+ If N% in byte 3 is 0, specify the project-programmer number of the file you wish to modify.
If N% is non-zero, these bytes are ignored.
- 7-10+ If N% in byte 3 is 0, specify the filename (in Radix-50 format) of the file you wish to modify.
If N% is non-zero, these bytes are ignored.

- 11-12+ If N% in byte 3 is 0, specify the file extension (in Radix-50 format) of the file you wish to modify.
If N% is non-zero, these bytes are ignored.
- 13-16 The specifications that you make in these bytes are dependent on the function code specified in byte 4, as follows:

If byte 4 AND 8% <> 0%, then bytes 13 through 16 contain the new run-time system name field in Radix-50 format.

Otherwise;

If byte 4 AND 16% <> 0%, then bytes 13 and 14 contain the low order word of the VBN you wish to locate, byte 15 contains 0% or is used by another operation, and byte 16 contains the high order word of the VBN you wish to locate.

If byte 4 AND 1%+64%+128%, then bytes 13, 14 and 16 contain zeros or are used by another operation, byte 15 contains flags for the following operations:

2% new value for the placed bit if byte 4 AND 1% <> 0%.

4% new value for sequential bit if byte 4 AND 64% <> 0%.

128% new value for cached bit if byte 4.
- 17-18 If the change file backup statistics function is selected in byte 4 (code 4), these bytes specify a new date of last access for the file. If no change of date is desired, specify 0.

If the statistics function is not selected, these bytes are ignored.

- 19-20 If the change file backup statistics function is selected in byte 4 (code 4), these bytes specify a new date of creation for the file. If no change of date is desired, specify 0. If the statistics function is not selected, these bytes are ignored.
- 21-22 If the change file backup statistics function is selected in byte 4 (code 4), these bytes specify a new time of creation for the file. If no change of time is desired, specify 0. If the statistics function is not selected, these bytes are ignored.
- 23-24+ If N# in byte 3 is 0, specify the name of the device that contains the file you wish to modify. The device must be a disk and a specification of 0 in bytes 23 and 24 indicates the public disk structure.
- If N# is non-zero, these bytes are ignored.
- 25-26+ If N# in byte 3 is 0, specify the unit number and unit number flag associated with the file you wish to modify.
- If N# is non-zero, these bytes are ignored.
- 27-30 Not used.

The file's characteristics are returned as follows:

Byte(s)	Meaning
1	Not used
2	byte 1 = 2 file is placed
	byte 1 = 4 file will be cached sequentially if at all
	byte 1 = 16 file is contiguous
	byte 1 = 128 file will be cached on open
3-4	If the file's VBN was passed in byte 16 and file retrieval information (code 16) was requested in byte 4 (see Data Passed), these bytes contain the DCN of the file's VBN. Note that these bytes return 0 if the specified VBN is larger than the file size or if the file was not placed and function code 2 was not passed in byte 4.
5-26	File attribute data; unused words are filled with zeroes.
27-30	The file's run-time system name in Radix-50 format.

Possible Errors:

Meaning	ERR Value
?CAN'T FIND FILE OR ACCOUNT	5
The file or account specified in bytes 5 through 12 is not present on the disk.	

?I/O CHANNEL NOT OPEN 9

The channel specified in byte 3 is not open.

?PROTECTION VIOLATION 10

The file open on the channel specified in byte 3 is not a disk file or the job lacks the privilege required for the specified operation.

?ILLEGAL SYS() USAGE 18

The file open on the specified channel is not a disk file or is a user file directory.

Discussion:

This call supplements the functions of the change file run-time system name call (FIP SYS Call -17) and the change file statistics call (FIP SYS Call -11) to support large files and to add support for file placement. You can also use this call to obtain a file's run-time system name and attribute data without opening the file.

The run-time system name field (see Data Passed, bytes 4 and 27 through 30) in the accounting entry of the file's User File Directory contains file size information for large files. The two-word run-time system name field is decoded as follows:

1. If the first word is non-zero, the data in both words is the run-time system name. The file size is limited to 65535 blocks.
2. If the first word is 0, the low order byte of the second word contains the most significant bits of the file size. The file size is limited to $2^{23}-1$ blocks. The high order byte of the second word is reserved and must be 0.

The following restrictions apply to large files:

1. Because a runnable file can not have both a run-time system name and a most significant byte indication in the same field, large files are not runnable.
2. You can not extend a compiled file beyond block 65535. An attempt to extend a compiled file past block 65535 results in a ?PROTECTION VIOLATION error (ERR=10).
3. You can not rename a file that is larger than 65535 blocks with the intent of assigning a compiled protection code. The attempt is rejected with no error and the compiled bit remains off.
4. When you extend a file past block 65535, it loses its run-time system name.
5. You can not change the run-time system name of a file that is larger than 65535 blocks. The attempt results in a ?PROTECTION VIOLATION error.
6. You can not change the run-time system name of a compiled file to two words of zeroes. The attempt results in a ?PROTECTION VIOLATION error. Note that you can perform this operation on a non-compiled file.
7. You can not change the run-time system name of an file to a zero word followed by a non-zero word.

To place a file in a particular position of the disk, you specify the desired disk DCN (Device Cluster Number) as returned in bytes 3 and 4 of this call in the file specification /POSITION switch (see the RSTS/E System User's Guide). The Monitor attempts to place the first block of the file at or after the specified DCN. If the file placement is successful, the placed bit (bit 1, mask value 2) in the file's UFD entry is set (see SYS calls -10 and -23). If the file placement is not successful, the first block of the file is placed at the lowest free block on the disk, the UFD placed bit is not set, and no error is returned.

This page is for notes.

Resident Library SYS Calls

Add a Resident Library - Privileged (F0--18)

Data Passed:

Byte(s)	Meaning
1	CHRS(6%), the SYS call to FIP
2	CHRS(-18%), the resident library manipulation code
3	CHRS(16%), add a resident library
4	Not used
5-6+	The project-programmer number of the file to add
7-10+	The resident library name in Radix-50 format
11-12	CHRS(A%)+CHRS(SWAP%(A%)), where A% is the 1K-word section of memory at which the resident library is to be loaded. The numbering begins at the first available 1K-word section and ends at n-1 (where n is the total number of 1K-word sections of memory on the system). A% must be specified and cannot be zero.
13-17	Not used
18	CHRS(S%), where S% is the stay flag. If S% is 128% (the high bit set), the resident library is made permanently resident. If S% is 0%, the memory occupied by the library can be freed for user space whenever the usage count of the library is zero (no active task is accessing the library).

- 19-20 CHR\$(F%) + CHR\$(SWAP\$(F%)), where F% is the flag word that defines the characteristics of the library. Only the high byte is used for flag bits. F% is the sum of the bits set, as follows:
- F%=256% Not used
- F%=512% The resident library is available to only one user. It is not shared by multiple users.
- F%=1024% The resident library allows read/write access to its memory, rather than read only access.
- F%=2048% The resident library does not record errors in its code in the system error log.
- F%=4096% The resident library is immediately removed from memory when its usage count equals zero.
- F%=8192% Not used
- F%=16384% Not used
- F%=32768%+1% Not used
- 21-22+ Protection code for the installed resident library. To specify a protection code, byte 21 is non-zero and byte 22 contains the desired protection code. To accept the default protection code, byte 21 is 0. The default protection code is <42>, which means that the Monitor grants read access to all users but denies write access.

- 23-24+ The name of the disk device on which the resident library is to be stored. If no name is specified, SY: is used.
- 25+ Unit number
- 26+ Unit number flag
- 27-30 Not used

Data Returned: No meaningful data is returned.

Possible Errors:

Meaning	ERR Value
?NO ROOM FOR USER ON DEVICE	4
You specified an address in bytes 11 and 12 that would cause the Monitor to load the library in such a manner that memory would be fragmented and a swapping violation would occur. Refer to the <u>RSTS/E System Generation Manual</u> for guidelines on avoiding memory fragmentation.	
?CAN'T FIND FILE OR ACCOUNT	5
You specified a filename in bytes 7 through 10 that cannot be found in the account specified in bytes 5 and 6 on the device specified in bytes 23 through 26. Ensure that the filename you specify has a .LIB extension and is located in the specified account and device.	
?PROTECTION VIOLATION	10
The file you wish to add is in improper format. For example, this error occurs if you specify a file that is not contiguous or has illegal entries in the SIL index.	

?NAME OR ACCOUNT NOW EXISTS 16

You specified the filename of a resident library that already exists.

?ILLEGAL BYTE COUNT OF I/O 31

You did not specify a load address in bytes 11 and 12 or the address you specified is not available. Refer to the memory status report of a display program to determine an available range of memory.

?NO BUFFER SPACE AVAILABLE 32

A small buffer is required for the description block of an added resident library. This error is returned if a small buffer is not available.

Discussion:

The Monitor SYS call to add a resident library is privileged and allows you to add a specified library to the Monitor's list of resident libraries. This call is similar to that used to add a run-time system except that you must specify a load address for the library. That is, unlike run-time systems, the Monitor does not automatically decide where to load a resident library. Also, the library file does not have to reside in account [0,1]; however, the file extension must be .LIB.

For additional information on the creation and use of resident libraries, refer to the RSTS/E Task Builder Reference Manual.

Remove a Resident Library - Privileged (P0=-18)

Data Passed:

Byte(s)	Meaning
1	CHRS(6%), the SYS call to FIP
2	CHRS(-18%), the resident library manipulation code
3	CHRS(20%), remove a resident library
4-6	Not used
7-10+	The resident library name in Radix-50 format
11-30	Not used

Data Returned: No meaningful data is returned

Possible Errors:

Meaning	ERR Value
?ACCOUNT OR DEVICE IN USE	3
You attempted to remove a library that is being loaded into memory or is in use by the currently running job. A resident library cannot be removed while a job is still attached to it.	
?CAN'T FIND FILE OR ACCOUNT	5
You specified a resident library name in bytes 7 through 10 that is not currently defined.	

Discussion:

The Monitor SYS call to remove a resident library is privileged and allows you to remove the library from physical memory, delete the Monitor structure that defines the library, and close the library file.

Load a Resident Library - Privileged (F0--18)**Data Passed:**

Byte(s)	Meaning
1	CHRS(6%), the SYS call to FIP
2	CHRS(-18%), the resident library manipulation code
3	CHRS(18%), load a resident library
4-6	Not used
7-10+	The resident library name in Radix-50 format
11-12	CHRS(A%) + CHRS(SWAP%(A%)), where A% is the 1K-word section of memory at which the library is to be loaded. The numbering begins at the first available 1K-word section and ends at n-1 (where n is the total number of 1K-word sections of memory on the system). If A% is 0%, the Monitor uses the load address that was initially set for the library when it was added. If A% is non-zero, the value specified replaces that initially set for the library.
13-17	Not used
18	CHRS(S%), where S% is the stay flag. If S% is 128% (the high bit set), the library is made permanently resident (i.e., the usage count is always non-zero). If S% is 0%, the memory occupied by the library is freed for user space whenever the usage count is zero.
19-30	Not used

Data returned: No meaningful data is returned

Possible Errors:

Meaning	ERR Value
7NO ROOM FOR USER ON DEVICE	4
You specified an address in bytes 11 and 12 that would cause the Monitor to load the library in a position where memory would be fragmented and a swapping violation would occur. Refer to the <u>RSTS/E System Generation Manual</u> for guidelines on assigning and allocating memory to avoid fragmentation.	
7CAN'T FIND FILE OR ACCOUNT	5
You specified a filename in bytes 7 through 10 for a resident library that is not currently defined.	
7ILLEGAL BYTE COUNT FOR I/O	31
The load address was not specified in bytes 11 and 12 or the address specified is not available. Refer to the memory status report of a display program to determine an available range of memory.	

Discussion:

The Monitor SYS call to load a resident library is privileged and allows you to specify a position in memory for the library. That is, you can use this call to move a resident library from its current position in memory to a position of your choice. Note that if you wish to move a library to a permanent position in memory, you must set the stay bit (see format description). The stay bit ensures that the library will remain in the specified location for the duration of the current time-sharing session.

Unload a Resident Library - Privileged (F0=-18)

Data Passed:

Byte(s)	Meaning
1	CHRS(6%), the SYS call to FIP
2	CHRS(-18%), the resident library manipulation code
3	CHRS(22%), to unload a resident library
4-6	Not used
7-10+	The resident library name in Radix-50 format
11-30	Not used

Data Returned: No meaningful data is returned.

Possible Errors:

Meaning	ERR Value
7ACCOUNT OR DEVICE IN USE	3
You attempted to unload a resident library that is in the process of being loaded or is in use by the currently running job. A library cannot be unloaded while a job is still attached to it.	
7CAN'T FIND FILE OR ACCOUNT	5
You specified a filename in bytes 7 through 10 for a library that is not currently defined.	

DIGITAL

RSTS/E V7.0 SWS UPDATE
Software and RSTS/E V7.0

2-1179

RSTS/E V7.0 SWS UPDATE
Software and RSTS/E V7.0

Discussion:

The Monitor SYS call to unload a resident library is privileged and allows you to remove the library from memory and free that portion of memory for use by other jobs.

Implementing New RSTS/E Features under BASIC-PLUS

Creating and Accessing Large Files

- o Pre-extending a large file

1000 OPEN "file.ext/S1:70000 "FOR OUTPUT.....

Note

The switch on the file specification must be used since the FILESIZE option of the OPEN statement accepts only integers.

- o Reading from and writing to a large file

GET #N%, BLOCK X

PUT #N%, BLOCK X

Note

The BLOCK modifier accepts floating point numbers. RECORD modifier accepts integers.

Specifying Data Caching

Note

Data caching must be configured into the monitor.

Data Caching

When a user job executes a read request, the monitor performs a disk access and transfers the requested data from the disk to the user job's buffer. On systems that have many jobs requiring large amounts of data, the large number of disk data transfers can have an impact on response time.

Data caching is a method whereby a privileged user can reduce data transfers from disk. When caching is enabled, the monitor stores the most recently read (accessed) data blocks in an area of memory called the cache. If the user job then requests a data block that is present in the cache, the monitor copies the requested data directly from the cache into the job's buffer. Thus, the monitor avoids a physical disk access operation.

Data caching is most useful for read operations in that it can minimize disk transfers. In a write operation that modifies existing data, the data is updated in the cache and on disk but no new data is installed in the cache.

The system manager configures caching on the system and optionally sets its parameters during system initialization. A privileged user enables or disables caching and determines the size of the cache by means of the disk cache enable SYS call or the UTILTY system program. The SYS call or UTILTY are also used to specify caching for a particular file by marking the file's UFD entry. If a new file is to be created and accesses to it are to be cached, use the MODE values in the OPEN statement. However, because the use of caching MODE values is privileged, a privileged user should designate a file for caching by means of the file's UFD entry (i.e., use the file utility SYS call or UTILTY).

Once a file's UFD is marked for caching, it will be cached on OPEN regardless of the user's privilege as long as caching is enabled on the system.

When caching is enabled the cache receives all data transfer requests that are directed to the disk driver. Because the cache contains data, read operations on data that is in the cache occur without placing a load on the disk driver. The monitor constantly updates the cache such that it contains the most recently requested data.

During system initialization (or with the disk cache enable sys call), the system manager sets the cache cluster size. The cluster size (1, 2, 4, or 8 blocks) determines the smallest amount of data transferred into the cache on a disk access. For example, when the cache cluster size is set at 8 blocks, a read operation that installs data in the cache causes the monitor to determine the number of cache clusters needed to fulfill the request. If a single cache cluster is required, 8 physically contiguous blocks (including the requested blocks) are installed. If more than one new cache cluster is required to full fill the request, a separate disk access is performed to install the data into each of the required cache clusters. However, if the number of new cache clusters required is more than two, the data is not installed in the cache but is transferred directly into the user buffer with a single disk access. Note that the cache cluster size should not usually be greater than the pack cluster size that was set during disk initialization. The Monitor allocates space for the cache from the Extended Buffer Pool (XBUF).

User Data Caching (Random Mode) - MODE 256

A privileged user can specify MODE 256 in an OPEN statement to cause data transfers from a file to be cached in random mode. MODE 256 has effect only if data caching is enabled on the system.

The monitor schedules data transfers into the cache on the basis of time since last access. That is, new data replaces data currently in the cache if the currently cached data:

1. is the data with the longest time since last access, and
2. has been in the cache for more than the minimum residency as established by the system manager.

If a data block meets these requirements, it is replaced when the monitor seeks to place a new data block in the cache.

As an example of random cache operation, consider a get statement that is executed on a file opened with mode 256%. When a read on the cached file occurs, the monitor examines the contents of the cache to determine if the requested data item is present. If the data is in the cache, the data is copied from the cache cluster that contains it. The data is made available to the program and the cache cluster is linked to the beginning of the list of cache clusters. The list of cache clusters is maintained in order of increasing time since last access.

If the requested data item is not present in the cache, the monitor examines the list of cache clusters to determine the time of last access for the oldest item in the cache. If the time is less than the minimum residency, the requested data cannot be installed in the cache and a normal disk read is performed. If the time is greater than the minimum residency, the requested data is installed in that cache cluster and made available to the program.

User Data Caching (Sequential Mode) - MODE 2048%

MODE 2048% has effect only if the file is being cached. That is, MODE 256% is set, the file's UFD entry is marked for caching, or caching is set for all data on the system. Note that sequential mode data caching requires a cluster size greater than 1, although no error is returned if the cluster size is 1.

When MODE 2048% is specified in an OPEN statement, data blocks from that file, placed in the cache as a result of that OPEN, are not subject to the minimum residency. However, MODE 2048% overrides the minimum residency only for that file's data blocks, not for the entire cache. That is, when data blocks from a sequentially cached file are placed in the cache, a read on the last block of the cluster causes it to become the "oldest" data block in the cache. It is the first to be replaced when the monitor seeks to place new data blocks in the cache. MODE 2048% has the effect of speeding the replacement process in the cache and thus minimizes the space required by the cache.

As an example of sequential cache operation, consider a GET statement that is executed on a file opened with MODE 2304%(256% + 2048%). When a read on the cached file occurs, the monitor examines the contents of the cache to determine if the requested data item is present. If the data is in the cache, the data is copied from the cache cluster that contains it. The data is made available to the program. If the data is in the last block of a cache cluster, that cluster is made available for replacement in the cache.

When a read operation is performed on the first block of a cache cluster, the monitor installs a full cluster of data in the cache. That is, if the system manager sets a clustersize of 8 for the cache, the requested data block plus the next 7 contiguous blocks are read from the disk into the cache. When the last block of the cache buffer is read, the cluster is immediately made available for the installation of new data as requested by other read operations.

If the GET statement causes more than one cache cluster to be read, all of the requested data blocks are made available to the program but only the last cache cluster is installed. However, if the last data block read is the last block in a cache cluster, no data is installed in the cache. Thus, if the cache cluster size is defined as 1 and sequential mode is specified, no data blocks will be installed in the cache (i.e., every data block is the last block in a cache cluster).

Table SOP-1. (Page 1 of 2)

Manager Specification through UTILTY	Non-Privileged Programming Options	
	Random Cache	Sequential Cache
DISABLE CACHE	No data caching * available	
ENABLE CACHE/ALL	Open file with no data caching modes	Open file with MODE 2048
ENABLE CACHE/FILE	No data caching * available	
FLAG file/CACHE**	Open file with no data caching modes	Open file with MODE 2048
FLAG file/NOCACHE**	No data caching * available	
FLAG file/CACHE/SEQ**	No random cache available	Open file with No data cache mode
FLAG file/CACHE/RAN	Open file with no data caching modes	Open file with mode 2048

* If data caching modes are specified, no error is issued.

** These commands set and unset data caching bits in the disk directory structure for the file.

Table SOP-1. (Page 2 of 2)

Manager Specification through UTILTY	Privileged Programming Options	
	Random Cache	Sequential Cache
DISABLE * CACHE	1. Enable cache SYS call ** 2. Open file with MODE 256 ***	1. Enable cache SYS call ** 2. Open file with MODE 256 + 2048 ****
ENABLE CACHE/ALL	Open file with no data caching mode	Open file with MODE 2048
ENABLE CACHE/FILE	Open file with MODE 256 ***	Open file with MODE 256 + 2048 ****
FLAG file/CACHE	Open file with no data caching mode	Open file with MODE 2048
Flag file/NOCACHE	Open file with MODE 256	Open file with MODE 256+2048
FLAG file/CACHE/SEQ	No random caching available *****	Open file with no data caching mode.
FLAG file/CACHE/RAN	Open file with no data caching mode	Open file with MODE 2048

* USE LIST CACHE command to check if disabled.

** Enables data caching for the system.

- *** The FLAG file/CACHE UTILTY command or the file utility SYS call can be used, but this makes random and sequential caching of the file available to all users.
- **** The FLAG file/CACHE/SEQ UTILTY command or the file utility SYS call can be used, but the file will be cached sequentially for anyone using the file.
- ***** The FLAG file/RAN UTILTY command or the file utility SYS call can be used.

Extending Files Opened in Update Mode

- o To extend the file:
 - a. Open the file in update mode (MODE 1 or MODE 5)
 - b. Read in the first block of the file.
 - c. Explicitly lock the first block using the SPEC\ function.
 - d. Extend the file by writing beyond the end of file.
 - e. Release the explicit lock on block 1 using the SPEC\ function, the UNLOCK statement or the CLOSE statement.

Note

As extended blocks are written, they are made available to other users.

Programs attempting to read block 1 during this process receive ?Disk Block is Interlocked errors.

- o Example

Create the file

```

1000 OPEN "UPDATE.DAT<48>" FOR OUTPUT AS FILE 11 &
      \FIELD 11, 20 AS N.AMES
1010 FOR I1 = 11 TO 21 &
      \INPUT "RECORD"; N5 &
      \LSET N.AMES = N5 &
      \PUT I1 &
      \NEXT I1
32767 CLOSE 11 &
      \END

```

Ready

Figure SOF-7. Create the File

Look at Directory

```
DIR UPDATE.DAT/S
Name .Ext Size Prot Access Date Time Clu RTS Pos Op/rr
SY:[100,7]
UPDATE.DAT 2 < 48> 28-Nov-78 28-Nov-78 12:32 PM 8 BAS4P 23121 8/8
```

Total of 2 blocks in 1 file in SY:[100,7]

Ready

Figure SOF-8. Look at Directory

Extend the file

```
1800 ON ERROR GOTO 19000
1810 OPEN "UPDATE.DAT" AS FILE 10, MODE 50
1815 FIELD #10, 200 AS N.AMES
1820 GET #10, BLOCK 1 &
      \10 = SPEC0(10,00,10,00) IEXPLICIT LOCK ON BLOCK 1 &
1830 LSET N.AMES = "NEW RECORD" &
      \PUT #10, BLOCK 170
1840 10 = SPEC0(30,10,10,00) IRELEASE BLOCK 1 &
      \GOTO 32767
19000 PRINT ERR &
      \RESUME 32767
32767 CLOSE 10 &
      \END
```

Ready

Figure SOF-9. Extend the File

Look at Directory

```
DIR UPDATE.DAT/S
Name .Ext Size Prot Access Date Time Clu RTS Pos Op/rr
SY:[100,7]
UPDATE.DAT 17 < 48> 28-Nov-78 28-Nov-78 12:32 PM 8 BAS4P 23121 1/80
```

Total of 17 blocks in 1 file in SY:[100,7]

Ready

Figure SOF-10. Check Directory

Placed files

o Specify the device cluster number on the /POSITION switch of the file specification.

```
LISTEN
1800 EXTEND I EXTEND MODE 0
      \ ON ERROR GOTO 19000 ! EPROR TRAPPER 0
1810 INPUT "FILENAME.EXT";FILS I GET NAME.EXT 0
      \ INPUT "FILESIZE(BLOCKS)";SIZ0 I GET SIZE 0
      \ INPUT "DISK POSITION";DIPS I GET POSITION 0
      \ SPES=FILS*(48)*"/POSITION;"*DIPS I SET UP SPIC 0
      \ OPEN SPES FOR OUTPUT AS FILE 10. I OPEN SPIC 0
      \ FILESIZE SIZ0,MODE 160 I MODE AND FILESIZE 0
      \ CLOSE 10 I CLOSE IT 0
1820 PRINT "FILE HAS BEEN CREATED!" I DONE 0
      \ GOTO 32767 I EXIT 0
1830 PRINT "NO ROOM ON DISK FOR FILESIZE "+ I NOT ENOUGH ROOM 0
      \ NUM0(SIZ0)*" TRY AGAIN" I TELL USER 0
      \ GOTO 1810 I GO TRY AGAIN 0
19000 IF ERR=40 THEN RESUME 18300 I NO DISK SPACE 0
19010 ON ERROR GOTO 0 I DISABLE TRAP 0
32767 END
Ready
```

Figure SOF-11. Placed File

Using Tentative Files

- o Specify MODE 32 in an OPEN FOR OUTPUT statement.

```
OPEN "file.ext" FOR OUTPUT AS FILE 1%, &
MODE 32%
```

- o System Actions

If the file specified already exists, the tentative file:

- is "marked for deletion"
- shows only in a slow listing of DIRECT
- Example

```
DIR UPDATE.*
Name .Ext Size Prot Date ST:[100,7]
UPDATE.DAT 17 < 48> 28-Nov-78
```

total of 17 blocks in 1 file in ST:[100,7]

Ready

```
DIR UPDATE.*/*S
Name .Ext Size Prot Access Date Time Cld RTS Pos Op/rc
ST:[100,7]
UPDATE.DAT 17 < 48> 28-Nov-78 28-Nov-78 12:32 PM 0 DAS4P 23121 0/0
UPDATE.DAT* 0 < 48> 28-Nov-78 28-Nov-78 02:54 PM 0 DAS4P ----- 1/0M
```

total of 17 blocks in 2 files in ST:[100,7]

Ready

Figure SOF-12. Tentative Files

When the tentative file is explicitly CLOSED, the original file is deleted and the tentative file is renamed.

Note

If the tentative file is not explicitly CLOSED, is CLOSED with a negative channel number or if the system should crash, the original file remains intact and the tentative file is deleted.

- o Example

```
1000 OPEN "BOOK.LST" FOR OUTPUT AS FILE 1%, MODE 32% &
      \OPEN "BOOK.LST" AS FILE 2%, RECORDSIZE 32767%+1%-2%
1010 FIELD #1%, 30% AS BOOK.TITLE% &
      \GET #2% + SWAP%(1%) &
      \LSET BOOK.TITLE% = "PDP-11 COBOL REFERENCE MANUAL" &
      \PUT #1%
1020 CLOSE 1%
32767 END
```

Figure SOF-12A. Tentative File Program

DIGITAL

RSTS/E V7.0 SMS UPDATE
Software and RSTS/E V7.0

000000

RSTS/E V7.0 SMS UPDATE
Software and RSTS/E V7.0

Making Contiguous Files Non-Contiguous

OLD NONCNT

Ready

```

LISTMH
130  EXTEND
500  ON ERROR GOTO 19000 &
     \ES = -1&
1000 OPEN "CONT.DAT" AS FILE 1& &
     \FIELD #1&, 20& AS N.AMES
1010 INPUT "NAME": N& &
     \LSET N.AMES = N&
1020 PUT #1&, BLOCK 20 &
     \GOTO 32766
2000 Y& = SYS(CHRS(6&)+CHRS(-26&)+CHRS(1&)+CHRS(32&)) &
     \GOTO 1020
19000 IF ENL = 1020 &
     THEN PRINT "ERROR NO.": ERR: "WITH ES = "; ES &
     \ES = 0& &
     \RESUME 2000
19010 CLOSE 1& &
     \ON ERROR GOTO 0
32766 CLOSE 1&
32767 END

```

Ready

```

DIR CONT.DAT/S
Name .Ext Size Prot Access Date Time Clu RTS Pos Op/rr
ST:[1,235]
CONT .DAT 16C < 60> 22-Mar-79 22-Mar-79 11:31 AM 8 BASRF 13702 0/0
Total of 16 blocks in 1 file in ST:[1,235]

```

Ready

```

RUN NONCNT
NAME? HARRY JAMES
ERROR NO. 10 WITH ES = -1

```

Ready

```

DIR CONT.DAT/S
Name .Ext Size Prot Access Date Time Clu RTS Pos Op/rr
ST:[1,235]
CONT .DAT 20 < 60> 22-Mar-79 22-Mar-79 11:31 AM 8 BASRF 13702 0/0
Total of 20 blocks in 1 file in ST:[1,235]

```

Ready

Figure SOF-13. Contiguous Files

Using Non-File Structured Block Mode

- o Available on large file systems only
- o Specify MODE 128 on OPEN
- o Each GET/PUT using the BLOCK modifier accesses a logical block rather than a device cluster.

Note

Block number 1 is the first block of the MFD on RSTS/E disks.

```

1000 EXTEND          | EXTEND MODE  &
      \ ON ERROR GOTO 10000 | ERROR TRAPPER &

1005 FOR 21-10 TO 90 | CAR IDENT'S  &
      \ CARS(10)-CARS(20) NUMS(10) FOR 24-10 TO 30 | THREE EACH  &
      \ NEXT 10 | 9 CAR IDENT'S &
      \ CARS(20)='000' | AND THE TENTH &
      \ FORMLINES='000 \'+SPACES(30)+'\'+ | FORMATS &
      " 0000 $500.00 00 $50,000.00" | REST OF IT &
      \ HEADS='CAR MAKE & MODEL & STYLE'+ | HEADER &
      SPACES(120)+' PART PRICE OR EXTENDED VALUE'+ | MORE OF SAME &
      \ VALFORS='TOTAL VALUE'+SPACES(40)+'$50,000,000.00'+ | VALUE FORMAT &

1010 INPUT 'READ FROM WHICH DISK(DON:)?';DK1 | GET INPUT DISK &
      \ OPEN DSKS AS FILE 10,MODE 1200 | BLOCK MODE NFS &

1020 FIELD #10,20 AS CAR.IDS, | KEY FIELD &
      21 AS PART.NOS, | PART CODE &
      300 AS DESCS, | DESCRIPTION &
      80 AS UNIT.PRIS, | PRICE EACH &
      21 AS QUAN.AVGS, | QUANTITY OR BAND &
      4640 AS RESIS, | REST OF RECORD &
      20 AS LST.BLKS, | LAST BLK THIS ID &
      20 AS NXT.BLKS | NEXT BLK THIS ID &

1030 INPUT 'CAR ID CODE?';CARIDS | GET WHAT TO READ &
      \ FOR X4=10 TO 100 | START VALIDATION &
      \ GOTO 1040 IF CARS(X4)-CARIDS | FOUND IT &
      \ NEXT X4 | .TRY AGAIN &
      \ PRINT '*? INVALID CAR ID CODE.' | WRNCO &
      \ GOTO 1030 | TRY AGAIN &

1040 BLK=X4+10 | STARTING BLOCK &
      \ INPUT 'NUMBER OF RECORDS TO PRINT ?';N.RECS | PRINT COUNT &
      \ INPUT 'PRINT REPORT TO ?';REPOS | OUTPUT DEV/FILE &
      \ OPEN REPOS AS FILE 20 | REPORT FILE &
      \ PRINT 020,HEADS | PRINT HEADER &
      \ PRINT 020 | BLANK LINE &

1050 FOR C10=10 TO N.RECS | LOOP FOR IREC &
      \ GET #10,BLOCK BLK | GET TYPE 1ST BLK &
      \ GOTO 1070 IF CVT50(NXT.BLKS)=-10 | END OF BLOCKS &
      \ Q1=CVT50(QUAN.AVGS) | CONVERT QUANTITY &
      \ P=CVT50(UNIT.PRIS) | CONVERT PRICE &
      \ V=P*Q1 | COMPUTE VALUE &
      \ PRINT 020 USING FORMLINES, | FORMAT &
      CVT50(CAR.IDS), | PRINT CAR TYPE &
      DESCS+' ', | DESCRIPTION &
      CVT50(PART.NOS), | PART NUMBER &
      P, | UNIT PRICE &
      Q1, | QUANTITY &
      V | COMPUTED VALUE &
      \ CNT0=CNT0+1 | BUMP BLOCK COUNT &
      \ VALUE=VALUE+V | SUM UP VALUES &
      \ BLK=CVT50(NXT.BLKS) | NEXT BLOCK &
      \ NEXT C10 | NEXT PLEASE &
      \ BLANK LINE &
  
```

Figure SOP-14. NFS Block Mode (Page 1 of 2)

```

1040 PRINT 020,SPACES(100)+'-----' | PRINT DIVIDER &
      PRINT 020,020,020,020,020,020 | PART &
      PRINT 020,TOTAL,REPOS,020,020 | NUMBER BUS &
      \ GOTO 1070 | CLOSE FILES &
      \ GOTO 1070 | END IT &
  
```

```

10700: ERROR HANDLING &

10810 IF ERR=110 AND ERR=10300 THEN RESUME 1090 | ABORT JOB &

10999 ON ERROR GOTO 0 | JUST IN CASE &
  
```

12767 END

Ready

```

RUN
READ FROM WHICH DISK(DON:)? DK1:
CAR ID CODE ? 888
NUMBER OF RECORDS TO PRINT ? 15
PRINT REPORT TO ? FB:
CAR MAKE & MODEL & STYLE
  
```

CAR MAKE & MODEL & STYLE	PART	PRICE	QU	EXTENDED VALUE
888 FIAT 124 4DR SEDAN PART	4511	\$78.89	67	\$5,265.63
888 FIAT 124 4DR SEDAN PART	4397	\$72.49	39	\$2,826.72
888 FIAT 124 4DR SEDAN PART	4211	\$61.36	89	\$5,461.04
888 FIAT 124 4DR SEDAN PART	2066	\$21.74	44	\$956.56
866 FIAT 124 4DR SEDAN PART	3785	\$98.63	51	\$5,930.13
886 FIAT 124 4DR SEDAN PART	7165	\$33.75	57	\$1,923.75
868 FIAT 124 4DR SEDAN PART	4156	\$30.90	11	\$339.90
868 FIAT 124 4DR SEDAN PART	9590	\$17.88	44	\$786.72
806 FIAT 124 4DR SEDAN PART	5555	\$71.59	47	\$3,364.73
800 FIAT 124 4DR SEDAN PART	2223	\$73.09	38	\$2,777.42
868 FIAT 124 4DR SEDAN PART	3087	\$4.40	48	\$211.20
888 FIAT 124 4DR SEDAN PART	5899	\$36.94	90	\$3,324.60
888 FIAT 124 4DR SEDAN PART	7410	\$43.52	94	\$4,970.88
888 FIAT 124 4DR SEDAN PART	4605	\$41.55	34	\$1,412.70
888 FIAT 124 4DR SEDAN PART	4769	\$94.81	39	\$3,697.59

```

TOTAL VALUE
TOTAL RECORDS: 15
$41,489.57
  
```

Ready

Figure SOP-14. NFS Block Mode (Page 2 of 2)

RMS-11 V1.8 and RSTS/E V7.0

General

DEFERRED WRITE

- o Available to user through MACRO and COBOL
- o Used by RMSCNV utility automatically

MASS INSERT

- o Available to user through MACRO and COBOL
- o Used by RMSCNV utility through a switch

NEW DATA TYPES FOR KEYS

- o Packed decimal
- o 2 and 4 byte signed and unsigned binary

FILE SPECIFICATION

- o Standard RSTS/E file specification without switches supported

OVERLAY DESCRIPTION FILES

- o RMSRLS.ODL Resident Library for sequential files only
- o RMSRLX.ODL Resident Library for sequential/relative/indexed files
- o RMS11S.ODL Disk Library for sequential/relative files

- o RMS11X.ODL Disk Library for sequential/relative/indexed files. (9Kb)
- o RMS12X.ODL Disk Library for sequential/relative/indexed files. (11Kb)

Implementing New RSTS/E Features under RMS-11 V1.8

Table SOF-2.

Feature	How Supported
Large files	<ul style="list-style-type: none"> o Pre-extension through RMSDEF. o Explicit or implicit extension of non-contiguous files.
Data Caching	Through UTILTY
Extending non-contiguous files Opened in update mode	RMS-11 automatically handles
Placed files	Through RMSDEF
Tentative files	Unsupported
Making contiguous files non-contiguous	Through UTILTY
Non-file structured Block Mode	Unsupported

BASIC-PLUS-2 V1.60 and RSTS/E V7.0**Commands****SHOW - V1.60**

- o Lists the current compiler defaults
- o Example

```

RSTS/E BASIC PLUS 2 V01.60 using PPM - 27 -
Installed on 22-Nov-79 at 23:00PM
Mixed for RUM or /TSC to FPACC2
Mixed for BUILD to LB:FPACOM
BASIC-2 Resident Library to none
Disk Library to LB:FPACOM
RMS DLL File to LB:RMSRSL
PMS Resident Library to LB:RMSRES
Data extend size = 120K
Scale factor = 3
Precision = Single
System settings:
  %D:RSP
  %D:DUMP
  %D:CHRG
  %D:LINE
  %D:DEBUG
Output:ISK with override to OBI
Rate arg:Basic Plus compatible only
FPACC2

```

Figure SOF-15. Show Command

ASSIGN/REASSIGN

- o ASSIGN DEV:[ppn]LIB Associates the logical name LIB with a device and account number

BRLRES

- o Allows the BASIC-PLUS-2 resident library to be linked to your program during task building
- o Example

```

BRLRES
FILE SPEC [NONE] -- BASICS

```

DSKLIB

- o Allows the BASIC2 or BP2COM disk library to be linked to your program during task building
- o Example

```

DSKLIB
FILE SPEC [LB:BASIC2]--

```

ODLRMS

- o Allows selection of RMS-11 ODL files
- o Example

```

ODLRMS
FILE SPEC [LB:RMS11S] -- LB:RMSRLX

```

RMSRES

- o Allows an RMS-11 resident library to be linked to your program during task building
- o Example

```

RMSRES
FILE SPEC [NONE] -- RMSRES

```

BUILD

- o Accepts new switches

```

/BRLRES:filespec
/DSKLIB:filespec
/ODLRMS:filespec
/RMSRES:filespec
/EXTEND:n

```

DIGITAL

RSTS/E V7.0 CUSP UPDATE
Software and RSTS/E V7.0

RSTS/E V7.0 CUSP UPDATE
Software and RSTS/E V7.0

Statements

PRINT, RECORD

- o PRINT with RECORD option supported

EXTEND/NO EXTEND - V1.5

- o Recognized by compiler
- o No action taken
- o Allows RSTS/E CUSP compatibility

CLOSE -N

- o Supported for files opened as tentative with a MODE switch on the file specification

System Function Calls

- o All the following new and enhanced system function calls are supported
 - File Name String Scan
 - Get Monitor Tables II
 - Change System Date/Time
 - Directory Lookup
 - Assign/Reassign
 - Deassign
 - Wild card PPN Lookup
 - Enable/Disable Disk Caching
 - File Utility
 - Resident Library
 - Return Job Status
 - Create a Job
 - Exit and Clear Program
- o Unsupported System Function Call
 - Execute CCL

Implementing New RSTS/E Features under BASIC-PLUS-2 V1.60
Table SOP-3.

Feature	How Supported
	Native Mode * RMS
Large Files	Same as BASIC-PLUS (Virtual arrays limited to 65535 blocks) o Pre-extension through RMSDEF. o Explicit and implicit extension of non-contiguous files
Data Caching	Through UTILTY or SYS calls
Extending non-contiguous files Opened for Update Placed files	Same as BASIC-PLUS RMS automatically handles
Tentative files	Same as BASIC-PLUS Through RMSDEF or TEMPORARY Keyword**
Making contiguous files non-contiguous	Through UTILTY or SYS call
Non-file structured Block Mode	Same as BASIC-PLUS Unsupported

* BASIC-PLUS compatible

** New handling for RSTS/E V7.0

Using the RMS Resident Library

- o Preliminary requirements
 - a. Resident library support must be configured into the monitor.
 - b. The resident library must have the correct protection code.
 - c. The resident library must be added to the system via UTILTY or the system function call.
 - d. RSX directive emulation in the monitor is not required by preferred.
- o Procedures for using the resident library
 - a. BUILD and task build procedures

```

000 00000
FPUIC2

LISTEN
100
300
450
500
32767
FPUIC2
NISEG
Name (FPUICM)--NONE
Account (LB:)--
FPUIC2
RMSRES
File spec (LB:RMSRES)--
FPUIC2
ODLRMS
File spec (LB:RMSRLX)--
FPUIC2

BUILD I1020/1ND
FPUIC2
COM/08J/000
FPUIC2
TRB 011020
FPUIC2

RUN I1020
IDENTIFICATION? NALS
CODE? 16P
LOCATION? C35
AMOUNT? 33220
ANOTHER RECORD? YES
IDENTIFICATION? NALS
CODE? 10P
LOCATION? 339
AMOUNT? 56745
ANOTHER RECORD? NO

```

Figure SOF-16. BASIC-PLUS-2 (page 1 of 2)

DIGITAL

RSTS/E V7.0 SMS UPDATE
Software and RSTS/E V7.0

DIGITAL

RSTS/E V7.0 SMS UPDATE
Software and RSTS/E V7.0

COBOL V4 and RSTS/E V7.0

General

PERFORMANCE

- o Improvements for CPU bound tasks through COMPUTATIONAL data formats.

Note

I/O bound tasks are affected very little.

DATA FORMATS

- o COMP -- PDP-11 binary format
- o COMP-6 -- Called COMP under previous releases
- o COMP-3 -- Packed Decimal

FPUC2

DSP ONLY.MAS

```

SY: (120, 120) DLT.MAS FILE ORGANIZATION: INDEXED
PROCEDURE VERSION NUMBER: 1
CREATED: 22-MAR-1979 09:53
FILE PROTECTION: 400
RECORD FORMAT: FIXED-52
RECORD ATTRIBUTES: CARRIAGE RETURN
FILE ATTRIBUTES:
ALLOCATION: 1 EXTEND QUANTITY: 0 BUCKET SIZE: 1
NUMBER OF SETS: 2

```

FPUC2

PIP

```

*1102*.CMD
ST: 1102/FP-ST: 1102/HP
UNITS = 12
ADD + ST: 5: 8: 7: 8: 9: 10: 11: 12
RESLID=LB:RMSRES/RO
EXTTSK= 102*
**
*1102*.DDL
ROOT BASIC2-RMSFOT-USER,RMSALL
USER: .FCTR ST: 1102*-LIBR
LIBR: .FCTR LB: FPUCOM/LB
PLB: FPUC2
PLB: RMSRLE
**2
.END

```

FPUC2

Figure SOP-16. BASIC-PLUS-2 (page 2 of 2)

Removal of Data Item Length Limitation

Previous to this version, the maximum length of any data item (group item, elementary item, or table) was 4095 characters. This limitation has been removed. Data item sizes are therefore limited only by the amount of memory available to the executable task and with the condition that all Data Division entries must be co-resident (non-overlayable).

Improved OTS Error Messages

COBOL-11, Version 4.8, includes a completely new object-time system. Error messages have been revised to provide clearer and more complete information. The new messages replace those contained in Appendix J of the PDP-11 COBOL User's Guide, Order No. AA-1757C-TC. The next revision of the User's Guide will include the new messages; however, they have been designed to require no additional reference.

Error message 15:

CBL -- 15: STOP RUN

no longer appears at program termination.

COBOL Interactive Debugger (CID)

The COBOL Interactive Debugger (CID) is an interactive debugging aid released with this version of PDP-11 COBOL. By using CID, which you include during the task-build process, you can:

- o Examine and change data in your program during execution.
- o Set and cancel breakpoints.

- o Change program flow.

CID allows you to find program errors without recompiling or changing your program. When you want to use CIF, you need only task-build the program with the CID module, then execute it.

Statements

ACCEPT

The ACCEPT identifier FROM DAY statements has been changed to agree with its description in the PDP-11 COBOL Language Reference Manual. Therefore, section 6.7.1 of the PDP-11 COBOL User's Guide will be revised to reflect this change.

Implementing New RSTS/E Features under COBOL V4

Table SOP-4.

Feature	How Supported
Large Files	<ul style="list-style-type: none"> o Pre-extension through RMSDEF. o Explicit or implicit extension of non-contiguous files.
Data Caching	Through UTILTY
Extending non-contiguous files Opened in Update Mode	RMS-11 automatically handles
Placed Files	Through RMSDEF
Tentative Files	Unsupported
Making Contiguous Files Non-Contiguous	Through UTILTY
Non-File Structured Block Mode	Unsupported

Using the RMS Resident Library

- o Preliminary requirements
 - a. Resident library support must be configured into the monitor.
 - b. The resident library must have the correct protection code.
 - c. The resident library must be added via UTILTY.
 - d. RSX directive emulation in the monitor is not required but preferred.
- o Procedures for using the resident library.
 - a. Merge and task build procedures

```

RUN *MPG
PLEASE ENTER FILE SPECIFICATION FOR OUTPUT FILE
IALLIR.ODL
DO YOU WANT A DEFAULT MERGE?
PLEASE ANSWER (YES), (NO), OR (HELP) N

DO YOU WANT AN ABBREVIATED OR MERGED ODL FILE?
PLEASE ANSWER (ABBREVIATED), (MERGED), OR (HELP) A

DO YOU WANT TO INCLUDE THE COMAL DEMONSTR (CID)?
PLEASE ANSWER (YES) OR (NO) N

DO YOU WANT TO OVERLAY I/O SUPPORT ROUTINES
OR USE THE RESIDENT I/O LIBRARY?
PLEASE ANSWER (YES), (NO), OR (HELP) Y

DO YOU WANT TO USE THE RESIDENT I/O LIBRARY, THE DEC
SUPPLIED I/O OVERLAY OF YOUR OWN I/O ODL?
PLEASE ANSWER (I/O OVERLAY), (OWN OVERLAY), (RESIDENT LIBRARY),
OR (HELP) P
PLEASE ENTER FILE SPECIFICATION FOR INPUT ODL FILE
IALLIC
OBJECT PROGRAM REFERENCED IN ODL FILE IS:
IALLIC.OBJ
PLEASE ENTER OBJECT FILE DEVICE AND PPN IN THE FORMAT: DEVICE:PROJECT:PROGRAMMER]

ANY MORE INPUT ODL FILES?
PLEASE ANSWER (YES) OR (NO) Y
PLEASE ENTER FILE SPECIFICATION FOR INPUT ODL FILE
SLFER
OBJECT PROGRAM REFERENCED IN ODL FILE IS:
SLFER.OBJ
PLEASE ENTER OBJECT FILE DEVICE AND PPN IN THE FORMAT: DEVICE:PROJECT:PROGRAMMER]

ANY MORE INPUT ODL FILES?
PLEASE ANSWER (YES) OR (NO) N
ODL FILE MERGE COMPLETE
MERGED ODL FILE IS: IALLIR.ODL

Ready

```

Figure SOF-17. COBOL Merge (Resident Library)

```

TY IALLIR.ODL
MERGED ODL FILE CREATED ON 23-MAR-79 AT 07:32:28
@IALLIC.ODL
@SLFER.ODL
CJOBJS: .FCTR IALLIC.OBJ-SLFER.OBJ-LB:COBOVR/LB:UIORSX
CBOTS: .FCTR LB:COBOVR/LB
      .NAME OTS%OV
CNVRT: .FCTR LB:COBOVR/LB:CONVER
LDCVT: .FCTR LB:COBOVR/LB:LDCVT
USRIO: .FCTR LB:COBOVR/LB:USRSIO
OTSOV: .FCTR OTS%OV-(CNVRT%-(LDCVT%)*USRIO%)
@LB:RMSRLX.ODL
RMS: .FCTR RMSROT-RMSALL
OBJRT: .FCTR CJOBJS-CBOTS-RMS
      .ROOT OBJRT%,OTSOV%
      .END

```

Ready

```

TKB
TKB>TKB IALLIR,IALLIR=IALLIR/MP
ENTER OPTIONS:
TKB>LIBR=RMSRES:RO
TKB>>>

```

Ready

Figure SOF-18. COBOL ODL and Task Build (Resident Library)

DIGITAL

RSTS/E V7.8 SWS UPDATE
Software and RSTS/E V7.8

SOFTWARE V7.8 SWS UPDATE
Software and RSTS/E V7.8

TRIAL.TSR MEMORY ALLOCATION MAP TAB 07.005 PAGE 13
23-4AK-79 07:36

*** SEGMENT: USHSIO

R/W MEM LIMITS: 061144 066113 004750 02536.
DISK BLK LIMITS: 000076 000102 000005 00005.

MEMORY ALLOCATION SYNOPSIS:

SECTION	TITLE	IDENT	FILE
PLK: (R, I, LCL, REL, COM)	061144	000000	00000.
BUSHIO: (RU, I, GBL, REL, UVN)	061144	004750	02536.
SSALYC: (R, D, LCL, REL, COM)	061144	000000	00000.
SSRTS: (R, I, GBL, REL, UVN)	026406	000002	00002.
	USHSIO 1A.18		COBOVR.OLB

GLOBAL SYMBOLS:

BALPHA 061764-R SIOU1L 061144-R SPSU1L 061202-R

*** TASK BUILDER STATISTICS:

TOTAL WORK FILE REFERENCES: 338815.
WORK FILE READS: 0.
WORK FILE WRITES: 0.
SIZE OF CORE POOL: 21194. WORDS (62. PAGES)
SIZE OF WORK FILE: 14544. WORDS (74. PAGES)

ELAPSED TIME: 00:01:13

Figure SOP-21. Task Build Statistics (Resident Library)

TRIAL.TSR MEMORY ALLOCATION MAP TAB 07.005 PAGE 38
23-4AK-79 07:26

*** SEGMENT: USHSIO

R/W MEM LIMITS: 075000 101747 004750 02536.
DISK BLK LIMITS: 000206 000272 000005 00005.

MEMORY ALLOCATION SYNOPSIS:

SECTION	TITLE	IDENT	FILE
BLK: (R, I, LCL, REL, COM)	075000	000000	00000.
BUSHIO: (RU, I, GBL, REL, UVN)	075000	004750	02536.
SSALYC: (R, D, LCL, REL, COM)	075000	000000	00000.
SSRTS: (R, I, GBL, REL, UVN)	030674	000002	00002.
	USHSIO 1A.18		COBOVR.OLB

GLOBAL SYMBOLS:

BALPHA 075620-M SIOU1L 075000-M SPSU1L 075036-R

*** TASK BUILDER STATISTICS:

TOTAL WORK FILE REFERENCES: 344509.
WORK FILE READS: 919.
WORK FILE WRITES: 541.
SIZE OF CORE POOL: 21194. WORDS (62. PAGES)
SIZE OF WORK FILE: 37376. WORDS (146. PAGES)

ELAPSED TIME: 00:01:50

Figure SOP-22. Task Build Statistics (With Resident Library)

DIGITAL

RSTS/E V7.0 RMS OPERATOR
Software and RSTS/E V7.0

MACRO and RSTS/E V7.0

General

MONITOR CALLS

- o All monitor calls are documented in the RSTS/E System Directives Manual
- o All RSTS/E V7.0 features available through monitor calls

RMS Features

- o Full RMS capability

Note

Use of RT11 directives are not documented and therefore are not supported.

RSTS/E UPDATE

BKP

Backup Programs

Backup Programs

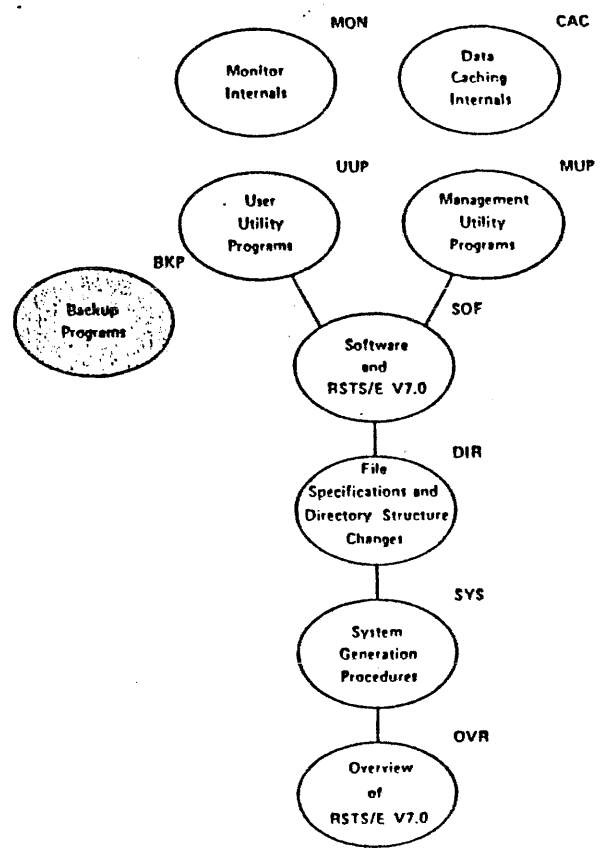
CONTENTS

Introduction.....	BKP-1
PIP Program.....	BKP-2
General.....	BKP-2
New Features.....	BKP-2
New Switches and Options.....	BKP-3
Multi-Volume Example.....	BKP-7
Notes on PIP.....	BKP-8
BACKUP Package.....	BKP-12
New Features.....	BKP-12
Example BACKUP in Interactive Mode.....	BKP-13
Example of LOADINDEX Mode.....	BKP-17
Example of LIST Mode.....	BKP-19
Example of BACKUP Batch Control File.....	BKP-22
Example of BACKUP Batch Log.....	BKP-23
SAVE/RESTORE Program.....	BKP-28
Features.....	BKP-29
Rules for Using Media.....	BKP-29
SAVE/RESTORE Options.....	BKP-31
Example of IDENTIFY and SAVE Functions.....	BKP-32
Example of RESTORE Function.....	BKP-34
Example of IMAGE Function.....	BKP-36
Additional SAVE/RESTORE Documentation.....	BKP-38
Definitions of SAVE/RESTORE Terms.....	BKP-39
Running SAVE/RESTORE.....	BKP-40
SAVE/RESTORE Functions.....	BKP-41
SAVE/RESTORE Options.....	BKP-43
Dialogue.....	BKP-45
SAVE/RESTORE Volumes.....	BKP-46
Device Specifications.....	BKP-47
Checking the Input Volume.....	BKP-48
Checking the Output Volume.....	BKP-49
General Comparison of Backup Programs.....	BKP-53
Transferring RMS Relative and Indexed Files to Other non-RSTS/E Operating Systems.....	BKP-55

CONFIDENTIAL

BACKUP PROGRAMS

COURSE MAP



Backup Programs

INTRODUCTION

Since backup is a major concern for all system managers, RSTS/E supplies several backup utility programs which handle the individual needs of each RSTS/E site.

The new or enhanced RSTS/E V7.0 utilities:

- o Backup and restore
 - large files
 - placed files
 - contiguous files
 - attributed files
 - entire disks
- o Produce bootable backup media with bad block handling

Decisions must be made concerning which utility to use under a given set of circumstances.

In this module you will learn:

- o The features of the new utility SAVE/RESTORE,
- o The enhancements to PIP and the BACKUP package, and
- o When to use SAVE/RESTORE, PIP, BACKUP and RMSBCK

PIP PROGRAM

General

- o Has expanded from 8Kw to 12Kw,
- o Has replaced PIPSML and PIPEXT,
- o Accepts the /POSITION:n switch to create a placed file as long as the switch is adjacent to the filename,
- o Is the only way to selectively back up large files which cannot fit on a single tape, and
- o Zeroes large files faster than mode 128.

New features

- o Multi-volume ANSI magnetic tape support, and
- o Specification of density and parity when zeroing a magnetic tape.

New Switches and Options

File Access Switches

Switch	Format	Meaning
/ACCESS	/AC	For disk input only, this switch causes PIP to change the file's last access date to the current date. If you do not specify this switch, PIP preserves the last access date.
/ASCII	/AS	Does a formatted ASCII transfer, causing PIP to ignore nulls, parity bits, and rubout characters.
/BSIZE:n		Used to specify a non-standard blocksize for file transfers to magtape. "n" is the blocksize in bytes to be output. "n" must be a positive even integer from 18 to 4096. For ANSI format magtapes intended for interchange with foreign operating systems, the blocksize should not exceed 2048. For interchange with RT11 operating systems, the default of 512 should be used.
/NEWfile	/NE	For disk output only, this switch creates a new file with the current date of creation and access.
/NOAttributes	/NOA	This switch causes PIP to transfer the file without writing attributes to the output file.
/PR:NOWARN		This switch causes PIP to ignore the operation if an output file already exists and to suppress the printing of the error message.

Switch	Format	Meaning
/RETAIN		For disk output, this switch preserves the file's creation, access dates, and creation time. This is the normal default for PIP, but it may be changed by an optional patch.
/RTS:name		This switch sets the output file's run-time system name to that specified in name. If no output file is specified, PIP renames the input file's run-time system.
/CREATION	/CRE	This switch uses the creation date for limit checking with the date switches (/AF, /BE, /ON, /SI, /TO). This is the normal default for PIP, but it may be changed by an optional patch.
/DLA		This switch uses the date of last access for limit checking with the date switches.

Attributed File Translation Switch

/RMS	/RMS:FTN	For input files, translates FORTRAN carriage control to formatted ASCII. For non-disk output files, translation is automatic.
/ANSI	/ANSI:FTN	
	/ANSI:PRN	For input disk files, translates RMS print files to formatted ASCII.
	/RMS:PRN	

File Operation Switches - General

/IN	/IN	Same as V6C.
/	/IN:S	Prints the filename, extension, filesize, protection code, creation date, and last access date.

File Operation Switches - Deletion

Switch	Format	Meaning
/DE	/DE	Same as V6C.
	/DE:NO(WARN)	Causes PIP to suppress the printing of an error message if the file to be deleted does not exist.

File Operation Switches - Listing

/DI:option	:AL	Prints the filename, extension, and the number of blocks allocated to the file.
/LI:option	:AT	Prints the filename, extension and symbolic representation of file attributes.
	:SA	
	:FU	Full listing; print all information except file attributes.
	:OA	Prints the filename, extension, and octal representation of file attributes.
	:RT	Prints the filename, extension, and associated run-time system.
	:SZ	Prints the filename, extension and file size (same as :SI).
	:WI	Prints filenames only across the page or screen.
/SLOW	:SL	Same as /DI:S .

File Operation Switches - Zeroing

Switch	Format	Meaning
/DENsity	/ZE/DEN:n	For magtape only, this switch sets the tape density prior to the zero operation.
/PARity	/ZE/PAR:ODD /ZE/PAR:EVEN	For magtape only, this switch sets the tape parity prior to the zero operation.

Multi-Volume Example

Consider the following example which transfers the contents of a user account to three magnetic tape volumes:

```
RUN $PIP
*MM0:*.]=[20,254]*.*/BL
```

```
%END OF ANSI MAGTAPE OUTPUT VOLUME HAS
%BEEN REACHED
```

```
%PLEASE TYPE THE DEVICE NAME AND UNIT
%NUMBER OF THE DRIVE WHERE THE NEXT
%VOLUME MAY BE FOUND
```

```
?MM1:TAPE2
```

```
%END OF ANSI MAGTAPE OUTPUT VOLUME HAS
%BEEN REACHED
```

```
%PLEASE TYPE THE DEVICE NAME AND UNIT
%NUMBER OF THE DRIVE WHERE THE NEXT
%VOLUME MAY BE FOUND
```

```
?MM2:TAPE3
```

In this example, you specify the block mode transfer of all files in account [20,254] on the system disk to the tape mounted on MM0:. When PIP reaches the end of the output tape, it prints the dialogue that requests the next volume. In response to the prompt, type the device name and unit number (MM1:) and the optional volume ID (TAPE2). PIP checks the output volume ID, if specified, and that the tape is ANSI labeled and initialized. If the ID is valid, and the tape is ANSI and free of data, then PIP continues with the transfer. This process continues until all of the specified files on the input device are transferred.

When the transfer is complete, PIP prints its asterisk prompt.

Notes on PIP

1. Appending files with attributes

- o Do not merge attributed files with PIP.

2. RMS Relative and Indexed Files

- a. Use the /BL switch for transfer from disk to ANSI magtape.
- b. Use no switch for transfer from ANSI magtape to disk.

Note

The above procedure is for RSTS/E to RSTS/E transfers only.

- c. No switches are needed for disk-to-disk transfers.

3. RMS Sequential Files

- a. No switches are needed for file transfers to and from ANSI magtape.
- b. No switches are needed for disk-to-disk transfers.

4. /ANSI and /RMS switches

- a. /ANSI:FA or /RMS:FA on the input file specification:

Purpose - Translate RMS Sequential files to stream ASCII files

- o Takes an RMS sequential file with fixed or variable length records as input.
- o Strips off the byte count (if necessary), squeezes out null and rubout characters, and appends <CR><LF> to each record.

- o Creates a stream ASCII output file without attributes.

Note

Do not use this switch if the RMS sequential file contains binary data.

- b. /ANSI:FA or /RMS:FA on the output file specification:

Purpose - Translate stream ASCII files to RMS sequential files with variable length records.

- o Takes a stream ASCII file as input.
- o Places a byte count at the beginning of each record and strips off <CR><LF>.
- o Creates a sequential file with variable length records.
- c. /ANSI:FTN or /RMS:FTN on the input file specification:

Purpose - Translate files with FORTRAN carriage control characters to stream ASCII files.

- o Takes a file with FORTRAN carriage control attributes as input.
- o Strips off the FORTRAN carriage control characters from each record and appends <CR><LF>.
- o Creates a stream ASCII file without attributes.

Note

Specifying this switch on an output file specification produces a ?Bad Switch Value error.

d. /ANSI:FB or /RMS:FB on the input file specification:

Purpose - Transfer RSX type object modules to DOS labelled magnetic tape.

- o Takes RSX type object modules as input.
- o Translates the file into RT11 object module format.

e. /ANSI:FB or /RMS:FB on the output file specification:

Purpose - Transfer RSX type object modules from DOS magnetic tape to disk.

- o Translates the file into RSX object module format.

f. /ANSI:PRN or /RMS:PRN on the input file specification:

Purpose - Translate RMS print format files to stream ASCII files.

Note

Currently VAX uses this format for its log files. If these files are transferred to RSTS/E by magnetic tape or through DECnet/E, the /ANSI:PRN should be used since the SPOOL program does not recognize this format.

- o Strips carriage control information from records and inserts appropriate control characters.

Note

Specifying this switch on the output file specification produces a ?Bad Switch Value error.

g. /ANSI:IM or /RMS:IM on the input file specification:

Purpose - Print map files created by the task builder at terminals.

Note

/ANSI:FA on the input file also performs this operation.

- o Takes an RMS sequential file as input.
- o Strips off the byte count at the beginning of each record (if necessary)

Note

Specifying this switch on the output file specification produces a ?Bad Switch Value error.

BACKUP PACKAGE

New Features

- o Copying of the index from the BACKUP set to a disk,
- o Listing a directory index,
- o Ability to run under BATCH,
- o Caching of the workfile,
- o Sequential caching of the files being backed up, and
- o Residing in an account other than [1,2].

```

PUB [1,10]BACKUP
BACKUP 17.0-05 RSTS 07.0-05 The ARE
BACKUP], RES[TORE], LOA[DINDEX] OR LIB[IT] ? BACKUP
WORK_FILE NAME<_BT> [ 1,235]B20PER.J25 ?
LISTING FILE<_KB> ?
FROM DISK<_BT> ?
FROM FILES [ 1,235]*.* ? [1,235]*.*/*EXC:(*.BAR,*.J25,BCRBAC.LST)
TO DEVICE<_MT> ?
BEGIN AT<!*.*> ?
DELETE FILES<NONE> ?
COMPARE FILES<NONE> ?

```

Backup Run Listing

```

Backup from '_BT' to '_MT'
Run started on 20-Feb-79 at 09:10 AM
Work-File is _BT [1,235]B20PER.J25
Transfer : [1,235]???????.???/EXC:([1,235]???????.BAR,[1,235]???????.J25,[1,235]BCRBAC.LST)
Begin at : [*.*.*
Delete : (none)
Compare : (none)
*
PHASE : LIST COMPLETE
ERRORS : 0
ELAPSED TIME : 1 SECONDS
CPU TIME : .4 SECONDS
BYTES : 60
*
PHASE : LOOKUP/SELECT COMPLETE

```

Figure BKP-1. BACKUP in Interactive Mode (Page 1 of 4)

DIGITAL

RSTS/E UPDATE
Backup Programs

VOLUME 0 : 1
ACCOUNTS : 1
FILES : 48
BLOCKS : 378
ERRORS : 0

ELAPSED TIME : 23 SECONDS
CPU TIME : 4.7 SECONDS
RCTS : 785

PLEASE ENTER BACKUP SET NAME(B20FEB) -
PLEASE ENTER EXPIRATION DATE(20-Feb-80) -
PLEASE ENTER DENSITY IN BPI(800) -
PLEASE ENTER THE PARITY(ODD) -

PHASE : MOUNT / DISMOUNT COMPLETE
ERRORS : 0

IDENTIFICATION WILL BE FINAL UPON SUCCESSFUL MOUNT
DEVICE? RMO:
THIS VOLUME HAS NO BACKUP LABEL!
MOUNT IT ANYWAY(Y/N) ? Y

PHASE : MOUNT / DISMOUNT COMPLETE
ERRORS : 0

ELAPSED TIME : 51 SECONDS
CPU TIME : .8 SECONDS
RCTS : 128

PHASE : TRANSFER COMPLETE
VOLUME 0 : 1
ACCOUNTS : 1
FILES : 48
BLOCKS : 378
ERRORS : 0

CURRENT VOLUME : _RMO :

ELAPSED TIME : 23 SECONDS
CPU TIME : 4 SECONDS
RCTS : 836

PHASE : INDEX DUMP COMPLETE
ERRORS : 0

CURRENT VOLUME : _RMO :

ELAPSED TIME : 6 SECONDS
CPU TIME : .6 SECONDS
RCTS : 94

DIGITAL

RSTS/E UPDATE
Backup Programs

DISMOUNT DEVICE: RMO 1
ID: B20FEB
SCSI: 11INDEX1
DENSITY: 800 BPI
PARITY: ODD
EXPIRATION DATE: 20-Feb-80
PLEASE LABEL THIS VOLUME!

PHASE : MOUNT / DISMOUNT COMPLETE
ERRORS : 0

ELAPSED TIME : 3 SECONDS
CPU TIME : .3 SECONDS
RCTS : 48

Backup Set Name : B20FEB Backup Device : _RMO
Volume Sequence # : 1(INDEX) Owner : [1.235]
Creation Date : 20-Feb-79 Expiration Date : 20-Feb-80
Density : 800 BPI Parity : ODD

		Account : [1.235]		Quota : 5000					
		Cluster size : 16							
Name	Ext	Size	Prot	Creation Date	Time	Access Date	Clu	RTS	TSCD
(SOURCE DISK - DB1:)									
RFD	.DMP	39	60	09-Nov-78	02:59 PM	09-Nov-78	0	BASIC	T
ONPFIL	.BAS	3	60	09-Nov-78	03:32 PM	09-Nov-78	0	...PSX	T
RFD1	.DMP	52	60	09-Nov-78	03:30 PM	09-Nov-78	0	BASIC2	T
SYS	.LST	3	60	16-Nov-78	01:16 PM	15-Nov-78	0	...PSX	T
SYSD	.LST	3	60	16-Nov-78	01:23 PM	15-Nov-78	0	...PSX	T
SEO	.CBL	3	60	28-Nov-78	11:03 AM	28-Nov-78	0	...PSX	T
SEDS	.ODL	2	60	29-Nov-78	12:15 PM	29-Nov-78	0	...PSX	T
CONT	.DAT	16C	60	07-Dec-78	09:57 AM	07-Dec-78	0	BASIC	T
PLACE	.BAS	1	60	07-Dec-78	09:07 AM	07-Dec-78	0	...PSX	T
11024	.B2B	1	60	29-Nov-78	09:14 AM	29-Nov-78	0	BPFCOM	T
11024	.ODL	1	60	15-Dec-78	02:11 PM	15-Dec-78	0	BPFCOM	T
ONLY	.RMS	11	60	15-Dec-78	02:10 PM	15-Dec-78	0	BASIC2	T
DMP	.BAS	2	60	01-Jun-78	11:47 AM	01-Jun-78	0	BASIC	T
LOOP	.BAS	1	60	12-Jul-78	10:41 AM	12-Jul-78	0	BASIC	T
ENTIC	.BAS	2	60	05-Sep-78	04:50 PM	05-Sep-78	0	BASIC	T
TCOMP	.BAS	1	60	24-Oct-78	03:31 PM	24-Oct-78	0	BASIC	T
CCONT	.BAS	1	60	21-Nov-78	02:16 PM	21-Nov-78	0	BASIC	T
NOH11	.BAS	1	60	21-Nov-78	02:38 PM	21-Nov-78	0	BASIC	T
CCPOS	.BAS	1	60	21-Nov-78	02:38 PM	21-Nov-78	0	BASIC	T
CHAIN	.BAS	1	60	21-Nov-78	02:19 PM	21-Nov-78	0	BASIC	T
DEF	.BAS	1	60	21-Nov-78	02:39 PM	21-Nov-78	0	BASIC	T
MOD12B	.BAS	1	60	27-Nov-78	02:22 PM	27-Nov-78	0	BASIC	T
VIRT1	.BAS	1	60	27-Nov-78	02:30 PM	27-Nov-78	0	...PSX	T
SEO	.BAS	1	60	27-Nov-78	03:10 PM	27-Nov-78	0	BASIC	T
VIRTUL	.BAS	1	60	28-Nov-78	10:40 AM	28-Nov-78	0	BASIC	T
UPDCRT	.BAS	1	60	28-Nov-78	12:09 PM	28-Nov-78	0	BASIC	T
UPDACC	.BAS	1	60	28-Nov-78	12:11 PM	28-Nov-78	0	BASIC	T

Figure BKP-1. BACKUP in Interactive Mode (Page 2 of 4)

Figure BKP-1. BACKUP in Interactive Mode (Page 3 of 4)

DIGITAL

RSTS/E UPDATE
Backup Programs

DIGITAL

RSTS/E UPDATE
Backup Programs

```

UPDERT.BAS  1  60  28-Nov-78 12:18 PM  28-Nov-78  0 BAS4F  T
UPGRED.BAS  1  60  28-Nov-78 02:20 PM  28-Nov-78  0 BAS4F  T
TEPT1.BAS   1  60  28-Nov-78 02:34 PM  28-Nov-78  0 ...PSM*T
TEPT1.BAS   1  60  28-Nov-78 02:47 PM  28-Nov-78  0 ...PSM*T
TEPT2.BAS   1  60  23-Dec-78 04:11 PM  23-Dec-78  0 ...PSM*T
TEPT2.BAS   1  60  06-Dec-78 10:20 AM  06-Dec-78  0 ...PSM*T
PMSC.BAS    1  60  07-Dec-78 09:17 AM  07-Dec-78  0 BAS4F  T
MORCHT.BAS  1  60  15-Dec-78 03:27 PM  15-Dec-78  0 BAS4F  T
TTT45.BAS   1  60  12-Dec-78 05:10 PM  12-Dec-78  0 BAS4F  T
EVLNFS.BAS  10  40  22-Dec-78 12:55 PM  22-Dec-78  0 BAS4F  T
EVLNFS.BAS  13C 124 22-Dec-78 12:55 PM  22-Dec-78  0 BASIC2  T
EVLNFS.CRP  10  60  22-Dec-78 01:12 PM  22-Dec-78  0 BASIC2  T
ANALYS.DMP  10  60  02-Jan-79 09:52 AM  02-Jan-79  0 RT11  T
SYSR.DMP    6  60  15-Jan-79 03:43 PM  15-Jan-79  0 RT11  T
SSIABL.IDX  6  60  17-Jan-79 09:58 AM  17-Jan-79  0 RT11  T
SSIABL.DMP  6  60  17-Jan-79 10:09 AM  17-Jan-79  0 RT11  T
SYSGEN.LOG  40  60  01-Feb-79 03:26 PM  01-Feb-79  0 ...RSX*T
RTSGEN.CYL  7  60  04-Feb-79 10:32 AM  04-Feb-79  0 BAS4F  T

```

* Attributes associated with this file

Account total of 278 blocks in 44 files on account (1,235)

Volume total of 278 blocks in 44 files in 1 accounts on volume _RM

```

*****
RUN total of 278 blocks in 44 files on 1 accounts on 1 Volumes
RUN total of 0 errors
RUN terminated on 28-Feb-79 at 09:16 AM
*****

```

```

*
PHASE : LIST COMPLETE
VOLUME 0 : 1
ACCOUNTS : 1
FILES : 44
BLOCKS : 278
ERRORS : 0

```

```

ELAPSED TIME : 141 SECONDS
CPU TIME : 2.9 SECONDS
ECTS : 435

```

Ready

Figure BKP-1. BACKUP in Interactive Mode (Page 4 of 4)

BKP-15

COMPANY CONFIDENTIAL

```

NOW(1,10)BACKUP
BACKUP 27.0-05 RSTS V7.0-05 THE ARR

BAC(RUP), RES(TORE), LOA(DINDEX) OR LIS(T) ? LOADINDEX
WORK FILE NAME<_BY>: (1,235)020PEB.325 ?
LISTING FILE<_MT> ?
FROM DEVICE<_MT> ?
TO FILE<_BY>: (1,235)01WD20.IND ?

```

Backup Run Listing

```

LoadIndex from "_MT"
Run started on 28-Feb-79 at 09:21 AM
Work-File is _BY : (1,235)020PEB.325
To File : _BY : (1,235)01WD20.IND

```

```

*
PHASE : LIST COMPLETE
ERRORS : 0

```

```

ELAPSED TIME : 8 SECONDS
CPU TIME : .3 SECONDS
ECTS : 44

```

```

*
PLEASE ENTER BACKUP SET NAME<020PEB> -
PLEASE ENTER DENSITY IN BPI<800> -
PLEASE ENTER THE PARITY<ODD> -
MOUNT DEVICE: _MT
          ID: 020PEB
          SCOD: INDEX
          DENSITY: 800 BPI
          PARITY: ODD
PLEASE MOUNT VOLUME WRITE LOCKED!
DEVICE? HMO:

```

```

*
PHASE : MOUNT / DISMOUNT COMPLETE
ERRORS : 0

```

Figure BKP-2. LOADINDEX Mode (Page 1 of 2)

BKP-17

COMPANY CONFIDENTIAL

DIGITAL

RSTS/E UPDATE
Backup Programs

ELAPSED TIME : 48 SECONDS
CPU TIME : .8 SECONDS
RCTS : 144

PHASE : INDEX LOAD COMPLETE
ERRORS : 0

CURRENT VOLUME : _MMS :

ELAPSED TIME : 1 SECONDS
CPU TIME : .3 SECONDS
RCTS : 74

DISHMOUNT DEVICE: _MMS :

PHASE : MOUNT / DISMOUNT COMPLETE
ERRORS : 0

ELAPSED TIME : 3 SECONDS
CPU TIME : .1 SECONDS
RCTS : 14

Ready

Figure BKP-2. LOADINDEX Mode (Page 2 of 2)

DIGITAL

RSTS/E UPDATE
Backup Programs

RUN 11.10|BACKUP
BACKUP V7.0-05 RSTS V7.0-05 The ARR

BAC[KUP], RES[TORE], LOA[D]INDEX OR LIS[T] ? LIST

WORK FILE NAME<_BY>| 1.235|020FEB.225 ?

LISTING FILE<_RD> ?

INDEX FILE<PRIMARY> ? 01MD20.TWO

Backup Run Listing

List

Run started on 20-Feb-79 at 09:24 AM

Work-File is _BY |1.235|020FEB.225

PHASE : LIST COMPLETE
ERRORS : 0

ELAPSED TIME : 1 SECONDS
CPU TIME : .2 SECONDS
RCTS : 30

PHASE : INDEX LOAD COMPLETE
ERRORS : 0

ELAPSED TIME : 3 SECONDS
CPU TIME : .2 SECONDS
RCTS : 97

Backup Set Name : 020FEB
Volume Sequence : 1(index)
Creation Date : 20-Feb-79
Density : 000 BPI

Backup Device : _MMS :
Owner : 11.235|
Expiration Date : 20-Feb-80
Parity : ODD

Figure BKP-3. LIST Mode (Page 1 of 3)

DIGITAL

RESTORE UPDATE
Backup Programs

DIGITAL

RESTORE UPDATE
Backup Programs

Account : [1.235] Quota : 5000
Clusterwise : 10

Name	Ext	Size	Prot	Creation Date	Creation Time	Access Date	Clu	STS	TCCO
MPD .DMP	30	60	00	09-Nov-70	02:59 PM	09-Nov-70	0	0	BAS4P T
DMPFIL.BAS	3	60	00	09-Nov-70	03:32 PM	09-Nov-70	0	...	PSX*T
MPDI .DMP	32	60	00	09-Nov-70	03:30 PM	09-Nov-70	0	0	BASIC2 T
STS .LST	3	60	00	16-Nov-70	01:16 PM	16-Nov-70	0	...	PSX*T
STSO .LST	3	60	00	16-Nov-70	01:23 PM	16-Nov-70	0	...	PSX*T
SEQ .CBL	3	60	00	29-Nov-70	11:03 AM	29-Nov-70	0	...	PSX*T
SEQ2 .DDL	2	60	00	29-Nov-70	12:13 PM	29-Nov-70	0	...	PSX*T
COMT .DAT	10C	60	00	07-Dec-70	08:57 AM	07-Dec-70	0	0	BAS4P T
PLICE .BAS	1	60	00	07-Dec-70	09:17 AM	07-Dec-70	0	...	PSX*T
II021 .B2B	1	60	00	29-Nov-70	09:14 AM	29-Nov-70	0	0	BP2COM T
II024 .DDL	1	60	00	15-Dec-70	02:11 PM	15-Dec-70	0	0	BP2COM T
ONLY .PAS	11	60	00	15-Dec-70	02:19 PM	15-Dec-70	0	0	BASIC2*T
DMP .BAS	2	60	00	01-Jun-70	11:07 AM	01-Jun-70	0	0	BAS4P T
LOOP .BAS	1	60	00	12-Jul-70	10:41 AM	12-Jul-70	0	0	BAS4P T
SPIC .BAS	2	60	00	03-Sep-70	04:50 PM	03-Sep-70	0	0	BAS4P T
TCOMP .BAS	1	60	00	14-Oct-70	03:31 PM	26-Oct-70	0	0	BAS4P T
CCOMT .BAS	1	60	00	21-Nov-70	02:36 PM	21-Nov-70	0	0	BAS4P T
MON11 .BAS	1	60	00	21-Nov-70	02:36 PM	21-Nov-70	0	0	BAS4P T
CCPOS .BAS	1	60	00	21-Nov-70	02:30 PM	21-Nov-70	0	0	BAS4P T
CHAIN .BAS	1	60	00	21-Nov-70	02:30 PM	21-Nov-70	0	0	BAS4P T
DEF .BAS	0	60	00	21-Nov-70	02:30 PM	21-Nov-70	0	0	BAS4P T
ROD120 .BAS	1	60	00	27-Nov-70	02:22 PM	27-Nov-70	0	0	BAS4P T
VIRT1 .BAS	1	60	00	27-Nov-70	02:30 PM	27-Nov-70	0	...	PSX*T
SEQ .BAS	1	60	00	27-Nov-70	03:10 PM	27-Nov-70	0	0	BAS4P T
VIRTUL .BAS	1	60	00	28-Nov-70	10:40 AM	28-Nov-70	0	0	BAS4P T
UPDCPT .BAS	1	60	00	28-Nov-70	12:09 PM	28-Nov-70	0	0	BAS4P T
UPCACC .BAS	1	60	00	28-Nov-70	12:11 PM	28-Nov-70	0	0	BAS4P T
UPCEST .BAS	1	60	00	28-Nov-70	12:16 PM	28-Nov-70	0	0	BAS4P T
UPCEDO .BAS	1	60	00	28-Nov-70	02:20 PM	28-Nov-70	0	0	BAS4P T
TEMP1 .BAS	1	60	00	28-Nov-70	02:30 PM	28-Nov-70	0	...	PSX*T
TEMP .BAS	1	60	00	28-Nov-70	02:47 PM	28-Nov-70	0	...	PSX*T
TEMP2 .BAS	1	60	00	03-Dec-70	04:11 PM	03-Dec-70	0	...	PSX*T
PNSC .BAS	1	60	00	06-Dec-70	10:20 AM	06-Dec-70	0	...	PSX*T
MONCR1 .BAS	1	60	00	07-Dec-70	09:17 AM	07-Dec-70	0	0	BAS4P T
TT144 .BAS	1	60	00	15-Dec-70	03:27 PM	15-Dec-70	0	0	BAS4P T
EVLNFS .BAS	30	40	12	12-Dec-70	03:10 PM	12-Dec-70	0	0	BAS4P T
EVLNFS .BAC	35C	124	22	22-Dec-70	12:55 PM	22-Dec-70	0	0	BAS4P T
EVLNFS .COP	30	60	00	22-Dec-70	01:12 PM	22-Dec-70	0	0	BASIC2 T
ANALYS .DMP	10	60	00	02-Jan-79	09:52 AM	02-Jan-79	0	0	BASIC2 T
BSTS .DMP	6	60	00	15-Jan-79	09:43 PM	15-Jan-79	0	0	RT11 T
PSTARL .IDX	4	60	00	17-Jan-79	09:50 AM	17-Jan-79	0	0	RT11 T
PSTARL .DMP	4	60	00	17-Jan-79	10:09 AM	17-Jan-79	0	0	RT11 T
BISGEN .LOG	40	60	00	01-Feb-79	03:26 PM	01-Feb-79	0	...	PSX*T
BISGEN .CYL	7	60	00	06-Feb-79	10:32 AM	06-Feb-79	0	0	BAS4P T

* Attributes associated with this file

Account total of 270 blocks in 44 files on account [1.235]

Volume total of 270 blocks in 44 files in 1 accounts on volume _HME

.....

Figure BKP-3. LIST Mode (Page 2 of 3)

RUN total of 270 blocks in 44 files on 1 accounts on 1 volumes
RUN total of 0 errors
RUN terminated on 20-Feb-79 at 09:25 AM

.....

PHASE : LIST COMPLETE
VOLUME # : 1
ACCOUNTS : 1
FILES : 44
BLOCKS : 270
ERRORS : 0

ELAPSED TIME : 50 SECONDS
CPU TIME : 3 SECONDS
ICTS : 450

Ready

Figure BKP-3. LIST Mode (Page 3 of 3)

DIGITAL

RSTS/E UPDATE
Backup Programs

```
$JOB
$RUN [1,10]BACKUP
$DATA
BACKUP
```

```
[1,235]*.*/EXC:(*.BAK,*.J25)
```

```
MM8:
YES
$EOD
$EOJ
```

Figure BKP-4. Batch Control File

DIGITAL

RSTS/E UPDATE
Backup Programs

```
-----$JOB
09:30:01 AM HELLO
09:30:03 AM RSTS V7.0-05 The ARR Job 25 001 20-Feb-79 09:30 AM
09:30:03 AM 01/235
09:30:04 AM Password:
09:30:04 AM 1 other user is logged in under this account
```

09:30:04 AM Ready

```
-----$RUN [1,10]BACKUP
09:30:04 AM RUN [1,10]BACKUP
09:30:05 AM BACKUP V7.0-05 RSTS V7.0-05 The ARR
```

```
-----$DATA
09:30:06 AM BAC(RUP), REB(TORE), LOA(DINDER) OR LIS(ET) ? BACKUP
09:30:07 AM WORK FILE NAME<_SY> [ 1,235]020FEB.J25 ?
09:30:07 AM LISTING FILE<_RB> ?
09:30:08 AM FROM DISK<_ST> ?
09:30:09 AM FROM FILE< 1,235>*. * ? [1,235]*.*/EXC:(*.BAK,*.J25)
09:30:10 AM TO DEVICE<_MT> ?
09:30:11 AM BEGIN AT<[*.*]> ?
09:30:12 AM DELETE FILES<NONE> ?
09:30:13 AM COMPARE FILES<NONE> ?
09:30:15 AM *
```

09:30:15 AM Backup Run Listing

```
09:30:15 AM Backup from '_RY' to '_MT'
09:30:15 AM Run started on 20-Feb-79 at 09:30 AM
09:30:15 AM Work-File is _SY (1,235)020FEB.J25
09:30:15 AM Transfer : [1,235]???????.???/EXC:(1,235)???????.BAK,(1,235)???????.J25)
09:30:15 AM Begin at : [*.*]
09:30:15 AM Delete : <none>
09:30:15 AM Compare : <none>
09:30:15 AM *
```

Figure BKP-5. Batch Log (Page 1 of 5)

DIGITAL

RSTS/E UPDATE
Backup Programs

DIGITAL

RSTS/E UPDATE
Backup Programs

```

09:30:15 AM PHASE : LIST COMPLETE
09:30:15 AM ERRORS : 0

09:30:15 AM ELAPSED TIME : 0 SECONDS
09:30:15 AM CPU TIME : .4 SECONDS
09:30:15 AM RCTS : 0

```

```

09:30:33 AM *
09:30:33 AM PHASE : LOOKUP/SELECT COMPLETE
09:30:33 AM VOLUME 0 : 1
09:30:33 AM ACCOUNTS : 1
09:30:33 AM FILES : 58
09:30:33 AM BLOCKS : 322
09:30:33 AM ERRORS : 0

```

```

09:30:33 AM ELAPSED TIME : 13 SECONDS
09:30:33 AM CPU TIME : 4.6 SECONDS
09:30:33 AM RCTS : 784

```

```

09:30:36 AM *
09:30:36 AM PLEASE ENTER BACKUP SET NAME(B20FEB) -
09:30:37 AM PLEASE ENTER EXPIRATION DATE(20-Feb-80) -
09:30:38 AM PLEASE ENTER DENSITY IN BPI(800) -
09:30:39 AM PLEASE ENTER THE PARITY(ODD) -
09:30:40 AM MOUNT DEVICE: MT 0
09:30:40 AM ID: B20FEB
09:30:40 AM SEQ#: 1
09:30:40 AM DENSITY: 800 BPI
09:30:40 AM PARITY: ODD
09:30:40 AM IDENTIFICATION WILL BE FINAL UPON SUCCESSFUL MOUNT
09:30:40 AM DEVICE? RMB:
09:30:41 AM EXPIRATION DATE HAS NOT YET ARRIVED!
09:30:41 AM ID: B20FEB
09:30:41 AM SEQ#: 1
09:30:41 AM DENSITY: 800 BPI
09:30:41 AM PARITY: ODD
09:30:41 AM EXPIRATION DATE: 20-Feb-80
09:30:41 AM MOUNT BY ANYWAY(CHO)TTES

```

```

09:30:44 AM *
09:30:44 AM PHASE : MOUNT / DISMOUNT COMPLETE
09:30:44 AM ERRORS : 0

```

```

09:30:44 AM ELAPSED TIME : 8 SECONDS
09:30:44 AM CPU TIME : 1.2 SECONDS
09:30:44 AM RCTS : 388

```

```

09:30:01 AM *
09:30:01 AM DATA UNRELIABLE - FILE OPENED BY ANOTHER USER IN FILE_DBI ([1.235]BACKU
09:30:01 AM P.LOG
09:30:01 AM (ON TRANSFER)

```

```

09:30:02 AM *
09:30:02 AM LENGTH CHANGE IN FILE_DBI ([1.235]BACKUP.LOG
09:30:02 AM (ON TRANSFER)
09:30:02 AM *
09:30:02 AM LENGTH AT LOOKUP - 1 LENGTH AT TRANSFER - 5
09:30:03 AM *
09:30:03 AM PHASE : TRANSFER COMPLETE
09:30:03 AM VOLUME 0 : 1
09:30:03 AM ACCOUNTS : 1
09:30:03 AM FILES : 58
09:30:03 AM BLOCKS : 322
09:30:03 AM ERRORS : 2

```

```

09:30:03 AM CURRENT VOLUME : _RMB :
09:30:03 AM ELAPSED TIME : 19 SECONDS
09:30:03 AM CPU TIME : 4.7 SECONDS
09:30:03 AM RCTS : 748

```

```

09:30:09 AM *
09:30:09 AM PHASE : INDEX DUMP COMPLETE
09:30:09 AM ERRORS : 0

```

```

09:30:09 AM CURRENT VOLUME : _RMB :
09:30:09 AM ELAPSED TIME : 5 SECONDS
09:30:09 AM CPU TIME : .7 SECONDS
09:30:09 AM RCTS : 118

```

```

09:30:14 AM *
09:30:14 AM DISMOUNT DEVICE: _RMB :
09:30:14 AM ID: B20FEB
09:30:14 AM SEQ#: 1(INDEX)
09:30:14 AM DENSITY: 800 BPI
09:30:14 AM PARITY: ODD
09:30:14 AM EXPIRATION DATE: 20-Feb-80
09:30:14 AM PLEASE LABEL THIS VOLUME!

```

```

09:30:14 AM *
09:30:14 AM PHASE : MOUNT / DISMOUNT COMPLETE
09:30:14 AM ERRORS : 0

```

```

09:30:14 AM ELAPSED TIME : 3 SECONDS
09:30:14 AM CPU TIME : .3 SECONDS
09:30:14 AM RCTS : 48

```

```

09:30:17 AM *

```

```

09:30:17 AM Backup Set Name : B20FEB Backup Device : _RMB :
09:30:17 AM Volume Sequence 0 : 1(Index) Owner : [1.235]
09:30:17 AM Creation Date : 20-Feb-79 Expiration Date : 20-Feb-80
09:30:17 AM Density : 800 BPI Parity : ODD

```

```

09:30:17 AM Account : [1.235] Quota : 5000
09:30:17 AM Clusterize : 16 Creation Access
09:30:17 AM

```

Figure BKP-5. Batch Log (Page 2 of 5)

Figure BKP-5. Batch Log (Page 3 of 5)

DIGITAL

RSTS/E UPDATE
Backup Programs

DIGITAL

RSTS/E UPDATE
Backup Programs

```

09:39:17 AM Name Ext Size Prot Date Time Date Lno Hls Tals
09:39:17 AM (SOURCE DISK = 001)
09:39:17 AM NRC .CMP 39 60 09-Nov-70 02:50 PM 09-Nov-70 0 BAS4F T
09:39:17 AM DMFFIL.BAS 3 60 09-Nov-70 03:32 PM 09-Nov-70 0 ...PSX*T
09:39:17 AM NRC1 .CMP 52 60 09-Nov-70 03:30 PM 09-Nov-70 0 BASIC2 T
09:39:17 AM SIS .LET 3 60 16-Nov-70 01:16 PM 16-Nov-70 0 ...PSX*T
09:39:17 AM SIS0 .LET 3 60 16-Nov-70 01:23 PM 16-Nov-70 0 ...PSX*T
09:39:17 AM SEQ .CBL 3 60 19-Nov-70 11:03 AM 19-Nov-70 0 ...PSX*T
09:39:18 AM SEQ2 .OOL 2 60 19-Nov-70 12:15 PM 19-Nov-70 0 ...PSX*T
09:39:18 AM COMT .DAT 16C 60 07-Dec-70 00:57 AM 07-Dec-70 0 BAS4F T
09:39:18 AM PLACE .BAS 1 60 07-Dec-70 09:47 AM 07-Dec-70 0 ...PSX*T
09:39:18 AM IIO24 .B2P 1 60 19-Nov-70 09:14 AM 19-Nov-70 0 DP7COM T
09:39:18 AM IIO24 .OOL 1 60 15-Dec-70 02:11 PM 15-Dec-70 0 DP7COM T
09:39:18 AM ONLT .BAS 11 60 15-Dec-70 02:19 PM 15-Dec-70 0 BASIC2*T
09:39:20 AM CMP .BAS 2 60 01-Jun-70 11:47 AM 01-Jun-70 0 BAS4F T
09:39:20 AM LOOP .BAS 1 60 12-Jul-70 10:01 AM 12-Jul-70 0 BAS4F T
09:39:20 AM SPVIC .BAS 2 60 05-Sep-70 04:50 PM 05-Sep-70 0 BAS4F T
09:39:20 AM TCOMP .BAS 1 60 26-Oct-70 01:31 PM 24-Oct-70 0 BAS4F T
09:39:20 AM CCOMT .BAS 1 60 21-Nov-70 02:36 PM 21-Nov-70 0 BAS4F T
09:39:20 AM MONII .BAS 1 60 21-Nov-70 02:36 PM 21-Nov-70 0 BAS4F T
09:39:20 AM CPOS .BAS 1 60 21-Nov-70 02:30 PM 21-Nov-70 0 BAS4F T
09:39:20 AM CPAIW .BAS 1 60 21-Nov-70 02:35 PM 21-Nov-70 0 BAS4F T
09:39:20 AM DCT .BAS 1 60 21-Nov-70 02:35 PM 21-Nov-70 0 BAS4F T
09:39:20 AM MCD18 .BAS 1 60 21-Nov-70 02:32 PM 21-Nov-70 0 BAS4F T
09:39:20 AM VIRT1 .BAS 1 60 21-Nov-70 02:30 PM 21-Nov-70 0 ...PSX*T
09:39:20 AM SEQ .BAS 3 60 27-Nov-70 03:10 PM 27-Nov-70 0 BAS4F T
09:39:20 AM VIRTUL.BAS 1 60 20-Nov-70 10:40 AM 20-Nov-70 0 BAS4F T
09:39:20 AM DPDCRT.BAS 3 60 20-Nov-70 12:09 PM 20-Nov-70 0 BAS4F T
09:39:20 AM UPDACC.BAS 3 60 20-Nov-70 12:11 PM 20-Nov-70 0 BAS4F T
09:39:20 AM UPDEXT.BAS 1 60 20-Nov-70 12:16 PM 20-Nov-70 0 BAS4F T
09:39:20 AM LDRDRE.BAS 1 60 20-Nov-70 02:20 PM 20-Nov-70 0 BAS4F T
09:39:21 AM TENT1 .BAS 1 60 20-Nov-70 02:34 PM 20-Nov-70 0 ...PSX*T
09:39:21 AM TENT .BAS 1 60 20-Nov-70 02:47 PM 20-Nov-70 0 ...PSX*T
09:39:21 AM TENT2 .BAS 1 60 05-Dec-70 04:11 PM 05-Dec-70 0 ...PSX*T
09:39:21 AM PMS .BAS 1 60 06-Dec-70 10:20 AM 06-Dec-70 0 ...PSX*T
09:39:21 AM MCHCWF .BAS 1 60 07-Dec-70 09:17 AM 07-Dec-70 0 BAS4F T
09:39:22 AM ITI66 .BAS 1 60 15-Dec-70 03:27 PM 15-Dec-70 0 BAS4F T
09:39:22 AM EVLHFS.BAS 14 40 12-Dec-70 05:10 PM 12-Dec-70 0 BAS4F T
09:39:22 AM EVLHFS.BAC 15C 124 22-Dec-70 12:55 PM 22-Dec-70 0 BAS4F T
09:39:22 AM EVLHFS.CRP 10 60 22-Dec-70 01:13 PM 22-Dec-70 0 BASIC2 T
09:39:22 AM ANALYS.DMP 10 60 02-Jan-79 09:52 AM 02-Jan-79 0 BASIC2 T
09:39:22 AM RSTS .DMP 6 60 15-Jan-79 03:43 PM 15-Jan-79 0 RTII T
09:39:22 AM BSTABL.IDX 4 60 17-Jan-79 09:50 AM 17-Jan-79 0 RTII T
09:39:22 AM BSTABL.DMP 4 60 17-Jan-79 10:09 AM 17-Jan-79 0 RTII T
09:39:22 AM SYSGEN.LOG 40 60 01-Feb-79 03:25 PM 01-Feb-79 0 ...RG*T
09:39:22 AM SYSGEN.CIL 7 60 06-Feb-79 10:32 AM 06-Feb-79 0 BAS4F T
09:39:22 AM BCYBAC.LST 11 60 20-Feb-79 09:12 AM 20-Feb-79 0 BAS4F T
09:39:22 AM BCYBAC.LST 3 60 20-Feb-79 09:10 AM 20-Feb-79 0 BAS4F T
09:39:22 AM BMD28 .IND 16 60 20-Feb-79 09:22 AM 20-Feb-79 0 BAS4F T
09:39:22 AM BCKEIS.LST 10 60 20-Feb-79 09:24 AM 20-Feb-79 0 BAS4F T
09:39:22 AM BACUP.CYL 1 60 20-Feb-79 09:31 AM 20-Feb-79 0 BAS4F T
09:39:22 AM BACUP.LOG 5 60 20-Feb-79 09:30 AM 20-Feb-79 0 BAS4F T

09:39:22 AM Error 1
09:39:22 AM DATA UNRELIABLE - FILE OPENED BY ANOTHER USER on FILE
09:39:22 AM LENGTH CHANGE on FILE
09:39:24 AM Total of 2 errors encountered on FILE

09:39:24 AM * Attributes associated with this file

09:39:24 AM Account total of 326 blocks in 50 files on account [1,319]
09:39:24 AM Total of 2 errors encountered on ACCOUNT

```

```

09:39:26 AM Volume total of 326 blocks in 50 files in 1 accounts on volume _MM
09:39:26 AM Total of 2 errors encountered on VOLUME

```

```

09:39:24 AM .....

```

```

09:39:24 AM RUN total of 326 blocks in 50 files on 1 accounts on 1 Volumes
09:39:24 AM RUN total of 2 errors
09:39:24 AM RUN terminated on 20-Feb-79 at 09:39 AM

```

```

09:39:24 AM .....

```

```

09:39:24 AM *
09:39:24 AM PHASE : LIST COMPLETE
09:39:24 AM VOLUME 0 : 1
09:39:24 AM ACCOUNTS : 1
09:39:24 AM FILES : 50
09:39:24 AM BLOCKS : 326
09:39:24 AM ERRORS : 0

```

```

09:39:24 AM ELAPSED TIME : 0 SECONDS
09:39:24 AM CPU TIME : 3.4 SECONDS
09:39:24 AM RCTS : 510

```

```

09:39:25 AM *
09:39:25 AM Ready

```

```

-----SE02
-----SE02
09:39:25 AM DVE
09:39:26 AM Confirms: YES
09:39:27 AM Saved all disk files; 633 blocks in use; 4368 free
09:39:27 AM Job 25 User 1,335 logged off RDI at 20-Feb-79 09:39 AM
09:39:27 AM 1 other user still logged in under this account
09:39:27 AM System RSTS V7.0-05 The AAK
09:39:27 AM Run time was 23.5 seconds
09:39:27 AM Elapsed time was 1 minute
09:39:27 AM Good morning

```

Figure BKP-5. Batch Log (Page 4 of 5)

Figure BKP-5. Batch Log (Page 5 of 5)

DIGITAL

Backup Programs

SAVE/RESTORE PROGRAM

Features

- o Disk to disk image copying with bad block handling,
- o Bootable save set of disk-to-tape or disk-to-disk,
- o Runnable off-line in the INIT.SYS environment, and
- o Runnable on-line under the RT11 Run-Time System.

DIGITAL

RSTS/E UPDATE
Backup Programs

Rules for Using Media

1. The IMAGE Function
 - a. IMAGE copies can only be made from disk to disk.
 - b. IMAGE can only copy between like disks (disks of the same size).
2. The SAVE and RESTORE functions.
 - a. Volumes of a single save set must be all disks or all tapes.
 - b. A save set may consist of different types of disks.
 - c. Tapes in a single save set must all be written at the same density.
3. Disks
 - a. Must be formatted.
 - b. Must be DSKINTed if SAVRES is being run on-line.
 - c. Will automatically DSKINT if SAVRES is running off-line.
 - d. The following blocks must not be bad:
 - o The first 15 blocks for a save set disk, and
 - o The first device cluster and the next n blocks, where n is the MFD clustersize for an IMAGE copy disk.
 - e. Must not have more than 151 bad clusters.
 - f. Cannot be logically mounted.

4. Tapes

- a. Must be long enough to hold the:
 - o Save set label and header information
 - o Bootstrap, and
 - o INIT.SYS .
- b. A bad block is considered to be the end of a volume.
- c. The default density is the lowest allowed on the first magtape drive used to produce a save set.

SAVE/RESTORE Options

Table BKP-1. SAVE/RESTORE Options

Option	Valid for Functions			
	IMAGE	SAVE	RESTORE	IDENTIFY
*EXPIRATION EXPIRATION: date NOEXPIRATION		X		
VERIFY *NOVERIFY	X	X	X	
*STATS NOSTATS	X	X	X	X
*ERROR NOERROR	X	X	X	X

* Default Condition (the default expiration date is one year from the date of the SAVE operation).

DIGITAL

RSTS/E UPDATE
Backup Programs

HOW SAVRES

SAV/RES Function:HELP

The valid SAV/RES functions are:

SAVE Have a safe copy of a RSTS/E file-structured disk.
 RESTORE Restore a RSTS/E file-structured disk from a SAVE Set.
 TRACE Copy a volume image from a RSTS/E file-structured disk
 to a LINE disk.
 IDENTIFY List the characteristics of a SAVE Set or RSTS/E disk.
 HELP Print this list of acceptable responses.
 Only the first two characters need be typed.

SAV/RES Function: IDENTIFY

From Device? DM0:

Device:	DM0
Pack ID:	STSPAR
Pack Cluster Size:	8
Creation date:	Wednesday, 07-Feb-79

SAV/RES Function: SAVE

From RSTS disk? DM0:

*** Pack ID/default Save Set Name is STSPAR

To Device? HMO:SAVEST

*** Save Set Name is SAVEST

Expiration Date <20-Feb-80>

Verify (Yes or No) (NO)?

Proceed (Yes or No)? Y

*** Initializing first SAVE volume

*** Begin SAVE from DM0 to HMO at 11:08 AM

Disconnect Device:	HMO
Set Name:	SAVEST
Seq #::	1
Density:	1600
Creation date:	Tuesday, 20-Feb-79
Expiration date:	Wednesday, 20-Feb-80

Please label this volume!

--- SAVE operation completed at 11:32 AM

Summary Report

SAVE of DM0:STSPAR to Save Set SAVEST

Input Device:	DM0
Pack ID:	STSPAR
Pack Cluster Size:	8
Creation date:	Wednesday, 07-Feb-79

Output Device:	Reptape
Set Name:	SAVEST
# of volumes:	1
Density:	1600

Figure BKP-6. IDENTIFY and SAVE Functions (Page 1 of 2)

DIGITAL

RSTS/E UPDATE
Backup Programs

Creation date:	Tuesday, 20-Feb-79
Expiration date:	Wednesday, 20-Feb-80

SAVE started on Tuesday, 20-Feb-79, at 11:08 AM

Run Statistics

Transfer Totals:

Total of 14936 blocks transferred

Error Totals:

Total of 8 new bad blocks encountered on source.

Timing Totals:

Total elapsed time:	0 hrs..	4 mins..	7 secs.
Total wait time:	0 hrs..	0 mins..	0 secs.
Total process time:	0 hrs..	4 mins..	7 secs.

SAV/RES Function: ?

Ready

Figure BKP-6. IDENTIFY and SAVE Functions (Page 2 of 2)

RUN \$SAVRES

SAV/RES Function: RESTORE

From Device? MM0:

*** Save Set Name/default Pack ID is SAVEST

To RSTS DM: Disk? DM0:SYSPAK

*** The volume on DM0: is a RSTS disk

*** Pack ID is SYSPAK

*** Pack will be reinitialized

Mount it anyway <NO>? Y

*** Pack ID is SYSPAK

Verify (Yes or No) <NO>?

Proceed (Yes or No)? Y

*** Begin RESTORE from MM0: to DM0: at 11:17 AM

Dismount Device: MM0:
Set Name: SAVEST
Seq #: 1
Density: 1600
Creation date: Tuesday, 20-Feb-79
Expiration date: Wednesday, 20-Feb-80

Dismount Device: DM0:
Pack ID: SYSPAK
Pack Clustersize: 8
Creation date: Wednesday, 07-Feb-79

Please label this volume!

--- RESTORE operation completed at 11:21 AM

Figure BKP-7. RESTORE Function (Page 1 of 2)

Summary Report

RESTORE of Save Set SAVEST to DM0:SYSPAK

Input Device: Magtape
Set Name: SAVEST
of volumes: 1
Density: 1600
Creation date: Tuesday, 20-Feb-79
Expiration date: Wednesday, 20-Feb-80

Output Device: DM0:
Pack ID: SYSPAK
Pack Clustersize: 8
Creation date: Wednesday, 07-Feb-79

RESTORE started on Tuesday, 20-Feb-79, at 11:17 AM

Run Statistics

Transfer Totals:

Total of 14824 blocks transferred

Error Totals:

Total of 0 new bad blocks encountered on destination.

0 files structurally altered.

Timing Totals:

Total elapsed time: 0 hrs., 4 mins., 6 secs.
Total wait time: 0 hrs., 0 mins., 0 secs.
Total process time: 0 hrs., 4 mins., 6 secs.

SAV/RES Function: ^Z

Ready

Figure BKP-7. RESTORE Function (Page 2 of 2)

DIGITAL

RSTS/E UPDATE
Backup Programs

RUN SSAVRES

SAV/RES Function: IMAGE/NOERROR

From RSTS disk? DB0:

*** Input Pack ID/default Output Pack ID is SOURCE

To RSTS DB: Disk? DB1:DESTIN

*** The volume on DB1: is a RSTS disk

*** Pack ID is OUTPUT

*** Pack will be reinitialized
Mount it anyway <NO>? YES

*** Output Pack ID is DESTIN

Verify (Yes or No)? YES

Proceed (Yes or NO)? YES

*** Begin IMAGE copy from DB0: to DB1: at 11:02 AM

*** Begin VERIFY pass from DB0: to DB1: at 11:04 AM

*** 0 differences found

Dismount Device: DB1:
Pack ID: DESTIN
Pack ClusterSize: 8
Creation date: Monday, 22-Jan-79

Please label this volume!

--- IMAGE copy operation completed at 11:07 AM

Figure BKP-8. IMAGE Function (Page 1 of 2)

BKP-16 COMPANY CONFIDENTIAL

DIGITAL

RSTS/E UPDATE
Backup Programs

Summary Report

IMAGE copy of DB0:SOURCE to DB1:DESTIN

Input Device: DB0:
Pack ID: SOURCE
Pack ClusterSize: 8
Creation date: Monday, 22-Jan-79

Output Device: DB1:
Pack ID: DESTIN
Pack ClusterSize: 8
Creation date: Monday, 22-Jan-79

IMAGE copy started on Monday, 22-Jan-79, at 11:02 AM

Run Statistics

Transfer Totals:

Total of 24400 blocks transferred

Error Totals:

Total of 0 bad compares.
Total of 0 new bad blocks encountered on source.
Total of 0 new bad blocks encountered on destination.

0 files structurally altered.

Timing Totals:

Total elapsed time: 0 hrs., 5 mins., 22 secs.
Total wait time: 0 hrs., 0 mins., 0 secs.
Total process time: 0 hrs., 5 mins., 22 secs.

SAV/RES Function: ^Z

Ready

Figure BKP-8. IMAGE Function (Page 2 of 2)

BKP-17 COMPANY CONFIDENTIAL

Additional SAVE/RESTORE Documentation

The RSTS/E SAVE/RESTORE system program is a disk backup and copy utility that provides you with four operational functions. SAVE, RESTORE, and IMAGE perform the backup, recreate, and copy functions respectively, while IDENTIFY extracts label information from a SAVE/RESTORE volume or RSTS/E disk. This documentation describes these four functions, how they are invoked and how and when they should be used.

When to Use SAVE/RESTORE

SAVE/RESTORE and BACKUP each possess similar but different system functions. SAVE/RESTORE was developed to provide a non-selective, fast volume backup, and copy capability that requires few operator responses during the operational dialogue. Unlike BACKUP, SAVE/RESTORE processes entire volumes only, and does not allow selective file transfers, or file deletes. SAVE/RESTORE does not require operator intervention when it encounters bad blocks. For these reasons, use the SAVE/RESTORE package when you need to:

1. Create a fast, reliable copy of an entire RSTS/E disk,
2. Backup files that are larger than 65,535 blocks, and
3. Make a fast copy to a disk that contains bad blocks which may not be reflected in the BADB.SYS file.

Definitions of SAVE/RESTORE Terms

The SAVE/RESTORE terms defined here are used throughout this documentation.

SAVE Format	The format of the output written by a SAVE operation and read by a RESTORE operation.
SAVE Set	The set of magtapes or disks created by a SAVE operation. A SAVE Set must be composed entirely of disks or tapes, not a combination of the two device types. However, different drive types may be mixed within one set, for example, two RK05's and one RK06.
SAVE Volume	One of the magnetic tapes or disks of a SAVE Set.
SAVE Set Name	One to six alphanumeric characters used to identify a SAVE set. By default, the SAVE Set Name is the same as the Pack ID from which it was created. You can specify another name for the SAVE Set rather than accept the default.
LIKE Disks	Like disks are units of the same device size. For example, SAVE/RESTORE considers the following devices LIKE: <ul style="list-style-type: none"> o An RP06 is like another RP06 o An RM02 is like an RM03 o An RP04 is like an RP05 o An RK05J is like one unit of an RK05F

As a further clarification, if you copy an RM02 to a SAVE Set, that set can later be RESTORED to either an RM03 or another RM02.

Running SAVE/RESTORE

You can invoke the SAVE/RESTORE program off-line while you are running the system initialization code (INIT.SYS). To do this, type SAVRES on the console terminal in response to the INIT "Option:" prompt. After you press the RETURN key, SAVE/RESTORE requests the current date and time then responds with its own "SAV/RES Function:" prompt. The following example illustrates the proper response to the INIT "Option:" prompt and the appearance of the SAVE/RESTORE program prompt.

Option: SAVRES

DD-YY-YY? 12-JAN-79
HH:MM? 12:15

SAV/RES Function:

To invoke SAVE/RESTORE on-line, you must type RUN SSAVRES, then press the RETURN key. After you issue these commands, RSTS/E prints the SAVE/RESTORE program prompt, "SAV/RES Function:". The example that follows represents the steps you take to initiate SAVE/RESTORE on-line.

RUN SSAVRES <CR>
SAV/RES Function:

Once the "SAV/RES Function:" prompt appears, either as a result of invoking SAVE/RESTORE off-line or on-line, you are under the control of SAVE/RESTORE and are ready for backup processing.

Next, select from the following table one of the following four operational functions: SA[VE], RE[STORE], IM[AGE], or ID[ENTIFY]. Unless you need to exit SAVE/RESTORE (by using <LF> or ^Z) or need further help (by typing HE[LP] or <CR>), type one of the four operational functions on the console terminal in response to the "SAV/RES Function:" prompt. The following example illustrates this procedure:

SAV/RES Function: SAVE

SAVE/RESTORE Functions

SA[VE]	Creates a copy of a RSTS/E file-structured disk. The SA[VE] function backs up to disk or tape.
RE[STORE]	Recreates a RSTS/E file-structured disk from a SAVE Set.
IM[AGE]	Copies a RSTS/E file-structured disk to a LIKE disk.
ID[entify]	Prints label information and other volume characteristics of a SAVE volume or a RSTS/E file-structured disk.
HE[LP]	Prints a table of SAVE/RESTORE functions.
<LF>	Press the LINE FEED key if you want to exit the SAVE/RESTORE program environment. If you invoke SAVE/RESTORE with a RUN command, pressing <LF> returns you to your default Run-Time System. If you are running SAVE/RESTORE off-line with INIT.SYS, pressing <LF> returns you to the "Option:" prompt issued by the INIT program.
<CR>	Press the RETURN key and the "Type HELP for help" message appears on your terminal.
^C	Type ^C to produce the same result as pressing the LINE FEED key.
^Z	Type ^Z to produce the same result as pressing the LINE FEED key.
other	Produces the same result as pressing the RETURN key.

Press the RETURN key when you finish entering the selected function. This procedure invokes the SAVE/RESTORE dialogue.

Note that the function response in the previous example represents only one of three different ways you can answer the "SAV/RES Function:" prompt. The response to the prompt determines the type of dialogue to be carried out as well as the operation to be performed. A description and an example of each type of response appears in the following table.

Table BKP-2. Response Types for
the "SAV/RES Function:" Prompt

Response Type	Example	Description
Function (by itself)	SAV/RES Function:SAVE	If only a function is given, the full dialogue follows
Function plus Option(s)	SAV/RES Function: SAVE/EXP:10-JUL-79	If the function is followed by options, dialogue questions will follow only for the devices to be used and the unspecified options.
Full Function Command Line	SAV/RES Function: IMAGE DM1:=DM0: MYPAK/VER/NOERR	If a complete command line is given, no dialogue questions will be asked.

During the run, SAVE/RESTORE informs you of its operational status by printing messages on your terminal. Some of the messages are intended for your information; others notify you of error conditions that might require your attention. This latter group consists of errors that can occur during the dialogue, mount, transfer, or verification phases of SAVE/RESTORE.

When the program encounters an error during a dialogue or mount operation, it instructs you to rectify the problem, then asks you to re-enter the appropriate command. These errors never affect the integrity of the data being transferred.

On the other hand, errors that occur while SAVE/RESTORE is transferring or verifying data can jeopardize this data. In this case, SAVE/RESTORE resolves the problem, aborts the run, or requests whether to continue or abort the program. In error situations of this type, the program gives you as much information as it can to help protect your data. That is, SAVE/RESTORE lets you know what it is doing by issuing informational messages as well as error messages to your terminal.

Finally, SAVE/RESTORE prints summary information about SAVE, RESTORE, or IMAGE operations. This report describes the operation performed, device information, run start date and time statistics in addition to run total statistics. The number of errors encountered during the operation are listed as part of the run total statistics. The program prints this report at the end of a run.

SAVE/RESTORE Options

You can include SAVE/RESTORE options (switches) in a function response or in a single line command. The options that appear in the following table are of two types. The first two pairs of options correspond to SAVE/RESTORE dialogue questions and have option defaults. If neither of a pair of options is attached to a function response, an appropriate question will appear in the dialogue. However, if one of the pairs is specified, the corresponding question will not appear in the dialogue. If neither of the pair is specified in a full function command line, the default is assumed.

The last two options in the table have no corresponding dialogue questions. The /ERROR option is a default setting which means that SAVE/RESTORE will print non-fatal error messages and continue processing. Consequently, if you want to have all errors abort the program, you must attach the /NOERROR option to the function response or the full function command line. The /STATS option is a default setting which means SAVE/RESTORE will automatically print a summary report at the end of each run. Consequently, if you want to suppress the summary report, you must attach the /NOSTATS option to your function response.

Table BKP-3. SAVE/RESTORE Options (page 1 of 2)

Option	Default	Description
EX[PIRATION][:date] NOEX[PIRATION]	EXPIRATION	<p>This option is legal only for a SAVE operation. The date argument that you insert is the date after which you may overwrite the destination volume. You are warned by SAVE/RESTORE of any attempt to overwrite the volume(s) any time prior to the specified date.</p> <p>If you specify /EX[PIRATION] with no date argument, SAVE/RESTORE establishes the expiration date as one year after the current date.</p> <p>If you specify /NOEX[PIRATION], SAVE/RESTORE uses the current date as the expiration date.</p>
VE[RIFY] NOVE[RIFY]	NOVERIFY	<p>This option is legal only for a SAVE, RESTORE, or IMAGE operation. If you specify the VE[RIFY] option, SAVE/RESTORE compares the volume(s) to ensure that the transfer was accurate.</p>
ST[ATS] NOST[ATS]	STATS	<p>SAVE / RESTORE prints a summary report at the end of a run, unless you specify the /NOST[ATS] switch with either a function response or a single line command.</p>

Table BKP-3. SAVE/RESTORE Option (page 2 of 2)

Option	Default	Description
NOER[ROR] ER[ROR]	ERROR	<p>If you omit the /NOERR[ROR] option and a non-fatal error occurs during a SAVE/RESTORE run, an error message appears and processing continues.</p> <p>If you specify NOER[ROR] and any run-time error occurs, the program terminates after printing an error message.</p>

Dialogue

The SAVE/RESTORE dialogue begins after you invoke the SAVE/RESTORE program and enter a function response to the "SAV/RES Function:" prompt. If you respond with SAVE, RESTORE, IMAGE, or IDENTIFY, the program initiates a set of dialogue questions. All questions in the dialogue have a short form or a long form. The short form appears automatically; the long form prints only if you press the RETURN key in response to the short form question. The long form provides information that may help you with an answer.

Some of the dialogue questions have default values; these are printed in angle brackets next to the question. You may select the default value by pressing the LINE FEED key or by typing the proper default response. If the LINE FEED key is pressed in response to a question with no default value, the question is repeated.

There are two valid responses, ^Z and ^C, in the SAVE/RESTORE package that can alter the existing operating status. For example, if you type ^Z at any point in the dialogue, SAVE/RESTORE returns you to the previous question. If you want to abort the run at any time, type ^C. After the abort, you will return to the "Option:" prompt if you are running off-line under the control of the INIT.SYS

program or to your default Run-Time System prompt if you invoked SAVE/RESTORE on-line with a RUN command. SAVE/RESTORE automatically returns to the "SAV/RES Function:" prompt after the execution of a specified function. This occurs when you are either operating on-line or off-line. You may exit from the INIT environment version of SAVE/RESTORE to the "Option:" prompt by typing LINE FEED or ^Z to the "SAV/RES Function:" prompt.

SAVE/RESTORE Volumes

You can perform SAVE/RESTORE operations on two classes of volumes: disks and tapes. The SAVE Set, a set of tapes or disks created from a SAVE operation, cannot include both disks and tapes. However, the SAVE Set may include one or more volumes of the same class. In other words, a tape volume SAVE Set could consist of two MM devices and one MT device while one RP04 and two RK05 disks constitute a legal disk SAVE Set. SAVE/RESTORE, to repeat, does not allow any disk/tape combination SAVE Sets.

A legal SAVE/RESTORE disk device is any disk supported by RSTS/E. Legal disk devices must meet the following requirements:

1. Must be formatted; that is, contain sector and track information.
2. All disks used by the on-line version of SAVE/RESTORE must have a valid RSTS/E file structure; that is, the disk device must be formatted and DSKINTed prior to its use.
3. The first 16 blocks on a disk must not contain any bad blocks.
4. RSTS/E output disk can contain no more than 161 bad clusters (the RSTS/E limit).

Magnetic tape is the only valid SAVE/RESTORE tape medium; DECTape or cassette are not supported. The statements here define what constitutes a legal tape in the SAVE/RESTORE package.

1. The magnetic tape must be long enough to contain the SAVE Set label and header information, a bootstrap, and INIT.SYS which is on the first volume only.
2. A bad block will be considered to be the end of the tape volume during the SAVE operation.
3. The default density is the lowest density allowed by the drive on which the first volume is mounted. Subsequent volumes cannot be written at any other density. Therefore, ensure that the density used on the first volume is compatible with all other tape drives to be used.

Device Specifications

Device specifications are required as responses to several dialogue questions. These specifications must be in the following general format.

<device>[:[<id>]][/<option(s)>]

where <device> is a two character device type (for disk either "Dx" or "Rx" is acceptable) followed by a single digit unit number in the range 0 to 7. The <id> represents either a SAVE Set name or a disk pack identification and the <options> are selected from the following table.

Table BKP-4. SAVE/RESTORE Device Specifications
(page 1 of 2)

Options	Description
SCR[atch]	The SCR[ATCH] option is legal only on an output volume device specification. If you use the SCR[ATCH] option, SAVE/RESTORE by-passes most volume label checking. SAVE/RESTORE always checks the label to see if an output volume contains a SAVE Set written at the current density and ensures that it is not a volume from the SAVE Set currently being written. It also checks destination disks for bad block information.

Table BKP-4. SAVE/RESTORE Device Specifications
(page 2 of 2)

Options	Description
DEN[SITY]: { 800 } { 1600 }	The DEN[SITY] option is legal only for tape. With this option you can specify the density at which the tape is to be written or read. If SAVE/RESTORE cannot read the tape at the specified density, it will try to read the tape at the other legal density setting.

Checking the Input Volume

SAVE/RESTORE checks the input volume prior to any actual data transfer and warns you of any problem that it foresees in saving, restoring, or copying the volume. All of the error messages that are printed at this point (at least concerning the input volume) are warning messages only; you can recover from them by following appropriate procedures. The error messages that result from a problem with the input volume are listed here and described separately below.

- o **!!!** Input disk has only nnn free clusters.
Mount it anyway <NO>?
- o **!!!** This is not the correct volume.
- o **!!!** This volume has no label.
- o **!!!** Input disk should be "CLEANed".

There must be at least as many good pack clusters on an output RSTS/E disk as there are clusters to be transferred from the original input disk. If SAVE/RESTORE determines that the input disk has very few clusters that are not allocated, it warns you that you may encounter problems copying or restoring the disk.

!!! Input disk has only nnn free clusters.
Mount it anyway <NO>?

Type NO or press the LINE FEED key to return to the previous device prompt. If you type YES, you are instructing SAVE/RESTORE to proceed with the SAVE or IMAGE operation. In the case of an IMAGE copy, the program also

checks the output volume to see if the transfer can be made. No further checking takes place in the case of the SAVE operation. At this point, you may want to free up space on the input disk by deleting unnecessary files. This process will increase your chances of eventually completing a successful RESTORE.

When you specify a Pack ID or a SAVE Set name with the input device specification, SAVE/RESTORE compares the specified name with the name already written on the volume. If the names do not match, the following warning message is printed and the input device prompt is asked again:

!!! This is not the correct volume.

A dismount request follows the above warning message. It lists useful information from the volume label but does so only if it is a SAVE Set or a RSTS/E file-structured disk.

If the input volume is neither a RSTS/E disk nor a SAVE Set, the terminal displays the following warning message and the input device prompt is asked again:

!!! This volume has no label

If you remove a disk from a drive without logically dismounting it and subsequently attempt to use it as the input for a SAVE or IMAGE copy operation, SAVE/RESTORE will print the message that follows:

!!! Input disk should be "CLEANed".

To perform the cleaning operation simply invoke the ONLCLN program if running on-line.

Checking the Output Volume

SAVE/RESTORE checks the output volume before SAVING, RESTOREing, or IMAGE copying a RSTS/E disk. This procedure ensures that the SAVE/RESTORE operations will proceed to completion with the least chance of error. The messages that SAVE/RESTORE issues, as a result of checking the output

volume, pertain to the expiration date, the label, and the available data space on the destination device. Some of the messages are printed only for your information; the rest warn you of serious output device problems. The paragraphs that follow exhibit the messages, describe why they appear, and explain how to respond to them.

When the expiration date of an output volume, labelled as a SAVE Set, has passed, SAVE/RESTORE prints this series of messages:

```
*** The volume on dev: is SAVE Set <xxxxxx>
    Density: nnn
    Creation date: day-of-week, dd-mmm-yy
    Expiration date: day-of-week, dd-mmm-yy
```

and proceeds to the next dialogue question. If you decide not to use the expired date volume, type "Z" to return to the "To Device?" output dialogue question.

If the output volume is labelled as a SAVE Set and its expiration date has not passed, the following series of messages are printed:

```
*** The volume on dev: is SAVE Set <xxxxxx>
    Density: nnn
    Creation date: day-of-week, dd-mmm-yy
    Expiration date: day-of-week, dd-mmm-yy
    !!! Its expiration date has not passed
    Mount it anyway <NO>?
```

If you type NO or press the LINE FEED key in response to the mount question, SAVE/RESTORE returns to the previous device prompt. SAVE/RESTORE will proceed to write over the existing SAVE Set if you type YES. If you press the RETURN key, the following message appears:

```
*** This is your last chance to prevent the
*** loss of any data on the output volume.
Mount it anyway <NO>?
```

SAVE/RESTORE notifies you with the following message that the output volume you are using is labelled as a RSTS/E disk:

```
*** The volume on dev: is a RSTS disk
*** Pack ID is <xxxxxx>
*** Pack will be reinitialized
Mount it anyway <NO>?
```

If you type NO, or press the LINE FEED key, you direct SAVE/RESTORE to return to the previous output prompt. A YES response, at this point, indicates that you wish to use the present volume.

If you RESTORE or IMAGE copy a disk, SAVE/RESTORE checks to see whether the total number of clusters on the disk minus the number of clusters with known bad blocks is greater than or equal to the number of allocated clusters on the original source disk. If SAVE/RESTORE determines that there is not enough space and the transfer cannot be made, it prints:

```
!!! Too many bad blocks on output disk
```

and then returns to the previous device prompt. If, however, the program determines that there are very few free blocks (<10%), it prints the following warning:

```
!!! Only nnt of the output disk clusters are available for relocation.
Mount it anyway <NO>?
```

You must decide at this point whether the number of free clusters is large enough to make possible a successful SAVE/RESTORE operation. If you decide it is not, type NO or press LINE FEED to return to the output prompt. Type YES to direct SAVE/RESTORE to proceed with the operation if you think there are enough free clusters available. Whenever this warning appears, DIGITAL recommends that, when it is possible, you use another disk for output. Follow this procedure since there exists a higher than usual possibility that the operation will not succeed. Note that DIGITAL does not guarantee that the operation will succeed even if this message does not appear.

DIGITAL

RSTS/E SWS UPDATE
Backup Utilities

SAVE/RESTORE normally transfers each allocated pack cluster from the original source disk to the same cluster on the output RSTS/E disk. If the output cluster is found to be bad, SAVE/RESTORE will try to relocate the data. If the cluster size of the item being moved is larger than the pack cluster size, there must be enough contiguous free pack clusters on the disk to accommodate the entire entity cluster. If there is not SAVE/RESTORE will abort the operation.

If the specified output is not a RSTS/E or a SAVE Set disk it prints the new SAVE Set name and proceeds without further notice. The program does not check the output volume label if you specified the /SCR[ATCH] option, except to determine whether it is a volume of the SAVE Set that is currently being created. It always attempts to recover bad block information from a RSTS file structured and SAVE disk.

DIGITAL

RSTS/E SWS UPDATE
Backup Utilities

GENERAL COMPARISON OF BACKUP PROGRAMS

Table BKP-5. Comparison of Backup Programs

	BACKUP	SAVE/ RESTORE	PIP	RMSBCK/ RMSRST
Copy Large Files	No	Yes	Yes	Yes (5)
Maintain Position of Placed Files	No	Yes(1)	Yes(2)	No
Selective Backup and Restore	Yes	No	Yes	Yes
Backup and Restore Public Structure On-line	Yes	No	Yes(3)	Yes
Backup and Restore Private Structure On-line	Yes	Yes	Yes(3)	Yes
Compensates for Bad Blocks on the Output Medium	Yes	Yes(4)	No	No
Creates Bootable Volumes	No	Yes	No	No
Runs Off-line	No	Yes	No	No
Restores Files to Different Accounts then Backed up From	No	No	Yes	Yes

- (1) SAVE/RESTORE maintains the position of placed files upon recovery as long as bad blocks on the output disk do not interfere with the placement of the first cluster of the file.
- (2) PIP maintains the position of a file if the user specifies that the output file is to be positioned. If PIP cannot position a file at the user specified block, it will attempt to create the file between the specified block and the end of the disk. If it still cannot find a space to create the file, it will start at the beginning of the disk and attempt to find space for the file. If the file is located at a lower numbered disk block than requested, it is not marked in the directory as a placed file. If it cannot be created at all, the "No room for user on device" error is generated.
- (3) PIP does not store project, programmer numbers in ANSI magtape labels. Therefore, it is difficult to recover files to multiple accounts from ANSI formatted magtapes. DOS magtape labels do store project, programmer numbers, but do not store RMS-11 attributes. DOS format also does not support multi-volume magtape files and therefore cannot support large files that do not fit on one volume.
- (4) Bad blocks on the output disk cause contiguous files to be made non-contiguous.
- (5) RMSBCK can copy large files as long as they can fit on the current output volume. There is no multi-volume capability allowing files to cross a volume boundary.

TRANSFERRING RMS RELATIVE AND INDEXED FILES TO OTHER NON-RSTS/E OPERATING SYSTEMS

1. File or files fit on one magtape
 - a. Use RMSBCK on the RSTS/E system.
 - b. Use RMSRST on the other system.
2. File requires more than one magtape
 - a. Use RMSCNV to do a disk to disk conversion to an RMS Sequential file.
 - b. Use PIP to transfer the sequential file to ANSI magtape with no switches.
 - c. On the other system
 1. Relative file - use RMSDEF and RMSCNV
 2. Indexed file - use RMSDEF and RMSIFL
3. Another possible consideration, for example:
 - a. If you have an 86,000-block indexed file (400,000 82-byte records), the back-up (RMSBCK) copy of the file requires at least two magnetic tape volumes.
 - b. However, if you convert (RMSCNV) the file to tape with 4096-character blocks, you need only one 2400-foot tape volume.
 - c. You can then restore the file quickly with RMSIFL.

RSTS/E V7.0 SWS Update

UUP

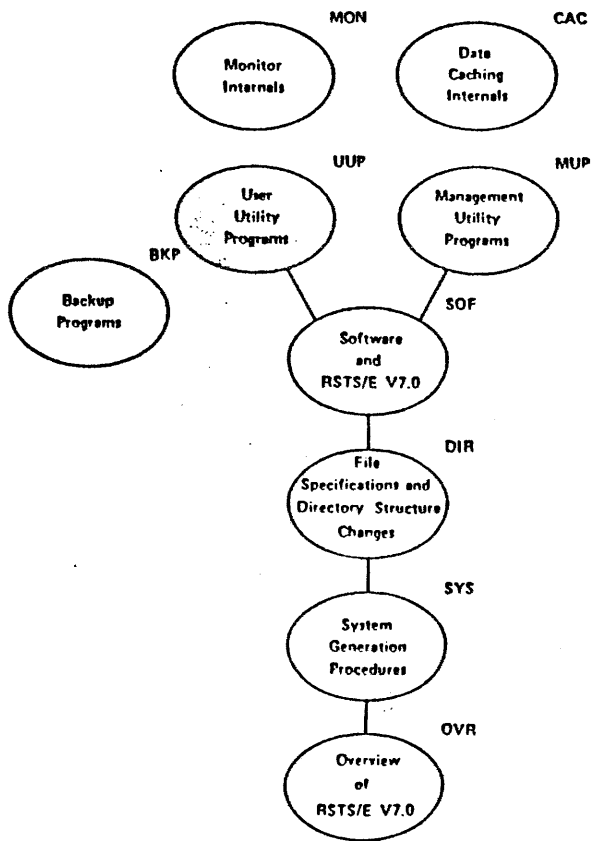
User Utility Programs

User Utility Programs

CONTENTS

Introduction.....	UUP-1
FIT Program.....	UUP-3
Additional FIT Documentation.....	UUP-3
Transferring Files with FIT.....	UUP-4
Maintaining RT11 Format.....	UUP-7
DOS Disk Directory Listings.....	UUP-11
RMSIFL Program.....	UUP-12
Additional RMSIFL Documentation.....	UUP-12
RSMIFL Utility Switches.....	UUP-20
RMSCNV Program.....	UUP-29
RMSDEF Program.....	UUP-30
DIRECT Program.....	UUP-32
BPCREF Program.....	UUP-33
COPY Program.....	UUP-37
FLINT Program.....	UUP-37
FILCOM Program.....	UUP-38
GRIPE Program.....	UUP-40
QUOLST Program.....	UUP-40

COURSE MAP



CONFIDENTIAL

RSTS/E V7.8 UPDATE
User Utility Programs

User Utility Programs

INTRODUCTION

As RSTS/E users you know the wide range of services offered by the CUSPs supplied with a RSTS/E system. Some of these services include:

- o File manipulation
- o File transfer
- o Directory listing
- o Batch processing
- o File spooling

RSTS/E V7.8 includes new and revised utilities which provide you with greater flexibility in handling your day to day and long term needs.

In this module you will learn these new and revised non-privileged utility programs.

This page is for notes.

FIT PROGRAM

- o Transfers files between
 - Disk and TUSB
 - Disk and floppy
 - RSTS/E disk and RT11 disk
 - RSTS/E disk and RT11 DECTape
- o Transfers files from a DOS disk to a RSTS/E disk
- o Lists directory of a TUSB and floppy
- o Lists directory of a DOS disk

Additional FIT Documentation

TUSB DECTape II and floppy disk files are maintained in the RT11 directory structure format. Thus, DECTape II and floppy disks are directly transferrable between RSTS/E and RT11 systems.

To invoke the FIT program, you must be logged into the system and type the following command:

```
RUN $FIT
```

If FIT is successfully invoked, it prints an identifying message followed by a FIT> prompt. The prompt indicates that FIT is prepared to accept a command line. To cause FIT to print a help text file, type the /HE switch in response to the prompt. To exit from FIT, type CTRL/Z in response to the prompt, for example:

```
FIT>^Z  
Ready
```

Transferring Files with FIT

To transfer a file with FIT, type a command line of the following form in response to the FIT prompt:

```
FIT>[output[/sw]=]input[/sw]
```

where output and input are file specifications and /sw is an optional switch that specifies the structure of the device on which the file resides (RSTS/E, RT11, or DOS).

The input and output file specifications have the following form:

```
dev:[prof,prog]name.ext<prot>
```

where:

dev: is a device specification. If none is specified, the default is the public structure. The output device must not be a DOS disk. The input device can be a RSTS/E, RT11, or DOS disk; or RT11 TU56 DECTape; or RT11 TU58 DECTape II; or RT11 floppy disk with RT11 file structure.

prof,prog is a project-programmer number. If none is specified, the default is your current account. FIT ignores the project-programmer number if the device has an RT11 file structure.

name.ext is a filename and extension. If none is specified for output, FIT uses the input filename and extension. An input filename must be specified.

<prot> is a protection code. If none is specified on output, FIT assigns your current default protection code. FIT ignores any input file protection code specification.

The legal input device specifications for a FIT file transfer are as follows:

```
RSTS/E disk
floppy disk (RT11 format)
TU58 DECTape II (RT11 format)
RT11 disk (with pack clustersize = 1) or TU56 DECTape
(RT11 format)
DOS disk
```

The legal output device specifications (except for DOS input) for a FIT file transfer are as follows:

```
RSTS/E disk
floppy disk (RT11 format)
TU58 DECTape II (RT11 format)
RT11 disk (with pack clustersize = 1) or TU56 DECTape
(RT11 format)
line printer, keyboard, etc.
DECTape (RSTS/E format)
```

If the input device is a DOS disk, the legal output devices are as follows:

```
RSTS/E disk or DECTape
line printer, keyboard
```

The switch specification /sw can be one of the following:

```
/RT11 The file is on an RT11 file structured device
/DOS The file is on a DOS format disk (legal only for an
input file specification)
/RSTS The file is on a RSTS/E disk or device. This
switch is illegal with floppy disk and TU58
DECTape II.
```

These switches can be abbreviated to the first two letters (for example, /RT, /DO, and /RS).

If you do not specify a switch on the input specification, FIT attempts to determine the file structure of the input file device. If the input device does not contain a valid RSTS/E, RT11, or DOS file structure, FIT prints the following error:

?INPUT DEVICE NOT FILE STRUCTURED

If the input device is a RSTS/E format DECTape, FIT prints the following error:

?ILLEGAL INPUT DEVICE

If the specified file does not exist on the device, FIT prints the following error:

?CAN'T FIND FILE OR ACCOUNT

If you do not specify a switch on the output specification, FIT assumes a default file structure based on the type of device. That is, if the device is TU58 DECTape II or floppy disk, FIT assumes the RT11 file structure; if the device is disk or TU56 DECTape, FIT assumes the RSTS/E file structure.

If FIT determines that the output device should have an RT11 file structure, the following errors are possible:

?NO RT11 STRUCTURE ON OUTPUT

The output device does not contain a valid RT11 file structure. Check that the right device was specified. An RT11 structure must already exist on the output device before files can be transferred to it. The /ZE switch can be used to create an RT11 structure on the device.

?DIRECTORY OVERFLOW

The output device has no space for directory entries. However, it may be possible for you to use the /SQ switch to free more space for directory. If this is not successful, you must delete some of the files on the device before more can be added. Note that if you intend to have many small files on the device, you should use the /N switch to allocate more than the default number of directory segments when the device is zeroed.

?DEVICE FULL

The output device has no free space for the file transfer. It may be possible for you to use the /SQ switch to create enough contiguous free space for the additional files. If this is not successful, you will have to delete enough files from the output device to add the files from the input device.

These errors are generated prior to the transfer of any data to the RT11 structured device. If an error is encountered during a transfer, a RSTS/E error is reported and the transfer is aborted. Because the directory entry for the RT11 file is not finalized until the transfer is complete, an error during transfer causes the space reserved for the file to be freed.

If an error is encountered during the open or transfer phase of a RSTS/E file operation, the appropriate RSTS/E error is generated and the transfer is aborted.

Maintaining RT11 Format

In addition to transferring files to and from an RT11 directory structured device, FIT can perform the following operations:

1. You can use the /ZE switch to zero and initialize a device as an RT11 device. This operation creates an RT11 directory structure on the device and must be performed prior to transferring files to and from the device.

DIGITAL

RSTS/E SMS UPDATE
User Utility Programs

RSTS/E SMS UPDATE
User Utility Programs

2. You can use the /LI switch to obtain a directory listing of the RT11 device.
3. You can use the /DE switch to delete a file on an RT11 device.
4. You can use the /SQ switch to compress the data and directory of the device and allow more files to be stored on that device. This operation is useful if the available space on the device is fragmented.

File Deletion on an RT11 Format Device -- To delete a file on an RT11 directory structured device, enter the following command format in response to the FIT> prompt:

```
FIT>dev: filename.ext/DE
```

where dev: identifies the device and filename.ext specifies the file to be deleted. If the specified device is not RT11 directory structured or if the file does not exist, FIT prints an error message. If an error is encountered by FIT as it accesses the device, a RSTS/E error is reported.

RT11 Device Switches -- You can perform zeroing, compressing, and listing operations on RT11 directories by entering the following command format in response to the FIT> prompt:

```
FIT>dev:/sw[/sw] [/sw]
```

where dev: indicates the device on which the operation is to be performed and /sw specifies the operation. The switches you specify can be one or more of the following:

/ZE:m Zero the device and build an RT11 directory structure. If :m is specified, m extra words of directory information for each file are added to the minimal RT11 directory structure. If m is not specified, FIT adds 13 extra words, by default, to the directory structure. These words are used to store the run-time system name and file attributes.

Note that RT11 systems currently ignore these extra directory entries. However, if future releases of RT11 attach significance to these words, a specification of /ZE:0 may be required for RT11 files that are read by RT11 systems. FIT currently ignores extra directory information unless there are 13 extra words (this may change in future releases). Therefore, you should only specify values of 0 or 13 in the /ZE switch.

/N:n

Used with the /ZE switch to cause n directory segments on the initialized device. Each segment consists of two 512-byte blocks. The value of n must be in the range of 1 to 31 and determines the size of the directory on the device (and, therefore, the number of files that can be stored on the device). The number of files that can fit on a device is determined by the following formula:

$$n * \text{INT}(507/m+7)$$

where n is the number of directory segments and m is the number of extra words for each directory entry (that is, /N:n and /ZE:m, respectively).

Note that you may have to compress the device with the /SQ switch before the maximum number of files is reached.

If you do not specify n, FIT applies defaults based on the device type. These defaults are indicated in the following table. The table also lists the maximum number of files allowed on a particular device corresponding to your specification in /ZE:n.

Table UUP-1. Defaults by Device Type

Device	Default number of Directory segments	Max. number of files /ZE:0	Max. number of files /ZE:13
TU58 DEctape	4	288	104
Floppy disk	4	288	104
RK05 disk	16	1152	416
RK06 disk	31	2232	806

/LI or /DI List the directory of the device. The information listed includes filenames, extensions, file size, protection codes, dates of creation, and position on the device. If the device has 13 extra words for each directory entry (/ZE:13, the default value), the run-time system name and file attributes are also printed. Those sections of the device which are free space are marked <UNUSED> in the listing.

/RT11 Used with the /LI or /DI switches. /RT11 indicates that the device must have RT11 directory structure. That is, if this switch is not present and the device is a DOS disk, a directory of the DOS disk is listed. If this switch is present and the device is a DOS disk, an error is returned.

/SQ Compress (squeeze) the device. This switch allows you to compress the files that are currently on the output device and make room for files that are to be transferred. The space on RT11 devices that is allocated to the directory or files can become fragmented. If this is the case, an attempt to add another file to the device can result in a ?DIRECTORY OVERFLOW or ?DEVICE FULL error. You can use the /SQ switch to compress the contents of the device and provide enough room for the additional files.

Note that because of the manner in which RT11 directory entries are made, you may need to compress the device to use the directory space even if files are added contiguously without deletions.

All of these switches can be abbreviated to one letter (for example, /Z, /N, /L, /D, /R, and /S).

DOS Disk Directory Listings

To list the directory of a DOS disk, enter the following command format in response to the FIT prompt:

```
FIT>dev: [/DO[S]]/L[I]
```

where dev: is the device designator of the DOS disk. The switch /D or /DI can be used in place of /L or /LI. If the /DOS switch is omitted, FIT lists the directory of the device whether it is DOS or RT11 directory structured. If the /DOS switch is present, an error is returned if the device is not a DOS disk.

The listing generated by this operation includes the following information: account numbers (UICs), filenames, extensions, file lengths, protection codes, dates of creation, the first and last block numbers of the files, and whether or not the file is contiguous.

RMSIFL PROGRAM

- o Provides fastest population of multi-key indexed files
- o Input file can have any RMS file organization
- o Output file must be previously created and must be empty

Additional RMSIFL DocumentationPurpose

The RMSIFL utility provides the fastest, most direct method of populating an empty RMS-11 Indexed file. The utility can read records from any type of RMS-11 file and load them into an Indexed file that you have already created. However, RMSIFL does not use the standard RMS-11 access methods to build the output file. It exploits the basic structure of Indexed files to insert data records and construct indexes as fast as possible.

Although RMSCNV may compete with this utility when populating a single-key indexed file, RMSIFL is definitely faster when it comes to multi-key file loading. However, you must use RMSCNV if the output file is not empty.

Effect

RMSIFL reads records from an input file and loads them into an output Indexed file in a series of phases.

1. The utility examines the command string, rejecting it for syntax and/or other errors. When it has a valid command string, RMSIFL determines if the input file must be sorted into ascending order by the output file's Primary Key.

RMSIFL bypasses the sort phase (phase 2) and skips to phase 3 if:

- o You included the Infile Switch NOSO.

- o The key of reference specified for an Indexed input file is equal to or contains the output Primary Key, based on position only. Nonstring key types must match exactly, in starting position and length. String keys must start in the same position, but the Primary Key can be the same length as or shorter than the input key of reference.

You specify key of reference with the KR Infile Switch.

Example The ten-byte string input Alternate Key 2 starts at byte 0 and is specified as the key of reference. The eight-byte string output Primary Key starts at byte 0. RMSIFL does not sort.

Example The ten-byte string input Primary Key starts at byte 0. The eight-byte string output Primary Key starts at byte 3. RMSIFL sorts the input file.

Example The three-byte packed decimal input Alternate Key 1 starts at byte 43 and is specified as the key of reference. The three-byte packed decimal output Primary Key starts at byte 42. RMSIFL sorts the input file.

RMSIFL notifies you that it has started processing by printing the following message:

PRIMARY KEY:

2. RMSIFL reads the input file in Sequential Access Mode, depending on the organization of the input file:

INPUT FILE ORGANIZATION	PROCESSING
SEQUENTIAL	RMSIFL reads the records in the Sequential Access Mode, starting with the first record in the file.
RELATIVE	RMSIFL reads the records in the Sequential Access Mode, starting with record cell 1.
INDEXED	RMSIFL reads the records in the Sequential Access Mode, following the key of reference specified in the command string (the Primary Key is default).

RMSIFL passes each record to modules extracted from the SORT-11 utility. These modules order the input records by ascending value in the Primary Key field of the output file.

At the beginning of this phase, RMSIFL allocates disk space for three sort work files, each of which is the size of the 1.5 times the input file.

Note

Performance in this phase depends entirely on the SORT-11 utility. See the PDP-11 SORT Reference Manual for information on the space required for temporary sort files and guidance on the factors affecting sort performance.

RMSIFL notes this phase by printing the following messages:

SORT HAS STARTED
SORT MERGE PHASE HAS FINISHED

3. RMSIFL loads the ordered input records into the output file, constructing Level 0 of the Primary index one bucket at a time.

The manner in which RMSIFL reads input records depends on whether the input file was sorted in phase 2:

- o If RMSIFL sorted the input file, the utility accepts the input records one at a time from the sort modules. This technique avoids the need for a temporary file containing the input records sorted in output Primary Key order.
- o If RMSIFL did not sort the input file, the utility reads the records from the input file.

RMSIFL examines each record to ensure that it is compatible with the output file:

- o If the record is not compatible, RMSIFL processes it as an exception record according to the switch you have specified in the command string. See the discussion of the ER: [filespec] and NOER Infile switches for the causes of exception records and their handling.
- o If the record is compatible, RMSIFL inserts it into a Level 0 bucket being built in memory.

As the utility places each record, the utility extracts the values for each of the Alternate Keys defined for the file, if any:

- a. RMSIFL converts nonstring key data types into a stringlike representation for sorting only.
- b. RMSIFL combines the Alternate Key value with the Record's File Address (RFA) of the user data record that supplied the key value. RFAs are six bytes long. The utility then appends a byte with the key number and the converted version of the key value.

- c. These fixed-length Alternate Key records are written into a separate temporary file named IFL0.TMP. RMSIFL creates this file with an initial allocation of 100 blocks and a Default Extension Quantity of 100 blocks. You can calculate the final size of IFL0.TMP (ALQ) as follows:

$$ALQ = NDRF * NAK * (LKS + 7 + PAD)$$

where NDRF is the number of user data records in the input file,

NAK is the number of Alternate Keys defined for the output file

LKS is the size of the longest Alternate Key defined for the output file (and shown by RMSDSP), in bytes, and

PAD is the number of null bytes added to word align the record:

$$PAD = 0 \text{ if LKS is an even number}$$

$$PAD = 1 \text{ if LKS is an odd number}$$

Remember to round ALQ to the nearest multiple of 100.

During this process, the utility assembles the index levels of the Primary index as well. RMSIFL puts an index record into a Level 1 bucket for each Level 0 bucket it writes to the file. When it writes a Level 1 bucket to the file, the utility inserts an index record into the Level 2 bucket, and so on, up to and including the Root.

4. RMSIFL sorts the Alternate Key temporary file into order by key number and ascending key value.

At the beginning of this phase, RMSIFL allocates the disk space for sort work files, each of which is 1.5 times the size of IFL0.TMP.

The utility notes this phase by printing the following messages:

```
ALTERNATE KEY
SORT HAS STARTED
SORT MERGE PHASE HAS FINISHED
```

5. RMSIFL builds the Alternate indexes one at a time, from Level 0 up to the Root, using the sorted temporary file for input.

The utility notes the construction of each alternate index by printing the following message:

```
ALTERNATE KEY(s) nnn
```

where nnn starts with 1.

6. RMSIFL prints the following messages when the output file is completely built, then terminates the processing of the current command string.

```
NUMBER OF INPUT RECORDS: nnnnn
NUMBER OF OUTPUT RECORDS: nnnnn
NUMBER OF EXCEPTION RECORDS: nnnnn
```

Note

RMSIFL extends the output file if it cannot contain the input records and the index(es). The utility minimizes this overhead by extending the file a set quantity each time it requires space. This quantity is the integral multiple of a bucketsize greater than 50.

Call and TerminationUsing the CCL command -

IFL [command string]

If you include a command string, the utility attempts to execute it and then returns control to your default run-time system.

If you do not specify a command string, RMSIFL assumes control and prints the prompt.

IFL>

You may enter a command string or CTRL/Z to terminate the utility. When RMSIFL has executed a command string, it prints the prompt.

Using the RUN command -

RUN SRMSIFL

The utility assumes control and prints the prompt:

IFL>

You may type a command string or CTRL/Z to terminate the utility. When RMSIFL has executed a command string, it reprints the prompt.

Command StringGeneral Form -outfile[/switch]..]=infile[/switch]..]

where outfile is the filespec of an existing Indexed file that is to receive the records of the input file. Wild card characters are not permitted in any field of the specification.

RMSIFL checks for the following conditions in the output file. If they are not met, the utility prints the indicated error message and terminates execution of the command string.

- o The output file must exist before you run the utility, otherwise, RMSIFL prints the message:
?IFL--FILE NOT FOUND - filespec
- o The output file must be empty except for the prologue written when the file was created. Otherwise, RMSIFL prints the message:
?IFL--INDEX OUTPUT FILE NOT EMPTY
- o The output file must be an RMS-11 Indexed file. Otherwise, RMSIFL prints the message:
?IFL--OUTPUT FILE MUST BE AN INDEX FILE
- o The output file must not have a bucket size greater than five blocks. Otherwise, RMSIFL prints the message:
?IFL--THERE IS NOT ENOUGH IFL MEMORY FOR THE CURRENT COMMAND LINE
- o The output file cannot have more than ten keys defined. Otherwise, RMSIFL prints the message:
?IFL--TOO MANY KEYS - filespec

Infile

is the filespec of the input file that is the source of records to be written to the output file. Wild card characters cannot appear in any field of the filespec.

You can specify only one input file, but it can have any RMS-11 file organization. RMSIFL does not affect the input file in any way.

The utility opens the input file access read, allow no-write. Therefore, no writing task can access the input file while RMSIFL is operating.

The input file cannot have more than ten keys defined. Otherwise, RMSIFL prints the message:

?IFL--TOO MANY KEYS - filespec

You cannot use a back-up file produced by RMSBCK as input. RMSIFL terminates with the following error message:

?IFL--OPEN ERROR ON FILE - filespec

?IFL--FATAL RMS ERROR STS = -01424, STV = 00000

switch may be one of the codes shown in the table below and described on the following pages.

Note

A command string may also consist of the word "HELP" or a question mark (?). RMSIFL responds with a HELP message.

RMSIFL Utility Switches

Type	Switch	Description	Default Process
String ? or HELP		Print HELP message.	No help.
Global /ID		Identify current version	No id.
Outfile /ER		Write exception records on terminal.	Same.
	/ER: <u>filespec</u>	Write exception records into specified file.	Write exception records on terminal.
	/LO	Honor fill numbers	Fill buckets.

	/NOER	Stop processing immediately if input record is incompatible.	Write exception records on terminal.
	/PD:[#]x	Pad input records to output record length.	Handle input as exception record if different lengths.
	/TR	Truncate input records to output record length.	Handle input as exception record if different lengths.
Infile	/DE:dvn[:dvn [dvn(:)]]	Reassign devices for sort work files.	Create and use sort work files on SYSDEV.
	/KR:n	Key of reference number.	Primary Key (n=0).
	/NOSO	Do not sort before loading.	Sort input file before loading.

Global Switches -

- o ID causes RMSIFL to print the version number of the utility that is operating in the form:

IFL -- VERSION 1.8nn

where nn is the patch level of the utility.

This switch may appear alone on a command string.

Outfile Switches

- o ER:[filespec] directs RMSIFL to continue processing if it encounters a record in the input file that cannot be written into the output file. Exception records can exist for one of the following reasons:

- o You specified the NOSO Infile Switch and the utility found a record that was not in ascending collating order by Primary Key value. The error message is:

?IFL--EXCEPTION RECORD: RECORD OUT OF SEQUENCE

- o You did not allow duplicates in at least one key of the output file, but an input record contains a duplicate value in that key value. The error message is:

IFL--EXCEPTION RECORD: DUPLICATE KEY WHERE NOT ALLOWED

- o An input record is not long enough to contain the Primary Key of the output file. The error message is:

IFL--EXCEPTION RECORD: RECORD TOO SHORT FOR PRIMARY KEY

- o An input record is too long to fit in a bucket of the output file. The error message is:

IFL--EXCEPTION RECORD: RECORD TOO LONG FOR BUCKET SIZE

- o An input record is not compatible with the output file's fixed-length record format and you have not specified the PD and/or TR switches. The error message is:

IFL--EXCEPTION RECORD: RECORD DOESN'T FIT FIXED LENGTH

- o An input record is longer than the output file's variable-length Maximum Record Size and you have not specified the TR switch. The error message is:

IFL--EXCEPTION RECORD: RECORD TOO LONG FOR MAXIMUM RECORD SIZE

If RMSIFL encounters an exception while it is processing the Alternate Index(es), it flags the data record in the Primary Level 0. The utility does not remove the record from the file. The record may be compressed during later use of the Indexed file.

You can add corrected exception records to the output file after RMSIFL terminates with either an application program or RMSCNV (possibly using your terminal as the input device).

If you do not specify an argument (filespec), RMSIFL prints the appropriate error message, then the exception record on your terminal.

If you specify a nondisk file or a terminal, the utility creates it as an RMS-11 Sequential file with variable length records when the first exception record is encountered. Then, RMSIFL writes both the error message and the exception record into the file. RMSIFL uses this procedure for every exception record until it finishes processing.

If you specify a disk file, the utility creates it as an RMS-11 Sequential file with VFC records when the first exception record is encountered. Then RMSIFL writes the exception record into the variable portion of the record and a four-byte exception code into the fixed control area. The codes are:

Code	Meaning
001	Record is out of Primary Key sequence.
002	Record contains an illegal duplicate value.
003	Record is too short to contain output Primary Key.
004	Record is too long to fit in output bucket.
005	Record is not correct size for output fixed length record.
006	Record is too long for output Maximum Record Size.

Also see the NOER switch.

Default If you do not include the ER switch, RMSIFL prints an appropriate error message first, then the exception record on your terminal.

- o LO directs RMSIFL to write records into buckets according to the fill numbers established when the file was created.

Default If you do not include the LO switch, RMSIFL inserts as many records as possible into each bucket.

- o NOER[:S] directs RMSIFL to print an error message, delete the output file, and terminate processing immediately if it encounters an input record that cannot be written into the output file; see the ER:[filespec] switch for reasons and error messages.

RMSIFL deletes the output file unless you specify the SAVE argument (:S) in the NOER switch. Termination at this point leaves the output file incomplete: it is not a valid file organization and cannot be used for any RMS-11 purpose.

Default If you do not include the NOER switch, RMSIFL responds according to the version of the ER switch you used.

- o PD[:{0}x] directs RMSIFL to pad records read from the input file to the output file's record length before writing them to the output file. Padding character is specified as follows:

Switch	Character
PD	NULL
PD:x	x is ASCII A-Z, 0-9, or special Character except \$, ?, and @
PD:#x	x is octal number 000-377 (40 for SPACE, 43 for \$, 77 for ?, and 100 for @)

You use the PD switch only when the output file specifies fixed-length records.

Default If you do not include the PD switch, and the input records are shorter than the output file's records, RMSIFL treats them as exception records.

See the ER and NOER switches.

- o TR directs RMSIFL to truncate records read from the input file to the output file's Maximum Record Size before writing them to the output file. The trailing bytes of the records are truncated.

Default If you do not include the TR switch, and the input records are longer than the input file's record RMSIFL treats them as exception records.

See the ER and NOER Switches.

Infile Switches

- o DE:dv[n]:dv[n]:dv[n][:]] directs RMSIFL to reassign the devices where it allocates the three sort work files. You can specify from one to three disk devices with the switch, where dv[n] is a physical device name and number or a logical name up to six alphanumeric characters. You separate device names by a colon (:); the final colon is optional.

If you include the DE switch in the command string, RMSIFL reassigns all three sort work files:

- o If you specify one device name, the utility reassigns the three work files to that device.
- o If you specify two device names, the utility reassigns one work file to the first device and two work files to the second.
- o If you specify three device names, the utility reassigns one work file to each device.

For optimum performance, you should assign each sort work file to a different device. However, if you have only two devices available, assign the first and third files to one device and the second file to the other device.

Example /DE:DM0:DM1:DM2

Example /DE:DM0:DM1:DM0

Default RMSIFL allocates the sort work files on SY:

- o KR:n directs RMSIFL to read an input Indexed file according to the key specified by n, where n is 0 for the Primary Key, 1 for the First Alternate Key, and so on.

The KR switch can eliminate the need for sorting the input file; see the "Effect" section.

If you use the KR switch with a Non-Indexed file, RMSIFL terminates with the following message:

?IFL--/KR NOT ALLOWED FOR SEQUENTIAL OR RELATIVE FILE

Default If you do not include the KR switch, and the input file is Indexed, RMSIFL reads the file via the Primary Index.

- o NOSO directs RMSIFL to bypass its sort phase of processing because you have already sorted the input file. If the input file is not in order by ascending value of the output file's Primary Key, RMSIFL's response depends on whether you have included the NOER switch or a version of the ER switch.

Default If you do not include the NOSO switch, RMSIFL sorts all records from the input file into the proper order. See the "Effect" section.

Cautions

- o RMSIFL ignores the fixed control area of VFC records in both the input and output files.
- o When both the input and output files have fixed-length records, RMSIFL requires either the TR or PD switch if the fixed record lengths are not equal. If neither switch is specified, the utility terminates with the error message:

?IFL--INPUT AND OUTPUT RECORD SIZES DO NOT CORRESPOND.

- o When the input file contains variable-length records and the output file contains fixed-length records, RMSIFL requires both the TR and PD switches. If both are not specified, the utility terminates with the error message:

?IFL--SWITCH /TR OR /PD OR BOTH ARE NEEDED FOR THIS RECORD

DIGITAL

RSTS/E SMS UPDATE
User Utility Programs

o Example

RUN RMSIFL
IFL>

```

          RMS-11 INDEXED FILE LOAD UTILITY
          -----
          FUNCTION: TO BUILD INDEXED FILES
          FORMAT : IFL OUTFILE[OPTIONS]-INFILE[OPTIONS]

GLOBAL OPTIONS:
  / OR HELP          TYPE THIS MESSAGE
  /ID                IDENTIFY CURRENT VERSION AND PATCH LEVEL

OUTFILE OPTIONS:
  /ER[:FILESPEC]    WRITE EXCEPTION RECORDS TO TERMINAL
                   (OR TO FILESPEC)
  /LO                HONOR FILL NUMBERS
  /NOER[:S]         STOP PROCESSING IF INPUT RECORD IS
                   INCOMPATIBLE AND DELETE THE OUTPUT FILE
                   (IF :S IS SPECIFIED SAVE OUTPUT FILE)
  /PD[:[#]X]        PAD INPUT RECORDS TO OUTFILE RECORD LENGTH
  /TR                TRUNCATE INPUT RECORDS TO OUTFILE RECORD LENGTH

INFILE OPTIONS:
  /DE:DYN[:DYN[:DYN[:]]) REASSIGN DEVICES FOR SORT WORK FILES
  /SR:MMN           KEY-OF-REFERENCE NUMBER
  /NOSO            DO NOT SORT INPUT BEFORE LOADING

IFL>NEW.IDX=ONLY.MAS

          KEY CHECK
          PRIMARY KEY:
          SORT HAS STARTED
          SORT MERGE PHASE HAS FINISHED
          NUMBER OF INPUT RECORDS: 1
          NUMBER OF OUTPUT RECORDS: 1
          NUMBER OF EXCEPTION RECORDS: 0
IFL>^Z
Ready

```

Figure UUP-1. RMSIFL

DIGITAL

RSTS/E SMS UPDATE
User Utility Programs

RMSCNV PROGRAM

- o Uses deferred write feature of RMS
- o Allows a switch to specify mass insert feature of RMS
- o Example

```

RUN $RMSCNV
CNV>ONLY1.MAS/MA/FO:IDX=ONLY.MAS
CNV>^Z
?Task exit status: SEVERE ERROR
Ready

```

Figure UUP-2. RMSCNV

RMSDEF PROGRAM

- o Allows specification of new RMS key data types
- o Allows placement of RMS files
- o Example

```

RMSDEF
DO YOU WANT TO GENERATE A COMMAND FILE FOR FUTURE USE(Y/N)?
ENTER FILE SPECIFICATION FOR COMMAND FILE(ONLY).CMD
DO YOU WANT TO CREATE THE FILE YOU WILL BE DESCRIBING(Y/N)?
ENTER FILE SPECIFICATION(ONLY).HAS
IF THE FILE ALREADY EXISTS, DO YOU WANT TO SUPPERSEDE IT(Y/N)?
ENTER FILE ORGANIZATION(SEQ/EDX)
ENTER RECORD FORMAT(1-4)
ENTER MAXIMUM RECORD SIZE
DO YOU WANT CARRIAGE RETURN CONTROL(Y/N)?
DO YOU WANT FORTRAN CHARACTER CONTROL(Y/N)?
IT'S TIME TO DEFINE THE PRIMARY KEY
ENTER DATA TYPE(S)
ENTER POSITION OF KEY
ENTER SIZE OF KEY
ENTER NAME OF KEY
WILL YOU ALLOW DUPLICATE KEYS(Y/N)?
DO YOU WANT TO DEFINE MORE KEYS(Y/N)?
DO YOU WANT TO DEFINE AREAS(Y/N)?
DO YOU WANT PLACEMENT CONTROL(Y/N)?
ENTER LOCATION
ENTER INITIAL ALLOCATION IN BLOCKS
ENTER BUCKET SIZE
ENTER DEFAULT EXTENSION QUANTITY IN BLOCKS
DO YOU WANT A CONTIGUOUS FILE(Y/N)?
THESE QUESTIONS ARE FOR THE PRIMARY KEY:
THE BUCKET SIZE IS 4 BLOCKS.
ENTER FILL NUMBER FOR DATA BUCKETS
ENTER FILL NUMBER FOR INDEX BUCKETS
ENTER CLUSTER SIZE
ENTER PROTECTION CODE
YOUR FILE HAS BEEN CREATED -- ST:(1,235)ONLY.HAS

```

Figure UUP-3. RMSDEF (Page 1 of 2)

```

YOUR FILE HAS BEEN PROCESSED AND A COMMAND FILE GENERATED -- ST:ONLY.CMD
DO YOU WANT TO CLOSE THE COMMAND FILE(Y/N)?
DO YOU WANT TO GENERATE A COMMAND FILE FOR FUTURE USE(Y/N)?
Ready

DIR ONLY.HAS/3
Name Len Size Prot Access Date Time Ciu Bts Pos Dp/r
ST:(1,235)
ONLY.HAS 100CL 4 RD 19-Mar-79 19-Mar-79 01:06 PM 120 ...RSE 01025 0/0
RFR:18:62 FDI:10X USED:101:0 RECS:67
BET:
Total of 100 blocks in 1 file in ST:(1,235)

Ready

```

Figure UUP-3. RMSDEF (Page 2 of 2)

DIRECT PROGRAM

o New Switches

/SZ List file names, extensions and sizes
(same as /SI)

/OP List open and read regardless file
access counts

/MD List files marked for deletion

/PO List file's position on disk

/AT or /SA List attributes symbolically

/OA List attributes in octal

o Example

```
DIR ONLY.MAS/S,SEQ2.BAK/S
Name .Ext Size Prot Access Date Time Clu RTS Pos Op/rr
ST:[1,235]
ONLY .MAS 11 < 60> 15-Dec-78 15-Dec-78 02:19 PM 8 BASIC2 14128 0/0
RF:FIX-62 FO:IDX USED:12:0 RECSI:62 CC:IMP
```

Total of 11 blocks in 1 file in ST:[1,235]

```
Name .Ext Size Prot Access Date Time Clu RTS Pos Op/rr
ST:[1,235]
SEQ2 .BAK 1L < 60> 14-Mar-79 14-Mar-79 10:45 AM 8 RT11 21998 0/0
```

Total of 1 block in 1 file in ST:[1,235]

Ready

Figure UUP-4. DIRECT

BPCREF PROGRAM

o Reports cross reference of local/global symbols

BPCREF command format:
"OUTFILE.CRF=INFILE.BAC[,INFILE.BAS]"
Note the following switches:

/SOURCE Include source code file

/QUEUE Queue file for printing

/NODELETE Don't delete file after queuing

/NOHEAD No header lines (for FILCOM)

/WIDTH:132 Line width for cref output

/PAGE:60 Number of lines per page

/NOCREF No standard CREF printout

/GLOW:0 Global low limit

/GHIGH:32767 Global high limit

/LLOW:0 Local low limit

/LHIGH:32767 Local high limit

The command "FILE" defaults to "FILE.CRF=FILE.BAC".
The command "FILE/S" defaults to
"FILE.CRF=FILE.BAC,FILE.BAS"

o Example

```
RUN $BPCREF
BPCREF X7.0-53 RSTS V7.0-53 The ARK / 5.3
BPCREF command? EVLNFS/WIDTH:80/LL:500/LH:2000/GL:2000/GH:5000
```

Ready

Figure UUP-5. BPCREF

Cross Reference Listing of EVLNFS on 22-Dec-78 at 01:12 PM Page 1

0	0	19999									
0 1005		1005	1070	1090							
0 1020		1020	1037								
0 1037		1035									
0 1040		1037									
0 1045		1050	1060	1060							
0 1070		1060									
0 1090		1032									
0 10100		1020	1045	1060							
0 10200		1030									
0 10250		1045									
0 10300		1060									
0 10600		1000									
0 10700		1050	10010								
BLK.OFFS		102100	102600	103100							
BLO.MASKS		10150									
BYTE.OFFS		102100	102600	103100							
BLO.MASKS		10150	10110								
CHTRS		1060	10600	1070	10700						
DCMA		10300	1030	1030	1030	10450	1045	1045	1045	10500	
DCMA		1050	1050	10600	1060	1060	1060				
DCM.UFDS		1045	102100								
DUMNYS		10150	10460								
DUMOFFS		10150	10460								
DUMUFDS		10500									
END.MASKS		10150	10110								
ERR		10010									
FILCHTS		1040	10600	1060	10600						
HL0.DCHS		10450	1050								
IN		10150	1015	1015	10460	1046	1046				
IGNORES		10100									
LINKS		10150	10350	10370	10450	10600	10110				
LINKS		101100	10150	101100	10110	10110	10110				
LNR.ACCS		1045	102100								
LNR.ATTS		102600									
LNR.EMTS		10100	1015								
LNF.NAMS		1035	1037	102100							
NCLUS		1045	102600								
NCOMS		102600									
NCPUS		102600									
NDEVS		102600									
NFD.DCHS(10)		10150	1030	1045							
NKCTS		102600									
NYSBS		102600									
NQLOS		102600									
OLDCHS		1030	10300	1045	10450	10500	1060	10600			
PACRID10		10100	1010								
PACRID20		10100	1010								
PASSS		102100									
PC3P		10100	1010								
PGI		10050	1005	1037	1050	1090					
PJJ		10050	1005	1037	1050	1090					
PPGGS		1037	1040	102100							
PROJS		1037	1040	102100							
STATS		10100									

Cross Reference Listing of EVLNFS on 22-Dec-78 at 01:12 PM Page 2

UACHTS 102100

Figure UUP-6. BPCREF Printout (Page 1 of 3)

LEADS	1040	10500									
LEADS	10410	1045									
LEADS	10400	1060									
LEADS	10400	1060	103100								
LEADS	10500	1050									
LEADS	10500	1050									
LEADS	1032	1040	101100	10210	10260	10310					
LEADS	1030	1032	1045	1060	1060	101100					
LEADS	1032	1040	101100	10210	10260	10310					
LEADS	1060	103100									
UCHARS	1060	103100									
UCHARS	102100										
USTATS	1035	102100									
WHO.CARESS	103100										
WORDS	102100										

Please check that the following variables have been referenced properly:

BLK.OFFS	BLO.MASKS	BYTE.OFFS	DUMNYS	DUMOFFS
DUMUFDS	IGNORES	LNR.ATTS	NCOMS	NCPUS
NDEVS	NFD.DCHS(10)	NKCTS	NHSSS	NQLOS
PASSS	STATS	UACHTS	UFD.DCHS(10)	UPROTS
WHO.CARESS	WORDS			

Cross Reference Listing of EVLNFS on 22-Dec-78 at 01:12 PM Page 3

Global range is 2000 to 5000

Cross Reference Listing of EVLNFS on 22-Dec-78 at 01:12 PM Page 4

Local range is 500 to 2000

0 1005		1005	1070	1090							
0 1020		1020	1037								
0 1037		1035									
0 1040		1037									
0 1045		1050	1060	1060							
0 1070		1060									
0 1090		1032									
0 10100		1020	1045	1060							
0 10200		1030									
0 10250		1045									
0 10300		1060									
0 10600		1000									
BLO.MASKS		10150									
CHTRS		1060	10600	1070	10700						
DCMA		10300	1030	1030	1030	10450	1045	1045	1045	10500	
DCMA		1050	1050	10600	1060	1060	1060				
DUMNYS		10150	10460								
DUMOFFS		10150	10460								
DUMUFDS		10500									
FILCHTS		1040	10600	1060	10600						
HL0.DCHS		10450	1050								
IN		10150	1015	1015	10460	1046	1046				
IGNORES		10100									

Figure UUP-6. BPCREF Printout (Page 2 of 3)

DIGITAL

RSTS/E RMS UPDATE
User Utility Programs

DIGITAL

RSTS/E RMS UPDATE
User Utility Programs

LHR.DMT8	10100	1015					
WFD.DCMS(10)	10150	1030	1045				
OLDDCM8	1030	10300	1045	10450	10500	1060	10600
PACRID18	10100	1010					
PACRID28	10100	1010					
PCS9	10100	1010					
PG9	10050	1005	1037	1050	1050		
PJ9	10050	1005	1037	1050	1050		
STAT8	10100						
UFD.CLS8	10050	1005					
UFD.DCMS(10)	10060	1000					
UFD.PG8	10000	1000					
UFD.PJ8	10000	1000					
UFD.UFD8	10000	1000					

Please check that the following variables have been referenced properly:

BLO.MASK8	DUMWY8	DUMOFF8	DUMUFD8	IGNORE8
WFD.DCMS(10)	STAT8	UFD.DCMS(10)		

37 Variables	170 References	29 Statements
865 Name bytes	509 Code bytes	
7 Total time	5 CPU time	

R-Words Reserved	Used	Free
Data	2.03	1.02
Code	1.17	.75
Total	4.00	2.40

Ready

COPY PROGRAM

- o Prints warning for disks with bad blocks
- o Reports device causing an error
- o Does not copy bad blocks
- o Does not copy RP06 disks

FLINT PROGRAM

- o Executes faster
- o Zeros floppies
- o Supports RX02 for single density only
- o Accepts a /NH switch to create an output file without a without a header record

Figure UUP-6. BPCREF Printout (Page 3 of 3)

DIGITAL

RSTS/E SWS UPDATE
User Utility Programs

FILCOM PROGRAM

- o Accepts wildcards
- o Has new switches
- o Includes defaults in its prompts
- o Example

```

RUN $FILCOM
FILCOM X7.0-53 RSTS V7.0-53 The ARK / 5.3
File Comparison Program 19-Mar-79 01:29 PM
Output to <KB:>?
Input File #1? FIX.CMD
Input File #2? FIX.BAK
How Many to Match <3>?
BASIC+ Lines <NO>?
Blank Lines <NO>?

Comparing: 1) [1,235]FIX.CMD to 2) [1,235]FIX.BAK

*****
1) [1,235]FIX.CMD
SY:SEQFIX=SY:SEQFIXA/MP
HISEG=BASIC2
*****
2) [1,235]FIX.BAK
SY:SEQFIX=SY:SEQFIX/MP
HISEG=BASIC2

71 Difference Found.

Output to <KB:>? ^Z

Ready

```

Figure UUP-7. FILCOM Defaults

DIGITAL

RSTS/E SWS UPDATE
User Utility Programs

```

RUN $FILCOM
FILCOM X7.0-05 RSTS V7.0-04 Timesharing
File Comparison Program 20-Dec-78 09:52 AM
Output to <KB:>? *.BAK,*.EXP/MA:1/NOBA/BL

Comparing: 1) [100,7]RMSTST.BAK to 2) [100,7]RMSTST.EXP

?Input File #2 '[100,7]RMSTST.EXP' not found

Comparing: 1) [100,7]IIO24A.BAK to 2) [100,7]IIO24A.EXP

?Input File #2 '[100,7]IIO24A.EXP' not found

Comparing: 1) [100,7]FILCOM.BAK to 2) [100,7]FILCOM.EXP

*****
1) [100,7]FILCOM.BAK
BASIC-PLUS CONTINUATION LINES WILL BE CONSIDERED PAR OF A
NUMBERED PROGRAM LINE (I.E., /MA:3/NOBL/NOBA). HOWEVER, IT
*****
2) [100,7]FILCOM.EXP
BASIC-PLUS CONTINUATION LINES WILL NOT BE CONSIDERED PART OF A
NUMBERED PROGRAM LINE (I.E., /MA:3/NOBL/NOBA). HOWEVER, IT
*****
1) [100,7]FILCOM.BAK
YOU SPECIFY THE /PATCH SWITCH, FILCOM DOES COMPARE BLANK LINES
AND DOES CONSIDER BASIC-PLUS CONTINUATION LINES. THAT IS,
A /P SPECIFICATION OVERRIDES THE NORMAL DEFAULTS AND SETS /BA
*****
2) [100,7]FILCOM.EXP
YOU SPECIFY THE /PATCH SWITCH, FILCOM DOES COMPARE BLANK LINES
AND DOES NOT CONSIDER BASIC-PLUS CONTINUATION LINES. THAT IS,
A /P SPECIFICATION OVERRIDES THE NORMAL DEFAULTS AND SETS /BA

72 Differences Found.

2 Differences Found in 1 File of 1 Total Files Compared.

Output to <KB:>? ^Z

Ready

```

Figure UUP-8. FILCOM Switches

DIGITAL

RSTS E DMS UPDATE
User Utility Programs

GRIFE PROGRAM

- o Allows a CTRL Z as well as ESC to end program

QUOLST PROGRAM

- o Shows more than 65535 blocks as >=65K

UMOUNT PROGRAM

- o Allows mounting of an unlabelled ANSI tape.

RSTS/E V7.0 SWS UPDATE

MUP

Management Utility Programs

Management Utility Programs

CONTENTS

Introduction.....	MUP-1
Device Testing Package (DEVTST).....	MUP-3
DEVTST Documentation.....	MUP-3
Introduction to DEVTST.....	MUP-4
Conventions.....	MUP-5
DEVTST Operating Procedures.....	MUP-6
DEVTST Dialogue Errors.....	MUP-10
CPEXER.....	MUP-11
CPEXER Tests.....	MUP-12
CPEXER Dialogue.....	MUP-13
CPEXER Errors.....	MUP-13
CPUTST.....	MUP-15
CPUTST Dialogue.....	MUP-15
CPUTST Errors.....	MUP-15
DSKEXR.....	MUP-17
DSKEXR Dialogue.....	MUP-17
DSKEXR Errors.....	MUP-17
DSKSEK.....	MUP-19
DSKSEK Dialogue.....	MUP-19
DSKSEK Errors.....	MUP-20
DTEXER.....	MUP-21
DTEXER Dialogue.....	MUP-21
DTEXER Errors.....	MUP-21
DXEXER.....	MUP-22
DXEXER Dialogue.....	MUP-23
DXEXER Errors.....	MUP-23
KBEXER.....	MUP-24
KBEXER Dialogue.....	MUP-24
KBEXER Errors.....	MUP-25

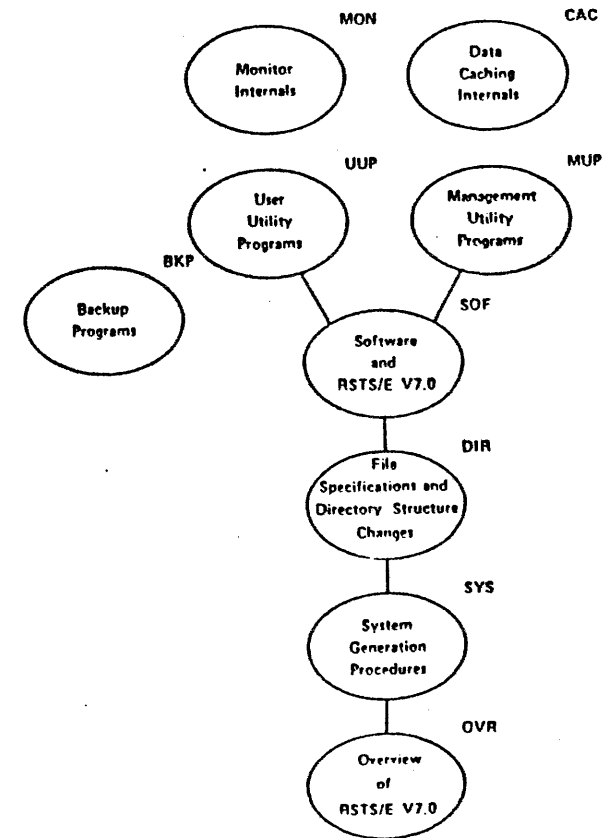
DIGITAL

RSTS/E UPDATE
MANAGEMENT UTILITY PROGRAMS

LPEXER.....	MUP-25
LPEXER Dialogue.....	MUP-25
LPEXER Errors.....	MUP-26
MTEXER.....	MUP-26
MTEXER Dialogue.....	MUP-27
MTEXER Errors.....	MUP-27
PPEXER.....	MUP-28
PPEXER Dialogue.....	MUP-28
PPEXER Errors.....	MUP-28
PREXER.....	MUP-29
PREXER Dialogue.....	MUP-29
PREXER Errors.....	MUP-29
ON-LINE CLEAN Program - ONCLN.....	MUP-31
Enhanced Management Utility Programs.....	MUP-32
Spooling Package.....	MUP-32
SYSTAT Program.....	MUP-35
VTSOPY (VTSOPY) Program.....	MUP-41
TTYSET Program.....	MUP-42
Error Logging Package.....	MUP-43
UTILTY Program.....	MUP-48
ANALYS Program.....	MUP-51
SHUTUP Program.....	MUP-58
LOGIN Program.....	MUP-59
MONEY Program.....	MUP-59
DSKINT Program.....	MUP-59
REACT Program.....	MUP-59
ODT Program.....	MUP-59
Programs That Use Data Caching (If Enabled).....	MUP-60

RSTS/E UPDATE
MANAGEMENT UTILITY PROGRAMS

COURSE MAP



Management Utility Programs

INTRODUCTION

Utility programs supplied with RSTS/E, which relate to system manager functions, allow the manager to:

- o Anticipate possible hardware problems and take appropriate action.
- o Track system activity during time sharing sessions.
- o Gather vital information in case of a system crash.

With RSTS/E V7.6 new management utilities have been supplied while others have been enhanced. With the exception of system generation and back up related programs, this module covers these new and enhanced programs.

This page is for notes.

Management Utility Programs

DEVICE TESTING PACKAGE (DEVTST)

1. Supersedes the UETP package
2. Includes exercisers for:
 - o CPU
 - o Disk
 - o TUS6 DECTape
 - o Terminal
 - o Floppy disk
 - o Line printer
 - o Magnetic tape
 - o Paper tape reader
 - o Paper tape punch
3. Executes under BASIC-PLUS, BASIC-PLUS-2, or RSX Run-Time Systems.
4. Limitations:
 - o CPU exercisers require trigonometric functions and matrix operations.
 - o No control programs exist

DEVTST Documentation

The DEVTST package exercises hardware on the RSTS/E V7.0 system during normal time-sharing operations. DEVTST is used to stimulate hardware errors on the system.

Introduction to DEVTST

DEVTST is a package of programs that provides the customer, System Manager or Digital Field Service personnel with tests that verify the reliability of RSTS/E supported hardware. Execution of any DEVTST program is limited to users with privileged accounts. Operational data, conventions, complete device testing information and error messages are explained in this documentation. After execution of individual DEVTST exercisers, users should check the System Error Log for additional errors.

The DEVTST exercisers are tests that can be initiated whenever hardware trouble is suspected. These device exercisers generate excess activity in an attempt to make the hardware fail during operation.

The DEVTST package should be used in conjunction with the RSTS/E Error Logging package. The Error Logging facility provides the means for determining detailed information on hardware errors. In addition, the DEVTST programs return error messages which may be useful in determining the failure; however, these messages are not as specific as the information provided by the System Error Log.

Conventions

The following conventions are used throughout the DEVTST package dialogue:

< >	Indicates default answer.
()	Shows all possible replies.
[]	Gives an optional reply; for example ASC[II], means enter either ASC, ASCII or ASCII.
<ALT MODE> or <ESC>	This key moves the program back one question. If you type this key as a response to the first question, DEVTST will return to the RSTS/E command level.
CTRL/Z	The CTRL/Z combination causes the program to stop execution and return to RSTS/E command level. CTRL/Z is detected only when the system checks for new input.
CTRL/C	Typing CTRL/C causes the program to stop execution immediately and return to a command level.
RET	Use the RET key to terminate the input line.
Detach <NO>	All of the device exercisers have the ability to detach; this allows you to run an exerciser while keeping your terminal free for other operations.
-1	Negative one, when used as a value for number of iterations, pages or lines, tells the program to run indefinitely, or until you enter CTRL/C.

DIGITAL

RSTS/E UPDATE
MANAGEMENT UTILITY PROGRAMS

DEVST Operating Procedures

To run a DEVST program contained within the system library account [1,2], type:

RUN \$program name

For example:

RUN \$KBEXER

To run a DEVST program from another privileged account, type:

RUN [account number]program name

For example:

RUN [1,215]KBEXER

Where program name is one of the exercisers in Table MUP-1.

DIGITAL

RSTS/E UPDATE
MANAGEMENT UTILITY PROGRAMS

Table MUP-1. DEVST Programs

Program Name	Hardware Tested
CPEXER	CPU
CPUTST	CPU
DSKEXR	FILE STRUCTURED DISK
DSKSEK	NON-FILE STRUCTURED DISK
DTEXER	DECTAPE
DXEXER	FLOPPY DISK
KBEXER	TERMINAL
LPEXER	LINE PRINTER
MTEXER	MAGNETIC TAPE
PPEXER	PAPER TAPE PUNCH
PREXER	PAPER TAPE READER

A complete list of RSTS/E supported hardware, including designators and the associated devices, appears in Table MUP-2.

Table MUP-2. RSTS/E Device Designators (Page 1 of 2)

Device Designator	Device
DF:, DS:, DK:, DL:, DM:, DP:, DR:, DB:, or SY:	RSTS/E public disk structure as a whole.
SYF:	System disk (the unit which was bootstrapped).
DFn:	RF11 disk (all platters).
DS0: to DS7:	RH11/RS03/RS04 fixed head disk units 0 through 7.
DK0: to DK7:	RK11/RK05 disk cartridge units 0 through 7.
DL0: to DL3:	RL01/RL02 disk cartridge units 0 through 3.
DM0: to DM7:	RK611/RK06/RK07 disk cartridge units 0 through 7.
DP0: to DP7:	RP11/RP02/RP03 disk pack units 0 through 7.
DRO: to DR7:	RH11/RM02/RM03 disk units 0 through 7.
DB0: to DB7:	RH11/RP04/RP05/RP06 disk pack units 0 through 7.

Table MUP-2. RSTS/E Device Designators (Page 2 of 2)

Device Designator	Device
PR:	High speed paper tape reader.
PP:	High speed paper tape punch.
CR:	CR11 punched or CM11 mark sense card reader.
CD:	CD11 punched card reader.
MT0: to MT7:	TE10/TU10/TS03 magtape units 0 through 7.
MM0: to MM7:	TE16/TU16/TU45 magtape units 0 through 7.
MS0: to MS7:	TS04 magtape units 0 through 7.
LP0: to LP7:	Line printer units 0 through 7.
DT0: to DT7:	TU56 DECTape units 0 through 7.
DD0: to DD7:	TU58 DECTape II units 0 through 7.

DEVTST Dialogue Errors

The following error messages may occur in DEVTST programs:

Message	Meaning
Warning: (Device Name) is a logical device: (Device Name) will be used.	DEVTST determines that the device name given is a logical name for another device. DEVTST performs a logical device translation and the translated device name is used.
?Illegal device: (Device Name) specified.	The device name given by the user is either incorrect or is a logical device name that cannot be translated (for example, incorrect characters or the wrong type of device). Retry the operation with the correct device name.
?Open failure on (Device Name): (Followed by an error message).	This error message can mean one of five things: <ol style="list-style-type: none"> 1. The device specified does not exist. 2. No volume was mounted on that device. 3. Another user has ASSIGNED the device. 4. The volume on the device has not been formatted.

5. The device is not on-line or the device is not write-enabled.

To resolve: retry the operation using the correct option as determined by the error message received.

?Illegal number of iterations selected.

Retry the operation using the correct number of iterations.

CPEXER

The central processor exerciser is designed to put a heavy load on the central processing unit. CPEXER runs compute bound for short bursts and then sleeps for five seconds so that average processor loading does not appreciably degrade the performance of the system. CPEXER was specifically designed to test the FPP option, but also verifies general CPU integrity and the PDP11/40 EIS operation. There is also a test of the PEEK function for kernel addresses 0 through 22000.

The tests performed by CPEXER are described in the next section. In most of the tests, results are compared to known correct values. Two heavily compute-bound tests are included to verify consistent results of duplicate calculations. Finally, tests are included that verify FPP divide by zero trap, integer divide by zero trap, integer conversion error and EIS operation.

CPEXER Tests

There are thirteen CPEXER tests included in the CPEXER program:

1. SIN(X) Uses SIN(X) extended function.
2. SIN(X) Uses polynomial approximation to SINE function.
3. LOG(X) Uses LOG(X) extended function.
4. EXP(X) Uses EXP(X) extended function.
5. SQR(X) Uses SQR(X) extended function.
6. SQR(X) Uses Newton-Raphson method to determine the square root.
7. LOG(EXP(X)) Compute-bound test.
8. ATN(TAN(X)) Compute-bound test.
9. A=1.0/0.0 Verify FPP divide by zero trap.
10. A%=1%/0% Verify integer divide by zero trap.
11. A%=60000 Verify integer conversion error.
12. INT(40.5621*100+0.5)/100=40.66 Verify EIS operation.
13. PEEK(X) Verify operation of PEEK function.

CPEXER Dialogue

The following is an example of the CPEXER dialogue:

```

RUN 11,2151 CPEXER
CPEXER V7.0-1 RSTS V07.0-07 The ARK

How many minutes to run <15>? 1

Detach <No>? NO

11.4 seconds of CPU time used in 1 pass.

CPEXER finished at 11:01 AM

Ready

```

Note

The CPEXER program will build correctly only if the version of BASIC-PLUS or BASIC-PLUS-2 includes TRIG and LOG. These mathematical functions must be included during System Generation.

CPEXER Errors

This program reports four types of errors:

1. Instructions failed to execute
2. Errors failed to trap
3. Computation errors
4. Computation consistency checks failed

Check the System Error Log for reported errors. These errors may indicate hardware problems.

When type 1 errors (instruction errors) occur, CPEXER prints a two line message: the first line gives the test that was being performed and the exact error message and the second line indicates that the test is continuing. For example:

```
?CPEXER PEEK failed at memory location m.
-( Error Message)
Continuing.....
```

Where: (Error Message) is the RSTS/E error message.

m is the memory location at which the PEEK function failed.

Examples of type 2 errors (trap failure) are:

```
?CPEXER Floating point divide by 0 didn't trap
Continuing.....
```

```
?CPEXER Integer divide by 0 didn't trap
Continuing.....
```

Type 3 errors (computation errors) give the type of test (SIN, PLYNOM, LOG, EXP, SQR or NEWTON SQR), error in the computed value, and the location of the possible error. For example:

```
?CPEXER - SIN test failed for the following cases:
?CPEXER - Computed value of SIN( 40 ) was .643, should be .64
?CPEXER CPU,EIS,FIS,FPP,MUL,DIV or SHIFT Error
Continuing.....
```

Type 4 errors (consistency failure) in the CPEXER are illustrated by the examples shown below:

```
?CPEXER - EXP/LOG Consistency check failed for the following cases:
?CPEXER - EXP(LOG(...( 1 )))) <> EXP(LOG(...(1 )))
?CPEXER - EXP(LOG(...( 4 )))) <> EXP(LOG(...(4 )))
?CPEXER CPU,EIS,FIS,FPP,MUL,DIV or SHIFT Error
Continuing .....
```

```
?CPEXER - ATN/TAN Consistency check failed for the following cases:
?CPEXER - ATN(TAN(...(-1.5 ))) <> ATN(TAN(...(-1.5 )))
?CPEXER CPU,EIS,FIS,FPP,MUL,DIV, or SHIFT Error
Continuing.....
```

CPUTST

The CPUTST dialogue first asks how many minutes you want the test to run. The program then checks for expiration of this request time at eight strategic points during each run sequence. CPUTST then clears and/or sets three 2-dimensional arrays to specified values. Matrix A is then transposed and the results placed in Matrix B. The transposition is checked by the statement $A(J,I) = B(I,J)$ while I and J are being varied. Any error is then reported; if none are encountered, the program returns to the matrix clear instruction. The program repeats this process until the run time has expired, at which time it prints the CPU time in seconds of expended job time.

CPUTST Dialogue

The following is an example of CPUTST dialogue:

```
Run [1,215]CPUTST
CPUTST V7.0-1 RSTS V07.0 0-07 The ARK

How many minutes to run <15>?1

Detach <no>? NO

6.9 seconds of CPU time used

CPUTST Finished at 10:59 AM

Ready
```

CPUTST Errors

This device exerciser reports errors as shown in the following examples:

?CPUTST-Error in matrix transposition routine.

?(Error message) at line mm in CPUTST V7.0-07.

Where: (Error message) is the RSTS/E error message.

mm is the line number in CPUTST where the error occurred.

Check the System Error Log for reported errors. These errors may indicate hardware problems.

DSKEXR

This disk exerciser tests the normal operation of all file structured disks. It is possible to run several copies of DSKEXR in order to test several drives simultaneously or to put a heavier load on any single drive. The disk must be physically and logically mounted. DSKEXR begins by asking questions to determine the disk drive to be tested and the number of test iterations to be performed. After this dialogue, the exerciser opens and extends a temporary file to a predetermined size which is dependent upon disk size and the number of free blocks. A pattern buffer is then loaded with one of four patterns (all 1's, all 0's or a pattern of alternating 1's and 0's) and the file is written. Each block is then read and compared for each of the patterns. Upon completion of all iterations for each drive specified, a status report is printed.

Note

This write-operation does not use current disk files to read and write. The temporary file will always be deleted after use, regardless of how the DSKEXR program is terminated. Be sure to turn off data caching before you run this test.

DSKEXR Dialogue

The following is an example of DSKEXR dialogue:

```
RUN [1,215]DSKEXR
DSKEXR V7.0-1 RSTS V07.0-07 The ARK
```

Disk drive to test? DB1:

Number of test iterations <8>? 1

Detach <No>? NO

No errors detected on _DB1:

Ready

DSKEXR Errors

Errors detected by DSKEXR are displayed as shown below:

?(Error message) error at line mm in DSKEXR V7.0-07.

?nn errors detected on DBn.

Where: (Error message) is the RSTS/E error message.

mm is the line number where the error occurred in the DSKEXR program.

nn is the number of errors detected by DSKEXR.

When errors are detected:

1. Insure that the volume is physically mounted and write-enabled.

2. Insure that the volume is logically mounted. This may be verified by the use of the SYSTAT program. If the disk is not mounted, use the mount command of the UTILTY program.

Caution

- A. Packs mounted as unlocked public disks will be available to other system users for creation and storage of their files. Subsequent removal of a public disk may result in the elimination of these user files from the system. For this reason, DIGITAL recommends the use of a private pack if a non-system disk is to be tested.
- B. Never mount a potentially bad disk on another drive until you are sure that the drive and pack are free of physical damage.

3. Persistent errors may be due to bad blocks on the pack. Any potentially bad blocks will be flagged in the System Error Log following unsuccessful access. To prevent further access to defective areas of the pack, use the BADS suboption of the REFRESH option of INIT.

DSKSEK

This disk exerciser tests the normal operation of non-file structured disks by randomly reading blocks from the disk. It is possible to use several copies of DSKSEK in order to test several drives simultaneously or to put a heavier load on any single drive. If the disk being tested is a mounted RSTS/E file structured disk, DSKSEK will not read any known bad blocks. If an error occurs on a read operation, it will be reported by DSKSEK and logged into the System Error Log.

Note

This is a read only operation. Turn off data caching before you run this test.

DSKSEK Dialogue

The following is an example of DSKSEK dialogue:

```

RUN [1,215]DSKSEK
DSKSEK V70-1 RSTS V07.0-07 The ARK

Disk drive to test? DB0:

Number of test iterations <30>? 1

Detach <No>? NO

No bad block detections in 1000 gets from _DB0:

Ready

```

DSKSEK Errors

Errors detected by DSKSEK are displayed as follows:

?(Error message) error at line mm in DSKSEK V7.0-07.

Where: (Error message) is the RSTS/E error message.

mm is the line number where the error occurred in the DSKSEK program

When errors are detected:

1. Check to see if the volume on the testing device is physically mounted.
2. If it is, mount another scratch volume on the drive and retry operation.

Caution

Never mount a potentially bad disk on another drive until you are sure that the drive and pack are free of physical damage.

3. If the test continues to fail, check the System Error Log to see if bad blocks are recorded.
4. If bad blocks are recorded, add them to the bad block list on the disk using INIT'S REFRESH/BADS suboption. This can only be done if the disk has been initialized (DSKINTed) as a RSTS/E disk.
5. If the problems continue, get a complete listing of the System Error Log (using the ERRDIS program).

DTEXER

The DECTape exerciser tests the normal operation of the TC11 DECTape control and TU56 DECTape drives. DTEXER begins by opening a file on the drive being tested and fills the file with floating point numbers. The program writes 420 blocks (of a possible 578 blocks) on the tape. Numbers written on the tape are read and checked, keeping a count of incorrect values. If at the end of the test the error count is not zero, DTEXER prints the error count.

Caution

Any information previously contained on the DECTape used for this test will be destroyed.

DTEXER Dialogue

The following is an example of DTEXER dialogue:

```
RUN [1,215]DTEXER
DTEXER V7.0-1 RSTS V07.0-07 The ARK
```

Warning: This exerciser will destroy data on the tested DECTape.

DEctape drive to test? DT0:

Number of iterations <2>? 1

Detach <No>? NO

No errors detected on _DT0:

DTEXER Finished at 11:00 AM

Ready

DTEXER Errors

This device exerciser shows the number of errors found during operation as shown in this example:

?mm errors on DTn during iteration nn.

Where: mm is the number of errors detected by the DTEXER program.

DTn is the name of device being tested.

nn is the number of the iteration during which the error occurred.

When errors are detected:

1. Check to see if the DECTape on the testing device is physically mounted and write-enabled.
2. If it is, mount another scratch DECTape on the drive and retry the operation.
3. Insure that the tape path and heads are free of oxide.

DXEXER

This exerciser tests the normal operation of the floppy controller and the specified floppy drives. DXEXER starts by opening a file 420 blocks long consisting of floating point numbers. These are written on the drive being tested, then read back and verified. The program then displays the number of words written and read, and the number of errors detected.

Caution

Any information previously contained on the floppy disk used for this test will be destroyed.

DXEXER Dialogue

The following is an example of DXEXER dialogue:

```
RUN [1,215]DXEXER
DXEXER V7.0-1 RSTS V07.0-07 The ARK
```

Warning: This exerciser will destroy data on the tested floppy disk.

Floppy drive to test? DX0:

Number of iterations <20>? 2

Detach <No>? NO

Floppy disk DXEXER I/O data

Device	Reads	Words	Writes	Words
DX0:	300	76.8K	300	76.8K

DXEXER Finished at 10:53 AM

DXEXER Errors

DXEXER shows the number of errors detected during operation as shown below:

?mm errors on DXn during iteration nn.

Where: mm is the number of errors detected by the DXEXER program.

DXn is the name of device being tested.

nn is the number of the iteration during which the error occurred.

When errors are detected:

1. Check to see if the floppy disk on the testing device is physically mounted and the door closed.
2. If it is, mount another scratch floppy disk on the drive and retry the operation.

KBEXER

The keyboard exerciser (KBEXER) is used to test local and remote terminals connected to the RSTS/E system. The four terminal tests are as follows:

1. The Space Test verifies that the terminal carriage will return reliably from any position.
2. The ASCII Pattern Test verifies that the terminal will print the standard ASCII character set in all print positions.
3. The Worst Case Test forces the head on ASR33 terminals to rotate a half revolution in every print position.
4. The Repeat Test repeats everything typed, one line at a time.

Note

The tests can be terminated at any time by typing CTRL/C. HELP can be obtained by typing HELP in answer to any dialogue question.

KBEXER Dialogue

The following is an example of KBEXER dialogue:

```
RUN [1,215]KBEXER
KBEXER V7.0-1 RSTS V07.0-07 The ARK

Keyboard to test <_KB58:>?

Test (SPA[CE], ASC[II], WOR[ST], REP[EAT], HEL[P]) ? ASC

Number of test iterations <8>? 1
```

Detach <No>? NO

*** ROTATING ASCII CHARACTERS TEST ***

```
BS%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNPOQRSTUVWXYZ
{/|_!~"#$%^
```

KBEXER Finished at 11:44 AM

Ready

KBEXER Errors

Errors detected by KBEXER appear as errors in the keyboard test patterns. Visually check the patterns for errors.

LPEXER

This exerciser prints a rotating pattern of ASCII characters on a specified number of pages and is used to test any line printer.

LPEXER Dialogue

The following is an example of LPEXER dialogue:

```
RUN [1,215]LPEXER
LPEXER V7.0-1 RSTS V07.0-07 The ARK

Line printer to test? LP0:

Number of pages to output <4>? 1

Detach <No>? No

LPEXER finished at 11:38 AM

Ready
```

LPEXER Errors

Problems found on the line printer appear as errors in the print-out of ASCII characters. Visually check the line printer listing and if an error message is printed by LPEXER, check to make sure that the line printer selected is on-line and not out of paper.

MTEXER

The magtape exerciser checks normal operation of the TM11 magtape control or RH11/TM02/TM03 controller and the specified number of 7-track or 9-track TE10, TE16, TU10, TS03, TU16, TU45 or TS04 drives. MTEXER allows you to select the drive to be tested, the length of tape to be written and the number of iterations to be performed. For each iteration, MTEXER zeroes the tape, opens a file and writes data until the specified length of tape has been used. MTEXER then rewinds the tape, opens the file for input, reads and verifies the data. If errors are detected, MTEXER prints a count of the number of bytes found to be incorrect; MTEXER then proceeds to the next iteration.

The data pattern used is a worst case NRZI pattern for 9-track drives; it is not worst case for 7-track recording. The pattern is loaded into a 512 byte buffer and, successively greater numbers of records are written to the magtape until the requested length of tape has been written. Furthermore, the pattern base varies with X so that the contents of the pattern buffer also varies on successive iterations.

Caution

This exerciser will overwrite data contained on the magtape used for the test.

MTEXER Dialogue

The following is an example of MTEXER dialogue:

```
RUN (1,215)MTEXER
MTEXER V7.0-1 RSTS V07.0-07 The ARK
```

```
Magtape drive to test? MT0:
```

```
Density (800 or 1600) <800>? 1600
```

```
Number of feet to test <100>? 25
```

```
Number of iterations <2>? 1
```

```
Detach <No>? NO
```

```
Magtape MTEXER I/O Data
```

Device	Reads	Words	Writes	Words
MT0:				
TA	24,576 K	96	24,576 K	

```
No errors detected on _MT0:
```

```
Ready
```

MTEXER Errors

Magtape errors are reported in these formats:

```
?Magtape select error at line nn in MTEXER V7.0-07.
```

```
?MTEXER-Selection error occurred.
```

Where: nn is the line number in MTEXER at which the error occurred.

When errors are detected:

1. Check to see if the volume on the testing device is physically mounted, write-enabled and assigned to you.

2. If it is, mount another scratch volume on the drive and retry the operation.
3. Insure that the tape path and heads are free of oxide.

PPEXER

This program tests the normal operation of the paper tape punch by punching random patterns on the paper tape for input into the PREXER.

PPEXER Dialogue

The following is an example of PPEXER dialogue:

```
RUN [1,215]PPEXER
PPEXER V7.0-1 RSTS V07.0-07 The ARK
```

This test will punch a binary count pattern tape. The tape will be equivalent to MAINDEC-00-D2G4 entitled *S p e c i a l Binary Count Pattern Tape*.

Detach <No>?

The generated paper tape may be used to test the paper tape reader.

PPEXER finished at 11:23 AM

Ready

PPEXER Errors

Problems in the paper tape punch will result in an incorrectly generated paper tape which is detected by PREXER when the tape is read. If PPEXER prints an error message, check to see that the paper tape punch is on-line and not out of paper tape.

PREXER

This program tests the normal operation of the paper tape reader by reading the paper tape punched by the PPEXER program.

PREXER Dialogue

The following is an example of PREXER dialogue:

```
RUN [1,215]PREXER
PREXER V7.0-1 RSTS V07.0-07 The ARK
```

Load MAINDEC-00-D2G4-PT or the binary count pattern tape generated by the punch exerciser into the reader with the leader under the read station.

Detach <No>? NO

No errors detected on paper tape.

PREXER finished at 11:29 AM

Ready

PREXER Errors

PREXER errors are reported in this format:

?nn errors detected on paper tape.

Where: nn is the number of errors detected by the PREXER program.

Note

If the paper tape was loaded backwards and the reader is operating correctly, the following error message is printed:
715360 errors detected on paper tape.

Any other errors detected by the PREXER may indicate a hardware problem.

ON-LINE CLEAN PROGRAM - ONLCLN

1. Provides the same 'CLEAN' as the REFRESH option of INIT.
2. Executes under the RT11 Run-Time System.

Example dialogue:

RUN \$ONLCLN

Disk? DM0:

Disk is being cleaned - wait...

Disk? ^C

Ready

Note

Disk must be physically, but not logically mounted.

ENHANCED MANAGEMENT UTILITY PROGRAMS

Spooling Package

1. Programs and their related files can reside in an account other than [1,2].

- o OPSER Program

Reports the keyboard number from which a message to the operator has been sent

For example:

```
MESSAGE      65 : 07-Mar-79 09:46 AM J0R:20 KB68: PLEASEC100,73
PLEASE MOUNT MAGTAPE LABELLED 'TESTS' ON MM1!
```

Figure MUP-1. OPSER Message

- o QUEMAN Program

New interrupt commands

Table MUP-3. New Interrupt Commands

Command	Option	Meaning
DIS[ABLE]	QUE[UING]<text>	QUE stops sending messages to QUEMAN and stops listing the queue.
	SPO[OLING] <text>	QUEMAN stops sending job requests to all spoolers.
	All <text>	Both QUEUING and SPOOLING are stopped.
ENA[BLE]	QUE[UING] <text>	QUE resumes sending messages to QUEMAN and enables listing the queue.
	SPO[OLING] <text>	QUEMAN resumes sending job requests to the spoolers.
	All <text>	Both QUEUING and SPOOLING are enabled.

DIGITAL

RSTS/E UPDATE
MANAGEMENT UTILITY PROGRAMS

o QUE program

New commands

F FLUSH A QUEUE*
S SHORT QUEUE LIST

CCL command

Recognizes the CCL command SUBMIT <filename>
and spools the file for batch processing.

* Privileged operation

DIGITAL

RSTS/E UPDATE
MANAGEMENT UTILITY PROGRAMS

SYSTAT Program

RUN SYSTAT
SYSTAT V7.0-03 RSTS V7.0-04 Timesharing
Output Status to?

RSTS V7.0-04 Timesharing status at 14-Nov-78, 03:00 PM Up: 31:09:37

Job	Who	Where	What	Size	State	Run-Time	RTS
1	[OPR]	Det	ZPRCPY	5/31K	SR D68	0:23.0	0/0 BAS4F
2	[OPR]	Det	OPSRUN	15/31K	SL	1:44.4	-0/4 BASIC2
3	[OPR]	Det	QUMRUN	16/31K	SL D48	0:22.9	-0/4 BAS4F
4	1,207	R066	DSKOMP	8/31K	RH	53.3	-2/2 BAS4F
5	[OPR]	Det	BATIRUN	15/31K	SL	20.7	-0/4 BAS4F
6	[OPR]	Det	BATIOL	13/31K	SL D61	0.7	-0/3 BAS4F
7	[OPR]	Det	BATIOL	13/31K	SL D57	4.2	-0/3 BAS4F
8	[OPR]	Det	MCU	14/31K	SR D50	9.9	-0/3 BAS4F
9	[OPR]	Det	MPKOVN	8/31K	SR	24.3	-0/2 BAS4F
10	[OPR]	Det	DYN.3R	2/31K	SL	17:21.7	120/10 ...PSR
11	1,224	R044	SYSTAT	10/31K	"C	3:26.0	0/3 SYSTAT
12	[OPR]	Det	SPLIOL	16/31K	SL D62	0.1	-0/4 BAS4F
13	2,215	R010	YEDIT	17/31K	RD	7:57.6	0/4 TECO
14	2,201	R043	NONAME	1/31K	"C	10.9	0/1 BP2COM
15	1,250	R033	...BP2	20/31K	"C A16	33:22.1	0/6 BP2COM
16	1,200	R022	NONAME	2/31K	"C A13	45.6	0/1 BAS4F
17	1,240	R024	SYSTAT	10/31K	"C	2:20.5	0/2 BAS4F
18	1,231	Det	STANEN	4/31K	SL	13.7	-0/1 BAS4F
19	1,219	R035	QUE	15/31K	"C A03	5.0	0/0 BAS4F
20	1,242	R021	VIEDIT	20/31K	RD A00	2:49.0	0/5 TECO
21	1,222	R035	NONAME	2/31K	"C A06	10.4	0/3 BAS4F
22	2,245	R022	NETWRK	8/31K	SL	5:17.0	-0/2 BAS4F
23	2,220	R020	VIEDIT	17/31K	RD A07	1:05.0	0/4 TECO
24	1,243	R040	...TRK	14/31K	RD A03	2:11.9	0/3 ...PSR
25	2,237	R047	VIEDIT	17/31K	RD A11	7.0	0/4 TECO
26	1,225	R030	VIEDIT	20/31K	RD	7:51.3	0/5 TECO
27	1,222	R026	TTYSET	10/31K	"C A04	6:10.7	0/3 BAS4F
28	1,207	R046	DSKOMP	13/31K	"C A05	10:54.3	0/3 BAS4F
29	1,237	R017	EVLNFS	4/31K	"C	1:50.6	0/1 BAS4F
30	2,210	R037	NONAME	2/31K	"C A01	21.0	0/1 BAS4F
31	2,230	R032	NONAME	2/31K	"C A12	21.9	0/1 BAS4F
32	2,236	R021	NONAME	2/31K	"C A15	0.9	-0/6 BAS4F
33	1,240	R025	SYMBLD	6/31K	RH	5:52.9	-0/2 BAS4F
34	1,245	R034	SYSTAT	10/31K	"C A10	29.5	0/3 BAS4F
35	1,244	R026	SYSTAT	10/31K	"C	15.1	0/3 BAS4F
36	[SELY]	R07.2	SYSTAT	10/31K	RD Lck	7.3	-0/3 BAS4F
37	1,253	P139	HPT.V2	16/31K	RH	2.1	-7/4 RMSV15

Busy Devices:

Device	Job	Why
PK0	5	AS*INIT
PK1	9	INIT
XM0	MSP	AS*INIT
XM1	MSP	AS*INIT
XM2	MSP	AS*INIT
HM0	35	AS*ANSI
HM2	35	AS*ANSI

Disk Structures:

Disk	Open	Free	Cluster	Errors	Name	Comments
DS0	13	600	0	0	SWAP0	File: DLW
DS1	1	1	0	0		File: NFS
DS4	1	1	0	0		File: P-CL NFS
DS9	0	5120	0	0	VEAC	File: P-CL DLW
DS1	17	3060	0	0	AB0	File: DLW
DS2	3	152006	0	0	VP*	File: DLW

Figure MUP-2. SYSTAT (page 1 of 2)

Small	Large	Jobs	Hung TTY's	Errors
41	1	37/63	8	388

Run-Time Systems:

Name	Ext	Size	Users	Comments
BAS4F	BAC	16(16)K	26	Perm, Addr:46, KBM, CSZ
HIS10G		28(4)K	0	Perm, Addr:268, KBM, CSZ, EMT:255
BASIC	BAC	16(16)K	0	Non-Res, KBM, CSZ
TECO	TEC	8(24)K	5	Temp, Addr:296, KBM
RT11	SAV	4(28)K	1	Temp, Addr:216, KBM, CSZ, EMT:255
RSX	TSK	3(28)K	0	Non-Res, KBM
RMS11	TSK	4(28)K	0	Non-Res
COBOL	TSK	4(28)K	0	Non-Res
BASIC2	TSK	16(16)K	1	Temp, Addr:163
BP2COM	TSK	4(28)K	2	Temp, Addr:179, KBM
BASICY	TSK	16(16)K	0	Non-Res
BPYCOM	TSK	4(28)K	0	Non-Res, KBM
RMSV15	TSK	4(28)K	1	Temp, Addr:184

Resident Libraries:

Name	Prot	Acct	Size	Users	Comments
RMSRES	< 42 >	[0,1]	23K	0	Non-Res, Addr:66

Message Receivers:

Name	Job	Msgs	Max	Senders
ERRLOG	1	0	40	Priv
OPSER	2	0	30	Local
QUEMAN	3	0	60	Local
BA0SPL	5	0	5	Priv
BA1SPL	6	0	5	Priv
BA2SPL	7	0	5	Priv
NCU	0	0	5	Priv, Network
NWPK09	9	0	5	Network
LP0SPL	12	0	5	Priv
NWTT22	22	0	5	Network
NFT.37	37	0	5	

Ready

Figure MUP-2. SYSTAT (page 2 of 2)

o Three new privileged switches

/O[:DEV] - list open files (large file systems only)
 /W[:DEV] - list open files and jobs accessing them (large file systems only)
 /C - list memory allocation

Table MUP-4. Status Codes

Code	Meaning
Pla	File is placed
Upd	File is opened in update mode
Nok	File cannot be renamed or deleted
Ufd	File is a UFD
Ctg	File is contiguous
Mdl	File is marked as deletable

Table MUP-5. Job Access Codes

Code	Meaning
Rd	Job has read access
Rr	File opened read regardless
Wr	Job has write access
Tent	File is a tentative file
Ca	File opened with data caching
Sq	File opened with sequential data caching
Up	File opened in update mode
Spup	File opened in special update mode

DISITAL

SYSTEMS UPDATE
MANAGEMENT UTILITY PROGRAMS

DISITAL

SYSTEMS UPDATE
MANAGEMENT UTILITY PROGRAMS

Open Files:

DS0: -- Null list

DM1: -- None

DB0: -- None

DB1: -- File	Op/RR	Size	Clu	Status
DB1: 1,235 SYSO .LST<60>	1/0	0	8	
DB1: 1,247 DISKID.BAS<60>	1/0	20	8	
DB1: 2,201 BUFMAP.B2S<60>	1/0	2	8	
DB1: 1,250	1/0	48	16	NoK, UFD
DB1: 2,201 TEMP14.TMP<60>	1/0	22	8	
DB1: 2,201 TMP214.TMP<60>	1/0	128	8	
DB1: 150,34 WRK .TMP<40>	1/0	0	8	
DB1: 150,34 CREP .TMP<40>	1/0	25	8	
DB1: 1,2 MACRO .SAV<104>	0/1	57	8	
DB1: 1,10 BATC06.WRK<60>	1/0	13	8	
DB1: 1,10 SPL004.WRK<60>	1/0	9	8	
DB1: 1,10 CHARS .QUE<60>	1/0	2	8	
DB1: 1,10 BATC05.WRK<60>	1/0	13	8	
DB1: 1,10 BATC07.WRK<60>	1/0	11	8	
DB1: 1,10 OPSLG1.LOG<60>	1/0	6	8	
DB1: 1,2 OPSER0.WRK<60>	1/0	2	8	
DB1: 1,2 OPSER1.WRK<40>	3/0	21	8	
DB1: 1,2 QUEUE .WRK<60>	1/0	23	8	
DB1: 1,2 QUEUE .SYS<40>	2/0	48	8	
DB1: 1,2 TEMP03.QUE<60>	1/0	105	8	

DB2: -- File	Op/RR	Size	Clu	Status
DB2: 1,2 BASIC2.TSK<104>	0/1	465	8	Ctg

Figure MUP-3. SYSTAT /O

Open Files and jobs accessing them:

DS0: -- Null list

DM1: -- None

DB0: -- None

DB1: -- File	Op/RR	Size	Clu	Status
DB1: 150,34 CREP .TMP<40>	1/0	0	8	
17 Rd, Wr, Tent				
DB1: 1,2 MACRO .SAV<104>	0/1	57	8	
17 Rd, RR				
DB1: 2,201 EVTREQS.B2S<60>	1/0	16	8	
14 Rd				
DB1: 1,10 BATC06.WRK<60>	1/0	13	8	
6 Rd, Wr				
DB1: 1,10 SPL004.WRK<60>	1/0	9	8	
4 Rd, Wr				
DB1: 1,10 CHARS .QUE<60>	1/0	2	8	
4 Rd				
DB1: 1,10 BATC05.WRK<60>	1/0	13	8	
5 Rd, Wr				
DB1: 1,10 BATC07.WRK<60>	1/0	11	8	
7 Rd, Wr				
DB1: 1,10 OPSLG1.LOG<60>	1/0	6	8	
2 Rd, Wr				
DB1: 1,2 OPSER0.WRK<60>	1/0	2	8	
2 Rd, Wr				
DB1: 1,2 OPSER1.WRK<40>	4/0	21	8	
19 Rd				
3 Rd				
26 Rd				
2 Rd, Wr				
DB1: 1,2 QUEUE .WRK<60>	1/0	23	8	
3 Rd, Wr				
DB1: 1,2 QUEUE .SYS<40>	3/0	48	8	
19 Rd				
26 Rd				
3 Rd, Wr				

DB2: -- File	Op/RR	Size	Clu	Status
DB2: 150,62 WRITE .MAC<40>	1/0	5	8	
12 Rd				

Figure MUP-4. SYSTAT /W

SY/C

Memory allocation table:

Start	End	Length	Permanent	Temporary
0K	-	46K (47K)	MONITOR	
47K	-	62K (16K)	BAS4F RTS	
63K	-	142K (80K)	** XBUF **	
143K	-	150K (8K)	TECO RTS	
151K	-	178K (28K)	HISTOG RTS	
179K	-	179K (1K)	(User)	
180K	-	202K (23K)	(User)	RMSRES LIB
203K	-	204K (2K)	(User)	COBOL RTS
205K	-	206K (2K)	(User)	COBOL RTS, FOO RTS
207K	-	207K (1K)	(User)	FOO RTS
208K	-	511K (304K)	(User)	
512K	-	*** END ***		

Ready

Figure MUP-4A. SYSTAT/C

VT50PY (VTSDPY) Program

1. New command for resident libraries

Command	Meaning
H	Displays resident library data.
-H	H preceded by a minus sign removes the data from the screen.

2. Memory display for resident libraries

-n- where n is the library's position in the resident library list.

DIGITAL

RSTS/E UPDATE
MANAGEMENT UTILITY PROGRAMS

TTYSET Program

1. Macros for new terminals include:

LA34
LA38
LA120
VT100

DIGITAL

RSTS/E UPDATE
MANAGEMENT UTILITY PROGRAMS

Error Logging Package

1. Recognizes and reports errors for new devices.
2. Programs and their related files can reside in an account other than [1,2].
3. Always reports the block number of the disk block that caused an error.

DR RH11/RM02-03 Seq #16 Occurred on 17-Nov-78 at 08:12:48 AM

User Description:

```

-----
Job Number          1E
KB Number           33
Account             [1,250]
Program Name        PACKID
User Job Physical Addr. 01100000
User Job Size       4K
Control Parameters  200
RIS Name            BAS4F
RIS Physical Address 00270000

```

Detailed Description:

```

-----
I/O Status          000
Timeout Indicator   000
Offset Position     Previous
Overlapped Seek Ind. 000
Unit Size in DC's  100220
Device Cluster Size 4
Pack Cluster Size   4
Logical Name
Physical Name       DR1:
Logical Block Number 4

```

```

DSQ          000900 174044 072344 000007
              000001 002171 106266 176000
              000000 013700 000000 000004
              000000 002002 127600 000900

```

```

=CB          002400 102044 000001 000005
              013734 000000 000000 000000
              000000 000000 000000 000000
              001007 001075 000004 000000

```

Figure MUP-5. ERRDIS (page 1 of 3)

CSR Address 176300

Contents of Registers

```

RMCS1  160200  MCPE Bus Err TRE Xfer Err
RMWC    000000
RMBA    000000
RMDA    000000
RMCS2   010101
RMDS    000000
RMER1   000000
RMAS    000000
RMLA    000000
RMHR1   000000
RMD1    000000
RMSN    000000
RMOF    000000
RMDC    000000
RMHR    000000
RMHR2   000000
RMER2   000000
RMEC1   000000
RMEC2   000000
RMBAE   000000
RMCS3   000000

```

Figure MUP-5. ERRDIS (page 2 of 3)

DIGITAL

RSTS/E UPDATE
MANAGEMENT UTILITY PROGRAMS

CONFIDENTIAL

RSTS/E UPDATE
MANAGEMENT UTILITY PROGRAMS

X1 DMC11 NPR Inter. Seq #613 Occurred on 16-Nov-78 at 02:59:59 PM

Detailed Description:

 Timeout Indicator 000000
 Physical Name XH1:

DDB 000010 000003 000000 000001
 102400 001035 000000 027640
 023006 000000 000000 000000
 073532 000000 027540 000200
 000000 000000 000000 000400
 000000 000000 000000 000000

DDB Extension 000000 027640 014200 021040
 000000 027650 000000 027654
 000000 027660 000000 027664
 000000 000000 002004 000000

CSR Address 160160

Contents of Registers
 SEL0 100000
 SEL2 046701
 SEL4 000000
 SEL6 000200

Table MUP-6. ERRDIS Mnemonics

Mnemonic	Description
UI	Undefined Interrupt
DR	RH11/RH02-03
DX	RX11/RX211
DD	TU58
TS	TS11/TS04
UM	Unexpected message*

* A message not received from the monitor.

Figure MUP-5. ERRDIS (page 3 of 3)

UTILITY Program

1. New Commands include:

Command	Meaning
SEIZE DEV:	Assign "DEV:" even if it is currently assigned to another job, unless a file is open on the device.
ADD LIBRARY LIBNAME	Add "LIBNAME" as a Resident Library.
/ADDR:n	Load Library in memory at address n, which is a 1K section of memory.
/STAY	The Library should be permanently resident.
/USER	The Resident Library is to handle only 1 user, rather than be shared code.
/RW	The Library should be mapped Read/Write, rather than Read-Only.
/NOLOGERR	Errors occurring within the Library should not be logged to the system error log.
/REMOVE	The Library should be immediately removed from memory when all jobs using it detach from the library.

Note

The /ADDR: switch must be specified with the ADD LIBRARY command.

REMOVE LIBRARY LIBNAME Remove "LIBNAME" as a Resident Library.

LOAD LIBRARY LIBNAME	Load Resident Library "LIBNAME" in memory.
/ADDR:n	Load Library in memory at address n which is a 1K section of memory.
/STAY	The library should be permanently resident.
UNLOAD LIBRARY LIBNAME	Unload Resident Library "LIBNAME" from memory.
LIST CACHE	List current cache settings.
ENABLE CACHE	Enable directory caching.
/ALL	Cache all data transfers regardless of OPEN mode.
/FILE	Enable data caching based on file and OPEN mode.
/NOFILE	Disable data caching.
/BUFF	Small buffers may be used for caching.
/NOBUFF	Small buffers may not be used for caching.
/CL:n	Cache Cluster Size
/LIMIT:n	Total # of Cache Clusters which will be used.
/DIR:n	Total # of Cache Clusters which can be used for directories.
/DATA:n	Total # of Cache Clusters which can be used for data.
DISABLE CACHE	Disable all caching.
RUNBURST n Q	Change job n's runburst to Q.
SIZE n Q	Change job n's maximum size to Q
FLAG filename	Modify file's directory based on the following switches.
/CACHE	Flag file to be cached automatically when the file is opened.

DIGITAL

RSTS/E UPDATE
MANAGEMENT UTILITY PROGRAMS

/NOCACHE Do not automatically cache the file when it is opened.

/SEQ Set file's default caching mode to sequential.

/RAN Set file's default caching mode to random.

/PLACE Flag file to be a placed file.

/NOPLACE Flag file as not being placed.

/NOCTG Flag file as not being contiguous.

DIGITAL

RSTS/E UPDATE
MANAGEMENT UTILITY PROGRAMS

ANALYS Program

1. New switch on output file produces an annotated version of a memory dump.
2. Additional information in the dump file.
3. New device support
4. Resident library support

RUN [1,10]ANALYS
ANALYS X7.0-05 RSTS V7.0-04 Timesharing
Input <[0,1]CRASH.SYS>?
Output <KB:ANALYS.DMP>?ANALYS.DMP/WIDE
Crash Error Log Filename<[1,10]ERRCRS.FIL>?

Ready

DIGITAL

RSTS/E UPDATE
MANAGEMENT UTILITY PROGRAMS

Analysis of 10.11CRASH.STS/SIL:STD:10.11STARTL.SIL
Taken on 20-Nov-78 at 09:30 AM

System was configured with 36Mnet/4
System was configured for large files

Crash Dump Status From 16-Nov-78 at 11:32 AM, Dpr 36:47:11

Job	Wpc	Where	What	Size	State	Run-Time	Pri/RD	RTS
1	1,2	Det	ERRCPY	5/31K	RM D56	0:56.0	0/6	BASMP
2	1,10	Det	CPSPUN	15/31K	SL	2:29.2	-6/0	BASIC2
3	1,10	Det	CLMPLN	16/31K	SL	10:57.9	-6/0	BASMP
4	1,10	Det	CPLODL	16/31K	SL D57	0.1	-6/0	BASMP
5	1,10	Det	ERRIDL	15/31K	SL	0.1	-6/3	BASMP
6	1,10	Det	ERRIDL	15/31K	SL	0.1	-6/3	BASMP
7	1,10	Det	BATPSA	15/31K	SL	9.4	-6/0	BASMP
8	1,10	Det	MCV	14/31K	SR D59	11.5	-6/3	BASMP
9	1,10	Det	MPRETP	8/31K	SR	12.5	-6/2	BASMP
10	1,10	Det	DYN-2N	2/31K	SL	19:49.0	120/10	...RSX
11	1,229	KE24	MCNAME	2/31K	"C	2:47.0	0/1	BASMP
12	1,229	PCJ7	B2CPLEP	25/31K	DB Lck	2:01.8	-40/5	...RSX
22	2,227	KB47	Y1ED11	17/31K	KB AIR	13.6	0/4	TECO
23	2,225	KB29	Y1ED11	16/31K	KB A06	6:02.6	0/4	TECO
24	1,240	KB38	R111	2/31K	"C	42.5	0/1	R111
32	1,244	KB31	MCNAME	2/31K	"C A10	28.0	0/1	BASMP

Busy Devices:

Device	Job	Way
PRC	-	AS-INIT
DN1	RSP	AS-INIT
DN2	USP	AS-INIT

Disk Structures:

Disc	Open	Free	Cluster	Errors	Name	Comments
SSC	12	0	0	0	SWAPD	Pri, DLV
SS1	1	0	1	0		Pri, NFS
EMO	0	1548K	1	0	BAS11M	Pri, R-O, DLV
SPS	0	4616	4	0	Y07A	Pri, DLV
DEC	0	5120	0	0	Y06C	Pri, R-O, DLV
ES1	23	12424	0	0	ARE	Pub, DLV
SB4	4	1548K	0	0	Y07	Pri, DLV

--- User Files ---
 SSC: --- Mail list
 SPC: --- None
 SPC: --- None
 SPC: --- None
 SPC: --- None
 SPC: --- None
 SPC: --- None

Figure MUP-6. ANALYS (page 1 of 6)

DIGITAL

RSTS/E UPDATE
MANAGEMENT UTILITY PROGRAMS

Job	Wpc	Where	What	Size	State	Run-Time	Pri/RD	RTS
1	1,229	KE24	B2CPLEP	25/31K	DB Lck	2:01.8	-40/5	...RSX
1	1,2	Det	ERRCPY	5/31K	RM D56	0:56.0	0/6	BASMP
1	1,2	Det	CPSPUN	15/31K	SL	2:29.2	-6/0	BASIC2
1	1,10	Det	CLMPLN	16/31K	SL	10:57.9	-6/0	BASMP
1	1,10	Det	CPLODL	16/31K	SL D57	0.1	-6/0	BASMP
1	1,10	Det	ERRIDL	15/31K	SL	0.1	-6/3	BASMP
1	1,10	Det	ERRIDL	15/31K	SL	0.1	-6/3	BASMP
1	1,10	Det	BATPSA	15/31K	SL	9.4	-6/0	BASMP
1	1,10	Det	MCV	14/31K	SR D59	11.5	-6/3	BASMP
1	1,10	Det	MPRETP	8/31K	SR	12.5	-6/2	BASMP
1	1,10	Det	DYN-2N	2/31K	SL	19:49.0	120/10	...RSX
1	1,229	KE24	MCNAME	2/31K	"C	2:47.0	0/1	BASMP
1	1,229	PCJ7	B2CPLEP	25/31K	DB Lck	2:01.8	-40/5	...RSX
1	2,227	KB47	Y1ED11	17/31K	KB AIR	13.6	0/4	TECO
1	2,225	KB29	Y1ED11	16/31K	KB A06	6:02.6	0/4	TECO
1	1,240	KB38	R111	2/31K	"C	42.5	0/1	R111
1	1,244	KB31	MCNAME	2/31K	"C A10	28.0	0/1	BASMP
1	1,229	KE24	B2CPLEP	25/31K	DB Lck	2:01.8	-40/5	...RSX
1	2,227	KB47	Y1ED11	17/31K	KB AIR	13.6	0/4	TECO
1	2,225	KB29	Y1ED11	16/31K	KB A06	6:02.6	0/4	TECO
1	1,240	KB38	R111	2/31K	"C	42.5	0/1	R111
1	1,244	KB31	MCNAME	2/31K	"C A10	28.0	0/1	BASMP
1	1,229	KE24	B2CPLEP	25/31K	DB Lck	2:01.8	-40/5	...RSX
1	2,227	KB47	Y1ED11	17/31K	KB AIR	13.6	0/4	TECO
1	2,225	KB29	Y1ED11	16/31K	KB A06	6:02.6	0/4	TECO
1	1,240	KB38	R111	2/31K	"C	42.5	0/1	R111
1	1,244	KB31	MCNAME	2/31K	"C A10	28.0	0/1	BASMP
1	1,229	KE24	B2CPLEP	25/31K	DB Lck	2:01.8	-40/5	...RSX
1	2,227	KB47	Y1ED11	17/31K	KB AIR	13.6	0/4	TECO
1	2,225	KB29	Y1ED11	16/31K	KB A06	6:02.6	0/4	TECO
1	1,240	KB38	R111	2/31K	"C	42.5	0/1	R111
1	1,244	KB31	MCNAME	2/31K	"C A10	28.0	0/1	BASMP
1	1,229	KE24	B2CPLEP	25/31K	DB Lck	2:01.8	-40/5	...RSX
1	2,227	KB47	Y1ED11	17/31K	KB AIR	13.6	0/4	TECO
1	2,225	KB29	Y1ED11	16/31K	KB A06	6:02.6	0/4	TECO
1	1,240	KB38	R111	2/31K	"C	42.5	0/1	R111
1	1,244	KB31	MCNAME	2/31K	"C A10	28.0	0/1	BASMP
1	1,229	KE24	B2CPLEP	25/31K	DB Lck	2:01.8	-40/5	...RSX
1	2,227	KB47	Y1ED11	17/31K	KB AIR	13.6	0/4	TECO
1	2,225	KB29	Y1ED11	16/31K	KB A06	6:02.6	0/4	TECO
1	1,240	KB38	R111	2/31K	"C	42.5	0/1	R111
1	1,244	KB31	MCNAME	2/31K	"C A10	28.0	0/1	BASMP
1	1,229	KE24	B2CPLEP	25/31K	DB Lck	2:01.8	-40/5	...RSX
1	2,227	KB47	Y1ED11	17/31K	KB AIR	13.6	0/4	TECO
1	2,225	KB29	Y1ED11	16/31K	KB A06	6:02.6	0/4	TECO
1	1,240	KB38	R111	2/31K	"C	42.5	0/1	R111
1	1,244	KB31	MCNAME	2/31K	"C A10	28.0	0/1	BASMP
1	1,229	KE24	B2CPLEP	25/31K	DB Lck	2:01.8	-40/5	...RSX
1	2,227	KB47	Y1ED11	17/31K	KB AIR	13.6	0/4	TECO
1	2,225	KB29	Y1ED11	16/31K	KB A06	6:02.6	0/4	TECO
1	1,240	KB38	R111	2/31K	"C	42.5	0/1	R111
1	1,244	KB31	MCNAME	2/31K	"C A10	28.0	0/1	BASMP
1	1,229	KE24	B2CPLEP	25/31K	DB Lck	2:01.8	-40/5	...RSX
1	2,227	KB47	Y1ED11	17/31K	KB AIR	13.6	0/4	TECO
1	2,225	KB29	Y1ED11	16/31K	KB A06	6:02.6	0/4	TECO
1	1,240	KB38	R111	2/31K	"C	42.5	0/1	R111
1	1,244	KB31	MCNAME	2/31K	"C A10	28.0	0/1	BASMP
1	1,229	KE24	B2CPLEP	25/31K	DB Lck	2:01.8	-40/5	...RSX
1	2,227	KB47	Y1ED11	17/31K	KB AIR	13.6	0/4	TECO
1	2,225	KB29	Y1ED11	16/31K	KB A06	6:02.6	0/4	TECO
1	1,240	KB38	R111	2/31K	"C	42.5	0/1	R111
1	1,244	KB31	MCNAME	2/31K	"C A10	28.0	0/1	BASMP
1	1,229	KE24	B2CPLEP	25/31K	DB Lck	2:01.8	-40/5	...RSX
1	2,227	KB47	Y1ED11	17/31K	KB AIR	13.6	0/4	TECO
1	2,225	KB29	Y1ED11	16/31K	KB A06	6:02.6	0/4	TECO
1	1,240	KB38	R111	2/31K	"C	42.5	0/1	R111
1	1,244	KB31	MCNAME	2/31K	"C A10	28.0	0/1	BASMP
1	1,229	KE24	B2CPLEP	25/31K	DB Lck	2:01.8	-40/5	...RSX
1	2,227	KB47	Y1ED11	17/31K	KB AIR	13.6	0/4	TECO
1	2,225	KB29	Y1ED11	16/31K	KB A06	6:02.6	0/4	TECO
1	1,240	KB38	R111	2/31K	"C	42.5	0/1	R111
1	1,244	KB31	MCNAME	2/31K	"C A10	28.0	0/1	BASMP
1	1,229	KE24	B2CPLEP	25/31K	DB Lck	2:01.8	-40/5	...RSX
1	2,227	KB47	Y1ED11	17/31K	KB AIR	13.6	0/4	TECO
1	2,225	KB29	Y1ED11	16/31K	KB A06	6:02.6	0/4	TECO
1	1,240	KB38	R111	2/31K	"C	42.5	0/1	R111
1	1,244	KB31	MCNAME	2/31K	"C A10	28.0	0/1	BASMP
1	1,229	KE24	B2CPLEP	25/31K	DB Lck	2:01.8	-40/5	...RSX
1	2,227	KB47	Y1ED11	17/31K	KB AIR	13.6	0/4	TECO
1	2,225	KB29	Y1ED11	16/31K	KB A06	6:02.6	0/4	TECO
1	1,240	KB38	R111	2/31K	"C	42.5	0/1	R111
1	1,244	KB31	MCNAME	2/31K	"C A10	28.0	0/1	BASMP
1	1,229	KE24	B2CPLEP	25/31K	DB Lck	2:01.8	-40/5	...RSX
1	2,227	KB47	Y1ED11	17/31K	KB AIR	13.6	0/4	TECO
1	2,225	KB29	Y1ED11	16/31K	KB A06	6:02.6	0/4	TECO
1	1,240	KB38	R111	2/31K	"C	42.5	0/1	R111
1	1,244	KB31	MCNAME	2/31K	"C A10	28.0	0/1	BASMP
1	1,229	KE24	B2CPLEP	25/31K	DB Lck	2:01.8	-40/5	...RSX
1	2,227	KB47	Y1ED11	17/31K	KB AIR	13.6	0/4	TECO
1	2,225	KB29	Y1ED11	16/31K	KB A06	6:02.6	0/4	TECO
1	1,240	KB38	R111	2/31K	"C	42.5	0/1	R111
1	1,244	KB31	MCNAME	2/31K	"C A10	28.0	0/1	BASMP
1	1,229	KE24	B2CPLEP	25/31K	DB Lck	2:01.8	-40/5	...RSX
1	2,227	KB47	Y1ED11	17/31K	KB AIR	13.6	0/4	TECO
1	2,225	KB29	Y1ED11	16/31K	KB A06	6:02.6	0/4	TECO
1	1,240	KB38	R111	2/31K	"C	42.5	0/1	R111
1	1,244	KB31	MCNAME	2/31K	"C A10	28.0	0/1	BASMP
1	1,229	KE24	B2CPLEP	25/31K	DB Lck	2:01.8	-40/5	...RSX
1	2,227	KB47	Y1ED11	17/31K	KB AIR	13.6	0/4	TECO
1	2,225	KB29	Y1ED11	16/31K	KB A06	6:02.6	0/4	TECO
1	1,240	KB38	R111	2/31K	"C	42.5	0/1	R111
1	1,244	KB31	MCNAME	2/31K	"C A10	28.0	0/1	BASMP
1	1,229	KE24	B2CPLEP	25/31K	DB Lck	2:01.8	-40/5	...RSX
1	2,227	KB47	Y1ED11	17/31K	KB AIR	13.6	0/4	TECO
1	2,225	KB29	Y1ED11	16/31K	KB A06	6:02.6	0/4	TECO
1	1,240	KB38	R111	2/31K	"C	42.5	0/1	R111
1	1,244	KB31	MCNAME	2/31K	"C A10	28.0	0/1	BASMP
1	1,229	KE24	B2CPLEP	25/31K	DB Lck	2:01.8	-40/5	...RSX
1	2,227	KB47	Y1ED11	17/31K	KB AIR	13.6	0/4	TECO
1	2,225	KB29	Y1ED11	16/31K	KB A06	6:02.6	0/4	TECO
1	1,240	KB38	R111	2/31K	"C	42.5	0/1	R111
1	1,244	KB31	MCNAME	2/31K	"C A10	28.0	0/1	BASMP
1	1,229	KE24	B2CPLEP	25/31K	DB Lck	2:01.8	-40/5	...RSX
1	2,227	KB47	Y1ED11	17/31K	KB AIR	13.6	0/4	TECO
1	2,225	KB29	Y1ED11	16/31K	KB A06	6:02.6	0/4	TECO
1	1,240	KB38	R111	2/31K	"C	42.5	0/1	R111
1	1,244	KB31	MCNAME	2/31K	"C A10	28.0	0/1	BASMP
1	1,229	KE24	B2CPLEP	25/31K	DB Lck	2:01.8	-40/5	...RSX
1	2,227	KB47	Y1ED11	17/31K	KB AIR	13.6		

Facil Large Jobs Hung TTY'S Errors
200 1 15 53 0 227

Run-Time Systems:

Name	Int	Size	Corro	Comments
BAZ-P	BAZ	16176K	10	Perm. Addr:16, KEM, CS2
BAZ-IC	BAZ	25144K	0	Perm. Addr:266, KEM, CS2, DMI:255
BAZ-CC	BAZ	16116K	0	Non-Res, KEM, CS2
BAZ-DC	BAZ	6127K	2	Perm. Addr:296, KEM
BAZ-EC	BAZ	4126K	1	Temp. Addr:357, KEM, CS2, DMI:255
BAZ-FC	BAZ	3120K	0	Temp. Addr:207, KEM
BAZ-GC	BAZ	4126K	0	Non-Res
BAZ-HC	BAZ	4126K	0	Non-Res
BAZ-IC	BAZ	16116K	1	Temp. Addr:210
BAZ-JC	BAZ	4126K	0	Non-Res, KEM
BAZ-KC	BAZ	16116K	0	Non-Res
BAZ-LC	BAZ	4126K	0	Non-Res, KEM
BAZ-MC	BAZ	4126K	0	Non-Res

Resident Libraries:

Name	Prot	Acct	Size	Users	Comments
RES-RES	4 429	1 0,1	23K	0	Non-Res, Addr:188

no Links were in use

Nodes:

Name	Addr	Line	Links/Max	Seg	State	Comments
BAZ	0	None	0/32	541	Run	Adj. Ana
BAZ	6	DM2	0/32	541	Run	Adj. Gr. Rat
BAZ	55	DM1	0/32	542	Run	Adj. Org. Rat

Lines:

Line	Node	Size	Quota	Count	Mode	State	DCR	1MG	OPR	Job
001	BAZ	541	0	0	FD2	On	0	0	0	BSP
002	BAZ	541	0	0	FD2	On	0	0	0	BSP

Message Receivers:

Receiver	Job	Obj	Msgs/Max	Lines/Max	Access
BAZ-IC	1	0	1/40	0/0	Prv
BAZ-DC	2	0	0/30	0/0	Lcl
BAZ-EC	3	0	0/60	0/0	Lcl
BAZ-FC	4	0	0/5	0/0	Prv
BAZ-GC	5	0	0/5	0/0	Prv
BAZ-HC	6	0	0/5	0/0	Prv
BAZ-IC	7	0	0/5	0/0	Prv
BAZ-KC	8	19	0/6	0/6	Net, Prv
BAZ-LC	9	249	0/5	0/1	Net
BSP	BSP	0	0/0	0/32	None

Detail Dump of Status

ERRCP CODE

CC018/ 117777

SAVED TO TO BS

CC018/ CC001 015776 000003 176000 120146 000002

EMPHAL SP

CC018/ 002004

VEFICAL PROGRAM COUNTER

CC018/ 012200

Figure MUP-6. ANALYS (page 3 of 6)

TABLE TOP STATUS
CC018/ 014440

NEXT 6. ITEMS ON KERNEL SP STACK

CC018/ 000000 000000 000000 000000 000000 000000 000000 000000 000000

USER SP

CC018/ 145004

TOP 6. ITEMS ON USER SP STACK

CC018/ 000043 000043 000043 000043 000043 000043 000043 000043

USER KEYWORD

CC018/ 105200

USER FIRGB

CC018/ 000000 003032 000062 000000 010602 070516 077430 000000
CC018/ 000000 000000 000000 000000 000000 000000 000000 000000

USER IRB

CC018/ 006434 000000 050024 000056 000115 000000 000000

	USER ADDR. REG.	USER DESC. REG.	KERNEL ADDR. REG.	KERNEL DESC. REG.
004532/	APRO 016740	077406	000000	077506
004542/	APR1 017140	077406	000200	077506
004552/	APR2 017340	077406	000400	077506
004562/	APR3 017540	077406	000500	077506
004572/	APR4 017740	077406	001000	077402
004602/	APR5 020140	057406	001710	077402
004612/	APR6 000000	000000	017440	077006
004622/	APR7 000000	000000	177600	077506

JOB NEXT
001006\ 000 000

JOBDA JOBFB IOSIS JCB4WK JOBBD2 JOBRTS CPUTIM JOB4DB
001010/ 000000 105052 062406 000000 062406 000000 062406 000000

F1JOB

001502\ 052

F1JBDA

001506/ 020040

JOB18L	043562/	000000	002700	010140	011040	011600	012240	012740	014600
043402/	016640	015140	016300	017240	000000	020040	020000	026200	026200
043422/	000000	000000	000000	000000	000000	000000	000000	000000	000000
043442/	054500	000000	000000	000000	000000	000000	000000	000000	000000
043462/	027500	000000	000000	000000	000000	000000	000000	000000	000000
043502/	000000	000000	000000	000000	000000	000000	000000	000000	000000
043522/	000000	000000	000000	000000	000000	000000	000000	000000	000000
043542/	000000	000000	000000	000000	000000	000000	000000	000000	000000

Figure MUP-6. ANALYS (page 4 of 6)

DIGITAL

RSTS/E UPDATE
MANAGEMENT UTILITY PROGRAMS

SHUTUP Program

1. Additional information in printout

RUN SSHUTUP
SHUTUP X07.0-1 RSTS V07.0-03 TIME SHARING

12:24 PM 22-Sep-78 ||||||| Set-up Dialogue Phase |||||||

Type 'ESC' ('ALT') to any query to backup one (1) step

'OPSER' not running

Minutes until system shutdown (0-99) <S>? 0

12:24 PM 22-Sep-78 ||||||| Warning Message Phase |||||||
Further LOGINS are now disabled

12:24 PM 22-Sep-78 ||||||| Initial Job Killing Phase |||||||

12:24 PM 22-Sep-78 ||||||| Remove RTS/RES LIB Phase |||||||

12:24 PM 22-Sep-78 ||||||| SWAP File Removal Phase |||||||

12:24 PM 22-Sep-78 ||||||| Disk DISMOUNT Phase |||||||

12:24 PM 22-Sep-78 ||||||| Final Shutdown Phase |||||||

Please wait for system to re-boot itself

DIGITAL

RSTS/E UPDATE
MANAGEMENT UTILITY PROGRAMS

LOGIN Program

1. Has a spawned entry point for use by the BUILD package.

MONEY Program

1. Reports accounts using more than 65535 blocks on the public structure as > = 65535 in the "blocks used" column.

DSKINT Program

1. Supports new disks.
2. Reports the current PACKID and Pack Cluster Size if disk already has a RSTS/E format.

REACT Program

1. Sets the protection code on ACCT.SYS to 128.

ODT Program

1. Supports large files

DIGITAL

POSTS:RE UPDATES
MANAGEMENT UTILITY PROGRAMS

PROGRAMS THAT USE DATA CACHING (IF ENABLED)

1. BATCH
2. ANALYS

RSTS/E V7.0 SWS UPDATE

MON

Monitor Internals

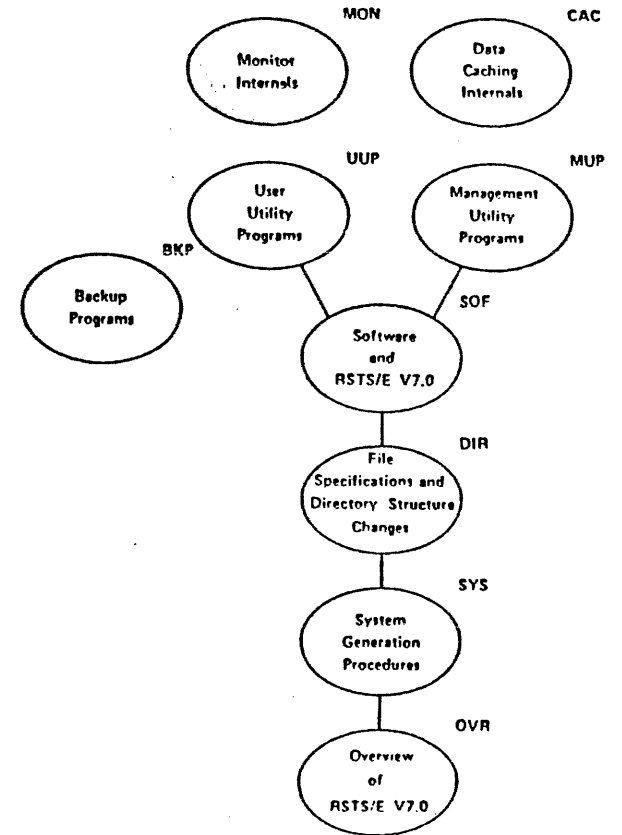
Monitor Internals

CONTENTS

Introduction.....	MON-1
Monitor Modules.....	MON-3
RSTS.....	MON-3
TER.....	MON-3
DSK.....	MON-3
EMT.....	MON-3
FIP.....	MON-3
RSX.....	MON-3
OVR.....	MON-4
ODT.....	MON-4
DEFAULT.....	MON-4
Monitor as a SIL on Disk.....	MON-5
Exercise for SIL Format.....	MON-11
Feedback for SIL Format.....	MON-15
The Resident Monitor.....	MON-17
Miscellaneous Code.....	MON-23
Map (PATCHS Module) V6C Change.....	MON-25
System Wide Logical Name Entries.....	MON-32
Job Related Structures.....	MON-34
JOBCLK Table - V6C Change.....	MON-34
FCB List Table (FCBLST).....	MON-50
"Contorted" Buffer Address.....	MON-55
Unchanged JOB Data Structures.....	MON-62
Device Related Structures.....	MON-65
New Handler Indexes.....	MON-65
New Driver Indexes.....	MON-65
Unit Count Table (UNTCNT).....	MON-73
Unit Options Table (UNTOPT).....	MON-75
Disk Indexes.....	MON-77
DSKMAP Table.....	MON-78
Device Name Table (DEVNAM).....	MON-81
Device Maximum Unit Number Table (DEVCNT).....	MON-83
Device Pointer Table (DEVPTR).....	MON-85

Device Table (DEVTBL)..... MON-88
 Determining FIP Unit Numbers (FUN) for a
 System..... MON-91
 Interfacing with Monitor from User Mode..... MON-94
 The Null (Disappearing) Run-Time System..... MON-100
 Exercise for Monitor..... MON-104
 Feedback for Monitor..... MON-154

COURSE MAP



CONFIDENTIAL

RSTS/E PWS UPDATE
Monitor Internals

Monitor Internals

INTRODUCTION

The RSTS/E monitor performs a multitude of functions for its users. These functions include:

- o Scheduling jobs to be run
- o Performing all device operations
- o Performing all file related operations
- o Managing all resources on the system

In actuality, the monitor is responsible for the harmony and security which RSTS/E users have come to take for granted.

In this module you will learn about the changes to the monitor in RSTS/E V7.0. These changes involve not only the format of a RSTS/E monitor as it resides on disk prior to system start up, but also the operational and structural changes which affect all users during a time-sharing session.

MON-1

COMPANY CONFIDENTIAL

DIGITAL

RSTS/E SWS UPDATE
Monitor Internals

This page is for notes.

RSTS/E SWS UPDATE
Monitor Internals

MONITOR MODULES

RSTS

- o Data structures for monitor, device and job management
- o Non-disk interrupt routines
- o Some monitor code

TER

- o Terminal driver routines
- o Mini SYSTAT routines

DSK

- o Disk interrupt routines
- o Disk driver code and error logging routines

EMT

- o Monitor call handling routines
- o Resident library handling and mapping routines
- o Scheduling and swapping routines
- o Memory management routines

FIP

- o Directory handling routines
- o File operation routines
- o FIP function handling
- o SAT handling routines

RSX

- o RSX directive emulation routines

DIGITAL

RSTS/E SAS UPDATE
Monitor Internals

DIGITAL

RSTS/E SAS UPDATE
Monitor Internals

OVR

- o Routines not made permanently resident during SYSGEN

ODT

- o Monitor ODT code

DEFAULT

- o Default settings for the monitor

MONITOR AS A SIL ON DISK

During system generation the monitor modules are assembled and linked. Following this process the SILUS program is run to convert the linked monitor into a save image library (SIL) format. It is this SIL format which makes autpatching of the modules possible.

This section covers the format and information contained in a save image library.

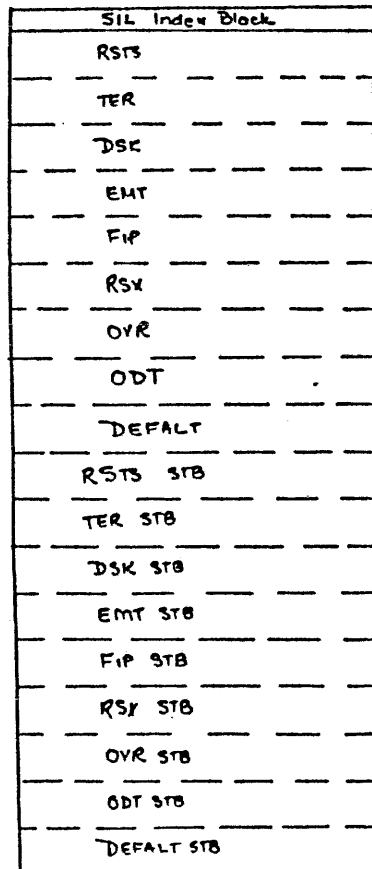


Figure MON-1. Monitor on Disk (A Save Image Library)

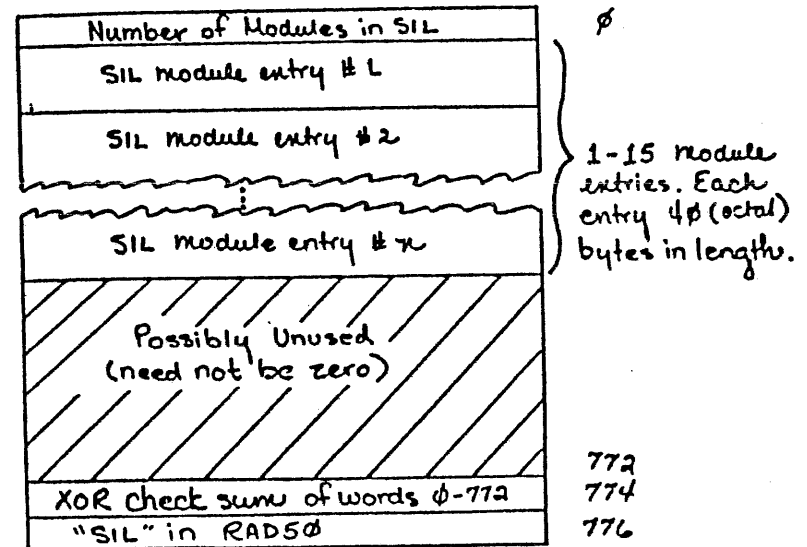


Figure MON-2. Index Block (First Block in a SIL)

SE.NAM	0
SE.IDN	2
SE.BLK	4
SE.STB	6
SE.STN	10
SE.LOD	12
SE.SIZ	14
SE.XFR	16
SE.SZD	20
SE.OVB	22
SE.OVN	24
Reserved (must be zero)	26
SE.XXX	30
	32
	34
	36

Figure MON-3. SIL Module Entry Format

SE.NAM Module name in RAD50
 SE.IDN Module's .IDENT in RAD50
 SE.BLK Block offset to module within SIL
 1 First module usually
 0 A SAV format SIL
 SE.STB Block offset to module's symbol table
 0 Module has no symbol table
 SE.STN Number of symbols in module's symbol table
 0 Module has no symbol table
 SE.LOD Virtual address into which module 'loads'
 Lowest address in the module's image
 Low limit Monitor SIL
 Low limit RTS
 Lowest limit Non-PIC LIB
 000001 PIC LIB
 000000 SAV format SIL and INIT.SYS
 SE.SIZ Virtual address 'size' of module

The first illegal address of a module is SE.SIZ plus SE.LOD
 High limit Monitor SIL
 177776 RTS
 Highest address Non-PIC LIB
 Highest address PIC LIB
 Highest address SAV format SIL
 SE.XFR Virtual transfer address of module
 Only SAV format SILs have transfer addresses.
 000001 means no transfer address
 SE.SZD Size of module on disk in blocks
 For non-overlaid modules:
 SE.SZD = (SE.SIZ + 777)/1000
 For overlaid modules (LIBs and overlaid RT11 programs in SAV format SILs)
 SE.SZD > SE.SIZ
 SE.SZD = 0 SIL format of previous RSTS/E releases
 SE.OVB Block offset to module's overlay descriptors
 SE.OVN Number of overlay descriptors for module
 SE.OVB } = 0 Monitors and RTS'
 SE.OVN }

Note

Any module which starts on a 1000 byte multiple and is not overlaid does not need overlay descriptors. Any module which starts on a non-1000 byte multiple (e.g. INIT.SYS), which has data below its start (e.g. all RT11 programs), or which is overlaid (e.g. LIBs) require overlay descriptors

SE.XXX Reserved for SAV format SILs. Contents 'random' to all but the RT11 emulator.

EXERCISE

- Using the following octal dump of a SIL index block,

Symbol in RAD50 (Must be unique and non-zero)	0
Symbol's overlay desc. no.	2
Symbol's value	4
	6

Figure MON-4. Symbol Table Entry Format

```

BLOCK NUMBER 00000
000/ 000012 071614 073300 140544 140122 000001 000377 002010
020/ 000000 140000 000001 000140 000000 000000 000000 000000
040/ 000000 076732 000000 140544 140122 000141 000420 002333
060/ 120000 021000 000001 000021 000000 000000 000000 000000
100/ 000000 016003 000000 140544 140122 000162 000444 002070
120/ 120000 010000 000001 000010 000000 000000 000000 000000
140/ 000000 020534 000000 140544 140122 000172 000465 002037
160/ 120000 017000 000001 000017 000000 000000 000000 000000
200/ 000000 023370 000000 140544 140122 000211 000506 002222
220/ 120000 027000 000001 000027 000000 000000 000000 000000
240/ 000000 071620 000000 140544 140122 000240 000531 002041
260/ 120000 007000 000001 000007 000000 000000 000000 000000
300/ 000000 055210 000000 140443 131716 000247 000552 002255
320/ 120000 020000 000001 000020 000000 000000 000000 000000
340/ 000000 060502 000000 140544 140122 000267 000575 002474
360/ 002000 073000 000001 000073 000000 000000 000000 000000
400/ 000000 057164 000000 140544 140122 000362 000622 000003
420/ 140000 012520 142522 000013 000000 000000 000000 000000
440/ 000000 014716 004064 140544 140122 000375 000623 000001
460/ 001000 002000 000001 000002 000000 000000 000000 000000
500/ 000000 000000 000000 000000 000000 000000 000000 000000
520/ 000000 000000 000000 000000 000000 000000 000000 000000
540/ 000000 000000 000000 000000 000000 000000 000000 000000
560/ 000000 000000 000000 000000 000000 000000 000000 000000
600/ 000000 000000 000000 000000 000000 000000 000000 000000
620/ 000000 000000 000000 000000 000000 000000 000000 000000
640/ 000000 000000 000000 000000 000000 000000 000000 000000
660/ 000000 000000 000000 000000 000000 000000 000000 000000
700/ 000000 000000 000000 000000 000000 000000 000000 000000
720/ 000000 000000 000000 000000 000000 000000 000000 000000
740/ 000000 000000 000000 000000 000000 000000 000000 000000
760/ 000000 000000 000000 000000 000000 000000 000000 117416 074064

```

Figure MON-5.

a. How many modules are in the save image library?

b. What is the name of the first module in the save image library?

c. What is the module's .IDENT?

d. What is the module's block offset within the SIL?

e. What is the module's STB block offset (in octal)?

f. How many symbols are in the symbol table (in octal)?

g. What is the virtual address into which this module 'loads'?

h. What is the virtual transfer address of this module?

2. Using the following octal dump of a module's symbol table,

```
BLOCK NUMBER 00377
000/ 003237 071640 000000 000014 006254 012265 000000 000027
020/ 006254 012404 000000 000037 006254 015172 000000 000001
040/ 006254 021042 000000 000171 006254 024327 000000 000022
060/ 006254 053665 000000 000002 006254 062063 000000 000032
100/ 006254 075154 000000 000103 006456 021042 000000 000064
120/ 006565 130272 000000 000006 006571 130272 000000 000006
140/ 006601 130272 000000 000012 006601 130360 000000 000024
160/ 007575 071520 000000 000056 007716 127664 000000 000776
200/ 007716 127670 000000 001000 007716 130272 000000 000200
220/ 007716 130360 000000 000200 007716 130425 000000 001000
240/ 007716 130474 000000 000002 007716 130613 000000 000200
260/ 007716 131021 000000 001000 007716 131315 000000 001000
300/ 010360 071671 000000 000046 010363 076463 000000 000044
320/ 010363 107770 000000 000045 010377 135600 000000 000042
340/ 010400 157560 000000 000043 010402 000000 000000 000041
360/ 012263 124564 000000 177777 012263 125325 000000 177775
400/ 012263 125721 000000 000001 012404 127642 000000 000003
420/ 012404 127653 000000 000002 012404 127655 000000 000002
440/ 012404 127662 000000 000002 012404 127663 000000 000002
460/ 012404 127664 000000 000002 012404 127670 000000 000002
500/ 012404 130272 000000 000130 012404 130360 000000 000001
520/ 012404 130425 000000 000003 012404 130613 000000 000017
540/ 012404 131315 000000 000010 012453 021042 000000 000113
560/ 012516 073546 000000 000133 012525 023710 000000 000135
600/ 012525 062246 000000 000136 012525 062314 000000 000137
620/ 012525 062402 000000 000134 012762 045575 000000 000034
640/ 014474 021042 000000 000015 014574 074170 000000 050000
660/ 014657 131330 000000 176356 014663 127664 000000 000040
700/ 014663 127670 000000 000034 014663 130272 000000 000072
720/ 014663 130360 000000 000034 014663 130425 000000 000046
740/ 014663 130474 000000 000010 014663 130613 000000 000014
760/ 014663 131315 000000 000060 014734 042641 000000 000033
```

Figure MON-6

a. What is the name of the first symbol in the table?

b. What is the value of the symbol (in octal)?

FEEDBACK

1. Using the octal dump of a SIL index block,

a. How many modules are in the save image library?

10. or 12 (octal)

b. What is the name of the first module in the save image library?

RSTS

c. What is the module's .IDENT?

07.004

d. What is the module's block offset within the SIL?

1

e. What is the module's STB block offset (in octal)?

377

f. How many symbols are in the symbol table (in octal)?

2010

g. What is the virtual address into which this module 'loads'?

0

h. What is the virtual transfer address of this module?

1

DIGITAL

RSTS/E SWS UPDATE
Monitor Internals

2. Using the octal dump of a module's symbol table,
 - a. What is the name of the first symbol in the table?
ABORT
 - b. What is the value of the symbol (in octal)?
14

THE RESIDENT MONITOR

At system start up image copies of specific monitor SIL modules are loaded into memory and make up the 'resident monitor'. Each of these memory resident modules is called a monitor 'phase'.

RSTS/E SWS UPDATE
Monitor Internals

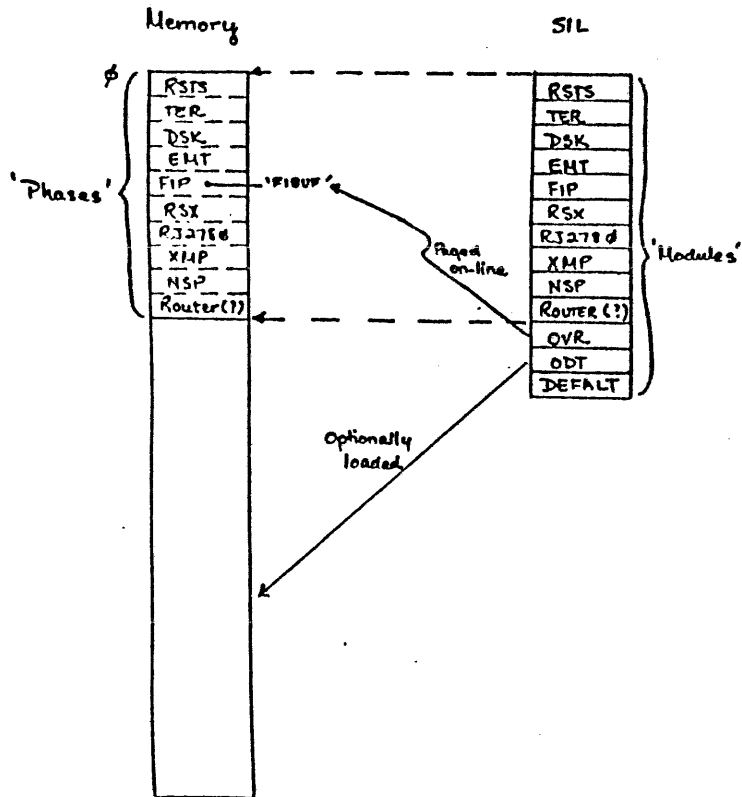


Figure MON-7. Resident Monitor

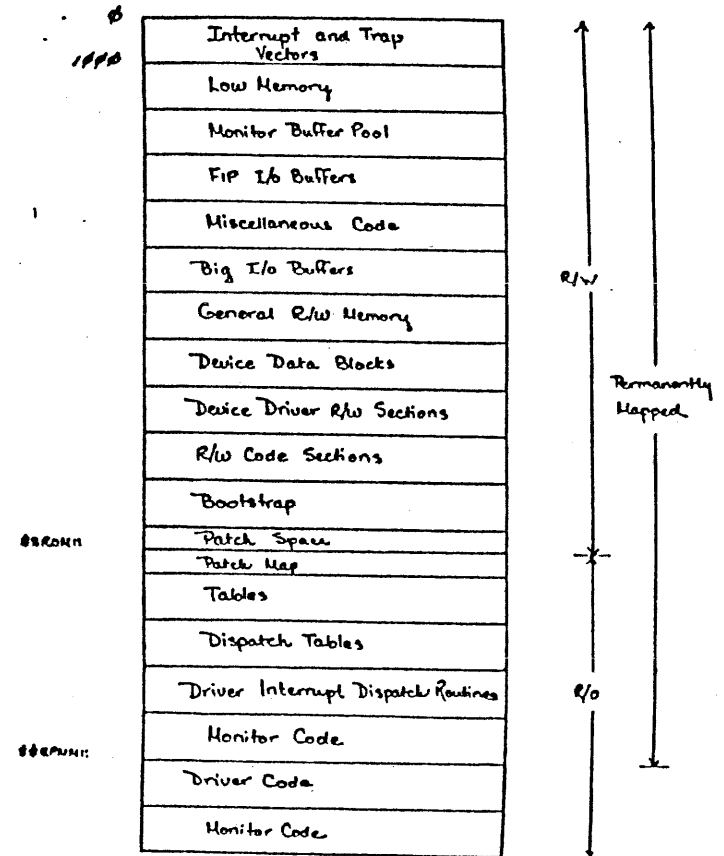


Figure MON-8. RSTS Phase

1000	Date and Time Information
1002	Current Job Information
1040	Root of Memory Control List (MEMLST)
1100	System Default RTS Block
1140	Tail of Memory Control List
1200	ERRCAP Receiver ID Block
1240	Null Run-Time System Block
1300	FIP Job Information
1736	Monitor Stack
2400	FIP STACK

Figure MON-9. Low Memory (Fixed Locations)

DATE		1000
TIME		1002
TIMCLK	TIMSEC	1004
NEXT	JOB	1006
JOBDA		1010
JOBF		1012
IOSTS		1014
JOBWRK		1016
JOBJD2		1020
JOBRTS		1022
CPUTIM		1024
JOBWDB		1026

Figure MON-10. Monitor Low Memory Area

DATE	Today's date in internal form
TIME	Current time in internal form
TIMSEC	Seconds to next minute
TIMCLK	Ticks to next second
JOB	Job running now
NEXT	Next job to run
JOBDA	Pointer to current job data block (JDB)
JOBF	Pointer to current job flags (JDB @ JDFLAG)
IOSTS	Pointer to current I/O status (JDB @ JDIOST)
JOBWRK	Pointer to job's work block (WRK)
JOBJD2	Pointer to job's second job data block (JDB2)
JOBRTS	Pointer to job's run-time system block (RTS)
CPUTIM	Pointer to current CPU time bucket (@ J2TICK)
JOBWDB	Pointer to job's window descriptor block (WDB @ W.WIN1)

*Filled in by the scheduler

R.LINK = \emptyset		1240
R.NAME (... RSX)		1242
R.DEXT = \emptyset		1244
M.PPRV \emptyset		1246
M.PNXT \emptyset		1250
M.TSIZ \emptyset		1252
M.SIZ \emptyset		1254
M.CTRL LCK	M.SIZE \emptyset	1256
M.PHYA \emptyset		1260
R.DATA \emptyset		1262
\emptyset		1264
R.FILE \emptyset		1266
\emptyset		1270
R.CNT		1272
R.MSIZ = 1	R.SIZE (SWPMAX)	1274
R.FLAG \emptyset		1276

Miscellaneous Code

- o 11/60 'MED' code
- o Unibus Memory Parity Code
- o Unibus Window Mapping Code
- o Asynchronous FIP code
- o Resident Library Subroutines

11

Figure MON-11. Null Run-time System Descriptor Block

R.LINK No pointer to next RTS
 R.NAME ...RSX in RAD50
 R.DEXT No default extension
 M.PPRV
 M.PNXT No previous or next pointers in MEMLST
 M.TSIZ No total size
 M.SIZ No RTS size
 M.CTRL Always locked in memory
 M.PHYA No physical address
 R.DATA Nothing on disk
 R.FILE Nothing in UFD
 R.CNT Access and residency counts
 R.SIZE SWPMAX filled in by INIT
 R.MSIZ Minimum legal job size
 R.FLAG No flags

Device Table (DEVTBL)	
Job Table (JOBSTBL)	
Job Status Table (JOBSTAT)	
Job Wait Table (JOBWAIT)	
Ptr. to RTN list (RTNLIST)	
Ptr. to LIO list (LIOLIST)	
Ptr. to Receiver ID list (RNDLIST)	
Job SLEEP time Table (JOBCLK)	
Drive Status Table and Time out Table (TIMTBL)	
Logical Name Table (LOGNAM)	
Drive Queue Roots Table	
Drive Optimization Table	
Unit Count Table (UNTCNT)	
Cluster Ratio Table (CLURAC)	Device Clusterize Table (DEVCLU)
Error Count Table (UNTERE)	File Clusterize Table (UNTCLU)
DEN for [1,2] table (UNTL12)	
Ptr. to FCB List (FCBLIST)	
Unit Options Table (UNTOPT)	Owning Job Table (UNFOWN)
Logical DEN Ptr. Table (SATPTR)	
Largest FCN Table (SATEND)	
Starting FCN (LSA) (SATSTL)	
Starting FCN (MSB) (SATSTM)	
Count of Free Blocks (LSA) (SATCTL)	
Count of Free Block (MSB) (SATCTM)	

Figure MON-12. General R/W Memory

PATCH Map (PATCHS Module) - V6C change

- o Bit map for mandatory and optional patches
- o Sixteen words long
- o Bit 'turned on' when patch is installed

\$\$\$01::

\$\$\$03::

\$\$\$05::

\$\$\$07::

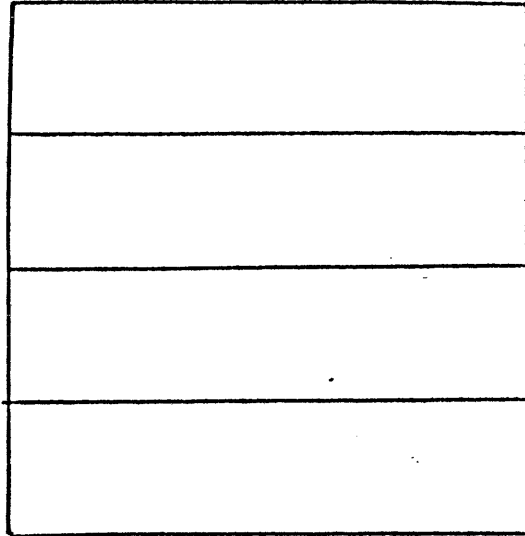


Figure MON-13. PATCHS Module Layout

\$\$\$01	(4 words)	Monitor patches 3.1.1 through 3.1.64
\$\$\$03	(4 words)	Terminal service patches 3.3.1 through 3.3.64
\$\$\$05	(4 words)	File processor patches 3.5.1 through 3.5.64
\$\$\$07	(4 words)	Device driver patches 3.7.1 through 3.7.64

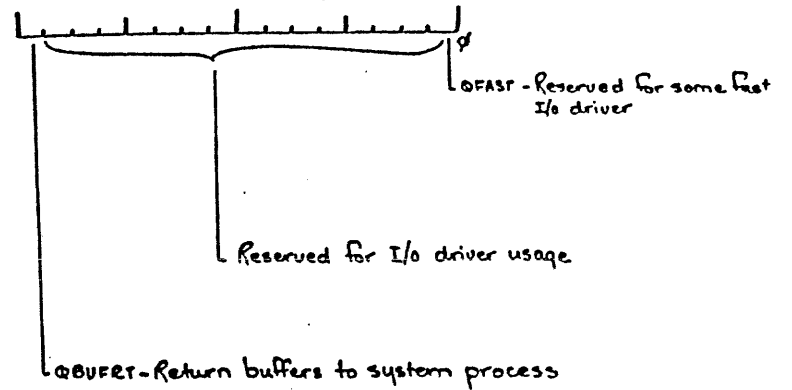


Figure MON-14. Level 3 Queue Bit Assignments (L3QUE)

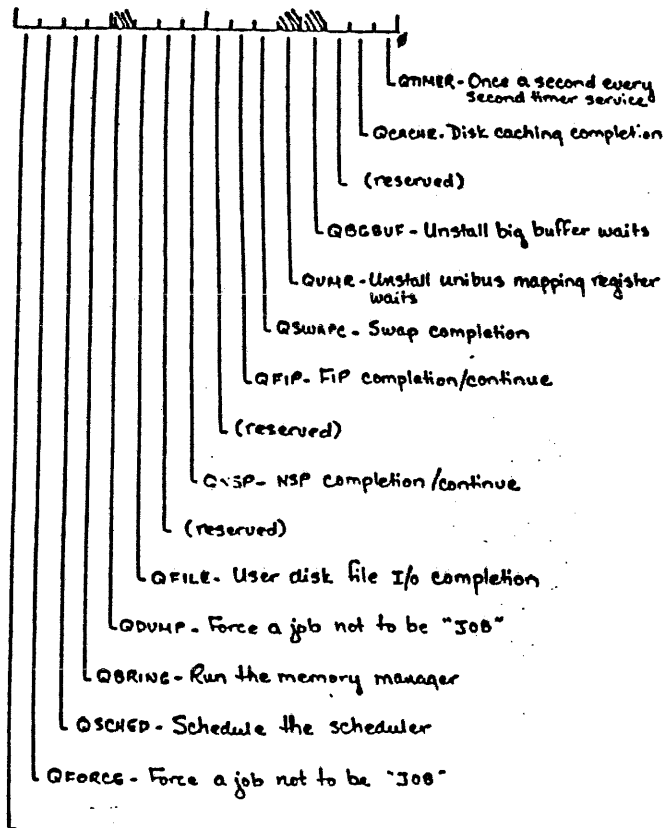


Figure MON-15. Level 3 Queue Bit Assignments (L3QUE2)

R.LINK	0
R.NAME	2
	4
L.PPN	6
R.MCTL	10
	12
	14
	16
R.KSIZ	
M.PHYA	20
R.DATA	22
	24
R.FILE	26
	30
R.CNT	32
L.PROT	34
L.STAT	34
R.FLAG	36

Figure MON-16. Resident Library Description Block (LIB)

R.LINK	Pointer to next resident library
R.NAME	Resident library name in RAD50
L.PPN	Resident library's owning PPN
R.MCTL	Memory control area
R.KSIZ	Resident library size in K words
M.PHYA	Physical address of start of library divided by 100 (octal)
R.DATA	FUN and FBN of file data (i.e. the library)
R.FILE	Name entry offset (/2) and FBN of UFD for file
R.CNT	Attach and map counts
L.STAT	Library's status
L.PROT	Library's protection code
R.FLAG	Resident library flags

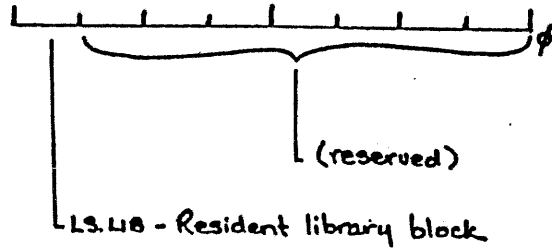


Figure MON-17. Flag Bits (L.STAT)

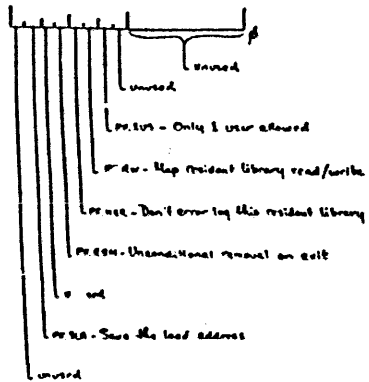


Figure MON-18. Resident Library Flags (R.FLAG)

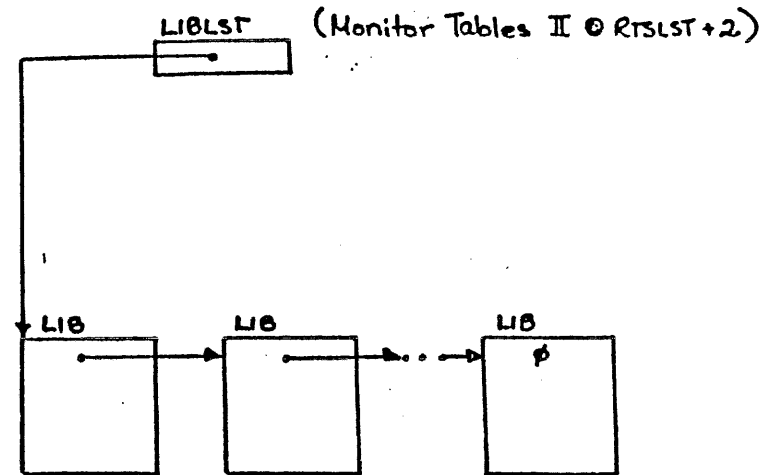


Figure MON-19. Monitor Tracking of 'Added' Resident Libraries

System Wide Logical Name Entries

- o Each entry requires 5 words

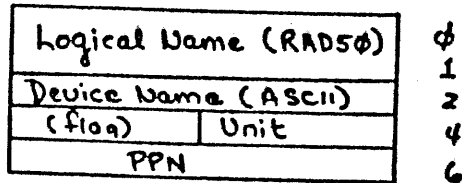


Figure MON-20

- o System wide logical name table
- o Each entry requires 5 words

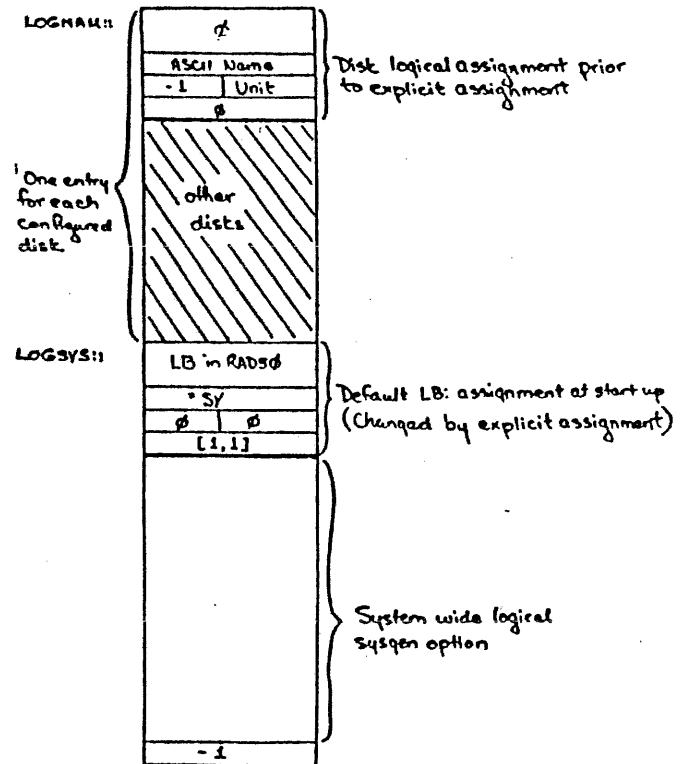


Figure MON-21. Logical Name Table (LOGNAM)

JOB RELATED STRUCTURES

JOBCLK Table -- V6C Change

- o Word table of sleep time for each job in seconds
- o Length is job maximum + 1
- o Indexed by job number times 2

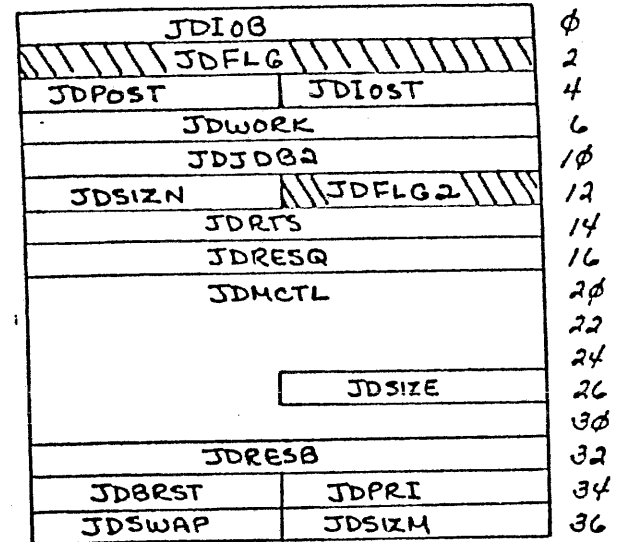


Figure MON-22. Job Data Block #1 (JDB)

JDI0B Pointer to the I/O block (I0B)
 JDFLG Job status flags
 JDI0ST I/O status for job, error codes
 JDPOST Index for bit pattern to post
 JDWORK Pointer to job work block (WRK)
 JDJOB2 Pointer to job data block extension (JDB2)
 JDFLG2 More job status flags
 JDSIZN Size for job to be on next residency
 JDRTS Pointer to run-time system block (RTS)
 JDRESQ Job's residency quantum
 JDMCTL Job's memory control
 JDRESB L3QUE bits to set on residency
 JDPRI Job's priority (-128. < = 127.)
 JDBRST Job's runnable burst
 JDSIZM Job's private memory maximum
 JDSWAP Job's swap parameter

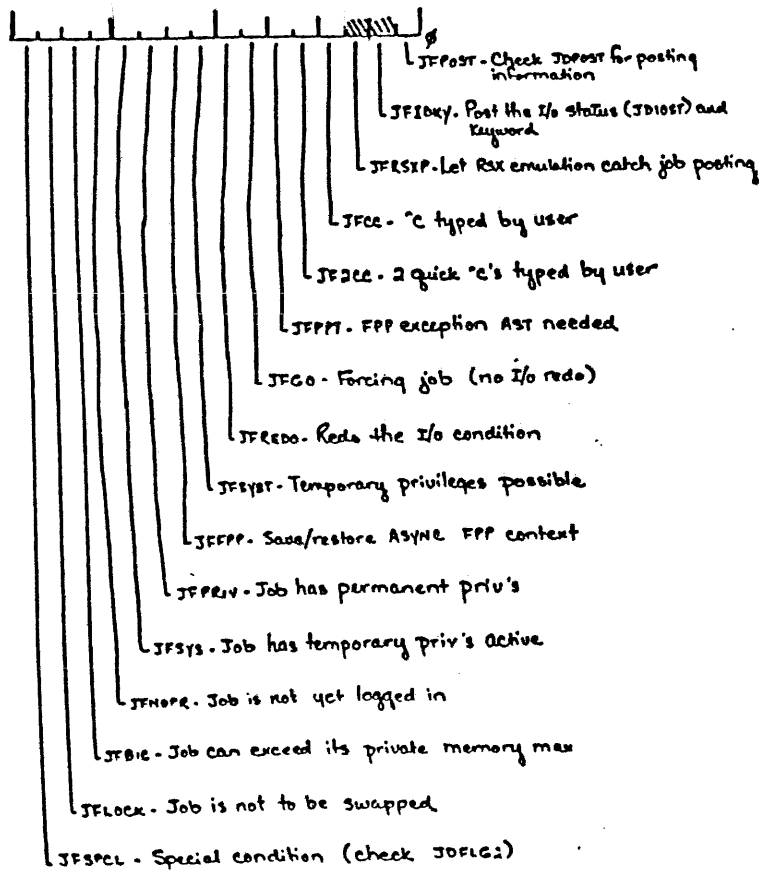


Figure MON-23. Job Flag Assignments (JDPLG)

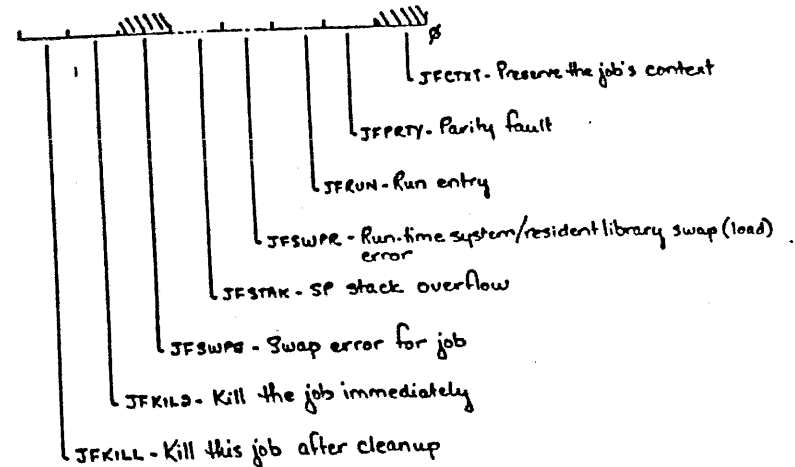


Figure MON-24. Job Flag Assignments (JDPLG2) -- V6C change

J2TICK		0
J2CPU		2
J2CON		4
J2KCT		6
J2DEV		10
J2CPUM	J2KCTM	12
J2NAME		14
J2DRTS		16
J2MPTR		20
J2PPTR		22
J2PCNT		24
J2PPN		26
J2UFDR		30
J2WPTR		32
J2CPUI		36

Figure MON-25. Job Data Block #2 (JDB2)

J2TICK	Number of clock ticks unconverted
J2CPU	CPU time collected for the job
J2CON	Connect time collected for the job
J2KCT	Kilo-core-ticks collected for the job
J2DEV	Device time collected for the job
J2KCTM	MSB of Kilo-core-ticks collected for the job
J2CPUM	MSB of CPU time collected for the job
J2NAME	Job's program name
(2 words)	
J2DRTS	Pointer to default run-time system for the job
J2MPTR	Pointer to job's receiver ID block (SND)
J2PPTR	Large data posting pointer
J2PCNT	Large data posting count
J2PPN	Job's project programmer number
J2UFDR	Retrieval pointer to job's system disk UPD
J2WPTR	Pointer to job's window descriptor block (WDB @ W.WIN1)
J2CPUI	Incremental CPU time calculation bucket

S\$STS	S\$IDX	0
S\$UNT	S\$PT	2
S\$SIZ		4
S\$NVB		6
S\$FVB		10
S\$FLG	S\$CLUS	12
S\$UFND		14
		16
S\$WND		20
		22
		24
		26
		30
		32
		34
		36

Figure MON-26. Small Control Block (SCB)

\$\$IDX	Driver Index
\$\$STS	Status bits for file
\$\$PT	Pending transfer count
\$\$UNT	FIP unit number
\$\$SIZ	Size of file
\$\$NVB	Next virtual block to read/write
\$\$FVB	First virtual block in window
\$\$CLUS	File Cluster size -1
\$\$FLG	SCB flag bits
\$\$UFND	Name entry offset (divided by 2) and FBN of UPD block
\$\$WND	Current retrieval window and flags

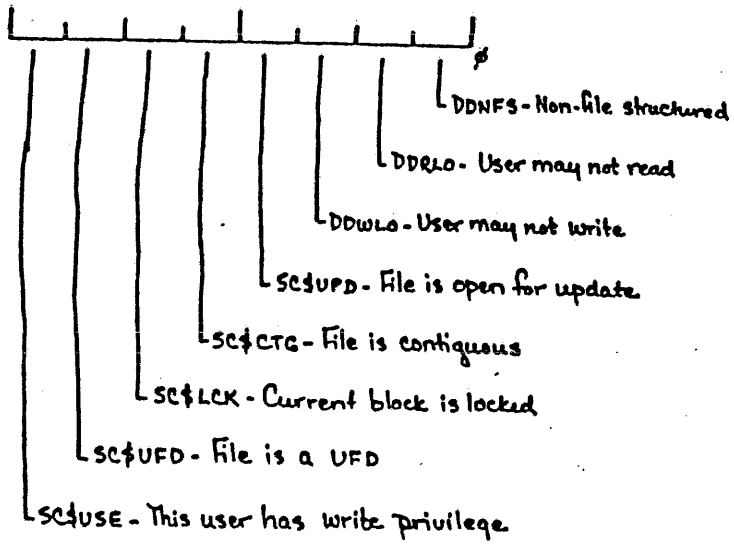


Figure MON-27. SSSTS

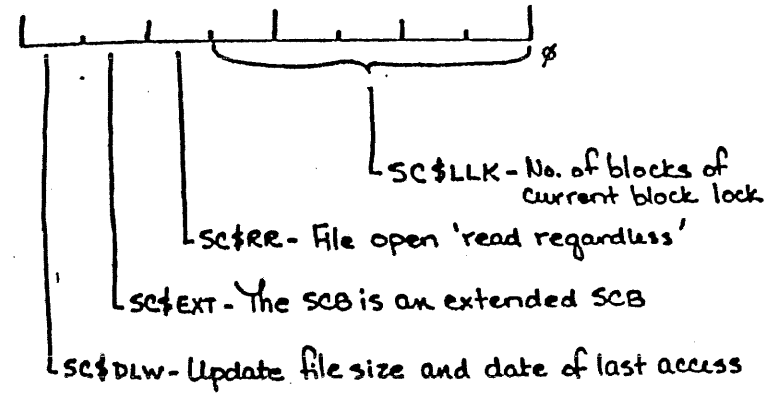


Figure MON-28. SSPLG

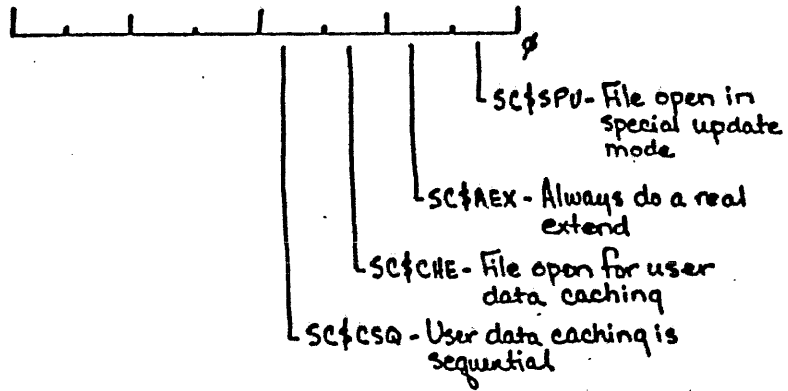


Figure MON-29. SSWND - flags

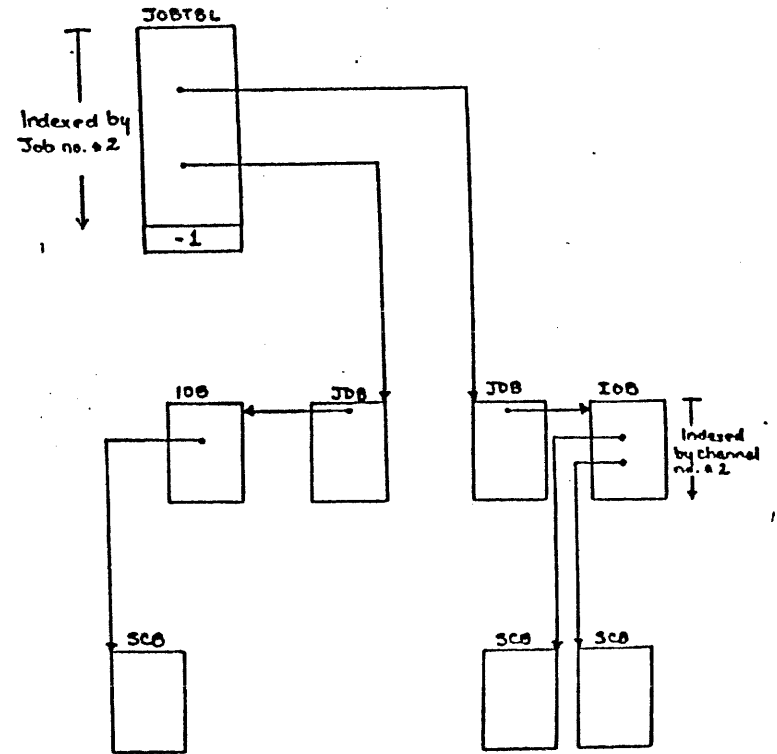


Figure MON-38. Monitor Tracking of Files Opened (Small File System)

- o Write access and update information is kept on disk in the directory.

F\$LINK		0
F\$FID		2
F\$PPN		4
F\$NAM		6
		10
		12
F\$PROT	F\$STAT	14
F\$RCNT	F\$ACNT	16
F\$WFND		20
		22
F\$UFND		24
		26
F\$SIZM	F\$SUNT	30
F\$SIZL		32
F\$CLUS		34
F\$WCB		36

Figure MON-31. File Control Block (FCB)

F\$LINK Pointer to next FCB on this FIP unit
 F\$FID File ID of file (link within UFD/MFD)
 F\$PPN PPN of file
 F\$NAM File name and extension in RAD50
 F\$STAT Status byte
 F\$PROT Protection code
 F\$ACNT Access count for normal/update opens
 F\$RCNT Access count for read regardless opens
 F\$WFND FBB of first retrieval entry
 F\$UFND FBB of name entry
 F\$SUNT FIP unit number
 F\$SIZM File size MSB (number of FBN's if NFS)
 F\$SIZL File size LSB
 F\$CLUS File cluster size (W\$FCB points here)
 F\$WCB Pointer to first WCB open on this file

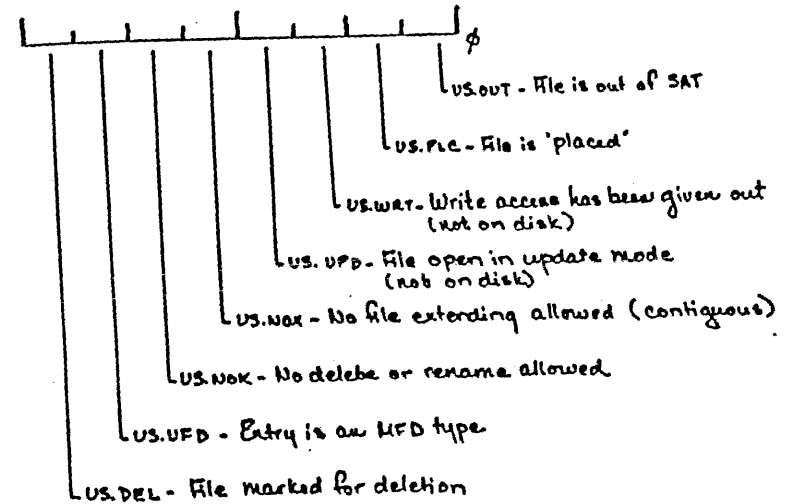


Figure MON-32. Status Byte (FSSTAT)

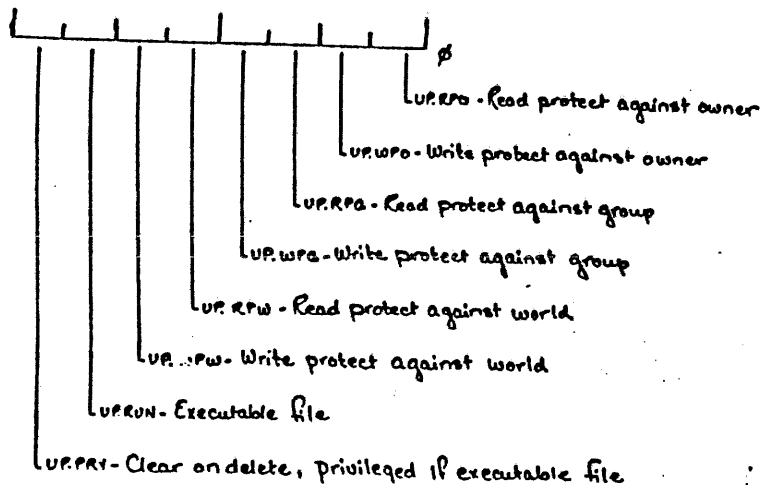


Figure MON-33. Protection Code (PSPROT)

W\$STS	W\$IDX	0
W\$FLAG	W\$JBNO	2
W\$NVBM	W\$PT	4
W\$NVBL		6
W\$FCB		10
W\$REN		12
W\$WCB		14
W\$NXT		16
W\$WND		20
		22
		24
		26
		30
		32
		34
		36

Figure MON-34. Window Control Block (WCB)

W\$IDX Driver index
 W\$STS Status bits for file
 W\$JBNO Job number *2 of owner
 W\$FLAG WCB flag bits
 W\$PT Pending transfer count
 W\$NVBM Next virtual block to read/write MSB (FBN if NFS)
 W\$NVBL Next virtual block to read/write LSB
 W\$FCB Pointer to FCB for file @ F\$CLUS
 W\$REN Retrieval entry number of current window
 W\$WCB Pointer to next WCB open on same FCB + flag bits
 W\$NXT FBN of next window
 W\$WND Current retrieval window

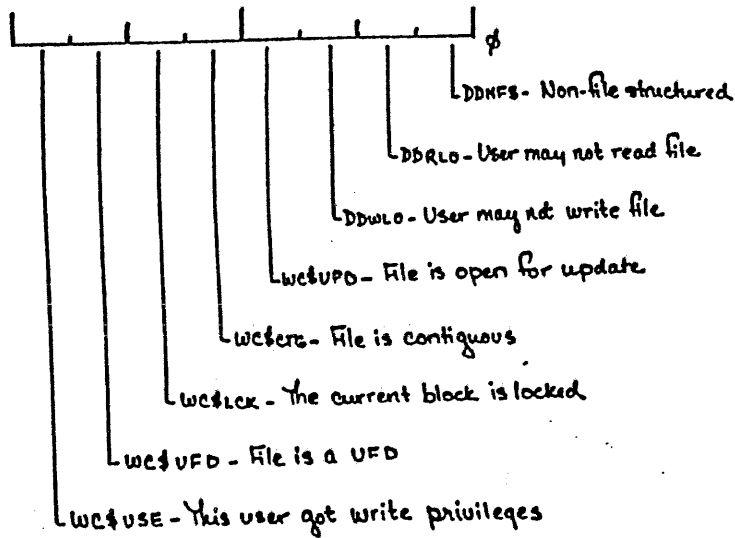


Figure MON-35. Status Bits (WSSTS)

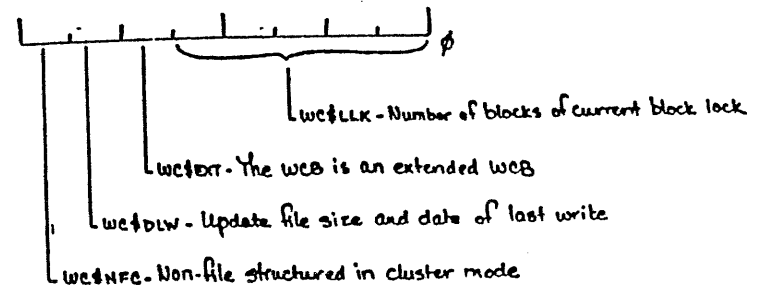


Figure MON-36. WCB Flag Bits (WSFLAG)

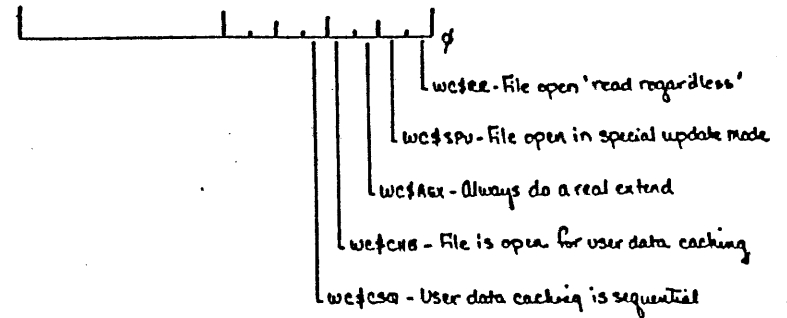


Figure MON-37. Flag Bits @ WSVCB

FCB List Table (FCBLST)

- o Word table of pointers to FCBS
- o One entry for every disk unit on the system
- o Indexed by FIP unit number (FUN) *2

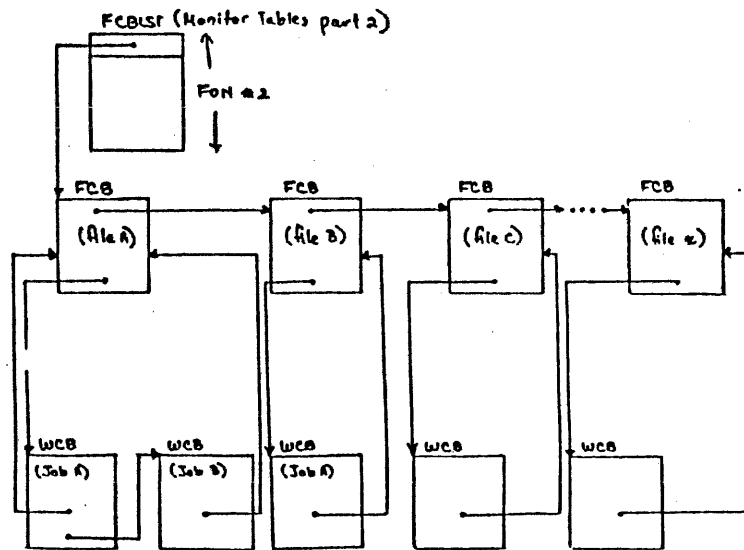


Figure MON-38. Monitor Tracking of Files Opened (Large File System)

- o Write access and update information kept in FCB @ FSSTAT, not in directory on disk

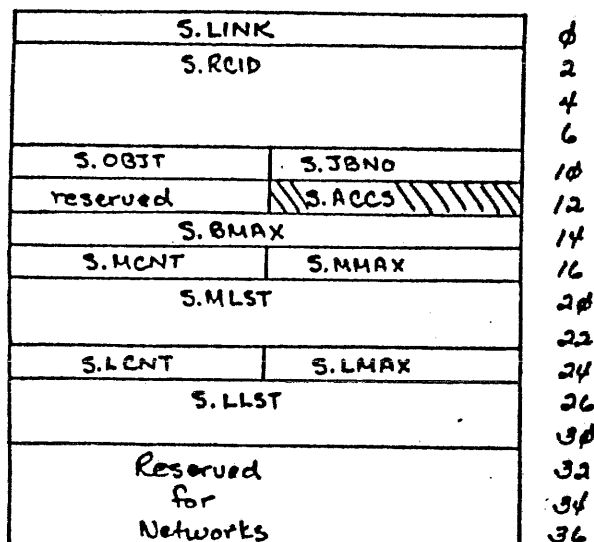


Figure MON-39. Receiver ID Block (RIB)

- S.LINK Link to next block
- S.RCID Receiver ID in ASCII
- S.JBNO Job number times two
- S.OBJT Object type (networks)
- S.ACCS Access control
- S.BMAX Buffer maximum in bytes
- S.MMAX Message maximum
- S.MCNT Message count
- S.MLST Message list root/tail pointers
- S.LMAX Link maximum
- S.LCNT Link count
- S.LLST Link list root/tail pointers

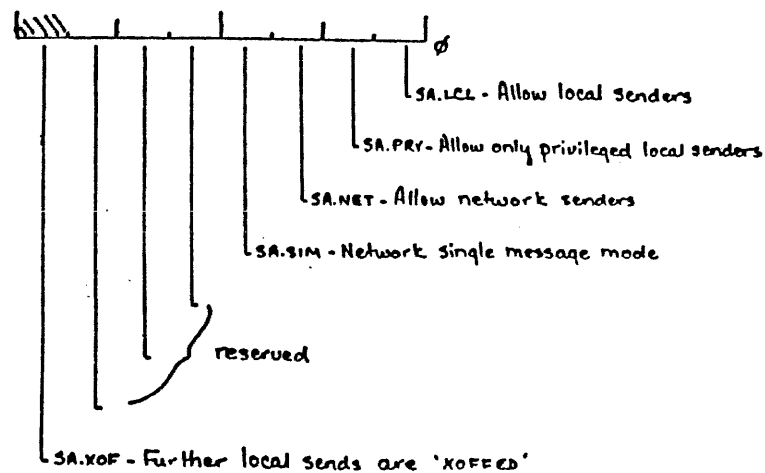
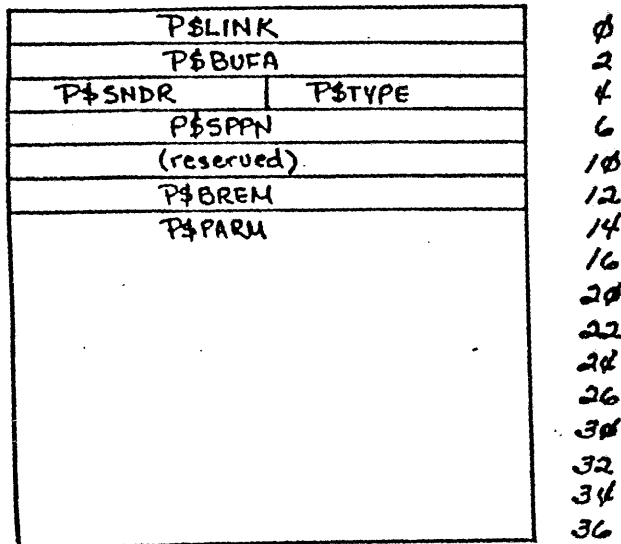


Figure MON-40. Access Control (S.ACCS)



"Contorted" Buffer Address

- o If low five bits are zero then a small buffer
- o If in XBUF, address has been rotated left seven bits so that low bits are not zero

Figure MON-41. Pending Message Block (PMD)

P\$LINK Link to next block
P\$BUFA "Contorted" buffer address (pointer to buffer containing message data)
P\$TYPE Message type code (DECNET/E)
P\$SNDR Job number times two of sender
P\$PPN Project Programmer number of sender
P\$BREM Number of bytes remaining in data message
P\$PARM (20 bytes) Parameters

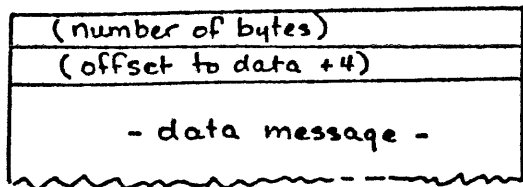


Figure MON-42. Buffer Containing Message (Long Messages Only)

(number of bytes) Length of data message
(offset to data + 4) Pointer to next character to be "received"

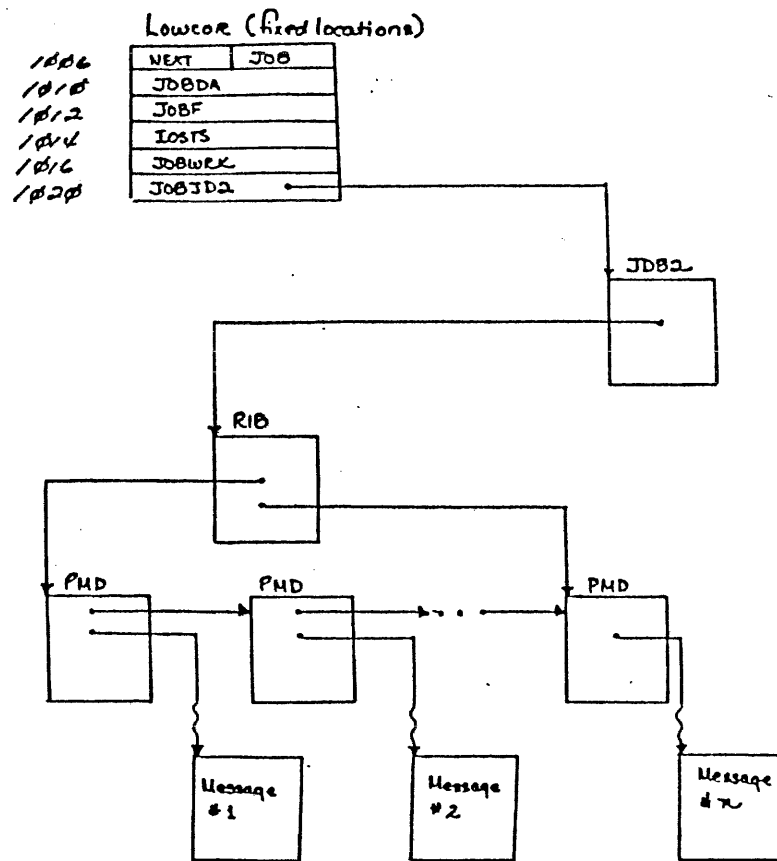


Figure MON-43. Monitor Tracking of Message Send/Receive for "JOB"

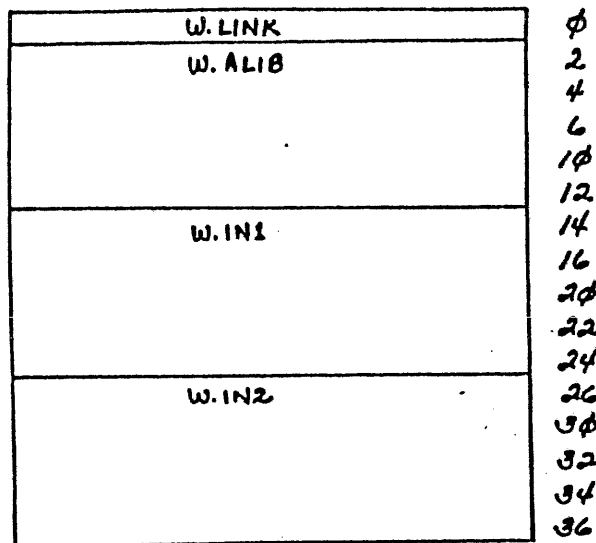


Figure MON-44. Window Descriptor Block (WDB)

W.LINK Link to next WDB @ offset + 2
W.ALIB Addresses of LIB blocks for attached
 libraries
W.IN1 Address window #1
W.IN2 Address window #2

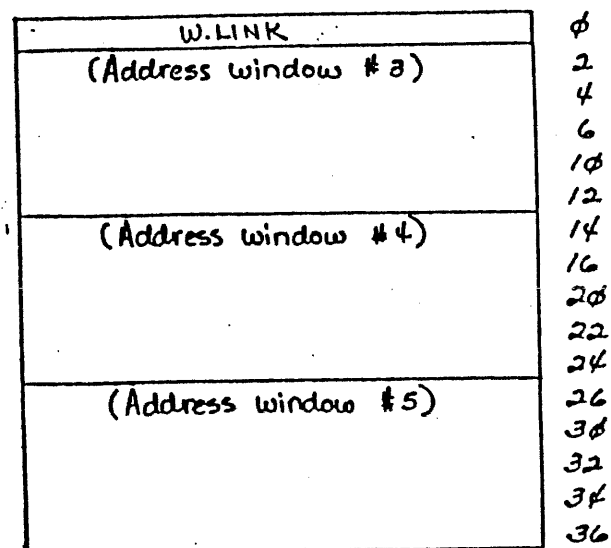


Figure MON-45. Window Descriptor Block #2 (WDB)

- o Exists only if more than two APRs are mapping a resident library (or libraries)

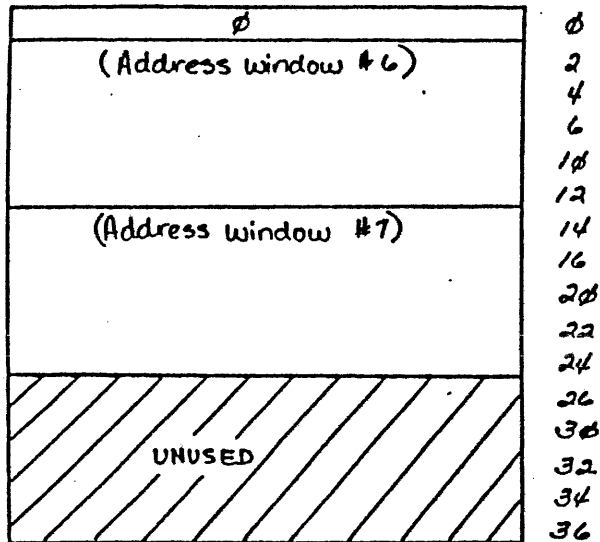


Figure MON-46. Window Descriptor Block #3 (WDB)

- o Exists only if more than five APRs are mapping a resident library (or libraries)

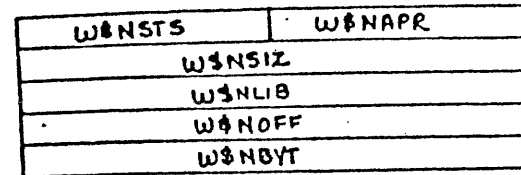


Figure MON-47. Address Windows

- W\$NAPR Window base APR (1-7)
- W\$NSTS Window status
- W\$NSIZ Window size/100
- W\$NLIB Pointer to library pointer in W.ALIB
- W\$NOFF Map offset/100 into library
- W\$NBYT Map length in bytes

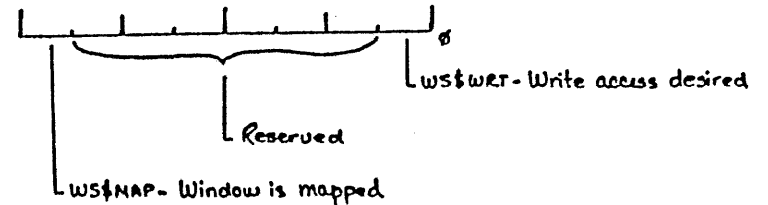


Figure MON-48. Flag Bits (W\$NSTS)

Unchanged JOB Data Structures

Tables

JOBTABL
JBSTAT
JBWAIT
JSBTBL

Blocks

RTS
PMD
IOB
WRK
MEMLST Sub block

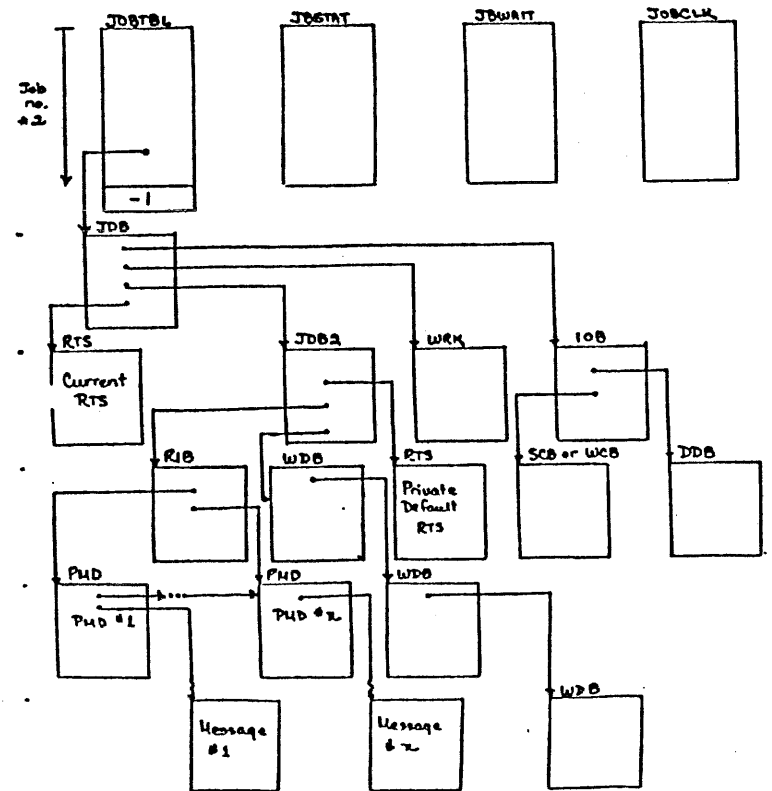


Figure MON-49. Job Structure

DEVICE RELATED STRUCTURES

New Handler Indexes

36	DT2HND	DEctape II
40	KMCHND	KMC11 Microprocessor
42	IBMHND	IBM Interconnect

New Driver Indexes

o Values dynamically assigned at SYSGEN

IDX.MS	TS04 Magtape
IDX.DD	TU58 DEctape II
IDX.XK	KMC11 Microprocessor

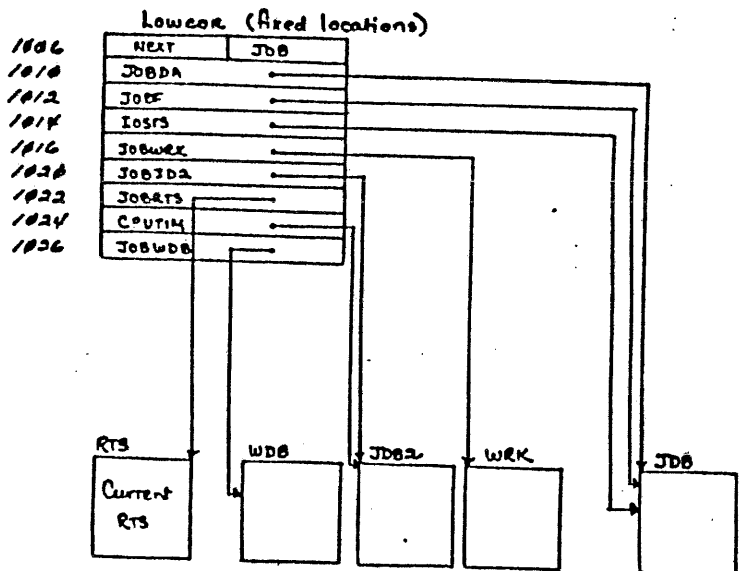


Figure MON-50. Monitor Tracking of "JOB"

Device Name	Alternate Name	DDB size (bytes)	Control C Interruptible	Buffer Quota	Check before SLEEP	Handles Unibus map entries
KB	TT	(varies)	Yes		Yes	-
NL	-	8.	No	-	No	-
PK	-	12.	Yes		Yes	-
ES	-	12.	No	-	No	-
XH	-	48.	Yes	-	Yes	-
LP	-	32.	Yes	2φ.	No	-
PE	-	16.	Yes	1φ.	No	-
PP	-	16.	Yes	1φ.	No	-
CE	-	178.	Yes	-	No	-
CD	CE	178.	Yes	-	No	-
HT	-	38.	No	-	No	Yes
UH	HT	38.	No	-	No	Yes
US	HT	9φ.	No	-	No	Yes
DT	-	32.	No	-	No	-
DX	Dy	28.	No	-	No	Yes
DD	-	24.	No	-	No	-

Figure MON-51. Device Information

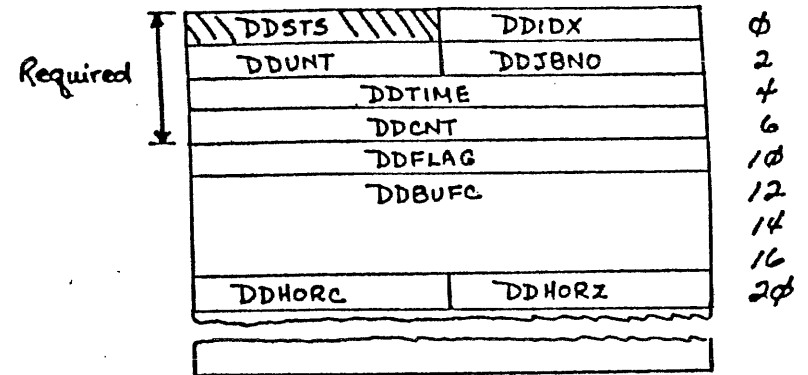


Figure MON-52. Device Data Block (DDB)

- DDIDX Driver Index
- DDSTS Status and access control
- DDJBNO Owner job number times 2 (0 if free)
- DDUNT Device unit number
- DDTIME Time assigned or inited
- DDCNT Init count and assignment control
- DDFLAG Device dependent flags
- DDBUFC Buffer chain control area
- DDHORZ Horizontal position
- DDHORC Characters per line + 1

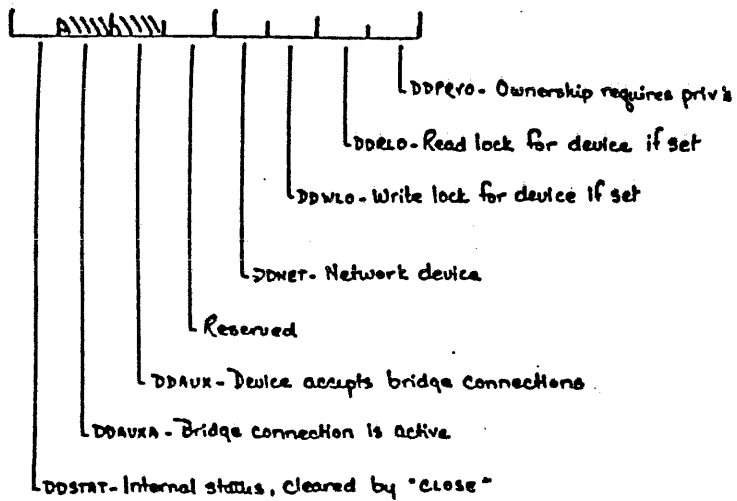


Figure MON-53. Status and Access Control (DDSTS)

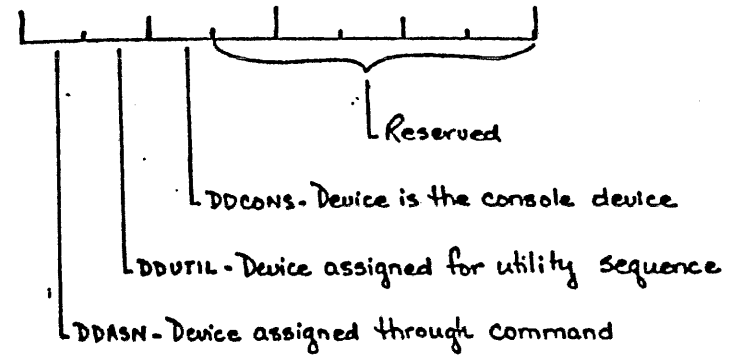


Figure MON-54. Init Count and Assignment Control

BBDOFF	BBSTS	0
BGGIRB		2
BBASND		4
BBAAPS		6
BBAUNT	BBDUNT	10
BBDSTA		12
BBDINT		14
BBDAPS		16
Reserved		20
		22
		24
		26
		30
		32
		34
		36

Figure MON-55. Device Driver to Auxiliary Processor Bridge Block (BB)

BBSTS	Status
BBDOFF	Data offset within data buffer
BGGIRB	Pointer to "GET IRB/data buffer" subroutine
BBASND	Pointer to "SEND IRB/data buffer" subroutine
BBAAPS	Pointer to PAR5 value for above two subroutines
BBDUNT	Device's control unit number
BBAUNT	Auxiliary's control unit number
BBDSTA	Pointer to "status change" entry in device driver
BBDINT	Pointer to "interrupt" entry in device driver
BBDAPS	Pointer to PAR5 value for above two entry points

IRBFUN		0
IRBSTA		2
IRB18M	IRBCHN	4
IRB18L		6
IRBMOD		10
IRBCON		12
IRBNXT		14
IRBBFL		16
IRBMMU		20
IRBSIZ		22
IRBDPT		24
IRBFPT		26
IRBEPT		30
Reserved		32
		34
		36

Figure MON-56. Interprocess Request Block (IRB)

IRBFUN	Function code and its modifier
IRBSTA	Status
IRBCHN	Channel number
IRB18M	MSB of 18-bit unibus address of data within buffer
IRB18L	LSB of 18-bit unibus address of data within buffer
IRBMOD	Data modifier
IRBCON	Contorted address of data buffer
IRBNXT	Link to next IRB (if chained)
IRBBFL	Byte size of data within data buffer
IRBMMU	MMU address of whole data buffer
IRBSIZ	Byte size of whole data buffer
IRBDPT	Virtual (through PAR5) done pointer
IRBFPT	Virtual (through PAR5) fill pointer
IRBEPT	Virtual (through PAR5) empty pointer

Unit Count Table (UNTCNT)

- o Word table of device flags and open counts
- o One entry for each disk unit configured on the system
- o Address returned in Monitor Tables I

Name	Description
UNTCNT	Device flags and open counts
DEVCLU	Device cluster size - byte table
CLUFAC	Clusterratio (Pcs/Dcs) - byte table at DEVCLU + 1
UNTCUW	Pack cluster size - byte table
UNTCUW	Error count - byte table at UNTCUW + 1
UNTLIB	Starting DCN for [1,2] UFDs
UNTSIZ	Unit size in DCNs
UNTOWN	Owning job number - byte table
UNTOPT	Disk status bits - byte table at UNTOWN + 1
HID\$XX	Massbus ID

Figure MON-57. Tables Indexed by FUN #2

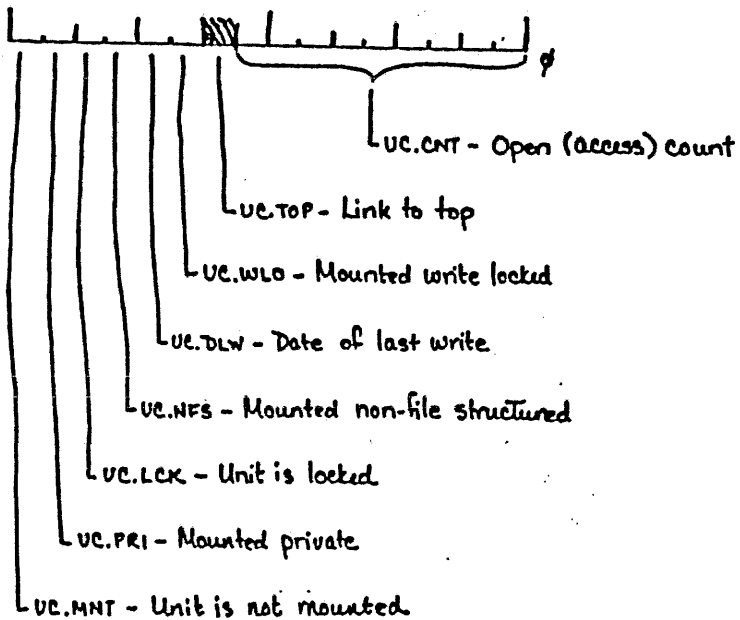


Figure MON-58. Unit Count Table (UNCNT) Status Bits
- V6C Change

Unit Options Table (UNTOPT)

- o Byte table
- o One entry for each disk unit configured on system
- o Located at job owning table (UNTOWN) + 1

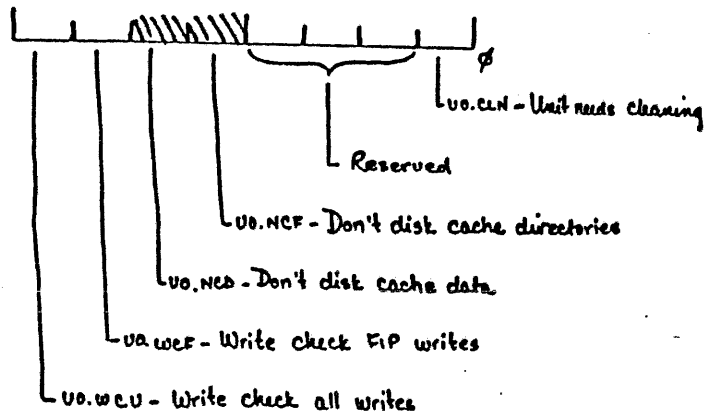


Figure MON-59. Unit Options Table (UNTOPT) Status Bits

Disk Indexes

- o Used internally by disk subsystem
- o Dynamically assigned at sysgen time

DIGITAL

RSTS/E SWS UPDATE
Monitor Internals

RSTS/E SWS UPDATE
Monitor Internals

DSKMAP Table

- o Word table of disk indexes
- o Indexed by FUN #2

Name	Description
DSKDSP	Word dispatch table to disk initiate routines
DSKINT	Word dispatch table to interrupt service routines
DSKTMO	Word dispatch table to timeout service routines
DSKCSR	Pointers to CSRs
DSKPTR	Pointers to D?DINT @ DAS\$00 - base queue root pointer

D?DINT layout

Figure MON-68. Tables Indexed by Disk Index

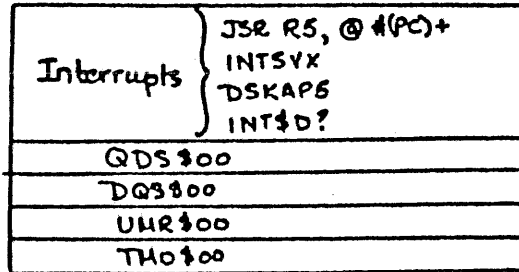


Figure MON-61. D?DINT Layout

QDS\$00 - Entry for setting up DSQ
 DQS\$00 - Base queue root pointer
 UMR\$00 - Entry for UMR freed up
 TMO\$00 - Entry for time outs

Device Name Table (DEVNAM)

- o Word table of ASCII device names
- o One entry for each disk type supported by RSTS/E
- o One entry for keyboards
- o One entry for the null device
- o One entry for pseudo keyboards
- o One entry for every other device type sysgenned
- o Entries for alternate names of sysgenned devices

Device Maximum Unit Number Table (DEVCONT)

- o Word table of maximum unit numbers
- o Length equals DEVNAM - 1
- o Entries correspond to entries in DEVNAM table

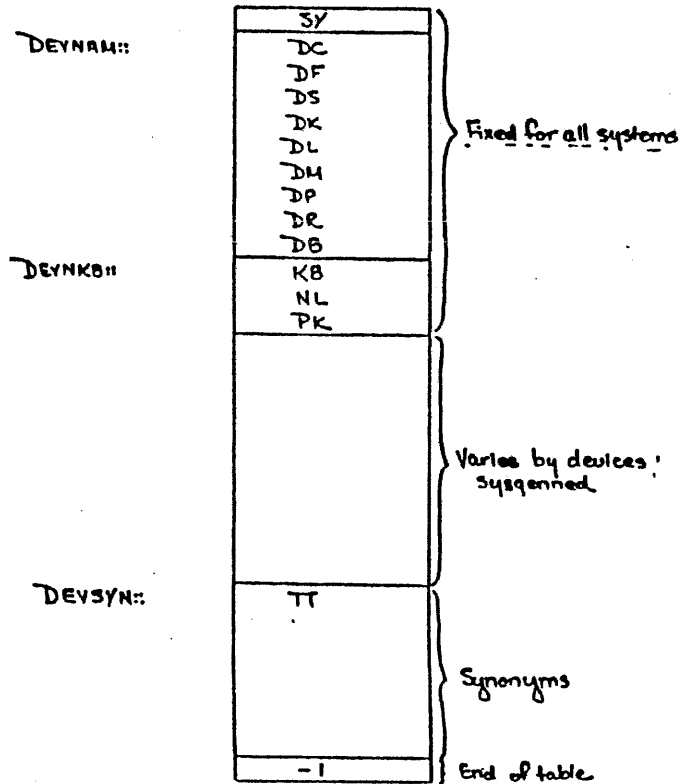


Figure MON-62. DEVNAM Table

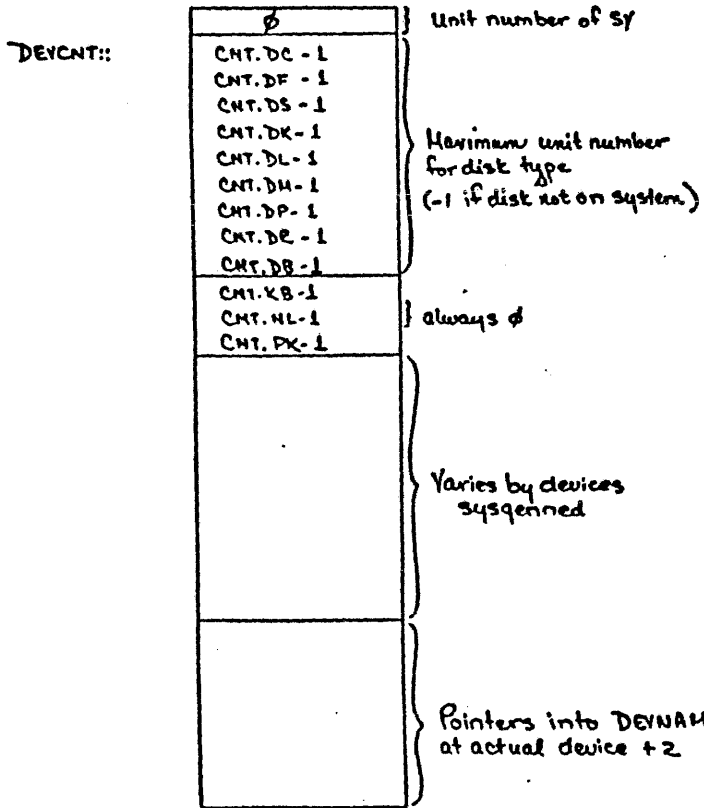


Figure MON-63. DEVICNT Table

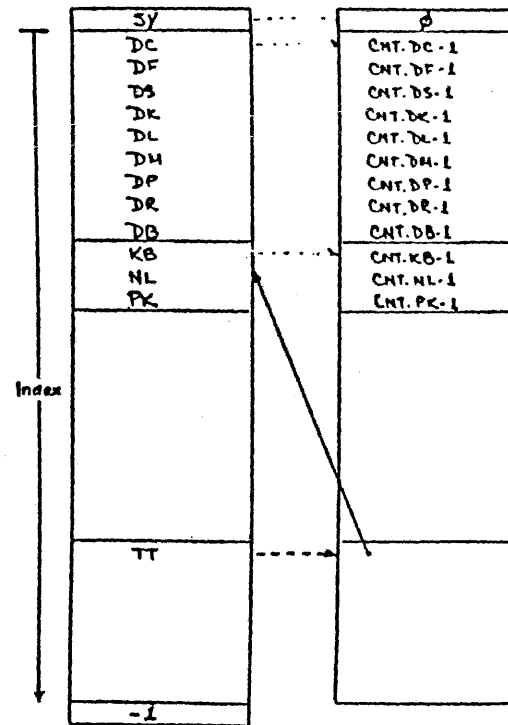


Figure MON-64. Indexing of DEVNAM and DEVICNT

Device Pointer Table (DEVPTR)

- o Word table of pointers
- o Length equals DEVCNT minus alternates
- o Entries correspond to entries in DEVNAM table
- o Disk entries point to unit 0 for each sygennd disk type in unit count table (UNTCNT)
- o Non-disk entries point to unit 0 entries in the device table (DEVTBL)

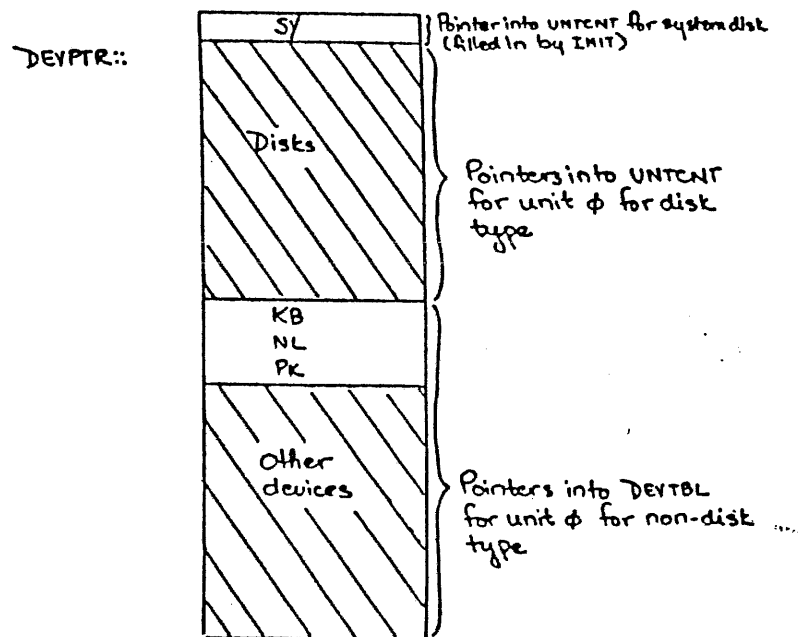


Figure MON-65. DEVPTR Table

Device Table (DEVTBL)

- Word table of pointers to device data blocks (DDBs) for each non-disk device sygened on the system.

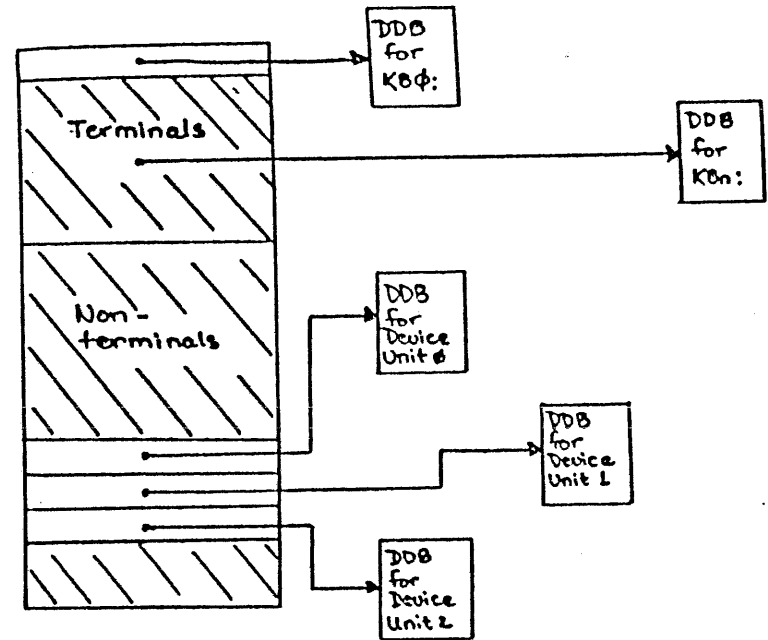


Figure MON-66. DEVTBL Table

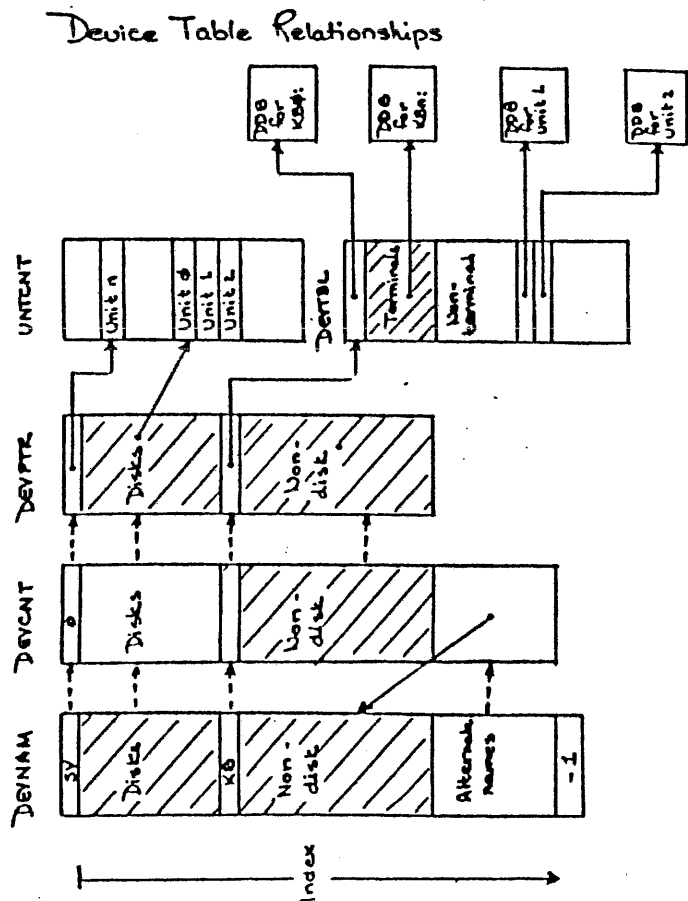


Figure MON-67. Device Table Relationships

Determining PIP Unit Numbers (FUN) for a System

- From information obtained from DEVNAM and DEVcnt tables determine the devices and number of units actually sysgen'd on the system.

Note that the order of the devices in the DEVNAM table is important.
- Based on this "order" FUN's are assigned starting with 0.

Note

Many tables referenced in this module and in the exercise need this information.

Queue link word		0
DSQERR	DSQJOB	2
DSQL3Q		4
DSQFBM	DSQUNT	6
DSQFBL		10
DSQMAN	DSQRFN	12
DSQMAL		14
DSQCNT		16
DSQFAR	DSQFUN	20
DSQMISC		22
DSQTOT		24
DSQPDA		26
DSQOPT		30
DSQSAV	DSQOUN	32
DSQPTO		34
DSQPUN	DSWCTO	36

DSQFAR Queue fairness count
 DSQMISC Miscellaneous pointer
 DSQTOT Total transfer count
 DSQPDA Physical disk address
 DSQOPT Disk optimization word
 DSQOUN Physical unit number #2
 DSQSAV Saved function
 DSQPTO Offset pointer
 DSWCTO Offset retry count
 DSQPUN Physical unit number

Figure MON-68. Disk Request Queue Block (DSQ)

DSQJOB Requesting job number #2
 DSQERR Retry count and error flag
 DSQL3Q Completion queue and L3Q bits to set
 DSQUNT FIP unit number
 DSQFBM FIP block number (MSB)
 DSQFBL FIP block number (LSB)
 DSQRFN RHL function
 DSQMAN Memory address of transfer (MSB)
 DSQMAL Memory address of transfer (LSB)
 DSQCNT Word count of transfer
 DSQFUN Function code (WFUN.C=0-Write with write check disk, WFUN=2-Write disk, RFUN=4-Read disk, RFUN.C=6-Write check disk)

INTERFACING WITH THE MONITOR FROM USER MODE

XRLN	0
XRBC	2
XRLOC	4
XRBLKM	6
XRCI	10
XRBLK	12
XRTIME	14
XRMOD	14

Figure MON-69. Transfer Request Queue Block (XRB)

XRLN Length of I/O buffer in bytes
 XRBC Byte count for transfer
 XRLOC Pointer to I/O buffer
 XRCI Channel number times two for transfer
 XRBLKM Random access block number (MSB)
 XRBLK Random access block number (LSB)
 XRTIME Wait time for terminal input
 XRMOD Modifiers

(reserved)	(error code)	0
FQFUN	FQJOB	2
FQSZM or FQERNO or FQFIL		4
FQPPN		6
FQNAM1		10
		12
FQEXT		14
FQSIZ		16
FQNAM2 or FQBUFL		20
FQNAM2 or FQMODE		22
FQNAM2 or FQFLAG		24
FQPROT	FQPFLG	26
	FQDEV	30
(flag)	FQDEVN	32
	FQCLUS	34
	FQAGENT	36

Figure MON-70. File Request Queue Block (FIRQB)

FQJOB Job number times two
 FQFUN Function requested
 FQERNO Error message code and text begin
 FQFIL
 (byte) Channel number times two
 FQSZM
 (byte) File size in blocks (MSB)
 FQPPN Project Programmer number
 FQNAM1
 (2 words) File name in RAD50
 FQEXT Extension in RAD50
 FQSIZ File size in blocks (LSB)
 FQNAM2
 (2 words) New filename and extension in RAD50

DIGITAL

RSTS/E SWS UPDATE
Monitor Internals

FQBUFL Default buffer length
 FQMODE Mode indicator
 FQFLAG Opened file's flag word as returned
 FQPROT 'Protection code real' indicator
 FQDEV New protection code
 FQDEVN ASCII device name
 FQDEVN Unit number
 (flag) 'Unit number real' indicator
 FQCLUS File cluster size for file creates
 FQNTENT Number of entries on directory lookup

DIGITAL

RSTS/E SWS UPDATE
Monitor Internals

New Monitor Calls (EMT's)

.PLAS	72	Resident library control
.RSX	74	Enter RSX emulation
.ULOG	76	ASSIGN/REASSIGN/DEASSIGN device/user logical

.PLAS (FIQRB @ FQERNO) Function Codes

ATRFQ	0	Attach region (resident library)
DTRFQ	2	Detach region (resident library)
CRAFQ	4	Create address window
ELAFQ	6	Eliminate address window
MAPFQ	10	Map address window
UMPFQ	12	Unmap address window

New .UUO/UUOFQ Subfunction Codes

UU.FIL	-26.	File utility
UU.CNV	20.	Convert date/time to ASCII (V6C)
UU.JOB	24.	Job creation (V6C)
UU.PPN	25.	Wild card PPN Lookup
UU.SYS	26.	Return job status
UU.KMC	27.	Connect KMC-11 to another device

Data Passed:
FIRQB

\\ \\ \\ \\ \\ \\ \\ \\ \\ \\		0
UU.JOB (=24.)	\\ \\ \\ \\ \\ \\ \\ \\	2
Must = 0	0 → Spawn 2 0 0 → Spawn regardless	4
Project number	Programmer No.	6
file name (2 words in RAD50)		10
file extension (RAD50)		12
10 bytes of any information. Placed in created job's core common		16 20 22 24 26
device name (2 ASCII chars.)		30
<> 0 Unit no. real	device unit no.	32
Parameter word (same as .cc)		34
\\ \\ \\ \\ \\ \\ \\ \\		36

Figure MON-71. UU.JOB - Job Creation

The Null (Disappearing) Run-time System

- 1. User runs an executable task image from a run-time system other than RSX.

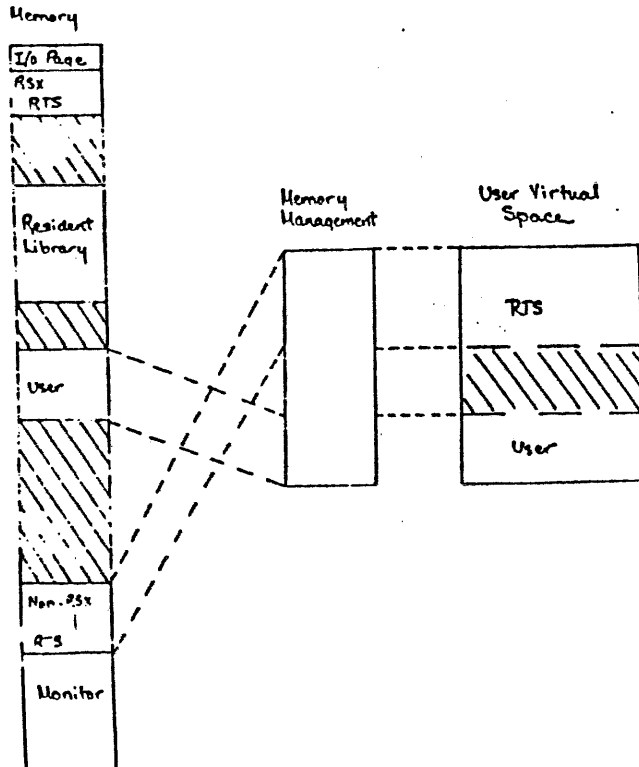


Figure MON-72.

- 2. Monitor renaps user space if necessary (loads run-time system if necessary).

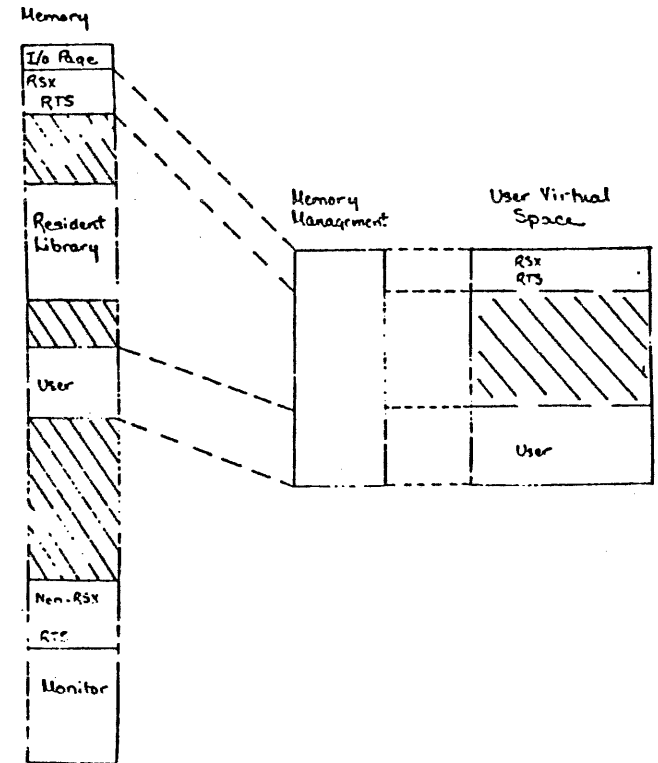
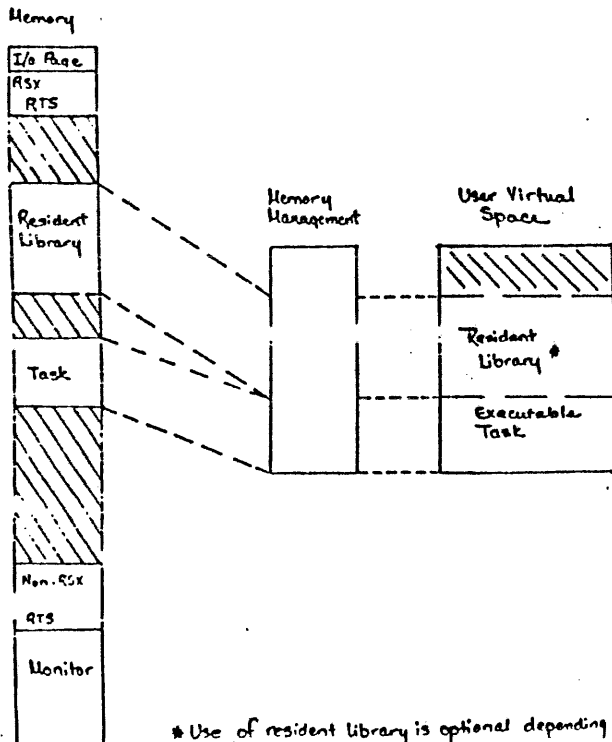


Figure MON-73.

3. RSX loads the executable task and issues the .RSX monitor call.



* Use of resident library is optional depending on the task
Location and size of resident library mapping varies based on APRs designated at task build time

Figure MON-74.

Logical #1	Logical Name in RAD5φ		74φ
	Physical Device in ASCII		
	Flag	Unit No.	
Logical #2			
Logical #3			
Logical #4 or Table of PPNs	*		776
	PPN for Logical # 1		
	PPN for Logical # 2		
	PPN for Logical # 3		

Figure MON-75. User Logical Table

- * 0 Available for logical assignment
- 1 Next three words for PPNs.
- Anything else RAD5φ for logical name

EXERCISE FOR MONITOR

Using the map and octal dump provided, answer the following questions.

- 1. Is 'JOB' attached to any resident libraries? _____

If the answer is yes,

- a. How many libraries are attached? _____
- b. How many APRs are being used to map the library or libraries? _____

- 2. Have any resident libraries been added? _____

If the answer is yes,

- a. How many resident libraries have been added? _____
- b. Answer the following questions about the library which is first in the list: _____

Resident library name _____

Its project, programmer number. _____

Its size in Kwords (decimal) _____

Its protection code _____

Which resident library flags are set _____

Load address of library (octal) _____

- 3. Are large files sysgennd on the system? _____

If the answer is yes, go to question 5.

- 4. Considering the file opened on channel 3 for job number 21:

a. Is the file opened for update? _____

b. Is the file contiguous? _____

c. Does the user have write privilege? _____

d. What is the size (decimal) of the file? _____

e. What is the clustersize (decimal) of the file? _____

f. Is data caching enabled for the file? _____

g. Is the data caching sequential or random? _____

Go to question 6.

5. Considering the first file open on the first disk unit of the system:

a. What is the file's PPN? _____

b. What is the file name and extension? _____

c. Is the file placed? _____

d. Has write access been given out? _____

e. Is the file open for update? _____

f. Is the file contiguous? _____

g. Is file marked for deletion? _____

h. What is the protection code? _____

i. What is the access count for opens in normal or update mode? _____

j. What is the clustersize? _____

k. What is the octal address of the first WCB for the file? _____

6. What disk types and how many units are sysgenned on this system? _____

7. Considering DB2:

a. Is it mounted? _____

b. Is it a date of last write disk? _____

c. Are files added at the top of the directory? _____

d. What is the current open count (octal)? _____

e. What is the pack clustersize and error count? _____

8. What is the device designator of the system disk? _____

Table MON-2. RAD 50

This page is for notes.

Single Char. or First Char.	Second Character	Third Character	
A	003100	A	000001
B	005200	B	000002
C	011300	C	000003
D	014400	D	000004
E	017500	E	000005
F	022600	F	000006
G	025700	G	000007
H	031800	H	000010
I	034100	I	000011
J	037200	J	000012
K	042300	K	000013
L	045400	L	000014
M	050500	M	000015
N	053600	N	000016
O	056700	O	000017
P	062800	P	000020
Q	065100	Q	000021
R	070200	R	000022
S	073300	S	000023
T	076400	T	000024
U	101500	U	000025
V	104600	V	000026
W	107700	W	000027
X	113800	X	000030
Y	116100	Y	000031
Z	121200	Z	000032
.	124300	.	000033
unused	127400	unused	000034
#	132500	#	000035
0	135600	0	000036
1	140700	1	000037
2	144800	2	000040
3	147100	3	000041
4	152200	4	000042
5	155300	5	000043
6	160400	6	000044
7	163500	7	000045
8	166600	8	000046
9	171700	9	000047

DIGITAL

RSTS/E SWS UPDATE
Monitor Internals

RT-11 Lib	VSYS	PS	Local Mac	Hex 87-79	PS:17120
RSTS	Lib	Titles	RSTS	Icons	77,995
Section	Peer	Size	Global	Value	Global
Value	Global	Value	Global	Value	Global
ARS	000000	000000	(4,1,CBL,RES,CVP)		
ARCAT	000010	BAOCLL	000027	BACCAT	
BACANM	000002	BACPAR	000032	BACTAT	
BFD,MP	000012	BFO,LP	000020	BSEOR	
BUF,LP	000009	BUF,MP	000004	HUP,AL	
B,PRTY	000006	B,STAD	000000	A,S-90	
CLSICT	177777	CLSTMP	177775	CLSTBY	
CNT,CR	000002	CNT,CS	000002	CNT,CT	
CNT,MP	000003	CNT,PH	000017	CNT,MP	
CPUPCF	000136	CPUPED	000137	CPUPFD	
CDC,XX	176362	CCS,NT	000000	CDS,CK	
CDS,AL	000010	CDS,PH	000010	CDS,MP	
CEVCSA	000030	CFCSA	000000	CFMCC2	
CLCS*	000000	CPDS*	000000	CFMCC2	
EDCACE	000157	EPCCAE	000162	FOEKA	
ECSEOR	000150	EPCLSP	000000	EPREAR	
FIELD*	000077	FIELD*	000020	FIELD*	
FLG,LP	000006	FLG,MP	000016	FLG,AL	
FLTER*	000000	F,TER*	000002	FUCCRE	
ICX,DI	000016	ICX,CR	000020	IDX,MB	
ICX,PH	000000	ICX,SY	000000	IDX,MP	
INUSE	000003	ICLERR	000070	JOB,4*	
JS,CT	000001	JS,MP	000002	JS,LP	
JS,SY	000001	JS,MP	000002	LINE*	
LCG14*	100200	LCGLP	100050	LCG14*	
MAGSEL	000007	MAUAT	000012	MAU*2	
NECERN	000073	ACRUPS	000002	ACCEVC	
ACRACS	000100	ACRCP*	000000	ACRPS	
ACTHAT	000025	ACTMTA	000021	ACJCP*	
CNAD	000072	CPNCT	177777	CRASH*	
PANLCA	000026	PHANT	000013	PH-CCE	
PHNER2	000170	PHVCL	000012	ODTONE	
CHVCH	100000	CHVCH	100000	CHVCH	
SASVNE	000102	SATTP*	000035	NCAER*	
SIZ,MB	000000	SIZ,LP	000000	SIZ,MP	
SIZ,MP	000107	SCREAR	000006	ST-ERR	
SIS,MB	000000	SIS,LP	000002	SIS,MP	
STZ01	000000	SUBERR	000007	SYSLOG	
TLJFE	000117	TLJACD	000132	TLJAN*	
TLACT*	000121	TLNZSP	000115	TLCPAV	
TLT*P	000131	TLPUT*	000105	TLACIM	
TICJ11	000012	TICL11	000000	TIC*PP	
TPL11	000000	TIMSCA	000000	TIPAR*	
VACFA1	000112	VACERR	000053	VCCERR	
VCCCN*	000000	VDFCA*	000000	SOLOIA	
Y127R*	000000				

Figure MON-76. RSTS.MAP

MON-110

RSTS/E SWS UPDATE
Monitor Internals

Value	Global	Value	Global	Value	Global
000037	PAPCIR	000001	PACERR	000171	MAPUD
000003	PCAE04	000000	EFL,MP	000006	BAI,MB
000056	ELF,CT	000076	ELF,CT	001000	BUF,MS
000007	ELF,PH	000002	ELF,SY	001000	BUF,MP
000005	P,LP	000002	P,25*	000003	A,0
000001	CNT,CR	000003	CNT,CR	000002	CNT,CR
000002	CNT,CR	000002	CNT,CR	000130	CNT,LP
000006	CCSEOR	000113	CPASOF	000133	CPUFAX
000130	CTOLCE	000030	CATEOR	000015	DCOSK
000030	CDS,ME	000006	CDS,LP	000000	UCS,MP
000001	CEINRY	000033	CEVAFS	000036	DEVCMS
000051	DTERR*	000170	CISERR	000172	DIVRYJ
176502	ESTMUL	000000	CICCE*	000021	EDASV
000000	EPCCAT	000006	EPCPAT	000000	EOF
000002	EXITAR	000111	EXITTY	000110	EXPER*
000003	FLG,CT	000000	FLG,DR	100022	FLG,MB
000020	FLG,PH	000020	FLG,SY	100000	FLG,MP
000107	FLAERR	000130	FACCEV	000010	HACTTY
000002	ICX,LP	000012	ICX,MP	000023	IDX,AL
000010	IE	000100	INTLCH	000000	INTPR
000077	JCR*2	000170	JSALL	000036	JS,DT
000022	JS,MP	000001	JS,AL	000000	JS,PH
000057	LCGER*	000005	LCG10*	100050	LCG10*
100052	LCG14*	100000	LRGFI*	000000	MAGRE*
000020	MAVER	000072	MCCLL*	000052	MOERR*
000000	ACGCTC	000153	ACRAT*	000175	NOCKM
000070	ACSUC*	000005	ACTAV*	000017	NOTCLS
000011	ARRERR	000101	AHMTS	000051	ODC
177775	CPASV	000001	CPCH*	000026	OUTCAS
000012	PHERAS	000100	PHVAT	000030	PHNER1
100000	OCIP*	100020	OCIPES	100100	ODHCC*
100002	RESEAR	000150	SIFACS	000103	SASCI*
000177	SIZCR*	000050	SIZ,DT	000107	SIZ,DR
000107	SIZ,AL	000107	SIZ,MP	000107	SIZ,SY
000107	STERR*	000173	SIS,DT	000000	SIS,CR
000000	SIS,AL	000000	SIS,MP	000000	SIS,MP
000001	TLCNT*	000155	TLCCN*	000100	TLICM*
000100	TLFAC*	000123	TLNCA*	000101	TLNCT*
000110	TLPH*	000150	TLGDU*	000122	TLNAN*
000106	TICCI*	000022	TICCI*	000000	TID*H
000000	TICJ11	000010	TIFCAT	000003	TIINCC
000052	TIPR11	000010	TIPR11	000050	UDERR*
000055	VCSEOR	000052	WCPAN*	000024	WCCOR*
000000	WCPAN*	000000	WCPAN*	000022	WCPAN*

Figure MON-76. RSTS.MAP (Cont.)

DIGITAL

RSTS/E SWS UPDATE
Monitor Internals

SLPTPL 75020 000022 (R-,I,GCL,REL,CVR)
 SLPTM 75020 000022 (R-,I,GCL,REL,CVR)
 SLGTH 75040 (R-,I,GCL,REL,CVR)
 SSIZT0 75070 000022 (R-,I,GCL,REL,CVR)
 RLPTPL 75112 000022 (R-,I,GCL,REL,CVR)
 SPLFTA 75112 (R-,I,GCL,REL,CVR)
 PSAT9 75130 000000 (R-,I,GCL,REL,OV9)
 SFIP10 75170 000050 (R-,I,GCL,REL,OV9)
 SFIP10 75170 (R-,I,GCL,REL,OV9)
 UUCTA 75200 000150 (R-,I,GCL,REL,OV9)
 UUCTA 75200 (R-,I,GCL,REL,OV9)
 UUCSAC 75200 000060 (R-,I,GCL,REL,OV9)
 4LUCN4 75220 (R-,I,GCL,REL,CVR)
 STSTAL 75512 (R-,I,GCL,REL,CVR)
 STSTAL 75512 (R-,I,GCL,REL,OV9)
 TERCSP 75522 000010 (R-,I,GCL,REL,OV9)
 CLATER 75520 CLATIA 075530 CHJCB
 STERCS 75522
 CSKCS 75530 000002 (R-,I,GCL,REL,CVR)
 GCCISX 75530 SCSACS 075530
 EMTCS 75540 000002 (R-,I,GCL,REL,OV9)
 CCPCST 75500 SEHTCS 075500
 PIPCS 75502 000000 (R-,I,GCL,REL,OV9)
 PIPERR 75500 PIPGC 075502 CVRPLP
 RSKAGN 75552 020010 (R-,I,GCL,REL,CVR)
 RSKAGN 75552 RSKENT 075550 RSKTAP
 DTACS 75570 000010 (R-,I,GCL,REL,OV9)
 SPTAOS 75560 (R-,I,GCL,REL,OV9)
 DTACS 75570 SSCRSZ 075610
 CSCIA 75610 000020 (R-,I,GCL,REL,CVR)
 CSCIA 75610 SCSOIA 075610
 DDCIA 75630 000020 (R-,I,GCL,REL,OV9)
 DDCIA 75630 SDCIA 075630
 DDCIA 75650 000020 (R-,I,GCL,REL,CVR)
 DDCIA 75650 SDCIA 075650
 DDCIA 75670 000020 (R-,I,GCL,REL,OV9)
 DDCIA 75670 SDCIA 075670
 DECIA 75710 000020 (R-,I,GCL,REL,OV9)
 DECIA 75710 SDCIA 075710
 STOIA 75730 000002 (R-,I,GCL,REL,OV9)
 FILGO 75732 FILIX 075742 FILSET
 ..DONT 75610
 HBCIA 75630 000010 (R-,I,GCL,REL,OV9)
 CHXPA 75730 CHXJAT 076650 CZXPA
 HBCAPS 75602 HBCFCM 076750 HXJAT
 SYCCTP 07670
 NLCIA 07700 000000 (R-,I,GCL,REL,OV9)
 NLCIA 07700 000000 (R-,I,GCL,REL,OV9)
 NLCIA 07700 (R-,I,GCL,REL,OV9)

Figure MON-76. RSTS.MAP (Cont.)

DIGITAL

RSTS/E SWS UPDATE
Monitor Internals

075532 EADPEY 075522 SETCTP 075530 SETPR 075520
 075506 SFIPCS 075502
 075550 SCSACS 075550
 076070 FLOAE 075730 SYCAPS 075700 SYCINT 075730
 076700 CZXJAT 076670 EATCOT 076700 EPLSND 076730
 076630 PTABLE 076700 JCFFCM 077000 SXBDIN 076630

Figure MON-76. RSTS.MAP (Cont.)

DIGITAL

RSTS MAP UPDATE
Monitor Intervals

RSTS MAP UPDATE
Monitor Intervals

```

LPCINT 077100 000100 (R=,I,GCL,REL,CVR)
AS4310 077100 CLS900 077100 CE4310
PCPCLE 077110 PPFCC2 077132 SE-300
WDSJUP 077110 WSDAPS 077210 W-CPAT
LPCINT 077100 000100 (R=,I,GCL,REL,CVR)
LPCDPS 077172 LPCDIA 077160
WDCIAT 077170 000100 (R=,I,GCL,REL,CVR)
WDCAPS 077202 WDCIAT 077170
DTCIAT 077212 000100 (R=,I,GCL,REL,CVR)
DTCAPS 077216 DTCIAT 077212
DCCIAT 077232 000100 (R=,I,GCL,REL,CVR)
DCCAPS 077236 DCCIAT 077232
ASPSL0 077202 000100 (R=,I,GCL,REL,CVR)
ASPCAS 077200 ASPSEN 077202 SASPSL
ALLJCP 177200 000100 (R=,I,GCL,REL,CVR)
ALLJCB 077200
ASPC00 077132 000100 (R=,I,GCL,REL,CVR)
ASPC00 077132
ASPT0C 077170 000100 (R=,I,GCL,REL,CVR)
ASPT0C 077170
RJET00 100010 000100 (R=,I,GCL,REL,CVR)
RJET00 100010
PRC00 100020 000100 (R=,I,GCL,REL,CVR)
PRC00 100020
RDC00 100020 000100 (R=,I,GCL,REL,CVR)
RDC00 100020
REC 100020 000100 (R=,I,GCL,REL,CVR)
REC 100020
LPE 100020 000100 (R=,I,GCL,REL,CVR)
LPE 100020
FPP 100020 000100 (R=,I,GCL,REL,CVR)
FPP 100020
RTI 100020 000100 (R=,I,GCL,REL,CVR)
CLNGCC 120250 CLCN.L
FONCA 121110 FSTPCF 122700 FSTPCL
FLX00 120110 FLX001 120122 FLX00
FLX00 120100 FLX007 120152 FLX00
ICEX01 121100 ICREFC 121102 LUGPCL
QES000 100500 RETPIG 121602 RETPCA
QSE001 101250 RTI 120020 RTI
QV100 100302 QVEX01 120004 QVEX00
SIC000 100200 SICCAT 120322 SICSEF
SIF000 100200 SIF000 120202 SIFLAY
SIC00E 100230 SICCT0 120220 SIFSCA
SIS000 101020 SIS000 120310
CHES0L 105100 000100 (R=,I,GCL,REL,CVR)
CHES0L 105100
CHES0A 105110 000100 (R=,I,GCL,REL,CVR)
CHES0A 105110
CHES0T 105110 000210 (R=,I,GCL,REL,CVR)
CHES0T 105110
CHES0C 105132 000100 (R=,I,GCL,REL,CVR)
CHES0C 105132

```

Figure MON-76. RSTS.MAP (Cont.)

```

000100 000100 077100 100500 077132 CE4310 077000
000100 000100 077102 SPC000 077250 100000 077050
000100 000100 077170 000010 077010

```

077202

```

120312 CLCN.P 120200 C-PPJC 101210 E-TAPS 121030
102730 FREF20 122000 FALF20 122250 FTLSCN 103600
100120 ATLX03 120132 FTLX00 100130 FTLX05 120102
100112 G17010 101722 INTS00 100500 INTS01 100070
122337 CANCL 102000 PMS000 103602 PEE000 105052
103100 RETUS0 101054 RETLS0 101000 RSHAPS 101630
120570 RTI07 100650 SET005 102070 STUP 103000
122732 LCC004 105700 SICCAT 100330 SICCS0 120202
120306 SIF000 102200 SIF000 120230 SIF15 100270
100200 SIF000 102032 SIFLAY 100652 SIFDIE 120710
103000 SIF000 103602 SIF000 103372 SIF000 100010

```

Figure MON-76. RSTS.MAP (Cont.)

DIGITAL

RSTS MAP MONITOR
Monitor Internals

RSTS MAP MONITOR
Monitor Internals

```

SYDLRG 11776P ##### (R=,I,GCL,REL,OVN)
UPDCM= 11776P ###330 (R=,I,GCL,REL,OVN)
LPCVAL 120210 ##### (R=,I,GCL,REL,OVN)
ALCVR 121020 ##### (R=,I,GCL,REL,OVN)
LPCVR= 121052 ##### (R=,I,GCL,REL,OVN)
LPCVR 121052 ##### (R=,I,GCL,REL,OVN)
OTCVR 123500 ##### (R=,I,GCL,REL,OVN)
MPCVR 125230 ##### (R=,I,GCL,REL,OVN)
XPCVR 130050 ##### (R=,I,GCL,REL,OVN)
OPCVR 130050 ##### (R=,I,GCL,REL,OVN)
BLFEXT 133670 ##### (R=,I,GCL,REL,OVN)
RUF 133670 ##### (R=,I,GCL,REL,OVN)
LIG-TS 137370 ##### (R=,I,GCL,REL,OVN)
UPCATE 137370 ##### (R=,I,GCL,REL,OVN)
UPDLG 137370 ##### (R=,I,GCL,REL,OVN)
IAB 137370 ##### (R=,I,GCL,REL,OVN)

```

```

1
120000 INTVAL 121020 ALCVR 121020 OPASAL 121020
121020
121000 INTSLP 123232 LPCVR 121052 OPASLP 121250
122100 ..LP60 121350
120002 OTCVR 123500 ERLSDT 120170 FIPADT 125102
120200 SPCACT 123500 TPCYDT 120150
125200 CAESPP 120100 ERLSPP 125000 INTSPH 125300
125700 TPCSP 125270 LPRSP 120100 ..MCP 120502
131000 CEASGX 131032 OTCVR 130050 ERLSGX 131110
131200 SPCGX 130000 TCGDX 131030 UMSGX 131002
130552 ..CHFT 130500
137370 STCIP 137020

```

Figure MON-76. RSTS.MAP (Cont.)

Figure MON-76. RSTS.MAP (Cont.)

DIGITAL

RSTS/E SMS UPDATE
Monitor Internals

1 1977

RSTS/E SMS UPDATE
Monitor Internals

```

007460/ 001100 007640 002000 000000 000412 007033 000000 003900
007500/ 011240 140640 000412 074514 140123 111233 036004 000001
007520/ 000330 070335 000320 070335 000012 000007 000010 007540
007540/ 100000 040412 000000 000001 007534 000000 000000 177777
007560/ 177777 000000 000000 000000 000000 000000 000000 000000
007600/ 000000 024014 004377 000412 051460 000000 000002 000000
007620/ 000000 000000 000000 000000 000000 000000 000000 000000
007640/ 010140 040502 051460 046120 000014 000003 000000 000005
007660/ 000000 007660 000000 000000 007666 000000 000000 000000
007700/ 000004 000047 000000 000724 000000 000412 007033 000000 000000
007720/ 001100 010140 000000 000000 000000 000000 000000 000000
007740/ 052720 007400 000000 000000 000000 000000 000000 000000
007760/ 000000 000000 000000 000000 000000 000000 000000 000000
010000/ 007740 003002 162400 007600 007440 006400 001100 000000
010020/ 006462 011510 000015 000015 016440 000000 001770 000037
010040/ 020140 000000 000000 000000 000000 064346 000000 000000
010060/ 171710 000035 000000 000000 013607 140000 007057 001035
010100/ 000000 024016 004377 000412 051461 000000 000002 000000
010120/ 000000 000000 000000 000000 000000 000000 000000 000000
010140/ 010540 040502 051461 046120 000016 000003 000000 000005
010160/ 000000 010160 000000 000000 010166 000000 000000 000001
010200/ 006640 170240 000412 006274 013625 111233 036004 000001
010220/ 000300 102421 000120 070340 000012 000011 000010 010500
010240/ 052720 010500 000000 000000 000000 000000 000000 000000
010260/ 000000 000000 000000 000000 000000 000000 000000 000000
010300/ 013440 013300 000377 000000 000000 000004 000007 000004
010320/ 000036 021533 000114 011017 000000 000000 000000 000000
010340/ 007500 000020 000412 012001 071570 066615 036000 000001
010360/ 000160 070335 000010 070321 000012 000002 000010 015200
010400/ 000000 053516 045520 030061 174424 000004 000000 000000
010420/ 000000 010420 000001 000000 010426 000000 000100 000000
010440/ 010240 003002 162400 010100 007700 006400 001100 000000
010460/ 007262 011060 000015 000015 020500 000000 001770 000037
010500/ 100000 040416 000000 000001 010234 000000 000000 177777
010520/ 177777 001515 020245 000000 000000 000000 000000 000000
010540/ 013000 040502 051462 046120 000003 000000 000000 000000
010560/ 000000 010560 000000 000000 010566 000000 000000 000000
010600/ 010200 031220 000412 006274 013626 111233 036004 000001
010620/ 000120 102424 000110 102424 000012 000011 000010 010700
010640/ 000000 000000 000000 000000 000000 000000 006274 034354
010660/ 001100 010540 000000 000000 000412 007033 000000 000000
010700/ 100000 040420 000000 000001 010634 000000 000000 177777
010720/ 177777 015136 015237 000000 000000 000000 000000 000000
010740/ 000000 024020 004377 000412 051462 000000 017402 000000
010760/ 000000 000000 000000 000000 000000 000000 000000 000000
011000/ 052720 012700 000000 000000 000000 000000 000000 000000
011020/ 000000 000000 000000 000000 000000 000000 000000 000000
011040/ 011000 003002 162400 010740 010640 006400 001100 000000
011060/ 010462 011360 000015 000015 021340 000000 001770 000037
011080/ 000000 000000 000000 000000 000000 000000 000000 000000
011100/ 000000 013000 020000 001000 000412 007033 000000 000000

```

Figure MON-77. RSTS Dump (Cont.)

MON-110

COMPANY CONFIDENTIAL

```

007460/ 001100 007640 002000 000000 000412 007033 000000 003900
007500/ 011240 140640 000412 074514 140123 111233 036004 000001
007520/ 000330 070335 000320 070335 000012 000007 000010 007540
007540/ 100000 040412 000000 000001 007534 000000 000000 177777
007560/ 177777 000000 000000 000000 000000 000000 000000 000000
007600/ 000000 024014 004377 000412 051460 000000 000002 000000
007620/ 000000 000000 000000 000000 000000 000000 000000 000000
007640/ 010140 040502 051460 046120 000014 000003 000000 000005
007660/ 000000 007660 000000 000000 007666 000000 000000 000000
007700/ 000004 000047 000000 000724 000000 000412 007033 000000 000000
007720/ 001100 010140 000000 000000 000000 000000 000000 000000
007740/ 052720 007400 000000 000000 000000 000000 000000 000000
007760/ 000000 000000 000000 000000 000000 000000 000000 000000
010000/ 007740 003002 162400 007600 007440 006400 001100 000000
010020/ 006462 011510 000015 000015 016440 000000 001770 000037
010040/ 020140 000000 000000 000000 000000 064346 000000 000000
010060/ 171710 000035 000000 000000 013607 140000 007057 001035
010100/ 000000 024016 004377 000412 051461 000000 000002 000000
010120/ 000000 000000 000000 000000 000000 000000 000000 000000
010140/ 010540 040502 051461 046120 000016 000003 000000 000005
010160/ 000000 010160 000000 000000 010166 000000 000000 000001
010200/ 006640 170240 000412 006274 013625 111233 036004 000001
010220/ 000300 102421 000120 070340 000012 000011 000010 010500
010240/ 052720 010500 000000 000000 000000 000000 000000 000000
010260/ 000000 000000 000000 000000 000000 000000 000000 000000
010300/ 013440 013300 000377 000000 000000 000004 000007 000004
010320/ 000036 021533 000114 011017 000000 000000 000000 000000
010340/ 007500 000020 000412 012001 071570 066615 036000 000001
010360/ 000160 070335 000010 070321 000012 000002 000010 015200
010400/ 000000 053516 045520 030061 174424 000004 000000 000000
010420/ 000000 010420 000001 000000 010426 000000 000100 000000
010440/ 010240 003002 162400 010100 007700 006400 001100 000000
010460/ 007262 011060 000015 000015 020500 000000 001770 000037
010500/ 100000 040416 000000 000001 010234 000000 000000 177777
010520/ 177777 001515 020245 000000 000000 000000 000000 000000
010540/ 013000 040502 051462 046120 000003 000000 000000 000000
010560/ 000000 010560 000000 000000 010566 000000 000000 000000
010600/ 010200 031220 000412 006274 013626 111233 036004 000001
010620/ 000120 102424 000110 102424 000012 000011 000010 010700
010640/ 000000 000000 000000 000000 000000 000000 006274 034354
010660/ 001100 010540 000000 000000 000412 007033 000000 000000
010700/ 100000 040420 000000 000001 010634 000000 000000 177777
010720/ 177777 015136 015237 000000 000000 000000 000000 000000
010740/ 000000 024020 004377 000412 051462 000000 017402 000000
010760/ 000000 000000 000000 000000 000000 000000 000000 000000
011000/ 052720 012700 000000 000000 000000 000000 000000 000000
011020/ 000000 000000 000000 000000 000000 000000 000000 000000
011040/ 011000 003002 162400 010740 010640 006400 001100 000000
011060/ 010462 011360 000015 000015 021340 000000 001770 000037
011080/ 000000 000000 000000 000000 000000 000000 000000 000000
011100/ 000000 013000 020000 001000 000412 007033 000000 000000

```

Figure MON-77. RSTS Dump (Cont.)

MON-110

COMPANY CONFIDENTIAL

DIGITAL

RSTS/E DMS UPDATE
Monitor Internals

016020/	177520	000000	000000	000000	020000	000000	016021	000020
016030/	016320	016056	016061	000402	102527	102604	177777	047515
016250/	247125	177524	000000	000000	000000	002000	016061	172460
016120/	216240	016116	016121	000404	047014	046537	006253	051514
015120/	045124	043517	000377	000000	000000	000000	016121	180000
015140/	020500	016156	016161	000402	007403	070516	177777	050102
016160/	051103	043105	000377	000000	000000	000000	016161	072460
016220/	015600	016216	016220	000404	020604	021020	006253	047105
016220/	042524	177522	000000	000000	000000	000000	016220	072460
016240/	016040	016256	016261	000402	076713	056700	076713	040515
016260/	042513	000377	000000	000000	000000	000000	016261	000000
016302/	016600	016316	016320	000402	076713	056700	076713	052515
016320/	042516	000377	000000	000000	000000	000000	016320	000000
016340/	016540	016356	016361	000402	046152	070200	073376	044514
016360/	051102	000377	000000	000000	000000	000000	016361	020020
016400/	016700	016416	016420	000402	062745	004475	177777	046120
016420/	040505	042523	000377	000000	000000	000000	016420	172460
016440/	016500	016456	016461	000402	046537	035160	177777	042510
016460/	046114	177517	000000	000000	000000	000000	016461	100000
016500/	016340	016516	016520	000402	042614	157700	073376	042513
016520/	032504	177462	000000	000000	000000	000000	016520	020020
016540/	017000	016556	016560	000402	100071	073634	177777	042523
016560/	177524	000000	000000	000000	000000	000000	016560	172460
016600/	016400	016616	016621	000402	062074	000000	100003	040520
016620/	177524	000000	000000	000000	000000	000000	016621	000000
016640/	016100	016656	016661	000402	046537	035160	177777	047514
016660/	044507	177516	000000	000000	000000	000000	016661	100000
016700/	016540	016716	016720	000402	066615	000000	177777	052521
016720/	052505	177505	000000	000000	000000	000000	016720	172460
016740/	017140	016756	016761	000402	066615	000000	177777	052523
016760/	046502	052111	000377	000000	000000	000000	016761	172460
017000/	017100	017016	017021	000402	023374	000000	177777	044506
017020/	177524	000000	000000	000000	000000	000000	017021	172460
017040/	017200	017056	017060	000402	075273	076474	177777	054623
017060/	052123	052101	000377	000000	000000	000000	017060	172460
017100/	016740	017116	017121	000402	074064	103070	073376	044523
017120/	052514	177523	000000	000000	000000	000000	017121	020000
017140/	017040	017156	017160	000402	075141	076600	177777	053523
017160/	052111	044103	000377	000000	000000	000000	017160	172460
017200/	017300	017216	017220	000404	076521	076400	006253	041124
017220/	052101	000377	000000	000000	000000	000000	017220	172460
017240/	017500	017256	017261	024050	054137	015464	100003	047104
017260/	177504	000000	000000	000000	051104	177400	017261	000000
017300/	017340	017316	017320	000402	076713	056700	076713	042524
017320/	047503	000377	000000	000000	000000	000000	017320	000000
017340/	017440	017356	017360	000402	100370	017500	076713	054524
017360/	042520	000377	000000	000000	000000	000000	017360	000010
017400/	000000	053005	032524	006462	033012	006460	047012	006517
017420/	054412	051505	005015	046502	041454	055123	045633	054633
017440/	017240	017456	017460	000402	103151	047071	177777	052125
017460/	046111	054524	000377	000000	000000	000000	017460	072460

Figure MON-77. RSTS Dump (Cont.)

DIGITAL

RSTS/E DMS UPDATE
Monitor Internals

017480/	000000	017496	017500	000000	000000	000000	017480	000000
017500/	000000	017516	017520	000000	000000	000000	017500	000000
017520/	000000	017536	017540	000000	000000	000000	017520	000000
017540/	000000	017556	017560	000000	000000	000000	017540	000000
017560/	000000	017576	017580	000000	000000	000000	017560	000000
017580/	000000	017596	017600	000000	000000	000000	017580	000000
017600/	000000	017616	017620	000000	000000	000000	017600	000000
017620/	000000	017636	017640	000000	000000	000000	017620	000000
017640/	000000	017656	017660	000000	000000	000000	017640	000000
017660/	000000	017676	017680	000000	000000	000000	017660	000000
017680/	000000	017696	017700	000000	000000	000000	017680	000000
017700/	000000	017716	017720	000000	000000	000000	017700	000000
017720/	000000	017736	017740	000000	000000	000000	017720	000000
017740/	000000	017756	017760	000000	000000	000000	017740	000000
017760/	000000	017776	017780	000000	000000	000000	017760	000000
017780/	000000	017796	017800	000000	000000	000000	017780	000000
017800/	000000	017816	017820	000000	000000	000000	017800	000000
017820/	000000	017836	017840	000000	000000	000000	017820	000000
017840/	000000	017856	017860	000000	000000	000000	017840	000000
017860/	000000	017876	017880	000000	000000	000000	017860	000000
017880/	000000	017896	017900	000000	000000	000000	017880	000000
017900/	000000	017916	017920	000000	000000	000000	017900	000000
017920/	000000	017936	017940	000000	000000	000000	017920	000000
017940/	000000	017956	017960	000000	000000	000000	017940	000000
017960/	000000	017976	017980	000000	000000	000000	017960	000000
017980/	000000	017996	018000	000000	000000	000000	017980	000000
018000/	000000	018016	018020	000000	000000	000000	018000	000000
018020/	000000	018036	018040	000000	000000	000000	018020	000000
018040/	000000	018056	018060	000000	000000	000000	018040	000000
018060/	000000	018076	018080	000000	000000	000000	018060	000000
018080/	000000	018096	018100	000000	000000	000000	018080	000000
018100/	000000	018116	018120	000000	000000	000000	018100	000000
018120/	000000	018136	018140	000000	000000	000000	018120	000000
018140/	000000	018156	018160	000000	000000	000000	018140	000000
018160/	000000	018176	018180	000000	000000	000000	018160	000000
018180/	000000	018196	018200	000000	000000	000000	018180	000000
018200/	000000	018216	018220	000000	000000	000000	018200	000000
018220/	000000	018236	018240	000000	000000	000000	018220	000000
018240/	000000	018256	018260	000000	000000	000000	018240	000000
018260/	000000	018276	018280	000000	000000	000000	018260	000000
018280/	000000	018296	018300	000000	000000	000000	018280	000000
018300/	000000	018316	018320	000000	000000	000000	018300	000000
018320/	000000	018336	018340	000000	000000	000000	018320	000000
018340/	000000	018356	018360	000000	000000	000000	018340	000000
018360/	000000	018376	018380	000000	000000	000000	018360	000000
018380/	000000	018396	018400	000000	000000	000000	018380	000000
018400/	000000	018416	018420	000000	000000	000000	018400	000000
018420/	000000	018436	018440	000000	000000	000000	018420	000000
018440/	000000	018456	018460	000000	000000	000000	018440	000000
018460/	000000	018476	018480	000000	000000	000000	018460	000000
018480/	000000	018496	018500	000000	000000	000000	018480	000000
018500/	000000	018516	018520	000000	000000	000000	018500	000000
018520/	000000	018536	018540	000000	000000	000000	018520	000000
018540/	000000	018556	018560	000000	000000	000000	018540	000000
018560/	000000	018576	018580	000000	000000	000000	018560	000000
018580/	000000	018596	018600	000000	000000	000000	018580	000000
018600/	000000	018616	018620	000000	000000	000000	018600	000000
018620/	000000	018636	018640	000000	000000	000000	018620	000000
018640/	000000	018656	018660	000000	000000	000000	018640	000000
018660/	000000	018676	018680	000000	000000	000000	018660	000000
018680/	000000	018696	018700	000000	000000	000000	018680	000000
018700/	000000	018716	018720	000000	000000	000000	018700	000000
018720/	000000	018736	018740	000000	000000	000000	018720	000000
018740/	000000	018756	018760	0				

DIGITAL

RSTS/E SMS UPDATE
Monitor Internals

024320/	023046	023046	023046	023046	025453	025453	025453	025453
024340/	024440	030200	022445	022445	022445	022445	022445	037076
024350/	037076	037076	037076	037076	056535	006535	177414	005015
024420/	020000	000000	037402	075510	000000	023204	000000	000000
024420/	004115	001100	141100	140714	140714	000000	000000	000000
024440/	000000	020000	075734	000406	026703	006571	025146	177400
024460/	000404	015100	000000	002006	000742	002000	127600	000000
024500/	000000	017740	000023	000014	000426	000125	100001	000001
024520/	000000	000000	000000	077206	016156	010004	000021	000361
024540/	001002	013660	001363	000000	014754	014600	001401	172440
024560/	002017	144660	000000	072162	000000	000100	150640	100000
024500/	000000	000000	001100	142011	062500	102301	000015	000000
024620/	037076	037076	056535	006535	177414	045415	045513	045767
024640/	000000	017600	075734	000012	012121	002571	130110	177400
024650/	000404	022500	000000	004020	000014	002004	127600	001115
024720/	000000	017540	046515	046515	176515	041503	173503	041503
024720/	005103	045513	173513	045513	176513	046515	046515	046515
024740/	000000	000000	046515	046515	176515	041503	173503	041503
024760/	005103	045513	175113	045513	175113	046515	176515	046515
025000/	000000	017440	000412	074751	006571	062156	174000	000000
025020/	000000	020540	000000	004022	000347	002004	127600	001113
025040/	000000	017400	000006	107521	006171	063312	174000	000000
025060/	000004	025000	000000	000424	000345	002000	127600	000000
025100/	000000	000000	000032	000010	000000	064346	000000	000000
025120/	000000	041035	000000	000000	073407	140000	007356	140014
025140/	000000	017300	075734	001412	120201	006571	062156	176400
025160/	000404	024600	000000	006020	001070	002004	127600	001060
025200/	000000	017240	022044	022044	022044	020441	020441	020441
025220/	020441	020441	020441	021443	021443	021443	021443	021443
025240/	025500	000200	030372	030060	176412	030461	030461	030461
025260/	030461	140451	030461	030461	030461	173461	030461	030860
025300/	026400	000000	030060	030060	030060	176460	176412	030461
025320/	030461	030461	030461	140461	030461	030461	030461	030461
025340/	026400	000000	030060	030060	030060	176460	005012	022044
025360/	055412	055533	036074	036074	036074	036074	036074	022044
025400/	030400	020441	020441	023046	023046	023046	022445	022445
025420/	025453	025453	025453	022445	022445	022445	022445	022445
025440/	026400	000000	037076	037076	037076	037076	056476	036074
025460/	056535	055412	055533	036074	036074	036074	036074	036074
025500/	026400	000100	037477	037477	037477	036475	036475	030060
025520/	036475	036475	036475	030060	030060	030060	030060	030060
025540/	026400	000000	025052	025052	025052	023046	023046	023046
025560/	023046	023046	023046	025453	025453	025453	025453	025453
025580/	026400	000000	022445	022445	022445	037076	037076	037076
025620/	037076	037076	037076	056535	005115	005012	051775	051523
025640/	026400	024142	051122	051122	051122	051122	175122	052124
025660/	052124	052124	052124	052124	052124	176524	052525	052525
025700/	026400	000000	052525	176525	050120	050120	050120	050120
025720/	050120	050120	050375	051523	173523	051523	051523	052124
025740/	026400	000000	050120	173520	050120	005120	051523	173523
025760/	051523	173523	052124	173524	051122	173522	051122	173522

Figure MON-77. RSTS Dump (Cont.)

DIGITAL

RSTS/E SMS UPDATE
Monitor Internals

050000/	024000	024000	173524	052525	173525	052525	176525	050120
050000/	024000	024000	051523	165523	052124	173524	051523	051523
050000/	024000	024000	173524	051122	173522	051122	173522	051124
050000/	024000	024000	052525	173525	052525	176525	050120	173520
050000/	024000	024000	051523	165523	052124	173524	051122	173522
050000/	024000	024000	052525	173525	052525	173525	052525	176525
050000/	024000	024000	051122	173522	052124	173524	052525	051523
050000/	024000	024000	052525	173525	052525	173525	051367	051122
050000/	024000	024000	051523	165523	052124	173524	051122	051122
050000/	024000	024000	052525	173525	052525	173525	052525	173525
050000/	024000	024000	050120	050120	050120	050120	050120	050120
050000/	024000	024000	051775	051523	051523	051523	051523	052364
050000/	024000	024000	051122	173522	051122	051122	051122	051122
050000/	024000	024000	050120	050120	050120	050120	050120	050120
050000/	024000	024000	051775	051523	051523	051523	051523	051523
050000/	024000	024000	051122	173522	051122	051122	051122	051122
050000/	024000	024000	050120	050120	050120	050120	050120	050120
050000/	024000	024000	051775	051523	051523	051523	051523	051523
050000/	024000	024000	051122	173522	051122	051122	051122	051122
050000/	024000	024000	050120	050120	050120	050120	050120	050120
050000/	024000	024000	051775	051523	051523	051523	051523	051523
050000/	024000	024000	051122	173522	051122	051122	051122	051122
050000/	024000	024000	050120	050120	050120	050120	050120	050120
050000/	024000	024000	051775	051523	051523	051523	051523	051523
050000/	024000	024000	051122	173522	051122	051122	051122	051122
050000/	024000	024000	050120	050120	050120	050120	050120	050120
050000/	024000	024000	051775	051523	051523	051523	051523	051523
050000/	024000	024000	051122	173522	051122	051122	051122	051122
050000/	024000	024000	050120	050120	050120	050120	050120	050120
050000/	024000	024000	051775	051523	051523	051523	051523	051523
050000/	024000	024000	051122	173522	051122	051122	051122	051122
050000/	024000	024000	050120	050120	050120	050120	050120	050120
050000/	024000	024000	051775	051523	051523	051523	051523	051523
050000/	024000	024000	051122	173522	051122	051122	051122	051122
050000/	024000	024000	050120	050120	050120	050120	050120	050120
050000/	024000	024000	051775	051523	051523	051523	051523	051523
050000/	024000	024000	051122	173522	051122	051122	051122	051122
050000/	024000	024000	050120	050120	050120	050120	050120	050120
050000/	024000	024000	051775	051523	051523	051523	051523	051523
050000/	024000	024000	051122	173522	051122	051122	051122	051122
050000/	024000	024000	050120	050120	050120	050120	050120	050120
050000/	024000	024000	051775	051523	051523	051523	051523	051523
050000/	024000	024000	051122	173522	051122	051122	051122	051122
050000/	024000	024000	050120	050120	050120	050120	050120	050120
050000/	024000	024000	051775	051523	051523	051523	051523	051523
050000/	024000	024000	051122	173522	051122	051122	051122	051122
050000/	024000	024000	050120	050120	050120	050120	050120	050120
050000/	024000	024000	051775	051523	051523	051523	051523	051523
050000/	024000	024000	051122	173522	051122	051122	051122	051122
050000/	024000	024000	050120	050120	050120	050120	050120	050120
050000/	024000	024000	051775	051523	051523	051523	051523	051523
050000/	024000	024000	051122	173522	051122	051122	051122	051122
050000/	024000	024000	050120	050120	050120	050120	050120	050120
050000/	024000	024000	051775	051523	051523	051523	051523	051523
050000/	024000	024000	051122	173522	051122	051122	051122	051122
050000/	024000	024000	050120	050120	050120	050120	050120	050120
050000/	024000	024000	051775	051523	051523	051523	051523	051523
050000/	024000	024000	051122	173522	051122	051122	051122	051122
050000/	024000	024000	050120	050120	050120	050120	050120	050120
050000/	024000	024000	051775	051523	051523	051523	051523	051523
050000/	024000	024000	051122	173522	051122	051122	051122	051122
050000/	024000	024000	050120	050120	050120	050120	050120	050120
050000/	024000	024000	051775	051523	051523	051523	051523	051523
050000/	024000	024000	051122	173				

DIGITAL

RSTS/E SWS UPDATE
Monitor Internals

```

044720/ 002412 007403 002412 007406 002412 007407 002412 007410
044740/ 002412 007411 002412 007412 002412 007413 002412 007414
044760/ 002412 007415 002412 007416 002412 007417 002412 007420
045000/ 002412 007421 002412 007422 002412 007423 002412 007424
045220/ 002412 007425 002412 007426 002412 007427 002412 007430
045240/ 002412 007431 002412 007432 002412 007433 002412 007434
045060/ 002412 007435 002412 007436 002412 007437 002412 007440
045120/ 002412 007441 002412 007442 002412 007443 002412 007444
045120/ 002412 007445 002412 007446 002412 007447 002412 007450
045140/ 002412 007451 010346 010246 005400 062700 007777 072027
045160/ 177764 042700 177760 016702 000160 022702 045432 103435
045280/ 010003 005722 001372 077303 010003 010442 077302 062702
045220/ 026542 006302 012622 010246 000405 016212 177774 062722
045240/ 020000 005501 010122 077007 012602 162702 170202 072227
045260/ 000003 000302 110201 105002 000404 012767 000020 000050
045320/ 012622 012603 000207 010046 016700 000036 020420 001002
045320/ 005069 177776 020027 045434 103771 012600 056767 000010
045340/ 020356 005067 000002 000207 000000 045374 177777 177777
045360/ 177777 177777 177777 177777 177777 177777 000000 000000
045400/ 020000 000000 000000 000000 000000 000000 000000 000000
045420/ 000000 000000 000000 000000 000000 000000 177777 012705
045440/ 073522 012701 073470 015137 172352 020127 073446 101407
045460/ 014502 000171 010146 004767 000042 012601 000764 016737
045500/ 040442 172352 014501 012100 016503 000032 011102 004767
045520/ 000012 020527 073470 101366 000137 104570 010546 012737
045540/ 034240 177776 000472 012737 034140 177776 012605 000207
045560/ 032711 001000 001433 022027 000172 103002 004737 111216
045600/ 172220 170127 040200 016720 020144 016720 020142 174020
045620/ 174120 174220 174320 172404 172505 174020 174120 170127
045640/ 042000 005067 020110 005067 020106 000207 005720 012720
045660/ 040020 005020 005020 000763 170127 040200 032711 001000
045700/ 001417 062700 000666 172540 172440 174105 174004 172740
045720/ 172640 172540 172440 014067 020026 014067 020020 170140
045740/ 000207 170367 020010 052777 000040 133036 000137 104656
045760/ 016200 000010 016202 000014 004715 016000 000034 001413
045800/ 014002 001406 032702 000037 001403 042702 000037 004715
046020/ 030027 000035 001365 005725 000205 016000 000034 001413
046040/ 005710 001401 004715 062700 000012 030027 000037 001370
046060/ 015000 177740 001365 005725 000205 004537 110562 016700
046100/ 132716 162703 000002 004767 000006 010366 000010 000207
046120/ 004567 177706 000401 000421 005710 100017 111002 072227
046140/ 000015 020302 103412 066002 000010 020302 103006 017003
046160/ 000004 042703 000037 005126 012605 000207 016102 000006
046200/ 016200 000034 001444 026700 132614 001002 005067 132606
046220/ 005062 000034 014002 001417 012746 000001 122761 000002
046240/ 020025 001405 032702 000037 001402 012716 000401 042702
046260/ 000237 162662 000032 030027 000035 001354 005740 101046
046300/ 010000 011003 000004 000040 012604 005700 001367 000207
046320/ 000000 000000 000000 000000 000000 000000 000000 000000
***** 31. Duplicate Lines (046340 - 047300) *****

```

Figure MON-77. RSTS Dump (Cont.)

MON-140

COMPANY CONFIDENTIAL

DIGITAL

RSTS/E SWS UPDATE
Monitor Internals

```

047320/ 052632 052720 052826 052924 053022 053120 053228 053326 053424
047340/ 053532 053620 053726 053824 053922 054020 054128 054226 054324
047360/ 054532 054620 054726 054824 054922 055020 055128 055226 055324
047380/ 055532 055620 055726 055824 055922 056020 056128 056226 056324
047400/ 056732 056820 056926 057024 057122 057220 057328 057426 057524
047420/ 057832 057920 058026 058124 058222 058320 058428 058526 058624
047440/ 059032 059120 059226 059324 059422 059520 059628 059726 059824
047460/ 060032 060120 060226 060324 060422 060520 060628 060726 060824
047480/ 061032 061120 061226 061324 061422 061520 061628 061726 061824
047500/ 062032 062120 062226 062324 062422 062520 062628 062726 062824
047520/ 063032 063120 063226 063324 063422 063520 063628 063726 063824
047540/ 064032 064120 064226 064324 064422 064520 064628 064726 064824
047560/ 065032 065120 065226 065324 065422 065520 065628 065726 065824
047580/ 066032 066120 066226 066324 066422 066520 066628 066726 066824
047600/ 067032 067120 067226 067324 067422 067520 067628 067726 067824
047620/ 068032 068120 068226 068324 068422 068520 068628 068726 068824
047640/ 069032 069120 069226 069324 069422 069520 069628 069726 069824
047660/ 070032 070120 070226 070324 070422 070520 070628 070726 070824
047680/ 071032 071120 071226 071324 071422 071520 071628 071726 071824
047700/ 072032 072120 072226 072324 072422 072520 072628 072726 072824
047720/ 073032 073120 073226 073324 073422 073520 073628 073726 073824
047740/ 074032 074120 074226 074324 074422 074520 074628 074726 074824
***** 4. Duplicate Lines (047760 - 050340) *****
050000/ 000000 000000 000000 000000 000000 000000 000000 000000
050100/ 020000 004000 000000 000000 000000 000000 177777 000000
050120/ 000000 000000 000000 000000 000000 000000 000000 000000
050140/ 000000 020000 020000 020000 020000 020000 020000 020000
050160/ 020000 020000 020000 020000 020000 020000 020000 020000
***** 5. Duplicate Lines (050360 - 050360) *****
050200/ 020000 020000 020000 020000 020000 020000 020000 020000
050300/ 020002 020002 020002 020002 020002 020002 020002 020002
050320/ 020002 010000 020002 020002 020002 020002 020002 020002
050340/ 000000 000000 000000 000000 000000 000000 000000 000000
***** 4. Duplicate Lines (050360 - 050440) *****
050400/ 000000 000000 000000 000000 000000 000000 000000 000000
050500/ 005040 001200 067206 000000 000000 000000 000000 001100
050520/ 010650 077470 074344 074360 074371 174444 000000 000270
050540/ 000020 000003 000000 000000 000000 000000 000000 000000
050560/ 000000 000000 000000 000000 000000 000000 000000 000000
***** 5. Duplicate Lines (050600 - 050700) *****
050720/ 000000 000000 175401 000000 000000 000000 000000 000000
050740/ 000000 000000 000000 000000 000000 000000 000000 000000
***** 11. Duplicate Lines (050700 - 051220) *****
051240/ 000000 000000 000000 000000 000000 000000 000000 000000
051260/ 000034 000003 000024 000000 000000 000000 000000 000020
051300/ 000000 000000 000000 000000 000000 000000 000000 000000
051320/ 177400 000000 000000 000000 000000 000000 075131 064260 051504
051340/ 000000 000000 111052 044630 051504 177401 000000 000000 000000
051360/ 000000 000000 000000 000000 046504 177400 000000 045504 177401
051400/ 046504 177401 000000 111052 042632 051104 177401 000000 000000
051420/ 040335 005000 051104 177401 000000 051104 177401 000000 000000
051440/ 177400 000000 000000 000000 177401 300000 000000 000000 041104
051460/ 000000 041104 177402 000000 000000 000000 000000 000000 000000
051500/ 000000 000000 000000 000000 000000 000000 000000 000000 000000
***** 12. Duplicate Lines (051520 - 052000) *****
052020/ 000000 000000 111146 000000 054523 000000 000410 077430

```

Figure MON-77. RSTS Dump (Cont.)

MON-141

COMPANY CONFIDENTIAL

DIGITAL

RSTS/E SMS UPDATE
Monitor Internals

052240/	022200	051104	177400	024252	025765	021474	021104	021104	021104
052240/	022200	177400	020220	051104	177420	020200	107124	011300	011300
052240/	051104	177400	020220	051104	177420	020200	051504	177401	000000
052240/	051104	051104	051504	177401	000000	060460	000000	051504	000000
052240/	177401	022200	177777	000000	000000	000000	000000	000000	000000
052240/	000000	022640	000000	000000	000000	000000	000000	000000	000000
052240/	022200	020000	030000	000000	000742	000070	000000	000000	000000
052240/	020017	044015	044000	120000	120000	100000	100000	044003	000000
052240/	044000	130000	100200	004017	176401	176401	000001	000001	000000
052240/	000001	000001	177404	177404	000010	000010	000010	000019	000000
052240/	030010	000000	000000	000401	000001	000010	000010	011010	000000
052240/	000010	000000	000000	000000	000000	000000	000000	000000	000000
052240/	000075	000000	000000	000000	002401	000000	000000	000000	000000
052240/	000000	000000	000000	000000	013740	000000	000000	010340	000000
052400/	000000	000000	000000	000000	000000	000000	000000	000000	000000
052400/	100000	100000	000000	000001	000001	000000	000000	000000	000000
052400/	000000	045653	000001	000000	000000	025725	000376	000376	000376
052400/	000000	000000	000000	000000	000106	040106	000000	000000	000000
052500/	121506	000111	000111	000000	000000	000000	000000	165141	000000
052500/	000000	000000	000000	070371	000000	000000	000000	000000	000000
052500/	000000	000000	000000	000000	000000	000000	000002	000000	000000
052500/	033730	000000	000000	000000	000000	027570	021370	000000	000000
052500/	000000	003110	000000	000000	000000	000000	000000	000000	000000
052500/	020220	000001	000000	000000	000000	000000	000000	000126	000000
052600/	020001	000001	000000	000000	000012	102605	000000	000000	000000
052600/	000000	000000	000000	106220	010000	000000	000000	000000	000000
052700/	020000	000000	000000	000000	006200	044400	000000	000000	000000
052700/	000002	000401	000124	020001	030000	000000	000000	000012	000000
052700/	044511	000000	000000	000006	000000	000000	122230	010010	000000
052700/	000000	000000	000000	000000	020000	000000	000000	122230	000000
053000/	044400	000000	000000	000002	001001	000150	020001	000000	000000
053000/	000000	000000	000012	044511	000000	000000	000006	000000	000000
053000/	000000	122230	010010	000000	000000	000006	020000	000000	000000
053000/	100300	000000	122230	044400	000000	000000	000002	001401	000000
053100/	000000	000000	000000	000000	000000	000012	044511	000000	000000
053100/	000000	000006	000000	000000	122230	010010	000000	000000	000000
053100/	000006	020000	000000	000000	000000	122230	044400	000000	000000
053100/	000000	000002	002001	000000	000000	000000	000000	000000	000000
053200/	000012	044511	000000	000000	000006	000000	000000	122230	000000
053200/	010010	000000	000000	000006	020000	000000	100000	000000	000000
053200/	122230	044400	000000	000000	000002	002401	000000	000000	000000
053200/	000000	000000	000000	000012	044511	000000	000000	000006	000000
053300/	000000	000000	122230	010010	000000	000006	020000	000000	000000
053300/	000000	100000	000000	122230	044400	000000	000000	000002	000000
053300/	003001	000000	000000	000000	000000	000000	000012	044511	000000
053300/	000000	000000	000006	000000	000000	122230	010010	000000	000000
053400/	000000	000006	020000	000000	100000	000000	122230	044400	000000
053400/	000000	000000	000002	003401	000000	000000	000000	000000	000000
053400/	000000	000012	044511	000000	000000	000006	000000	000000	000000
053400/	122230	010010	000000	000000	000006	020000	000000	100000	000000

Figure MON-77. RSTS Dump (Cont.)

DIGITAL

RSTS/E SMS UPDATE
Monitor Internals

053500/	000000	122230	044400	000000	000000	000000	000002	000001	000000
053500/	000000	000000	000000	000000	000000	000000	000012	044511	000000
053500/	000000	000000	000000	000000	000000	000000	000000	000000	000000
053500/	000000	000000	100000	000000	000000	122230	044400	000000	000000
053600/	000002	004401	000000	000000	000000	000000	000000	000000	000012
053600/	044511	000000	000000	000000	000000	000000	000000	000000	122230
053600/	000000	000000	000000	000006	020000	000000	100000	000000	000000
053600/	044400	000000	000000	000002	005001	000000	000000	000000	000000
053700/	000000	000000	000012	044511	000000	000000	000000	000000	000000
053700/	000000	122230	010010	000000	000000	000000	000000	000000	000000
053700/	100000	000000	122230	044400	000000	000000	000000	000000	000000
053700/	000000	000000	000000	000000	000000	000000	000000	000012	044511
054000/	000000	000000	000000	000000	000000	000000	000000	000000	000000
054000/	000006	020000	000000	100000	000000	000000	122230	010010	000000
054000/	000000	000002	006001	000000	000000	000000	000000	000000	000000
054000/	000012	044511	000000	000000	000000	000000	000000	000000	000000
054100/	010010	000000	000000	000000	000000	000000	000000	000000	000000
054100/	122230	044400	000000	000000	000000	000000	000000	000000	000000
054100/	000000	000000	000000	000000	000000	000000	000012	044511	000000
054100/	000000	000000	122230	010010	000000	000000	000000	000000	000000
054200/	000000	100000	000000	122230	044400	000000	000000	000000	000000
054200/	007001	000000	000000	000000	000000	000000	000000	000000	000000
054200/	000000	000000	000000	000000	000000	000000	000000	000000	000000
054200/	000000	000006	020000	000000	000000	000000	000000	000000	000000
054300/	000000	000000	000002	007401	000000	000000	000000	000000	000000
054300/	000000	000012	044511	000000	000000	000000	000000	000000	000000
054300/	122230	010010	000000	000000	000000	000000	000000	000000	000000
054300/	000000	000000	044400	000000	000000	000000	000000	000000	000000
054400/	000000	000001	000000	000000	000000	000000	000000	000000	000000
054400/	000000	000000	000000	000000	000000	000000	000000	000000	000000
054500/	050521	000000	000000	000000	000000	000000	000000	000000	000000
054500/	000000	000000	000006	020000	000000	000000	000000	000000	000000
054500/	044400	000000	000000	000000	000000	000000	000000	000000	000000
054500/	000000	000012	050521	000000	000000	000000	000000	000000	000000
054600/	000000	000000	000012	121014	030000	000000	000000	000000	000000
054600/	000000	006307	006200	044400	000000	000000	000000	000000	000000
054600/	000000	000000	000001	000000	000000	000000	000000	000000	000000
054600/	000000	000006	000000	000000	000000	000000	146260	121414	000000
054700/	000000	020000	033503	000000	006307	006200	044400	000000	000000
054700/	000000	000000	000000	000000	000000	000000	000000	000000	000000
054700/	000012	050521	000000	000000	000000	000000	000000	000000	000000
054700/	000012	050521	000000	000000	000000	000000	000000	000000	000000
054700/	122014	000000	000000	000000	000000	000000	000000	000000	000000
055000/	006200	044400	000000	000000	000000	000000	000000	000000	000000
055000/	000001	000000	000000	000000	000000	000000	000012	050521	000000
055000/	000000	146260	122414	030000	000000	000000	000000	000000	000000
055000/	000000	006307	006200	044400	000000	000000	000000	000000	000000
055100/	013000	000000	000000	000000	000000	000000	000000	000000	000000
055100/	000000	000000	000000	000000	000000	000000	146260	123014	000000

Figure MON-77. RSTS Dump (Cont.)

DIGITAL

RSTS/E SWS UPDATE
Monitor Internals

```

063360/ 000000 000000 000012 044511 000000 000000 000006 000000
063400/ 000020 006200 121016 000000 000000 000006 020000 011072
063400/ 000200 011072 006200 044400 000000 000000 000002 051400
063440/ 000200 000000 000001 000000 000000 000012 044511 000000
063480/ 000200 000000 000000 000000 000000 000000 000000
063520/ 000200 020000 011073 000000 011073 006200 044400 000000
063520/ 000000 000002 052000 000000 000000 000001 000000 000000
063560/ 000012 044511 000000 000000 000006 000000 006200
063600/ 122016 000000 000000 000006 020000 011074 000000 011074
063600/ 006200 044400 000000 000000 000002 052400 000000 000000
063640/ 000000 000000 000012 044511 000000 000000 000000 000006
063640/ 000000 006200 122416 000000 000000 000006 020000
063680/ 011075 006200 011075 006200 044400 000000 000002
063700/ 053000 000000 000001 000000 000000 000012 044511
063720/ 000000 000006 000000 000000 006200 123016 000000
063740/ 000000 000006 020000 011076 000000 011076 006200 044400
063760/ 000000 000002 053400 000000 000000 000001 000000
064000/ 000000 000012 044511 000000 000000 000006 000000
064020/ 006200 123416 000000 000000 000006 020000 011077
064040/ 011077 006200 044400 000000 000000 000004 000000
064080/ 000201 000306 000000 000124 000000 000002 000000
064100/ 002400 000150 000000 000000 000006 001000 000000
064120/ 000300 000306 000000 000000 000000 000012 000000
064140/ 000300 000306 000000 000000 000000 000012 000000
064160/ 002400 000300 000000 000014 000000 000006 000000
064200/ 000000 000016 000000 000000 000000 000000 000000
064220/ 000000 000006 000000 000000 000022 000000 000006
064240/ 002400 000000 000000 000024 000000 000006 000000
064260/ 000300 000006 000000 000006 005400 000000 000000
064300/ 000000 000006 000000 000000 000000 000000 000000
064320/ 006400 000300 000000 000000 000000 000006 000000
064340/ 000000 000006 000000 000000 000000 000001 102000
064360/ 000135 000300 015340 136210 000000 000000 077104
064400/ 132004 015240 000000 000000 000000 001400 000000
064420/ 000300 000300 000000 000010 000000 000000 102000
064440/ 001335 000000 020000 134410 000000 000000 077104
064480/ 132004 017700 000000 000000 000000 001400 000000
064500/ 000300 000000 000000 000010 001000 000000 102000
064520/ 000135 000000 020440 136210 000000 000000 077104
064540/ 132004 015240 000000 000000 000000 001400 000000
064560/ 000300 000300 000300 000010 001400 000000 102000
064600/ 001335 000000 021000 131010 000000 000000 077104
064620/ 132004 020700 000000 000000 000000 001400 000000
064640/ 000300 000300 000010 002000 000000 000000 000000
064660/ 000300 000000 000000 000000 000000 000000 000000
064700/ 000300 000000 000000 000000 000000 000000 000000
064720/ 000200 000000 000000 000000 000000 000000 000000
064740/ 000300 000000 000000 000000 000000 000000 000000
064760/ 000000 000000 000000 000000 000000 000000 000000
065000/ 000000 000000 001012 000000 000117 000000 064014
065020/ 013202 013202 000023 102605 041102 004300 000026 014400

```

Figure MON-77. RSTS Dump (Cont.)

DIGITAL

RSTS/E SWS UPDATE
Monitor Internals

```

065240/ 000000 000000 000000 000014 000000 000123 000000 000001
065260/ 000000 000000 000000 077206 017312 020001 000000 000001
065300/ 000000 000000 000000 000000 021614 021540 000014 000426
065320/ 000125 100000 000001 000000 000000 000000 077206 016156
065340/ 000001 000001 000361 001402 050664 004514 000000 014754
065360/ 000014 001000 000000 000000 000000 000400 000000 000000
065380/ 000000 000000 000000 000000 000000 000000 000000 000000
065400/ 000000 000000 000000 000000 000000 000000 000000 000000
065420/ 000000 000000 000000 000000 000000 000000 000000 000000
065440/ 252162 000000 000000 000000 000000 000000 000000 000000
065460/ 037416 000077 000077 000000 000000 001004 065524 001033
065500/ 000000 001401 065510 000000 002001 065534 002001
065520/ 000000 177777 000000 004113 177400 000033 000000 000001
065540/ 000000 000100 000000 000000 000000 000000 000000 000000
065560/ 000000 000340 000000 000000 000000 000000 000000 000000
065600/ 000000 000000 000000 000000 000000 000000 000000 000000
065620/ 000000 000000 000000 000000 000000 001312 000000 056700
065640/ 104004 000000 000000 000000 000000 000000 000000 000000
065660/ 000000 000000 000000 000000 002004 100716 000001 100320
065700/ 000000 120000 000000 000000 000000 000000 000000 000000
065720/ 020000 000000 000454 000000 003001 013540 000000 000000
065740/ 000000 000000 000000 103112 065746 103014 002270 000000
065760/ 000000 070000 000000 000000 012040 000000 000000 000000
066000/ 000001 000000 035117 032462 000056 031000 040020 041005
066020/ 033455 000071 000000 000000 000000 000000 000000 000000
066040/ 177750 001002 001002 001002 001002 001002 001002 001002
066060/ 001002 001034 001002 001002 001002 001002 001002 001002
066100/ 001002 013026 001002 001002 001002 001002 001002 001002
066120/ 001002 013026 001002 001002 001002 001002 001002 001002
***** 2. Duplicate Lines (066140 - 066160) *****
066200/ 001002 013026 001002 001002 001002 013026 001002 001002
066220/ 001002 013026 001002 001002 001002 016034 016002 020020
066240/ 001002 016034 001002 021042 001002 001002 001002 001002
066260/ 001002 001002 001002 001002 001002 001002 001002 001002
066300/ 001002 001002 001002 001002 001002 001002 001002 001002
***** 4. Duplicate Lines (066320 - 066400) *****
066420/ 001002 001002 001002 001002 001002 001002 001002 001002
066440/ 017002 012300 000006 165151 000070 000000 000000 000000
066460/ 000000 000000 000012 000441 000000 000000 000000 000000
066500/ 000310 000025 000000 000000 173440 105500 000000 166715
066520/ 005571 150000 000000 177404 000000 000000 001002 000574
066540/ 002000 127600 000000 177776 177777 177777 177777 000000
066560/ 000000 000000 000000 177776 000777 000000 000000 177776
066600/ 177777 177777 177777 000000 000000 000000 000000 000000
066620/ 177777 107740 010567 000400 000001 140706 140644 003740

```

Figure MON-77. RSTS Dump (Cont.)

DIGITAL

RSTS E SWG UPDATE
Monitor Internals

DIGITAL

RSTS E SWG UPDATE
Monitor Internals

073060/034340	177776	005037	177572	025077	012370	012786	071630
073120/005037	000102	120037	000167	104742	012737	074340	177776
073140/005037	177572	005277	012334	012706	071630	005767	177034
073140/005037	004767	175514	027727	012310	177777	001354	016746
073150/124564	004767	002020	100775	026726	105606	001403	162737
073220/000204	000206	005007	116767	176442	176442	012767	000400
073220/176424	005067	176766	000167	176430	054523	041504	043104
073240/051504	045504	046104	046504	050104	051104	041104	041113
073260/046116	045520	046530	050114	046515	052104	054104	052124
073320/052115	054504	177777	000300	177777	177777	000001	000001
073320/177777	000001	177777	000001	000002	000127	000000	000016
073340/000005	000005	000002	000001	000001	073272	073272	073276
073360/052246	052222	052222	052222	052226	052232	052232	052236
073420/052236	052242	047320	047600	047602	047640	047654	047656
073420/047664	047670	000001	000002	000001	000004	000010	000020
073440/000001	000001	000001	075740	076642	075740	076642	077014
073460/077172	077202	077216	077236	075626	075646	075666	075706
073520/075726	000000	000000	000000	000000	000000	126140	000000
073520/131202	172040	000000	177440	176300	176700	177560	065536
073540/065536	065536	065536	065536	065536	065536	065536	065536
073560/065536	065536	065536	065536	065536	065536	160020	160020
073600/160020	160020	160020	160020	160020	160020	160020	160020
073620/160020	160020	160020	160020	160020	160020	160040	160040
073640/162040	160040	160040	160040	160040	160040	160040	160040
073660/160040	160040	160040	160040	160040	160040	160060	160060
073700/160060	160060	160060	160060	160060	160060	160060	160060
073720/160060	160060	160060	160060	160060	160060	160250	160250
073740/160250	160250	160250	160250	160250	160250	160260	160260
073760/160260	160260	160260	160260	160260	160260	160270	160270
074000/160270	160270	160270	160270	160270	160270	000000	000000
074020/000000	000000	000000	000000	000000	000000	000000	000000
074040/000000	000000	000000	000000	000000	000000	160150	160160
074060/160170	160200	160210	160220	177514	172440	172440	172440
074100/177342	177342	000000	000000	000000	000000	000000	000002
074120/000004	000004	000006	000006	000010	000010	000010	000002
074140/000003	177777	177777	000000	000000	020024	020024	020022
074160/024022	024022	004000	004000	000000	000000	064740	064740
074200/100220	100220	121510	121510	121510	000000	000000	000000
074220/000000	000000	000000	000000	000000	000000	000000	000000
074240/000000	000000	000000	000000	000000	130060	122614	122606
074260/121536	121720	122762	122730	076716	076716	076716	076716
074300/076716	121044	121174	120712	120610	120750	127750	121156
074320/121412	121252	120460	122464	120574	130476	127440	126332
074340/130176	127752	136176	075554	130634	000000	000000	126150
074360/126160	125102	124042	124736	131440	000000	000000	000000
074400/000000	000000	000000	000000	123676	125616	135432	000000
074420/101744	045436	133440	122530	000000	120412	000000	117644
074440/101210	135334	132010	101210	000000	077014	077202	077202
074460/077216	077216	077216	077236	000000	000000	000000	000000
074500/000000	000000	000000	101430	106144	075740	000000	075740

Figure MON-77. RSTS Dump (Cont.)

074520/075740	101430	105610	000000	077244	000000	075740	101430
074540/101430	101430	101430	107632	107624	107606	107624	107624
074560/107624	107500	107516	000000	000000	000000	126242	120552
074600/121010	000000	122372	124112	000000	076730	077030	000000
074620/125466	124174	131114	116466	123120	121026	121246	077034
074640/121454	126030	124240	131256	000001	124506	121024	121106
074660/077040	121254	177775	177777	131152	000001	124636	121024
074700/121172	077044	122260	177775	177777	131252	120314	133722
074720/121020	120774	077054	121252	125760	123506	130460	000000
074740/133500	121022	121104	077064	121434	125236	123510	131030
074760/000000	133500	121024	121104	077070	121404	125266	123542
075000/131030	000000	121734	121024	121104	077050	122150	125274
075020/124154	131034	000000	130754	000000	121054	077060	000000
075040/000000	000000	000000	100000	074402	000426	070420	030430
075060/035406	000016	000004	100422	000107	000000	000107	000107
075100/000107	000000	000107	000107	000107	001000	000200	000002
075120/000200	001000	001000	000200	000776	001000	177777	131177
075140/101163	177777	177777	030007	000003	000003	177777	000007
075160/177777	177777	177777	101003	101163	101063	137372	130760
075200/134650	147000	147234	142420	144710	135172	137332	142642
075220/014000	014424	014520	134434	134634	133446	130742	130842
075240/140220	140556	141452	066000	143376	033310	062323	000301
075260/000001	000001	000001	145477	035001	024260	062225	033573
075300/024541	145325	145012	024263	000001	043155	145564	000001
075320/035631	065424	000001	144712	145507	145505	024001	024565
075340/026275	025165	033001	140221	034167	145215	022411	135172
075360/014000	014424	014520	023000	022000	142420	041000	142642
075400/000001	065000	145624	062001	000001	263007	145113	143276
075420/146330	000001	000000	000000	000377	177400	000000	000000
075440/177777	000377	177400	000377	177777	177400	177777	000000
075460/000000	000000	177400	000000	000000	000000	000000	000000
075500/177400	177777	000000	000377	000377	067552	067364	071614
075520/066754	125012	125446	133572	133506	125122	126176	126420
075540/120226	121736	122220	120736	121564	005000	013737	101634
075560/172352	000137	120772	054110	053720	053530	053000	046000
075600/047000	051266	052000	045002	050000	051000		

Figure MON-77. RSTS Dump (Cont.)

FEEDBACK FOR MONITOR

Using the map and octal dump provided, answer the following questions.

1. Is 'JOB' attached to any resident libraries? No
- o Look at address 1026 in the dump. This is JOBWDB in low memory which contains a pointer to the WDB @ W.WIN1 for 'JOB'.
 - o Since the value of JOBWDB is 0, the job has no resident libraries attached.
2. Have any resident libraries been added? Yes
- o Look at the map under the RAWMEM section and find the label LIBLST. The address associated with LIBLST is 50500.
 - o Now look at address 50500 (LIBLST) in the dump. This address contains 5040. Since the value is non-zero, at least one resident library has been added on the system. The value 5040 is a pointer to the first resident library description block (LIB).
 - a. How many resident libraries have been added? 1
 - o Look at address 5040 in the dump. The value at this address is the link word for the resident library description block. Since the word contains a 0, there is only one LIB block in the list.
 - b. Answer the following questions about the library which is first in the list:

Resident library name	RMSRES
-----------------------	--------

FEEDBACK

- o Look at addresses 5042 and 5044. They contain 71233 and 70533 respectively, which is the resident library name (R.NAME), in RAD50. Using the RAD50 table, the library name is RMSRES. (Remember all subtraction is octal).

```

71233
70200 - R
-----
1033
1010 - M
-----
 23
 23 - S
-----
  0

```

```

70533
70200 - R
-----
 333
 310 - E
-----
  23
  23 - S
-----
   0

```

Its project, programmer number [0,1]

- o Look at address 5046 (L.PPN) which contains 000001. This value has the project number in the high byte and the programmer number in the low byte.

```

00000000/00000001
-----
 0          1

```

Its size in Kwords (decimal) 23kw

- o Look at address 5056 (R.KS1Z) which contains 1027 in octal. The low byte of this word is the size of the resident library.

27 (in octal) = 23 (in decimal)

FEEDBACK

Its protection code

42

- o Look at address 5074 which contains 025200. The library status flags are in the low byte and the library protection code is in the high byte.

L.PROT L.STAT
00101010/10000000

52 (octal) = 42 (decimal)

Which resident library flags are set. Save the load address.

- o Look at address 5076 (R.FLAG) which contains 040000.

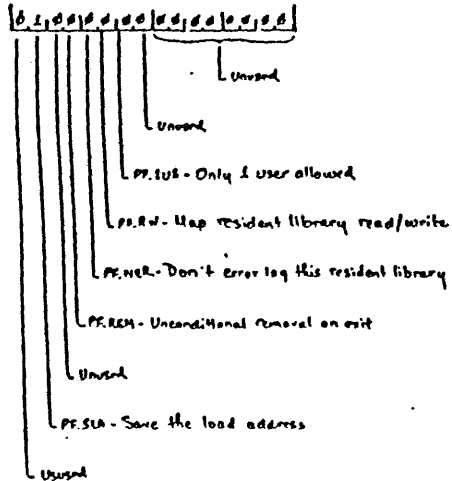


Figure MON-78.

FEEDBACK

Load address of library (octal)

1320000

- o Look at address 5060 (M.PHYA) which contains 13200. This is the actual physical address at which the library will load divided by 100 octal.
3. Are large files sysgened on the system? Yes
 - o Look at the map. Since the section SYDLRG appears in the map, this is a large file system.
 5. Considering the first file open on the first disk unit of the system:

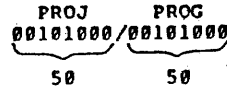
Note

To find the address of the FCB for the first file:

- o Look at the map under the RAWMEM section and find the label FCBLST. The value 52352 is the address of the FCB table.
- o Starting at address 52352 in the dump, scan for the first non-zero word. This is at address 52366 which contains 13740.
- o The 13740 is the address of the FCB for the first file.

FEEDBACK

- a. What is the file's PPN? (40,40)
- The file's PPN is the third word in the FCB. At address 13744 is the value 24050 which is the PPN.



Note
50 (octal) = 40 (decimal)

- b. What is the filename and extension? DNDA14.TMP
- The filename and extension are stored in RAD50 format starting at address 13746.
 - The values 15464, 5472 and 77430, converted from RAD50 using the table provided, produce DNDA14.TMP.
- c. Is the file placed? No
- Look at address 13754 which contains 36024. The low byte of the word is the FSSTAT byte.

FEEDBACK

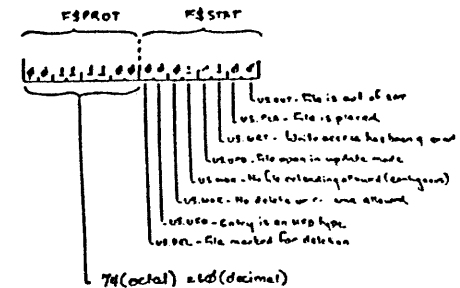


Figure MON-79.

- d. Has write access been given out? Yes
- e. Is the file open for update? No
- f. Is the file contiguous? Yes
- g. Is the file marked for deletion? No
- h. What is the protection code? 60
- Look at the high byte shown in c (FSSTAT) above. The 74 octal is 60 decimal.
- i. What is the access count for opens in normal or update mode? 1
- Look at address 13756 which contains 000001. The low byte of the word (FSACNT) contains the access count for normal or update mode.
- j. What is the clustersize? 8
- Address 13744 contains the clustersize.
- k. What is the octal address of the first WCB for the file? 17740

FEEDBACK

- o Address 13776 contains the WCB address.

6. What disk types and how many units of each are sysgennd on this system?

Disk	Number
DS	2
DK	2
DM	2
DR	2
DB	3

There are two ways to find the answer.

- a. o Look at the .ABS. section of the map for symbols of the form CNT.xx where xx is a disk type. (i.e. CNT.DB)
 - o The value of these symbols is the number of units sysgennd.
- b. o Look at the map under the TBL section for the label DEVCNT which contains the address of the device count table 73310.
 - o Starting at address 73310 the first nine words contain: -1, -1, 1, 1, -1, 1, -1, 1, 2. These are the highest valid unit numbers on this system for each disk type supported by RSTS/E.
 - o Referring to the DEVNAM table in the module, the first nine words represent the disk types supported by RSTS/E (DC, DF, DS, DK, DL, DM, DP, DR, DB)

FEEDBACK

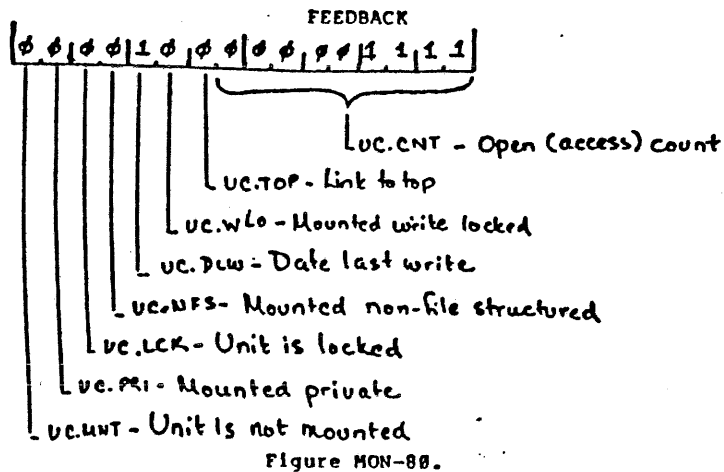
- o Since the entries in DEVNAM and DEVCNT correspond directly to each other:

DEVNAM entry	DEVCNT entry	# of units
DC	-1	0
DF	-1	0
DS	1	2
DK	1	2
DL	-1	0
DM	1	2
DP	-1	0
DR	1	2
DB	2	3

7. Considering DB2:

To answer this question you must first get the UNTCNT entry for DB2: on this system. To get this information:

- o Look at the map under the TBL section for the label DEVPTR which contains 73362, the address of the device pointer table.
- o Starting at address 73362 count in 9 words. The value (52242) is the address in the UNTCNT table for DB0:
- o Starting at address 52242 count in 3 words (this is the offset from DB0: to DB2: in the UNTCNT table). The value is 004017, which is the unit count status information for DB2:



- a. Is it mounted? Yes
- b. Is it a date of last write disk? Yes
- c. Are files added at the top of the directory? No
- d. What is the current open count (octal)? 17
- e. What is the pack clustersize and error count? 8
0

To answer this question you must:

- a. Determine the FIP unit number*2 for DB2: on this system.
- o Using the information in question 5, you know that the configured disks are:

FEEDBACK

Disk	FUN(octal)	FUN*2 (octal)
DS0:	0	0
DS1:	1	2
DK0:	2	4
DK1:	3	6
DM0:	4	10
DM1:	5	12
DR0:	6	14
DR1:	7	15
DB0:	10	20
DB1:	11	22
DB2:	12	24

Note

The order of the disks is very important. They must be in the order in which they appear in the DEVNAM table. Also the FUN begins with 0.

- o Therefore, the FUN for DB2: is 12 and FUN*2 is 24.
- b. Using the FUN*2 for DB2:, offset into the UNTCLU table to get the pack clustersize in the low byte and the error count in the high byte.
 - o Look in the map under the RAWMEM section for the label UNTCLU which contains the value 52276. This is the address of the UNTCLU table.
 - o To offset into the table by FUN*2, add 24 to 52276 (remember to do octal addition) to get the address 52322.
 - o Look at address 52322 in the dump which contains 000010.

FEEDBACK

UNTERR	UNTCLU
<u>00000000</u>	<u>/00001000</u>
0	10

- o The pack clustersize=10 (octal) which is 8 in decimal.
- o The error count is 0.

8. What is the device designator of the system disk?

DB2:

To answer this question, you must remember that the word just before the DEVPTR table points to the entry in the UNTCNT table for the system disk (see the module).

- o Since the address of the DEVPTR table is 73362 (see question 7), look at address 73360 which contains 52246.
- o Starting an address 73362 (DEVPTR) scan the first nine words for the number which is the closest to but not greater than 52246.

The ninth word (52242) is the one you're looking for.

- o From question 7, this entry points from the DEVPTR table into the UNTCNT table for DB0:
- o Therefore:

52242	—————>	DB0:
52244	—————>	DB1:
52246	—————>	DB2:

RSTS/E V7.0 SWS UPDATE

CAC

Data Caching Internals

DIGITAL

POST/E UPDATE
Data Caching Internals

Data Caching Internals

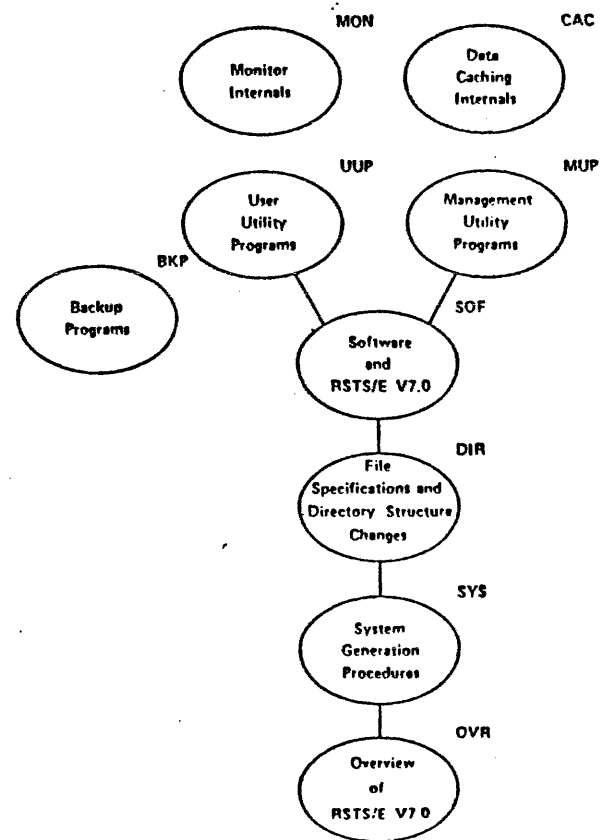
CONTENTS

Introduction.....	CAC-1
Monitor Structures - Without Sysgenning Data Caching.....	CAC-3
Monitor Structures - Data Caching Sysgenned.....	CAC-9
How Cache Views a Disk.....	CAC-19
How Cache Moves Data into Memory.....	CAC-22

CONFIDENTIAL

RSTS/E UPDATE
Data Caching Internals

COURSE MAP



Data Caching Internals

. INTRODUCTION

The internal structures and operations of caching on RSTS/E aids in understanding what actions are taken when a user requests disk information.

Knowing how caching works enables you to determine:

- Whether a file should be cached
- How a file should be cached
- What the cache cluster size should be

DIGITAL

RSTS/E UPDATE
Data Caching Internals

This page is for notes.

DIGITAL

RSTS/E UPDATE
Data Caching Internals

MONITOR STRUCTURES - WITHOUT SYSGENNING DATA CACHING

- Only directory caching (FIP buffering)
- Defaults at startup time
 - Cache enabled
 - Cache clustersize of 1
 - Use monitor buffer pool
 - Total number of cache clusters is 65535 (unlimited)
 - Total number of cache clusters for directories is 65535 (unlimited)

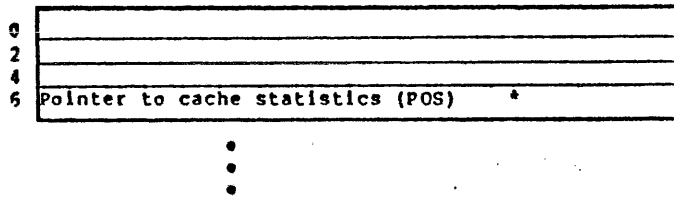
0	CHECTL
2	CHETIC
4	CHENXT
6	CHEPRV
10	CHEPAR*
12	CHETAG*
14	CHEUSM*
16	(unused)
20	CHELIM
22	CHEFLM
24	(unused)
26	(unused) 0-no data caching
30	(size of cache cluster in blocks)
32	SAVKA6
34	RSTRTH
36	REFLAG
40	CHEL3Q
42	
44	CACHEQ
46	POS **
50	**
52	**
54	**
56	HIT **
60	**
62	**
64	**
66	CHERST**
70	CHENUM
72	CHENUE

\$\$CACH::

CHECTL	400 → cache enabled
	0 → cache disabled
CHETIC	1 → charge cache time
	0 → no charge
CHENXT	Pointer to "first" cache tag in list 0+2
CHEPRV	Pointer to "last" cache tag in list 0 + 0
CHEPAR	PAR6 value for extended cache tags (INIT)
CHETAG	List of usable extended cache tags (INIT)
CHEUSM	-1 → monitor pool (INIT)
	0 → no use of monitor pool
CHELIM	Limit of total numbers of extended cache tags
CHEFLM	Limit of number of extended cache directory tags
SAVKA6	Saved KISAR6
RSTRTH	"Restart" return address
REFLAG	"Restart" flag (SP stack value)
CHEL3Q (word)	Pointer to cache completion queue root (CACHEQ)
	(word)
CACHEQ	Caching level 3 queue bit (QCACHE)
POS (2 words)	Caching completion queue root
	Total possible number of hits
HIT (2 words)	Total possible number of FIP hits
	Total number of hits
	Total number of FIP hits
CHERST	Number of times cache restarted
CHENUM	Number of monitor buffer cache elements
CHENUE	Number of extended pool cache elements

* FILLED IN BY INIT
** ONLY IF STATS ARE GENNED

Figure CAC-1. CHECTL AREA OF MONITOR (no data caching)

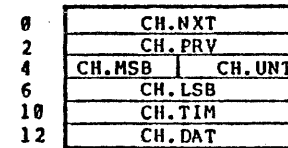


* Only if STATS are genned

Figure CAC-2. STSTBL entry for statistics (no data caching)

Cache tags

- Identify directory blocks currently in the cache clusters.
- Keep track of time of last reference.
- A doubly linked list with head and tail pointers in CHECTL.
- Exist in XBUF and/or monitor pool space.
- XBUF tags created by INIT at start up.



CH\$SIZ - Size of cache tag in bytes

Figure CAC-3. Cache Tag

- | | | |
|---------|------------------------------|-------------------------------------|
| CH.NXT | Pointer to next @ CH.PRIV | } If two high bits set, tag in XBUF |
| CH.PRIV | Pointer to previous @ CH.NXT | |
| CH.UNT | Unit number of block | |
| CH.MSB | Block's FBN MSB | |
| CH.LSB | Block's FBN LSB | |
| CH.TIM | Time of last reference | |
| CH.DAT | Pointer of cached data block | |

DIGITAL

RSTS/E UPDATE
Data Caching Internals

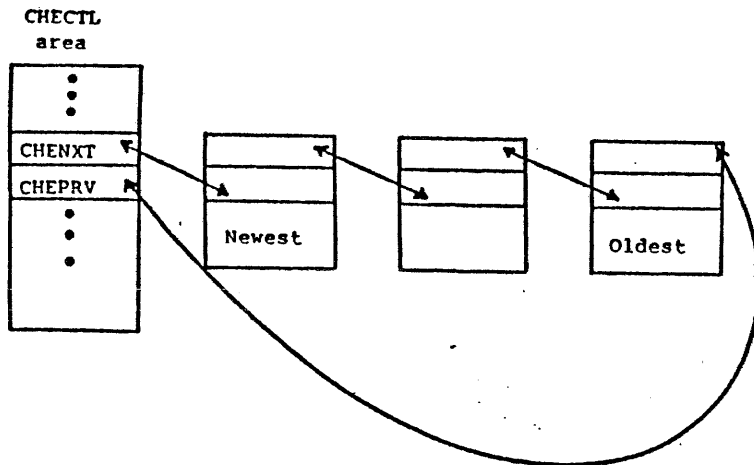


Figure CAC-4. Cache Tag List

DIGITAL

RSTS/E UPDATE
Data Caching Internals

MONITOR STRUCTURES - DATA CACHING SYSGENMED

- Directory and Data Caching available
- Defaults at startup time
 - Cache enabled
 - Data caching available through MODE
 - Cache cluster size of 4
 - Total number of cache clusters to be used is 65535 (unlimited)
 - Total number of cache clusters for directories is 65535 (unlimited)
 - Total number of cache clusters for data is 65535 (unlimited)
- All cache tags are in XBUF

DIGITAL

RSTS/E UPDATE
Data Caching Internals

0	CHECTL
2	CHE TIC
4	CHE NXT
6	CHE PRV
10	CHE PAR*
12	CHE TAG*
14	0 - No monitor buffers used*
16	CHE WTG*
20	CHE LIM
22	CHE FLM
24	CHE DLM
26	200-Data caching CHEMOD
30	CHE BSZ*
32	CHE SHF*
34	*
36	CHE DIC*
40	SAVKA6
42	RSTRTN
44	REFLAG
46	TMWQUE
50	CHLFXB
52	CHWTLJ
54	
56	CHWRTQ
60	CHRD LJ
62	
64	CHREDQ
56	FAKDSQ
70	
72	
74	
76	
100	
102	
104	
106	
110	
112	
114	
116	
120	

\$\$CACH==

\$\$CBSZ

* Filled in by INIT

Figure CAC-5. CHECTL Area of Monitor (data caching)

DIGITAL

RSTS/E UPDATE
Data Caching Internals

122	
124	
126	RDS.XC**
130	
132	DIR.CN**
134	
136	HRS.RH**
140	
142	DIR.RH**
144	
146	CHE RST**
150	
152	CHE NUE
154	CHE CNT
156	CHDCNT
160	DAT.CN**
162	
164	DAT.RH**
166	
170	RDS.NC**
172	
174	RDS.CI**
176	
200	RDS.LX**
202	
204	RDS.RX**
206	
210	RDS.SR**
212	
214	WTS.NX**
216	
220	WTS.CH**
222	
224	CAN.CH**
226	
230	HIT.CH**
232	

** Only if STATS Genned

Figure CAC-5. CHECTL Area of Monitor (data caching) (cont.)

DIGITAL

 RSTS/E UPDATE
 Data Caching Internals

CHECTL 400 → cache enabled
 0 → cache disabled
 CHEPIC 1 → charge cache time
 0 → no charge
 CHENXT Pointer to first in list @+2
 CHEPRV Pointer to last in list @+0
 CHEPAR PAR6 value for extended cache tags
 CHETAG List of usable extended cache tags
 CHEWTG List of usable extended cache tags
 CHELIM Limit of total number of extended cache tags
 CHEFLM Limit of number extended cache directory tags
 CHEDLM Limit of number of extended cache data tags
 CHEMOD 0 → No data caching
 1 → open mode caching
 -1 → cache all data
 CHEBSZ Size of a cache cluster in blocks
 CHESHF (1 word) Amount of shift to divide by cluster size
 (1 word) Mask to round FBN down to cache cluster
 CHEBIC Complement of above mask
 SAVKA6 Saved KISAR6
 RSTRTN "Restart" return address
 REFLAG "Restart" flag (SP stack value)
 TMWQUE Queue for writes awaiting free write tag
 CHLFXB Time of last failure to allocate buffer
 CHWTL3 (1 word) Pointer to caching queue root for writes
 (1 word) (CHWRTQ)
 CHWRTQ Level 3 queue bit for cache (QCACHE)
 CHRDJ3 (1 word) Caching queue root for writes
 (1 word) Pointer to caching queue root for reads
 (CHREDQ)
 CHREDQ Level 3 queue bit for cache (QCACHE)
 FAKDSQ Caching queue root for reads
 Fake DSQ for handling cross cluster transfers
 RDS.XC (2 words) Number of disk transfers which are cached
 DIR.CN (2 words) Number of above which are directory transfers
 HRS.RH (2 words) Number of cache hits
 DIR.RH (2 words) Number of above which are directory hits
 CHERST Number of times cacher restarted
 CHENUE Number of extended pool cache elements

CAC-12

COMPANY CONFIDENTIAL

DIGITAL

 RSTS/E UPDATE
 Data Caching Internals

CHFCNT Count of directory cache tags
 CHDCNT Count of data cache tags
 DAT.CN (2 words) Data transfers which were cached
 DAT.RH (2 words) Cache hits on data transfers
 RDS.NC (2 words) Transfers which are not cached
 RDS.CI (2 words) Fraction of above due to cache installs
 RDS.LX (2 words) Large transfers cut up into segments
 RDS.RX (2 words) Large transfers not cached because no tags
 RDS.SR (2 words) Segment reads produced when cut up
 WTS.NX (2 words) Total write transfers
 WTS.CH (2 words) Tags hit by writes
 CAN.CH (1 word) Pointer to STATS area for directory reads
 (DIR.CN)
 (1 word) Pointer to STATS area for data reads
 (DAT.CN)
 HIT.CH (1 word) Pointer to STATS area for directory hits
 (DIR.RH)
 (1 word) Pointer to STATS area for data hits
 (DAT.RH)

CAC-13

COMPANY CONFIDENTIAL

DIGITAL

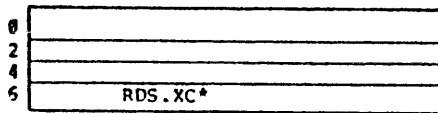
RSTS/E UPDATE
Data Caching Internals

DIGITAL

RSTS/E UPDATE
Data Caching Internals

Cache Tags

- Identify directory and data blocks currently in cache clusters
- Keep track of time of last reference
- A doubly linked list with head and tail pointers in CHECTL
- Exist only in XBUF
- Created by INIT at startup



* Only if STATS are genned

Figure CAC-6. STSTBL entry for statistics (data caching)

0	CH.NXT	
2	CH.PRV	
4	CH.MSB	CH.UNT
6	CH.LSB	
10	CH.DAT	
12	CH.BLS	
14	CH.TYP	
16	CH.TIM	

Figure CAC-7. Cache Tag

CH.NXT Pointer to next @ CH.PRV
 CH.PRV Pointer to previous @ CH.NXT
 CH.UNT Unit number of block
 CH.MSB Block's FBN MSB
 CH.LSB Block's FBN LSB
 CH.DAT Pointer to the cached data block
 CH.BLS Pointer to DSQ's waiting on tag
 CH.TYP Type pointer (data or directory)
 CH.TIM Time of last reference

Write Tags

- Used to store DSQ information while disk is transferring data to the cache

0	WT.MAM	WT.TYP
2	WT.MAL	
4	WT.CNT	
6	WT.L3Q	
10	WT.CHM	WT.COF
12	WT.CTP	
14	WT.MSC	

Figure CAC-8. Cache Write Tag

WT.TYP Write tag type (real or from tag list)
 WT.MAM Saved DSQMAM
 WT.MAL Saved DSQMAL
 WT.CNT Saved DSQCNT
 WT.L3Q Saved DSQL3Q
 WT.COF Saved transfer's block offset into cluster
 WT.CHM Saved cache mode of transfer
 WT.CTP Saved pointer to cache tag
 WT.MSC Saved DSQMSC

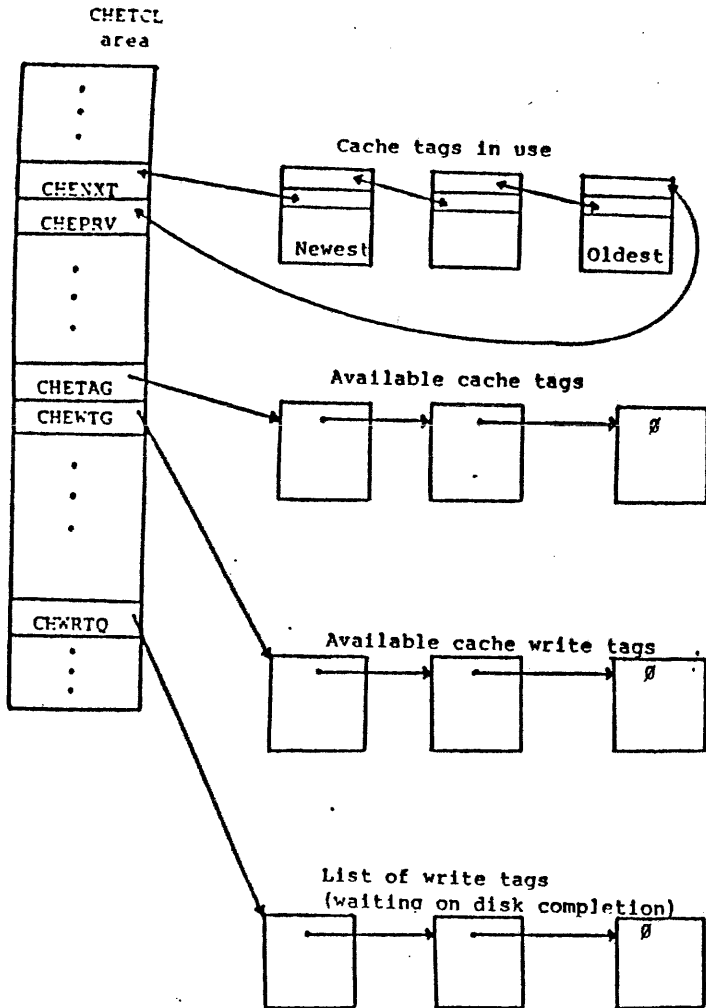
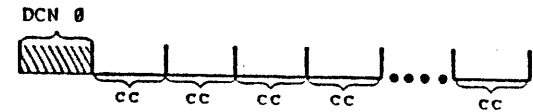


Figure CAC-9. Tag List (data caching)

HOW CACHE VIEWS A DISK

- Always divided into cache clusters (cc) starting at DCN 1.

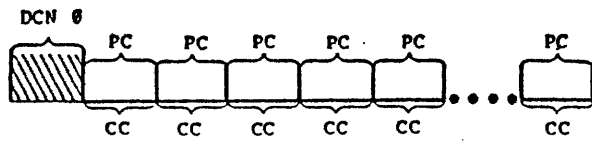


Note

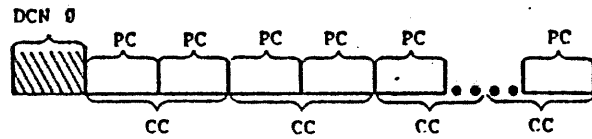
Cache will always transfer an entire cache cluster.

If there is no room in the cache a normal disk I/O to user buffer is done.

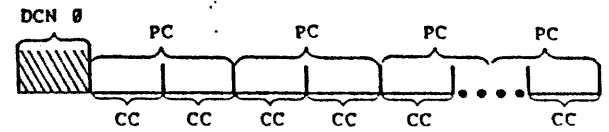
- When a cache cluster equals a pack cluster (pc)



- When a cache cluster is larger than a pack cluster

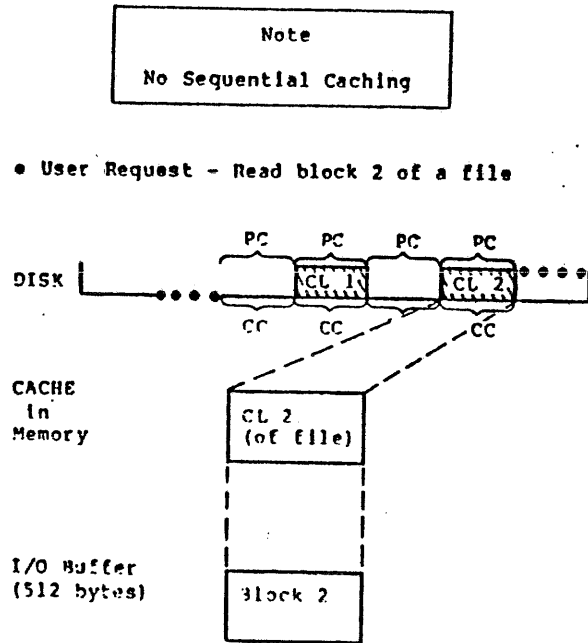


- When a cache cluster is smaller than pack cluster



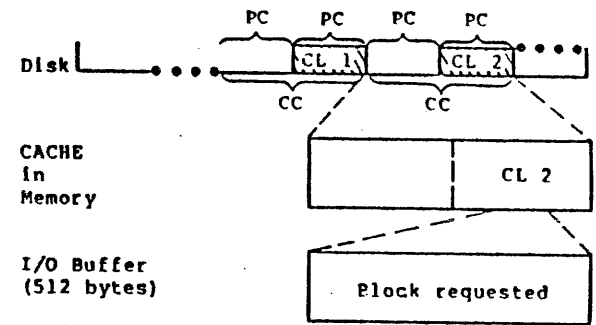
HOW CACHE MOVES DATA INTO MEMORY

Case 1: • Cache Cluster = Pack Cluster = File Cluster = 1



Case 2: • Cache cluster > Pack Cluster = file cluster

• User Request - Read Block 2 of a file in file cluster 2



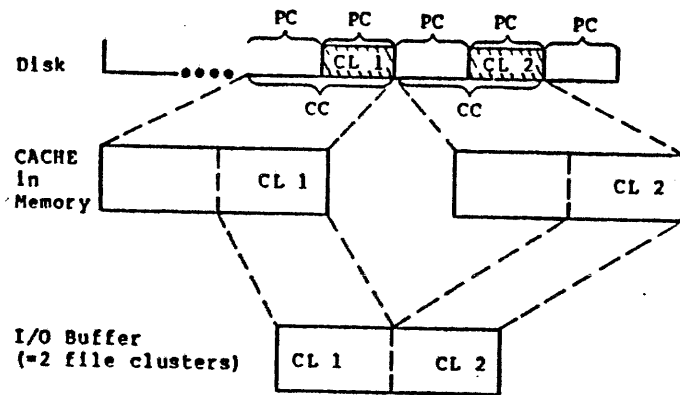
Note

Since an entire cache cluster is read into memory, part of the data is not useful to the user.

DIGITAL

RSTS/E UPDATE
Data Caching Internals

- Case 3: • Cache Cluster > Pack Cluster = File Cluster
• User Request - Read File Clusters 1 and 2



Note
Cache makes two disk reads to satisfy the request.