CONTENTS (Cont.)

FIGURES

TABLES

PREFACE

This manual describes the complete generation process for Version V06A-02 of RSTS/E, the PDP-11 Resource Sharing Time Sharing System/Extended. It is assumed that the reader is familiar with RSTS/E software and the hardware on which RSTS/E will run. The *RSTS/E System Reliability Test* document supplied with the RSTS/E software is designed to be used with this document.

For more information on RSTS/E documentation, consult the *RSTS/E Documentation Directory*.

For a quick reference to a subject in this guide, use the following list.

If you need to know about	See Section
System generation from magtape	2.2.1
System generation from disk cartridge	2.3.1
System generation from DECtape	2.2.2
Configuration question considerations	2.7
Configuration question examples	2.6
Bootstrapping a device	Appendix A
Starting time sharing operations	3.7
Changing system files	3.4.2
Adding more memory to RSTS/E	3.6.3.6
Changing the maximum number of users	3.6.2
Changing the maximum job size allowed	3.6.2
Organization of the system disk	3.4.1
Building system library files	Ch. 4
Creating user accounts	4.4
Reporting system troubles	Back of manual
Creating TTYSET.SPD file	2.7.3, 2.7.4, 4.3.4

If you need to know about	See Section
Enabling DH11 modem service	3.5
Formatting disks	2.3.3
Determining memory requirements	Appendix D
Building the auxiliary run time system	4.5
Building the COBOL compiler	4.6

CHAPTER 1

SYSTEM GENERATION OVERVIEW

RSTS/E software is distributed on either 7-track or 9-track magtape, disk cartridge, or DECtape. The RSTS/E system is generated from the distributed software. During the generation process, the system manager or other responsible individual specifies the hardware configuration, chooses optional software features, and tailors the system to local installation requirements.

The system generation process comprises the separate steps shown in Figure 1-1. The following is a summary of those steps.

1. Use the distribution media (if other than disk cartridge) to create a system generation monitor.
2. Use the system generation monitor to conditionally assemble and link the tailored components of RSTS/E into a RSTS/E Monitor capable of being bootstrapped.
3. Save a copy of the RSTS/E Monitor for backup purposes.
4. Load the RSTS/E Monitor onto the new RSTS/E system disk.
5. Bootstrap the RSTS/E disk to load the initialization code and use PATCH, one of several initialization options, to correct any known errors.
6. Use DSKINT to create the file directory structure on the new RSTS/E system disk.
7. Use REFRESH to create and tailor various system files for swapping, error messages, overlay code, and crash data.
8. Use SETKEY to enable DH11 modem service.
9. Use DEFAULT to establish default time sharing parameters such as the maximum number of jobs and maximum user job size.

System Generation Overview

1. Bootstrap DOS/BATCH-11

1½. COPY THIS BY LOGIN
RUN CILUS
ROLLIN OPTION

SAVE ORIGINAL

1¾. FORMAT SYSTEM DISK BY SYSLOD

2. Type BATCH SYSGEN and answer configuration questions

3. Save RSTS/E Monitor.

4. Put RSTS/E Monitor on RSTS/E System Disk

5. OPTION: PATCH install published patches

6. OPTION: DSKINT Create file directory structure

7. OPTION: REFRESH Create System Executive files

8. OPTION: SETKEY Establish keyboard defaults

9. OPTION: DEFAULT Establish default startup parameters

10. OPTION: START Start RSTS/E timesharing

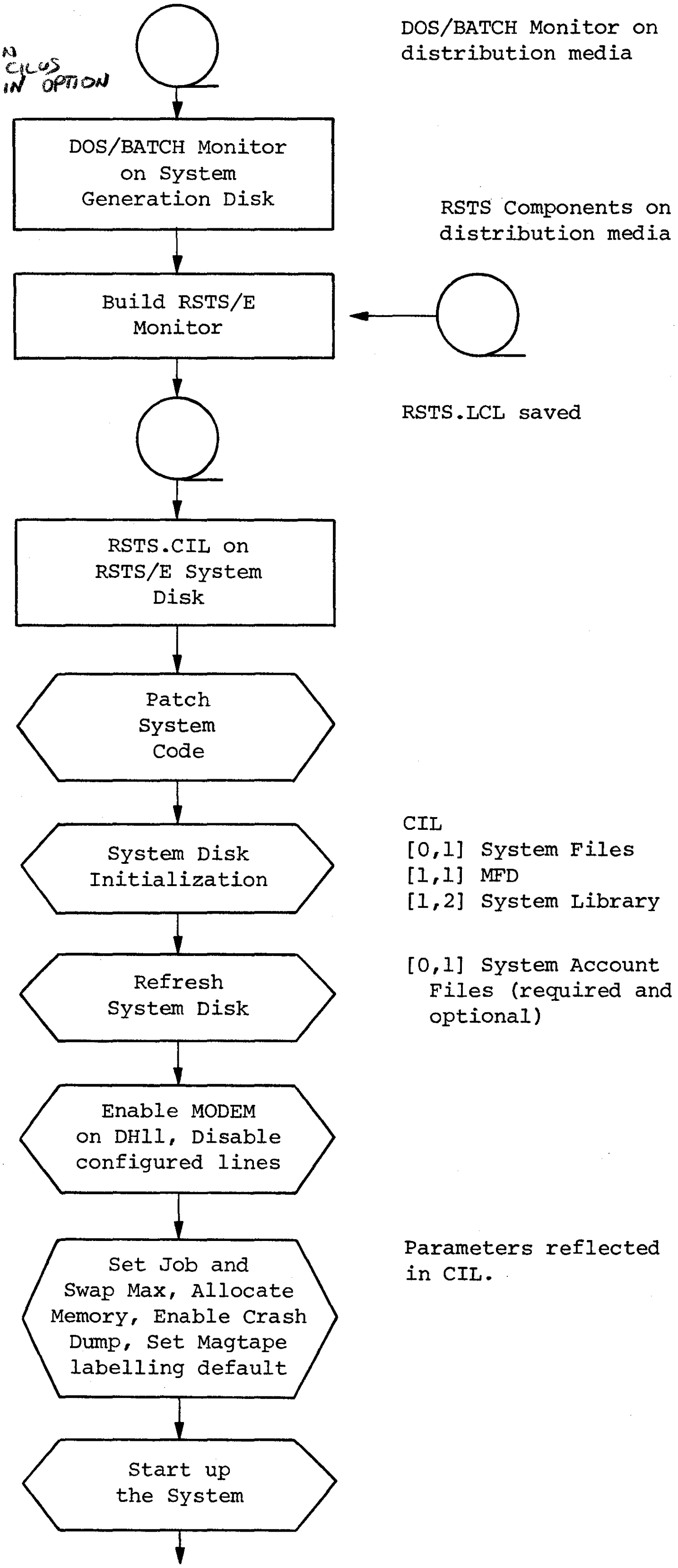


Figure 1-1
System Generation Flow Chart

11. Type RUN BUILD to compile system library programs and auxiliary RTS

12. Type RUN LNGBLD to create auxiliary language

13. Type RUN BUILD to create system test programs

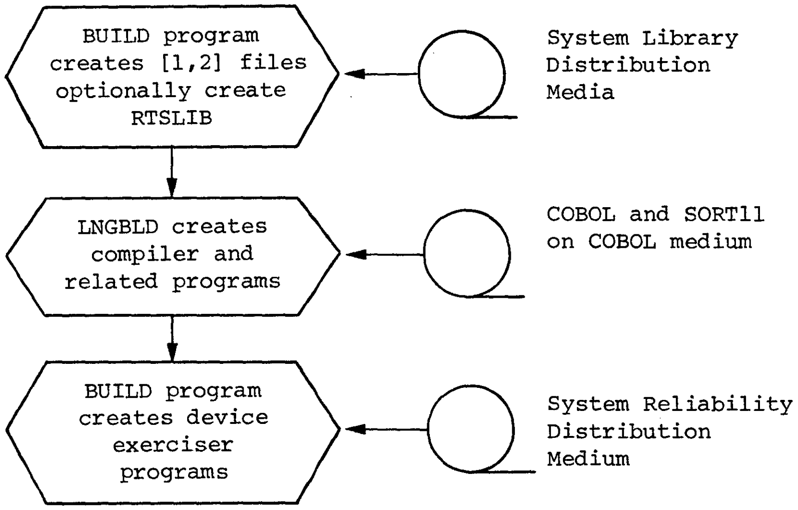
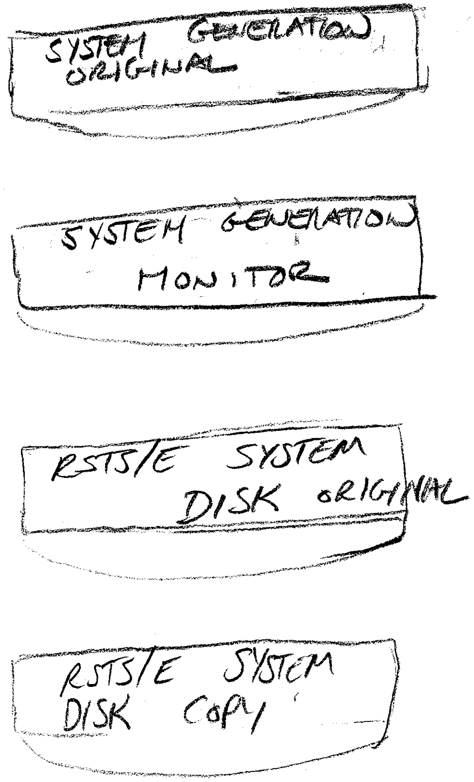


Figure 1-1 (Cont.)
System Generation Flow Chart

PHYSICAL DISKS INVOLVED



Save →

write on this?

Save →

will be → DB0:

10. Use START to load the RSTS/E Monitor and start time sharing.
11. Run the BUILD program from the system library distribution medium to create the general system library. Optionally, rerun BUILD to create other system library programs and the auxiliary run time system RTSLIB.
12. Optionally run LNGBLD to create the COBOL language code.
13. Run BUILD to create the system reliability test programs (device exercisers) and conduct system testing.

Upon conclusion of the system testing, RSTS/E is ready for normal time sharing operations.

Chapter 2 describes in detail the procedures to generate the RSTS/E system and BASIC-PLUS language code (steps 1 through 4 in Figure 1-1). Section 2.9 contains information needed to replace a currently existing RSTS/E system without destroying system and user files.

Chapter 3 describes the RSTS/E initialization options (steps 5 through 10 in Figure 1-1). Although the initialization options can be used at any time after the system generation process, each option is described in the order required for generating RSTS/E.

Chapter 4 describes the system library build procedures (steps 11 and 12 in Figure 1-1).

Step 13 of the system generation process (running the system reliability test) is described in the RSTS/E System Reliability Test document. The build procedure is described in Section 2.2 of that document and Chapters 3 and 4 contain guidelines for running the device exercisers and for determining the reliability of the system.

The appendixes of this manual give summary and reference information. Appendix A describes all hardware bootstrap information. Appendix B lists RSTS/E consistency errors and gives guidelines for recovery. The summary of short form questions and responses in Appendix C can facilitate system generation for an experienced user. The system module sizes in Appendix D can be used to calculate memory requirements. Appendix E summarizes system generation monitor errors. Appendixes F and G give data on hidden options and on floating address and vector assignments.

CHAPTER 2

RSTS/E SYSTEM GENERATION

This chapter describes the procedures to generate RSTS/E system code from distributed software. Section 2.1 presents an overview of the system generation process and contains comments concerning the media on which DIGITAL distributes the software.

Section 2.2 contains the SYSLOD procedures required for magtape and DECtape software. Section 2.3 describes the copy procedures for disk cartridge software. Procedures to start the system generation batch command file are in Section 2.4. Examples and guidelines to answering configuration questions are in Sections 2.5, 2.6, and 2.7 respectively.

2.1 SYSTEM GENERATION OVERVIEW

The process for generating the RSTS/E system code consists of the following general steps.

a) If the software is on magnetic tape, bootstrap a tape to load the stand alone program SYSLOD and transfer the system generation monitor to a disk using SYSLOD. If the software is on disk cartridge, bootstrap the system generation disk to load the system generation monitor and copy both the system generation cartridge and the system library cartridge.

b) Log into the system generation monitor and initiate execution of the batch command file.

c) Answer configuration questions printed during the system generation dialogue.

d) Follow the instructions printed by the system generation monitor and mount and dismount tapes and disks as required.

After the RSTS/E system code is created, proceed to Chapter 3 to initialize the RSTS/E system disk and continue with Chapter 4 to build the system library files.

The system generation procedure is performed using a non-standard DOS/BATCH monitor as the system generation monitor. If the software supplied by DIGITAL is on magtape or DECtape, this monitor must be loaded onto an RF11, RK11, RP02, RP03, or RP04 disk with the stand alone program SYSLOD. If the DIGITAL-supplied software is on disk cartridge, bootstrap the cartridge to load the system generation monitor. (In this document, the disk on which the system generation monitor resides is referred to as the system generation disk. Such nomenclature differentiates the disk in question from the disk on which the RSTS/E public file structure resides and which is referred to as the RSTS/E system disk.)

The procedures for magtape or DECtape media involve transferring files to the system generation disk. After the system generation monitor is loaded onto a disk, one batch command file must be transferred from the tape to the disk. The system manager then initiates execution of the batch commands which transfer all required system generation programs (MACRO, LINK, CILUS, EDIT, PIP, and SYSLOD) from tape to disk without further interaction.

The procedures for disk cartridge media are similar to those for tape except that none of the file transfer operations are necessary. However, it is advisable to copy the disk cartridges containing the system generation and system library programs and to use the copies instead of the originals to generate the system. Copying of the disk cartridges is performed with the stand-alone program ROLLIN. To begin system generation, the system manager bootstraps the copy of the system generation disk cartridge, answers the initial monitor DIALOGUE questions, and types one command which initiates execution of the batch command file. Commands in the batch file delete any old RSTS/E system (RSTS.LCL file) which may exist from a prior system generation.

After the system generation monitor transfers all files from tape or deletes files from the disk cartridge, it executes a command in the batch file which runs the system generation program SYSGEN. The program prints approximately 70 hardware and software configuration questions and creates two files based on the answers typed in response to each question. The answers must accurately reflect the hardware configuration on which RSTS/E will run and the software options desired. During the configuration dialogue, SYSGEN creates the configuration file CONFIG.MAC and a second batch command file SYSGEN.BAT, which are later used to create the RSTS/E system code tailored to local installation requirements.

After all configuration questions are answered, the system generation monitor executes the second batch command file. The monitor conditionally assembles the TBL module (system tables) and the TTY module (terminal service) using the configuration file created during the SYSGEN dialogue.

During execution of the second batch command file, the system generation monitor prints instructions to mount appropriate tapes or disks as each is required. If the 2780 package is required on the system, the monitor also prints messages to mount the 2780 tape or disk cartridge. Required 2780 files are transferred to the system generation disk and assembled.

The system generation monitor next links the monitor code (RSTS), overlay code (OVR), error messages (ERR), initialization code (INIT), and octal debugging tool (ODT) and copies files from tape as needed. Subsequently, the monitor links the BASIC-PLUS monitor (Run Time System) code to include all BASIC-PLUS language and mathematical package options.

The final step in the system generation process creates a linked core image library (LICIL) of the RSTS/E system from the load modules created by the linking process. For magtape and DECtape media, the step includes writing the LICIL (RSTS.LCL file), system load maps, batch and configuration files, and SYSLOD program to a scratch tape. For disk cartridge software, the LICIL is created and remains on the system generation disk.

During the final step for magtape and DECtape media, the monitor prints a message indicating the exact command to type to write the contiguous core image library, or CIL, onto the RSTS/E system disk. The batch command file terminates by loading SYSLOD into memory. The system manager can type the exact command to SYSLOD and write the CIL to the RSTS/E system disk or, at some later time, can bootstrap the tape to load SYSLOD into memory. After writing the CIL to the RSTS/E system disk, SYSLOD automatically bootstraps the device to load the RSTS/E initialization code into memory.

During the final step for disk cartridge media, the system generation monitor prints a message to mount and write enable the RSTS/E system disk. Typing one command continues execution of the batch file which runs CILUS to write the CIL onto the RSTS/E system disk. If the RSTS/E system disk is either an RP02, RP03, RP04, or an RF11 disk, the batch command file terminates by bootstrapping that device. If the RSTS/E system disk is an RK disk, the system generation monitor prints a message to move the disk to RK unit 0 and to bootstrap it by the hardware loader to load the RSTS/E initialization code.

The RSTS/E CIL contains the system initialization code, the monitor, the BASIC-PLUS Run Time System, overlay code, error messages, and, optionally, the stand alone program ROLLIN. When the RSTS/E system disk is bootstrapped, the initialization code is loaded into memory. The initialization options described in Chapter 3 must be used to install necessary patches (PATCH option), to initialize the system disk (DSKINT option), to create the system files (REFRESH option), to set keyboard defaults (SETKEY option), to establish default start up parameters (DEFAULT option), and to begin time sharing (START option).

If interrupted, the entire system generation procedure need not be redone. If one of two checkpoints has been passed, merely return to the previous checkpoint. The checkpoints and procedures for recovery are indicated by messages printed by the system generation monitor.

The drive selection standard implemented in V06A-02 enables the system disk to be mounted on and operated from a drive unit other than 0. However, the DOS/BATCH-11 monitor used to generate RSTS/E runs only on unit 0. Hence, the SYSGEN program requires unit 0 to operate. The resulting RSTS/E system disk must be bootstrapped once from unit 0 to install a new bootstrap routine which passes the correct unit information to the initialization code. Thereafter, the system disk can be mounted on any drive unit.

2.2 MAGTAPE AND DECTAPE PROCEDURES

Magtape and DECtape procedures differ in the bootstrap procedures required and in the device designators used in several keyboard commands.

2.2.1 Magtape Bootstrap Procedure

Procedures to operate the magtape device are described in Section 5.6 of the RSTS-11 System User's Guide.

To bootstrap the magtape, perform the following steps.

Mount the SYSTEM GENERATION TAPE labelled

DEC-11-ORSPA-D-MC9	for a 9-track TU10 drive
DEC-11-ORSPA-D-MC7	for a 7-track TU10 drive
DEC-11-ORSPA-D-WC9	for a TU16 drive

on unit 0 with the write enable ring removed.

Ensure that the tape is at its load point. (The LD PT indicator comes on.) The computer does not bootstrap the device unless the tape is at its load point.

Set the ON-LINE/OFF-LINE switch on the tape unit to ON-LINE and ensure that the RDY indicator is lit.

Ensure that the console terminal is on line.

Follow the instructions in Appendix A for the type of hardware bootstrap on the system.

Proceed to Section 2.2.3 to transfer the system generation monitor from tape to disk.

2.2.2 DECtape Procedures

Procedures to operate the DECtape device are described in Section 5.5 of the RSTS-11 System User's Guide. To bootstrap the DECtape, perform the following steps.

Mount the DECtape reel labelled DEC-11-ORSBA-D-UC1 (SYSTEM GENERATION TAPE 1 of 2) on unit 0.

Mount the DECtape reel labelled DEC-11-ORSBA-D-UC2 (SYSTEM GENERATION TAPE 2 of 2) on unit 1.

On DECtape units 0 and 1, set the REMOTE/OFF/LOCAL switch to REMOTE and the WRITE ENABLE/WRITE LOCK switch to WRITE LOCK.

Ensure that the console terminal is on line.

Bootstrap unit 0 by following the instructions in Appendix A for the type of hardware bootstrap device on the system.

Proceed to Section 2.2.3 to transfer the system generation monitor from tape to disk.

2.2.3 Loading the System Generation Monitor from Tape

When the tape is bootstrapped, the computer reads unit 0 and loads SYSLOD into memory. SYSLOD prints its identification line followed by the first in a series of queries as follows.

```
SYSLOD V08-01A
CONSOLE FILL COUNT=
```

If SYSLOD does not print its identification and the processor halts, a parity error possibly was detected in reading the tape. Retry the entire procedure, including rewinding the tape to load point (if magtape). If the bootstrap procedures fail repeatedly, obtain a new tape reel.

2.2.3.1 Answering the SYSLOD Questions - When SYSLOD runs, perform the following steps.

> If the system generation disk is either an RK03 or RK05 cartridge or an RP02, RP03, or RP04 pack, mount it on drive unit 0.
>
> Ensure that the system generation disk is on line, write enabled, and ready before proceeding. (After the DIALOGUE query is answered, SYSLOD does not recognize any devices previously not ready.)

Type the RETURN key in response to the CONSOLE FILL COUNT question and proceed as follows:

```
SYSLOD V08-01A
CONSOLE FILL COUNT=                  (Type RETURN key.)
DATE: 14-FEB-75                      (Type in dd-mmm-yy format.)
DIALOGUE?                            (Type RETURN key.)
#
```

SYSLOD indicates that it is ready to accept a command string by printing the # character. A single command string is necessary to format the disk, to check for bad blocks, and to transfer the system generation monitor to the disk. Use the following format for the SYSLOD command string.[1]

```
#xx:MONLIB.CIL/FO/CO:0/HO/BO<yy:MONLIB.LCL
```

where:

xx	is DB for RP04 disk pack DP for an RP02 or RP03 disk pack DK for an RK05 or RK03 disk cartridge DF for an RF type disk.
yy	is DT0 if DECtape medium is used or, MT0 if either 7- or 9-track TU10 magtape medium is used. MM0 if TU16 magtape software is used.

NOTE

> The /FO switch is not necessary for RF-type disks.

The /FO switch in the SYSLOD command causes a removable disk to be formatted. SYSLOD prints the message STARTING FORMAT PASS when it starts formatting and prints END FORMAT PASS when it is finished formatting. If any other messages appear, consult Appendix B to determine the nature of the error and the recovery procedure. The time between the two messages depends on the size and type of disk.

The /CO:0 switch writes one pattern on each block of the disk to ensure that no bad blocks are used. The /HO switch causes SYSLOD to place a pointer to the CIL in the bootstrap record. The /BO switch causes SYSLOD to bootstrap the device upon completing the transfer.

The entire process takes between 5 and 20 minutes depending upon the size and type of disk. If SYSLOD prints any error messages, consult Appendix E for the meaning and possible steps for recovery. Upon completing the transfer, SYSLOD prints the following messages.

```
SYSLOD COMPLETE

ANSWER WITH CARRET OR 'Y' CARRET: -IS YOUR LINE FREQUENCY 50 HERTZ?
DO YOU WANT TO DISABLE DIALOGUE FOREVER? NO
DOS/BATCH V9-20C
DATE:
```

---

[1] On an LA30(S) DECwriter, an LA36 DECwriter II, and a VT05(B) alphanumeric display terminal, it may be necessary to use the SHIFT key while typing alphabetic characters in order to ensure that upper case characters are transmitted.

Type the RETURN key (CARRET denotes carriage return) if the line frequency of the power used to run the PDP-11 is 60 Hertz. Type YES and the RETURN key only if the power frequency is 50 Hertz.

SYSLOD then prints a question asking whether to disable the dialogue forever. The system generation monitor begins with a dialogue similar to that used by SYSLOD. It is possible to disable this dialogue at this time by typing YES followed by the RETURN key. For RSTS/E system generation purposes, type NO so that the dialogue is not disabled. Proceed to Section 2.4.1 for instructions on answering the system generation monitor dialogue.

2.3 DISK CARTRIDGE PROCEDURES

Disk cartridge procedures involve bootstrapping the device and copying the original cartridges using the stand alone program ROLLIN.

To prevent possible destruction of the system generation and system library disk cartridges, it is advisable to copy the cartridges and use the copies for generating systems. The cartridges are created on properly aligned drives. Since drive alignment drifts slightly in shipping and with age, problems sometimes occur. If the cartridges cannot be copied, a DIGITAL Field Service representative must check the drive alignment before system generation can continue. The stand alone program ROLLIN is included on the system generation disk cartridge to facilitate the copy operation.

2.3.1 Disk Cartridge Bootstrap

To bootstrap the cartridge, perform the following steps.

Physically mount the cartridge labelled DEC-11-ORSPA-D-HC (SYSTEM GENERATION DECPACK) in the RK03 or RK05 unit 0.

Ensure that the RDY light is on.

If the cartridge has not been copied, ensure that the WR PROT light is on. (This condition write protects the disk.) If the copying is complete, ensure that the WR PROT light is off.

Ensure that the console terminal is on line.

Follow the instructions in Appendix A for the type of hardware bootstrap device on the system.

The system reads the disk and loads the system generation monitor which prints the following lines.

```
DOS/BATCH V9-20C
DATE:
```

If the monitor fails to identify itself, retry the entire operation, and carefully check the bootstrap procedure. After the monitor prints its identifying lines, proceed to Section 2.3.2 to copy the system generation and system library cartridges or proceed to Section 2.4 to start system generation.

2.3.2 Copying the Disk Cartridges Using ROLLIN

To copy the disk cartridges, perform the following steps.

Mount a new disk cartridge on drive unit 1.

Ensure that the RDY light comes on and that the WR PROT light for unit 1 is off.

Ensure that the WR PROT light for unit 0 is on. The original disk must be write protected to prevent inadvertent destruction.

Continue the dialogue in the following manner.

```
DOS/BATCH V9-20C
DATE:  11-JUL-74                 (Type in dd-mmm-yy format.)
TIME:  06:51                     (Type in hh:mm format.)
DIALOGUE?                        (Type the RETURN key.)
```

When the monitor prints the $ character, type the LO 1,1 command. The monitor prints the current date and time followed by the $ character.

```
$LO 1,1                          (Terminate with RETURN key.)
DATE:  11-JUL-74
TIME:  06:52
$
```

The program CILUS is used to load ROLLIN. Type the RUN CILUS command as shown.

```
$RUN CILUS                       (Terminate with RETURN key.)
CILUS V08-06A
#
```

CILUS runs and prints its header line followed by the # character. Type the command shown to run ROLLIN. Then type the command to format unit 1 and copy unit 0 to unit 1.

Run Rollin

```
#ROLLIN.CIL/BO                    (Terminate with RETURN key.)
```

Format [illegible] 1 [illegible] copy

```
ROLLIN V07
#DK1:/FO<DK0:/VE                  (Terminate with RETURN key.)
```

ROLLIN prints messages signalling the start and end of the format pass and the start of the verification pass. If no errors are encountered, ROLLIN prints the # character again as shown below.

```
STARTING RK FORMAT PASS

END RK FORMAT PASS

STARTING RK VERIFICATION PASS

#
```

If any errors are encountered, ROLLIN prints appropriate messages and the # character. A Field Service representative should be called to align the drive. If ROLLIN does not print any error messages, continue according to the following steps.

Move the LOAD/RUN switch to its LOAD position on both units 0 and 1.

When the LOAD light comes on, remove the cartridges from their respective drives.

Label the copied cartridge in such manner as SYSTEM GENERATION COPY. Store the original in a safe place.

Mount the disk cartridge labelled DEC-11-ORSLA-D-HA SYSTEM LIBRARY AND RELIABILITY DECPACK in unit 0. Ensure that the WR PROT light is on.

Mount a second new cartridge in unit 1. Ensure that the RDY light comes on and that the WR PROT light for unit 1 is off. Ensure that the WR PROT light for unit 0 is on. The original disk on unit 0 must be write protected.

Since ROLLIN is still waiting, type the following command in response to the # character.

```
#DK1:/FO<DK0:/VE
STARTING RK FORMAT PASS
END RK FORMAT PASS
STARTING RK VERIFICATION PASS
#
```

If any errors are encountered, ROLLIN prints appropriate messages and the # character. A Field Service representative should be called to align the drive. If ROLLIN does not print any error messages, continue according to the following steps.

> Move the LOAD/RUN switch to its LOAD position on both units 0 and 1.
>
> When the LOAD light comes on, remove the cartridges from their respective drives.
>
> Label the copied cartridge in such manner as SYSTEM LIBRARY COPY. Store the original system library cartridge with the original system generation cartridge.
>
> Mount the copy of the system generation disk in unit 0 and move the LOAD/RUN switch to its RUN position. Ensure that the RDY light comes on and that the WR PROT light is off. (The disk must be write enabled.)

Bootstrap unit 0 by typing the following command to ROLLIN.

```
#/BO:DK
DOS/BATCH V9-20C
DATE:
```

Proceed to Section 2.4.1 to start the system generation monitor.

2.3.3 Formatting Disks Using SYSLOD

During a system generation from disk cartridge distribution media, the CILUS program loads the RSTS/E Core Image Library onto the RSTS/E system disk. Since CILUS cannot format disks, the SYSLOD program

Cannot format SYSTEM DISKS!

should be used at this point to format the disk to be used as the RSTS/E system disk. Bootstrap the copy of the System Generation disk cartridge, answer the monitor dialogue, perform the login procedure and run CILUS to load SYSLOD as follows.

```
DOS/BATCH V9-20C
DATE:  11-JAN-75                    Type the date in format shown
TIME:  16:15                        Type the time in format shown
DIALOGUE?                           Type the RETURN key

$LO 1,1
DATE:  11-JAN-75
TIME:  16:15

$RUN CILUS
CILUS V08-06A

#SYSLOD.CIL/BO
```

The command shown bootstraps SYSLOD from the disk.

When SYSLOD runs, it prints a header line and the first query line. Mount the disk(s) to be formatted and answer the SYSLOD questions shown in the following sample. SYSLOD DOES NOT RECOGNIZE ANY DEVICE WHICH IS NOT READY WHEN THE DIALOGUE QUESTION IS ANSWERED.

```
SYSLOD V08-01A
CONSOLE FILL COUNT=                 Type the RETURN key
DATE:  11-JAN-75                    Type the date in format shown
DIALOGUE?                           Type the RETURN key

#
```

Type the device designation with the /FO option as shown.

```
#DB1:/FO
STARTING FORMAT PASS
END FORMAT PASS
#
```

In the command shown, DB1: can be replaced by DBx:, DPx:, or DKx: where x is between 0 and 7 inclusive. If any messages other than the formatting messages appear, consult Appendix E for possible recovery procedures. The operation takes between 5 and 20 minutes depending on the size and type of disk used.

2.4 STARTING SYSTEM GENERATION

Once the system generation disk is bootstrapped, the system generation monitor runs. The system manager must perform the monitor dialogue and the login procedure and initiate the batch command file. If magtape or DECtape distribution media are used, he must additionally transfer the batch command file from tape to the disk.

2.4.1 Monitor Dialogue

After the disk is bootstrapped, the system generation monitor prints its identification line followed by the first of several prompting lines. Type the current date in response to DATE: and the current time of day in response to TIME: as shown below. Terminate each response with the RETURN key.

```
DOS/BATCH V9-20C
DATE:  11-JAN-75                    (Type in dd-mmm-yy format.)
TIME:  10:12                        (Type in hh:mm format.)
DIALOGUE?
```

The monitor dialogue can be omitted if the line printer used for system generation (unit 0) is an LP11 with 80 columns and if the console fill count required is 0. To omit the dialogue, type the RETURN key in response to the DIALOGUE query. The monitor prints the $ character. Continue at Section 2.4.2 to perform the login procedure.

To continue the dialogue, type YES followed by the RETURN key in response to the DIALOGUE query and proceed as shown.

```
DIALOGUE?  YES
DO YOU WANT TO RESET CONSOLE FILL COUNT? YES
FILL COUNT=
```

Type the console fill count in response to the FILL COUNT= query according to the following values for the type of console terminal on the computer.

Console Fill Values	Console Terminal Types
0	ASR-33 and ASR-35 Teletype; LA30S and LA30P DECwriter (110 and 150 baud); LA36 (any baud rate), VT05, VT05B, or VT50 display (110, 150, and 300 baud).

Console Fill Values	Console Terminal Types
1	ASR-37 Teletype VT05B display at 600 baud.
2	VT05B display at 1200 baud.
4	VT05B display at 2400 baud. LA30P DECwriter at 300 baud.
12	LA30S DECwriter at 300 baud.

For example, type 12 for an LA30S DECwriter at 300 baud. If the line printer is an 132-column LP11, answer all remaining questions as shown below.

```
FILL COUNT=12
ARE ANY DEVICES DOWN? NO
DO YOU WANT TO CHANGE LINE PRINTER? YES
LS11? NO
HOW MANY COLUMNS? 132
LOWER CASE? NO
OVERPRINT? NO
DO YOU WANT TO CHANGE CARD READER DEFAULTS? NO
HAVE YOU GOT RK02 DISK? NO

$
```

If the line printer is an 80-column LP11 (the default), type the RETURN key in response to the first line printer question. The remaining line printer questions do not appear. The card reader question is printed only if a card reader is present. Proceed to the login procedure when the $ character is printed.

2.4.2 Performing the Login Procedure

To log into the system generation monitor, type the LO 1,1 command in response to the $ character as shown below.

```
$LO 1,1
DATE:  21-JUL-74
TIME:  10:56
$
```

If the login procedure is done properly, the monitor prints the current date and time followed by the $ character. Otherwise, the monitor prints an appropriate error message followed by the $ character. Simply try again.

At this point, procedures differ slightly for tape and disk media. Continue with Section 2.4.3 if the medium is either magtape or DECtape. Proceed to Section 2.4.4 if the medium is disk cartridge.

2.4.3 Transferring the Batch Command File from Tape

TAPE

SKIP IF NOT USING TAPE DN

The PIP program must be run from tape and one command must be typed to transfer the batch command file to the system generation disk. Use the following format for the command to execute PIP.

2.4.4, DISK

```
$RUN xx:PIP
```

where:

```
xx  is MT for either 7- or 9-track TU10 magtape,
       MM for TU16 magtape, or
       DT for DECtape.
```

When PIP prints the identification line and the # character as follows:

```
PIP V10-02
#
```

Use the following format to transfer the file.

```
#SY:<xx:SYSGEN
```

where:

```
xx   is MT for either 7- or 9-track TU10 magtape,
        MM for TU16 magtape, or
        DT1 for DECtape.
```

PIP signals completion by printing the # character. Type the CTRL/C combination to terminate PIP and the KI command in response to the dot character printed by the monitor. For example,

```
#↑C
.KI
$
```

When the monitor prints the $ character, proceed to Section 2.4.4 to execute the batch command file.

2.4.4 Initiating the Batch Command File

To initiate execution of the first batch command file, type the following command in response to the $ character.

```
$BATCH SYSGEN
```

The monitor executes the commands in the batch command file SYSGEN. There is a delay as the monitor processes the first commands in the batch file. When the message BEGINNING OF RSTS/E SYSTEM GENERATION is printed, SYSGEN has entered the configuration dialogue. Section 2.5 contains an explanation of the various forms of the questions and of the procedure to restart the questions.

2.5 CONFIGURATION QUESTIONS

After the batch command file starts, the system generation program SYSGEN runs and enters the configuration dialogue. The dialogue is a series of approximately 70 hardware and software configuration questions. The questions come in both a long form and a short form. With each form, SYSGEN can calculate the answers. This latter feature is called automatic answers and is described in Section 2.5.1.

Long form questions contain explanatory information and are useful to anyone who is unfamiliar with the system. For a sample printout of the long form questions, see Section 2.6.1. To save time when the dialogue questions are familiar, select the short form of the questions. A sample printout of the short form questions appears in Sections 2.6.2 and 2.6.3. If a question is unclear, simply type the RETURN key in response to a short form query; SYSGEN prints the long form of that particular question.

During the dialogue, SYSGEN checks the answers entered. If an answer is incorrect, SYSGEN reprints the query or series of queries regarding that subject. To restart the dialogue, use the checkpoint facility described in Section 2.5.2. Implications of the configuration questions are given in Section 2.7.

After all configuration questions are answered, the monitor begins executing the second batch command file. For information on this part of the procedure, consult Section 2.8.

2.5.1 Automatic Answers

The SYSGEN automatic answer facility is enabled by appending /A to the response for the short/long form query. SYSGEN allows the user to accept the answer, to supply a different answer, or to print the question again. With automatic answers enabled, SYSGEN physically checks the hardware configuration of the computer on which it is running. SYSGEN addresses each device to determine its existence and the numbers and types of units. This facility is useful when the system generation is performed on the computer on which RSTS/E will run. Automatic answers can be misleading if the system generation is performed for a different machine.

SYSGEN denotes the meaning of the automatic answer by printing special characters as shown below.

1	The answer 1 is correct for the current hardware.
##08##	The answer assumes something concerning the system.
##??##	The answer cannot be determined; an answer must be entered.

For example, the program can accurately determine whether the computer has either FIS or FPP and thus prints either **Y** or **NO** as the correct answer. However, for certain devices such as DECtape, Magtape, RK disk, and RP disk, SYSGEN can only verify the presence of the controller and must assume the existence of eight drives. The answer for such devices is either **00** or ##08## for the number of drives based upon the absence or presence of the controller. For an example of automatic answers, see Sections 2.6.1 and 2.6.2.

The following are valid responses to automatic answers.

LINE FEED key	Use the automatic answer as the response.
x and RETURN key	Use x as the response.
RETURN key	Reprint query or print long form of the query.

If SYSGEN prints an answer and it appears to be incorrect, the hardware is possibly configured incorrectly. For example, terminal interface wire jumpers are possibly cut improperly. In such circumstances, it is advisable to have a DIGITAL Field Service representative verify the hardware configuration.

Care must be taken if an automatic answer is overridden for a floating address device. The communications devices DJ11, DH11, DQ11, and DU11 have so-called floating addresses. This term means that the presence or absence of any of these devices alters the UNIBUS address assignments for the other devices. Before printing the automatic answer for such a device, SYSGEN recomputes the correct floating addresses based on responses to previous questions. For example, DJ11 multiplexers are assigned addresses on the UNIBUS before DH11 multiplexers. If 1 is the automatic answer for the DJ11 question but is overridden by typing 2, SYSGEN recomputes the floating address of and looks for the presence of a DH11 at an address based on two DJ11 multiplexers. This procedure is correct only if the address jumpers for any DH11 are cut correctly for a system with two DJ11 units. Similarly, if the same automatic answer is overridden by entering 0, SYSGEN recomputes the floating address of and looks for the presence of DH11 devices at an address based on no DJ11. Thus, in situations where the automatic answer is overridden for a floating address device, subsequent automatic answers are possibly incorrect. For more information on floating address assignments, see Appendix G of this document.

2.5.2 Checkpoints

Prior to printing the message to begin RSTS/E system generation, SYSGEN prints a message indicating the start of a checkpoint. The message gives instructions for restarting the system generation if the process must be interrupted for any reason. Any steps performed before the checkpoint need not be redone. Two checkpoint messages appear during the system generation process. The first message immediately precedes the configuration dialogue. The second message immediately precedes the execution of the second batch command file SYSGEN.BAT. Each message describes the correct procedure to restart system generation from the checkpoint.

For example, to abort SYSGEN and terminate the batch stream, type the CTRL/C combination and TE (terminate). One command restarts the process from the related checkpoint. Assume SYSGEN prints a configuration question and the CTRL/C combination is typed.

```
AC FREQ?  ↑C

.
```

The monitor echoes the CTRL/C combination and prints the dot. The user then types TE in response to the dot to terminate the current batch command file, after which the monitor prints the current time, its identification and the $ character. For example,

```
.TE
TIME:-12:36:01
DOS/BATCH V9-20C
$
```

To restart the dialogue from a checkpoint, type the appropriate command as described in the last checkpoint message. For example,

```
$BATCH SYSGN1
```

The monitor begins executing the checkpoint 1 batch command file SYSGN1.

2.6 SYSTEM GENERATION EXAMPLES

This section contains three samples of console printout produced during the generation of RSTS/E. The right hand margin of each sample has references to other sections which contain relevant descriptions. The samples show the system generation to the point where the software automatically bootstraps the resultant RSTS/E system disk and loads the initialization code into memory for the first time.

The first sample shows the output produced by using magtape software with an RP03 disk pack as the system generation disk and an RP03 disk pack as the resultant RSTS/E system disk. The sample shows all configuration questions in long form.

The second sample shows a system generation using disk cartridge (DECpack) software. The system generation disk and the resultant RSTS/E system disk are RK05 disk cartridges. The sample shows short form configuration questions and automatic answers.

The third sample shows a system generation using DECtape software and an RF disk as both the system generation disk and the resultant RSTS/E system disk. The printout shows the points in the procedure where the user must mount and dismount specified DECtapes.

2.6.1 Magtape Software Using Long Form Questions with Automatic Answers

hook bootstrap onto disk

Bootstrap the magtape with the hardware loader.

formatted copy

for bad blocks

no retries

hook

```
SYSLOD V08-01A                                                  (2.2.1)
CONSOLE FILL COUNT=0
DATE: 20-DEC-74                                                 (2.2.3.1)
DIALOGUE?

#DA:MONLIB.CIL/FO/CO:0/HO/BO<MT:MONLIB.LCL

STARTING FORMAT PASS

END FORMAT PASS
SYSLOD COMPLETE
```

tape dn being copied to disk

Only the monlib.cil (asks questions) is now copied

Boot the DOS system after it is on disk

NO

YES

```
ANSWER WITH CARRET OR 'Y' CARRET:- IS YOUR LINE FREQUENCY 50 HERTZ?
DO YOU WANT TO DISABLE DIALOGUE FOREVER? NO
```

'Y' = CARRIAGE RETURN

60 Hz - Europe

```
DOS/BATCH V9-20C                                                (2.4.1)
DATE: 20-DEC-74
TIME: 10:00
DIALOGUE? YES NO
DO YOU WANT TO RESET CONSOLE FILL COUNT? YES
FILL COUNT=0
ARE ANY DEVICES DOWN? NO
DO YOU WANT TO CHANGE LINE PRINTER? YES
LS11? NO
HOW MANY COLUMNS ? 132
LOWER CASE? NO
OVERPRINT? NO
HAVE YOU GOT RK02 DISK? NO
```

TIMING FACTOR ON MAIN CONSOLE

```
$LO 1,1
```

SIGN ON TO DOS SYMBOL

```
DATE:-20-DEC-74                                                 (2.4.2)
TIME:-10:00:20

$RUN MT:PIP

PIP V10-02                                                      (2.4.3)

#SY:<MT:SYSGEN

#^C
 KI

$BATCH SYSGEN
                                                                (2.4.4)
$JOB SYSGEN[1,1]
TIME:-10:00:50
$RUN MT:PIP
$RUN LINK
$RUN PIP
$RUN CILUS
$CH SYSGN3
```

```
SYSGEN:CHECKPOINT ONE:  DURING THE CONFIGURATION DIALOGUE       (2.5.2)
SYSGEN:WHICH FOLLOWS, YOU  MAY  ABORT  SYSGEN  BY  TYPING
SYSGEN:"CONTROL/C" AND THEN "TE". RESTART FROM THE BEGIN-
SYSGEN:NING OF THE DIALOGUE BY TYPING "BATCH SYSGN1".
$RUN SYSGEN

**20-DEC-74**

BEGINNING OF RSTS/E  SYSTEM  GENERATION.                        (2.4.4)

QUESTIONS COME IN LONG AND SHORT  FORMS.                        (2.5.1)
IF  YOU  ARE  FAMILIAR WITH THEM, ANSWER
"S" FOR SHORT; OTHERWISE, ANSWER "L" FOR
LONG FORM.  APPEND "/A" TO ENABLE  AUTO-
MATIC ANSWERS. FORM ?  L/A

THE RSTS/E SYSTEM CAN BE  DISTRIBUTED ON
DECTAPE,MAGTAPE, RK CARTRIDGE DISKS, AND
RP DISK PACKS.  IN THE CASE  OF MAGTAPE,
A DISTINCTION MUST BE MADE  BETWEEN  THE
TU10 (MT) AND  TU16 (MM) MAGTAPE DRIVES.
ENTER THE TYPE OF DISTRIBUTION MEDIA AND
DRIVE TYPE  (IF MAGTAPE)  BEING USED FOR
THIS SYSTEM GENERATION (DT,MT,MM,RK,RP):

MEDIA ? MT

NOW YOU MUST SPECIFY THE  HARDWARE  CON-
FIGURATION  ON  WHICH THIS RSTS/E SYSTEM
WILL RUN.

WILL THIS RSTS/E SYSTEM RUN ON PDP-11/70                        (2.7.1)
HARDWARE (YES OR NO) :

PDP-11/70 ?                       **NO**

WILL  THIS  SOFTWARE  RUN ON A PDP-11/45
COMPUTER WITH FLOATING  POINT  PROCESSOR
(YES OR NO) :

FPP (11/45) ?                     **NO**

WILL THIS SYSTEM RUN ON A PDP-11/40 COM-
PUTER  WITH THE FLOATING INSTRUCTION SET
(YES OR NO) :     <LF>= YES, ACCEPT  AUTOMATIC  ANSWER

FIS (11/40) ?                     **Y **

RSTS/E SYSTEMS RUNNING ON 11/40 OR 11/45
HARDWARE  CAN BE  CONFIGURED  FOR  UP TO
124K WORDS OF MEMORY.  11/70 SYSTEMS CAN
BE CONFIGURED FOR UP TO 1 MILLION(1024K)
WORDS.  YOU MAY INCLUDE ANY MEMORY WHICH
WILL BE ADDED IN THE FORSEEABLE  FUTURE.
ENTER THE  AMOUNT OF MEMORY (IN K WORDS)
YOU  WISH TO  SUPPORT WITH THIS SOFTWARE
(32 TO 1024) :

MAX MEMORY SIZE (K WORDS) ?        **124**
```

```
THE RSTS/E SYSTEM  CLOCK CAN BE EITHER A          (2.7.2)
KW11-L  LINE FREQUENCY CLOCK OR A KW11-P
PROGRAMMABLE CLOCK. THE KW11-P CLOCK HAS
AN INTERNAL CRYSTAL OSCILLATOR WHICH CAN
BE USED AS THE SYSTEM TIME BASE IN AREAS
WHERE  THE  AC POWER  FREQUENCY  IS  NOT
ACCURATE.  IF YOU HAVE THE KW11-L CLOCK,
ANSWER "L".  FOR THE KW11-P, ANSWER  "P"
TO USE THE AC LINE FREQUENCY AS THE TIME
BASE, OR ANSWER  "C"  TO USE THE CRYSTAL
OSCILLATOR (L, P, OR C) :

CLOCK ?                           **L **

THE AC POWER FREQUENCY  IS  NORMALLY  60          (2.7.3)
HERTZ  IN  THE  UNITED STATES, BUT ELSE-
WHERE IT CAN BE 50 HERTZ.  WHAT  IS  THE
AC  POWER FREQUENCY AT WHICH THIS SYSTEM
WILL RUN (50 OR 60) :

AC FREQ ?                         ##60##
```

this question really only applies to P answer to prior question.

```
THE KW11-P 10KHZ CRYSTAL OSCILLATOR WILL          (2.7.4)
BE USED  AS THE SYSTEM  TIME BASE.  THIS
FREQUENCY IS DIVIDED  IN THE HARDWARE TO
PROVIDE THE DESIRED INTERRUPT RATE.  FOR
RSTS/E  THE  INTERRUPT  RATE  CAN BE ANY
MULTIPLE OF 50HZ BETWEEN 50 AND 1000 HZ.
THE LARGER MULTIPLES PROVIDE BETTER TIME
RESOLUTION FOR JOB ACCOUNTING AND SCHED-
ULING  PURPOSES AT THE SMALL EXPENSE  OF
ADDITIONAL  CLOCK  INTERRUPT   OVERHEAD.
PLEASE  ENTER THE INTERRUPT RATE DESIRED
(50,100,150,...,1000) :

KW11P INTERRUPT RATE ?            ##100##

THE NEXT FEW  QUESTIONS  DEAL  WITH  THE
NUMBERS AND TYPES OF TERMINAL INTERFACES
ON THE SYSTEM.

THE  CONSOLE  TERMINAL (KB0) MAY  BE  AN          (2.7.5)
ASR33,  KSR33, ASR35, OR KSR35 TELETYPE;
LA30 (PARALLEL), LA30S (SERIAL), OR LA36
DECWRITER; OR A VT05 (300 BAUD OR LESS),
VT05B (600 BAUD OR MORE), VT50, OR  VT51
SCOPE.  PLEASE IDENTIFY THE TYPE OF CON-
SOLE  TERMINAL  ON  THIS  RSTS/E  SYSTEM
(ASR33, LA36, VT05B, VT51, ETC.) :

CONSOLE TYPE ?                    ##LA30S.## LA36
```

LA30P

Parall

assumed

```
SERIAL ASCII TERMINALS CONNECTED DIRECT-          (2.7.5.1)
LY TO THE COMPUTER AND  THOSE  CONNECTED
THROUGH  LEASED  PRIVATE TELEPHONE LINES
(NOT DIAL-UP)  MAY  USE  EITHER  OF  TWO
CLASSES  OF  SINGLE  LINE INTERFACES (OR
THE DH11 MULTIPLEXER, DISCUSSED  LATER).
THE FIRST CLASS INCLUDES THE KL11, LC11,
DL11A, AND DL11B  INTERFACES.  IF  THERE
ARE  NONE  OF THESE, ANSWER 0; OTHERWISE
WHAT IS THE TOTAL NUMBER OF THESE SINGLE
LINE INTERFACES (1 TO 16 -DO NOT INCLUDE
THE CONSOLE TERMINAL!):

KL11,LC11,DL11A,DL11B'S ?         **04**

THE SECOND CLASS OF SINGLE  LINE  INTER-          (2.7.5.1)
FACES  USED TO CONNECT TERMINALS LOCALLY
OR THROUGH LEASED TELEPHONE  LINES  (NOT
DIAL-UP)  INCLUDES  THE  DL11C AND DL11D
INTERFACES. HOW MANY DL11C'S AND DL11D'S
ARE  ON  THIS SYSTEM  (0 TO 31 -- DO NOT
INCLUDE THE CONSOLE TERMINAL) :

DL11C, DL11D'S ?                  **02**

THERE ARE TWO KINDS OF INDIVIDUAL INTER-
FACES FOR AUTOMATIC ANSWER DATASETS USED
ON THE DIAL-UP TELEPHONE  NETWORK.   THE
FIRST  OF  THESE  IS THE DC11.  HOW MANY
DC11-DA LINES DO YOU HAVE (0 TO 32) :

DC11'S ?                          **02**

THE  SECOND  KIND  OF  INDIVIDUAL  LINE
DIAL-UP INTERFACE IS THE DL11E. HOW MANY
DL11E'S ARE ON THIS SYSTEM (0 TO 31) :

DL11E'S ?                         **01**

THE DJ11 MULTIPLEXER  IS  NOT  SUPPORTED
UNDER RSTS/E.  HOWEVER, TO DETERMINE THE
CORRECT FLOATING  DEVICE  ADDRESSES  FOR
OTHER SUPPORTED HARDWARE, WE NEED TO KNOW
THE  NUMBER  OF  DJ11  MULTIPLEXERS CON-
NECTED TO THIS SYSTEM (0 TO 16) :

DJ11'S ?                          **00**

EACH DH11 MULTIPLEXER CAN CONNECT UP  TO
16 TERMINALS TO THE SYSTEM. PLEASE ENTER
THE TOTAL  NUMBER OF DH11'S ATTACHED  TO
THIS SYSTEM INCLUDING ANY WHICH WILL NOT
BE USED BY THE RSTS/E SOFTWARE (0 TO 16)

DH11'S ?                          **03**

THERE  ARE  02 DM11-BB COMPATIBLE  MODEM
CONTROL MULTIPLEXERS ON THIS SYSTEM.  YOU
HAVE ACCOUNTED FOR 00 SO FAR.  NOTE: ALL
DH11-AD'S INCLUDE DM11-BB MODEM CONTROL.
```

```
DH11 UNIT 00 TYPE ?              ##AA##

THERE ARE FIVE STANDARD DH11 VARIATIONS:

DH11-AA  WILL SUPPORT ANY COMBINATION OF
         LOCAL OR REMOTE LINES. IF AUTO-
         ANSWER DATASETS ARE  TO BE SUP-
         PORTED, A DM11-BB MODEM CONTROL
         MULTIPLEXOR IS ALSO REUUIRED.

DH11-AB  WIRED  FOR TELEGRAPH LINES  AND
         TREATED  AS A  DH11-AA  WITHOUT
         MODEM CONTROL BY RSTS/E.

DH11-AC  240 V,50 HZ VERSION OF DH11-AA.

DH11-AD  CONNECTS 16 EIA/CCITT (RS232-C)
         COMPATIBLE LINES INCLUDING FULL
         MODEM CONTROL.

DH11-AE  SAME AS  DH11-AD WITHOUT  MODEM
         CONTROL. USED FOR CONNECTION OF
         LOCAL EIA TERMINALS OR  PRIVATE
         LINE MODEMS.

PLEASE ENTER THE TYPE OF EACH DH11  UNIT
(AA,AB,AC,AD, OR AE) :

DH11 UNIT 00 TYPE ?              ##AA## AA

DOES DH11 00 INCLUDE A DM11-BB? ##??##

THERE  ARE  02 DM11-BB COMPATIBLE  MODEM
CONTROL MULTIPLEXERS ON THIS SYSTEM. YOU
HAVE ACCOUNTED FOR 00 SO FAR.  NOTE: ALL
DH11-AD'S INCLUDE DM11-BB MODEM CONTROL.

IF ANY DIAL-UP TELEPHONE LINES ARE TO BE
CONNECTED  THROUGH  AN  AUTOMATIC ANSWER
DATASET  TO  THIS DH11 MULIIPLEXER,  THE
DM11-BB MODEM CONTROL IS ALSO  REQUIRED.
DOES THIS  DH11  UNIT INCLUDE A  DM11-BB
(YES OR NO) :

DOES DH11 00 INCLUDE A DM11-BB? ##??## YES

DH11 UNIT 00 LINES ENABLED ?     ##16##

FOR THIS DH11 UNIT,  ENTER THE NUMBER OF
LINES  WHICH WILL  BE USED NOW OR IN THE
FORSEEABLE FUTURE. A RESPONSE OF 16 WILL
PERMIT ALL LINES TO BE USED. IF THE NUM-
BER OF LINES ENABLED (N) IS LESS THAN 16
RSTS/E WILL NOT  BE CONFIGURED  FOR  AND
WILL  NOT  RECOGNIZE  LINES N THRU 15 ON
THIS DH11 UNIT. A 0 RESPONSE IMPLIES THE
WHOLE DH11 UNIT IS TO BE IGNORED.  ENTER
THE NUMBER OF LINES ENABLED (0 TO 16) :

DH11 UNIT 00 LINES ENABLED ?     ##16## 16
```

RECOMMENDED - ENABLE ALL 16
THEN SETKEY CAN DISABLE

```
DH11 UNIT 01 TYPE ?              ##AA## AE.

DH11 UNIT 01 LINES ENABLED ?     ##16## 8

DH11 UNIT 02 TYPE ?              ##AA## AD

DH11 UNIT 02 LINES ENABLED ?     ##16## 0
```

```
THE NEXT  SEVERAL  QUESTIONS  DEAL  WITH
TERMINAL RELATED SOFTWARE FEATURES WHICH
MAY BE INCLUDED IN THE RSTS/E SYSTEM  AT
THE OPTION OF THE SYSTEM MANAGER.

PSEUDO KEYBOARDS PERMIT INTERACTIVE JOBS                   (2.7.6)
TO BE RUN WITHOUT TYING UP A REAL TERMI-
NAL. THEY ARE PRIMARILY INTENDED FOR USE
BY A BATCH CONTROL PROGRAM  WHICH  FEEDS
COMMANDS TO ONE OR MORE PSEUDO KEYBOARDS
DEDICATED TO RUNNING  BACKGROUND  TASKS.
HOW MANY PSEUDO KEYBOARDS WOULD YOU LIKE
TO HAVE (0 TO 63) :

PSEUDO KEYBOARDS ?              ##04##

RSTS/E IS CAPABLE OF SUPPORTING IBM 2741                   (2.7.7)
COMPATIBLE  TERMINALS  ON DL11D,  DL11E,
AND  DC11 SINGLE LINE INTERFACES, OR THE
DH11 MULTIPLEXER. IF YOU DO NOT WANT ANY
2741 SUPPORT THEN ANSWER "NO". OTHERWISE
ANSWER "SL" TO INCLUDE 2741 SUPPORT  FOR
SINGLE LINE INTERFACES ONLY; ANSWER "DH"
FOR 2741 SUPPORT ONLY ON DH11 LINES;  OR
ANSWER "BOTH" TO PROVIDE 2741 SUPPORT ON
BOTH SINGLE LINE INTERFACES AND THE DH11
MULTIPLEXER(S) :

2741'S ?                        ##NO## BOTH

2741 TERMINALS ARE AVAILABLE  WITH  MANY                   (2.7.8)
CODE/KEYBOARD  ARRANGEMENTS. FOUR OF THE
MORE COMMON CODES  ARE  SUPPORTED  UNDER
RSTS/E. THESE INCLUDE IBM CORRESPONDENCE
CODE (STANDARD SELECTRIC TYPEWRITER KEY-
BOARD),  EXTENDED  BINARY CODED DECIMAL,
STANDARD BINARY CODED DECIMAL,  AND CALL
360 BASIC CODE   ANY COMBINATION OF FROM
ONE TO FOUR OF THESE CODE TABLES CAN  BE
INCLUDED IN THE RSTS/E TERMINAL SERVICE.
PLEASE ENTER THE ABBREVIATED CODE  NAMES
(CORR,EBCD,SBCD,C360)  SEPARATED BY COM-
MAS FOR 2741 CODES YOU WISH TO SUPPORT :

2741 CODE(S) ?                  ##CORR. ## EBCD,CORR

AN OPTIONAL FEATURE OF THE RSTS/E TERMI-                   (2.7.9)
NAL SERVICE  ALLOWS  ONE JOB TO INTERACT
WITH SEVERAL TERMINALS  THROUGH  SPECIAL
FORMS  OF THE THE RECORD I/O GET AND PUT
STATEMENTS.  THIS FEATURE IS  USEFUL  IN
APPLICATIONS  WHERE THE SAME BASIC FUNC-
TION IS PERFORMED ON  SEVERAL  TERMINALS
AND A SEPARATE JOB FOR EACH IS UNDESIRE-
ABLE OR AT LEAST INEFFICIENT.  WOULD YOU
LIKE TO INCLUDE THIS FEATURE (YES OR NO)

MULTI-TTY SERVICE ?             ##Y ##
```

```
THE NEXT QUESTIONS DEAL WITH THE NUMBERS
AND KINDS OF DISK UNITS ON THIS SYSTEM.

THE RJS04/RJS03  DISK SYSTEM CONSISTS OF
AN RH11 DEVICE CONTROLLER  AND  FROM ONE
TO EIGHT RS03 (256K) OR RS04 (512K WORD)
FIXED-HEAD DISK DRIVES.  RSTS/E SUPPORTS
ANY  COMBINATION OF RS03'S AND RS04'S UP
TO A TOTAL OF FOUR DRIVES  FOR  SWAPPING
AND  NON-RESIDENT CODE (ADDITIONAL UNITS
MAY BE CONNECTED TO THE  RH11, HOWEVER).
PLEASE ENTER THE NUMBER OF EACH  TYPE OF
DRIVE (0 TO 4) :
```

(2.7.10)

```
RS03'S ?                          **00**
```

~~MUST~~

```
RS04'S ?                          **00**
```

IT KNOWS IF NO CONTROLLER

```
THE RF11 CONTROLLER IS USED  TO  CONTROL
UP TO 8 OF THE RS11 256K WORD FIXED-HEAD
DISKS.  IF THIS SYSTEM HAS NONE OF THESE
DISKS,  ANSWER  0;  OTHERWISE,  HOW MANY
256K WORD RS11 DISK PLATTERS  ARE  THERE
(1 TO 8) :

RF/RS11'S ?                       **00**

IT IS POSSIBLE TO USE THE RC11 DISK CON-
TROLLER WITH UP TO FOUR  RS64  64K  WORD
FIXED-HEAD   DISKS   FOR   SWAPPING  AND
NON-RESIDENT CODE.  IF THERE ARE NONE OF
THESE  DISK  UNITS, ANSWER 0; OTHERWISE,
HOW MANY 64K WORD RS64 DISK PLATTERS ARE
THERE (1 TO 4) :

RC11/RS64'S ?                     **04**
```

YOU MUST TELL HOW MANY DIS

```
THE  RJP04  DISK  SYSTEM  CONSISTS OF AN
RH11 CONTROLLER AND UP TO EIGHT  OF  THE
RP04  40 MILLION WORD  MOVING-HEAD  DISK
PACK DRIVES.  HOW  MANY  RP04 DRIVES ARE
INCLUDED  IN THIS CONFIGURATION (0 TO 8)
NOTE:  IF YOU  HAVE  TWO  OR  MORE  RP04
DRIVES BUT PREFER  TO CONSERVE MEMORY BY
USING THE NON  OVERLAPPED  SEEK  DRIVER,
APPEND "/NO" TO THE NUMBER OF DRIVES :

RP04'S ?                          **00**

THE RP11-C CONTROLLER IS USED TO CONTROL
ANY COMBINATION OF  UP TO EIGHT RP02 (10
MILLION WORD)  OR RP03 (20 MILLION WORD)
MOVING-HEAD DISK PACK DRIVES.  ENTER THE
TOTAL NUMBER  OF RP02 AND RP03 DRIVES ON
THIS SYSTEM (0 TO 8).  NOTE: IF YOU HAVE
TWO OR  MORE DRIVES BUT WANT TO CONSERVE
MEMORY BY USING THE NON OVERLAPPED  SEEK
DRIVER, APPEND "/NO" TO THE NUMBER OF RP
DRIVES :

RP02/RP03'S ?                     ##08## 2
```

~~RK05~~ new dual density non-removable disk— heads are in pack System treats it as 2 packs.

THE RK11 CONTROLLER IS USED TO CONTROL UP TO 8 OF THE RK03 OR RK05 1.2 MILLION WORD MOVING-HEAD CARTRIDGE DISK DRIVES. HOW MANY RK03 OR RK05 DRIVES ARE THERE (0 TO 8). NOTE: IF YOU HAVE TWO OR MORE RK DRIVES BUT WANT TO CONSERVE MEMORY BY USING THE NON OVERLAPPED SEEK DRIVER, APPEND "/NO" TO THE NUMBER OF DRIVES :

RK03/RK05'S ?   ##08## 2/NO

THE "SYSTEM DISK" CONTAINS THE SYSTEM CORE IMAGE AND THE FIRST PART OF THE PUBLIC DISK STRUCTURE. THE SYSTEM DISK CAN BE EITHER THE ~~RF11~~ (RF) FIXED-HEAD DISK, AN ~~RK03~~ OR RK05 (RK) ~~CARTRIDGE~~, OR AN RP02 (RP), RP03 (RP), OR RP04 (RB) DISK PACK. ON THIS SYSTEM WHICH SHALL IT BE (RF, RK, RP, OR RB) :

The system disk must be on drive where you boot.

SYSTEM DISK ?   **RP**

THE RX11 FLOPPY DISK SYSTEM CONSISTS OF A UNIBUS INTERFACE AND THE RX01 FLOPPY DISK SUBSYSTEM. THE RX01 INCLUDES A DISK CONTROLLER AND TWO FLOPPY DISK DRIVES. RSTS/E SUPPORTS UP TO FOUR RX11 SYSTEMS FOR A MAXIMUM OF EIGHT DRIVES. PLEASE ENTER THE NUMBER OF FLOPPY DISK DRIVES ATTACHED TO THIS SYSTEM (0, 2, 4, 6, OR 8):

RX01'S ?   ##??## 0

THE NEXT FEW QUESTIONS DEAL WITH THE PERIPHERAL DEVICES ATTACHED TO THIS RSTS/E SYSTEM.

(2.7.11)

THE TJU16 MAGTAPE SYSTEM CONSISTS OF AN RH11 MASSBUS INTERFACE, TM02 CONTROLLER, AND FROM ONE TO EIGHT TU16 TAPE DRIVES. IF THIS SYSTEM HAS NO TU16 MAGTAPE UNITS THEN ANSWER 0; OTHERWISE, HOW MANY TU16 DRIVES ARE ON THIS SYSTEM (1 TO 8) :

TJ implies mass-bus interface

TU16'S ?   **00**

THE TM11 MAGTAPE CONTROLLER CAN SUPPORT UP TO EIGHT TU10 MAGTAPE DRIVES. PLEASE ENTER THE NUMBER OF TU10 DRIVES ON THIS SYSTEM (0 TO 8) :

TU10'S ?   *##08## 2

It can only see controller

THE TC11 DECTAPE CONTROLLER CAN CONTROL UP TO 8 SINGLE DRIVES. IF THIS SYSTEM HAS NO DECTAPE AT ALL, ANSWER 0; OTHERWISE, HOW MANY SINGLE DECTAPE DRIVES, NOT TU56'S, ARE THERE (1 TO 8 - NOTE THAT EACH TU56 DUAL DRIVE HAS 2) :

DECTAPES ?   ##08## 4

```
THE RSTS/E SYSTEM CAN HAVE  UP TO  EIGHT
LINE  PRINTERS OF THE LP11 OR LS11 TYPE.
THESE PRINTERS ARE REFERRED  TO  BY  THE
DEVICE NAMES LP0: THROUGH LP7:. IF THERE
IS NO LINE PRINTER, ANSWER 0; OTHERWISE,
ENTER  THE  NUMBER OF PRINTERS OF EITHER
TYPE (1 TO 8) :

PRINTERS ?                         **03**

CERTAIN FEATURES OF  EACH  OF  THE  LINE
PRINTERS  ON THE SYSTEM MUST BE DEFINED.
THE NEXT FEW QUESTIONS DEAL  WITH  THOSE
FEATURES.

LP0: TYPE ?                        **LP**
```

CL TO GET LONG FORM QUESTION

```
FOR THE LINE PRINTER DESIGNATED  "LP0:",
PLEASE  DESIGNATE  THE  TYPE OF PRINTER:
"LP11" (DEC STANDARD DRUM PRINTERS),  OR
"LS11" (CENTRONICS SERIAL PRINTER) :

LP0: TYPE ?                        **LP**

LP0: WIDTH ?                       **132**
```

IT ONLY DEPENDS ON PAPER THAT WILL BE USED: 80 OR 132

```
FOR THE LP11 PRINTER DESIGNATED  "LP0:",
ENTER LINE WIDTH - 80 OR 132 CHARACTERS:

LP0: WIDTH ?                       **132**

LP0: LOWER CASE ?                  ##NO##

SOME LP11 PRINTERS HAVE A  96  CHARACTER
SET  INCLUDING  LOWER  CASE  CHARACTERS.
DOES THIS  PRINTER  INCLUDE  LOWER  CASE
(YES OR NO).

LP0: LOWER CASE ?                  ##NO##

LP1: TYPE ?                        **LP**

LP1: WIDTH ?                       **80**

LP1: LOWER CASE ?                  ##NO## YES

LP2: TYPE ?                        **LS**

IF THERE IS A LINE PRINTER AVAILABLE FOR
USE DURING SYSGEN (MUST BE LP0), IT WILL
BE USED TO PRINT THE  SYSTEM  LOAD  MAPS
AND, OPTIONALLY, TO PRINT ASSEMBLY LIST-
INGS OF THE SYSTEM TABLES  AND  TERMINAL
SERVICE  MODULES.  IF YOU HAVE A PRINTER
AND WANT THE  ASSEMBLY  LISTINGS  AND/OR
LOAD MAPS  PRINTED DURING SYSGEN, ANSWER
"YES", OTHERWISE ANSWER "NO".

LP0 FOR SYSGEN ?                   **Y **
```

(2.7.12)

COULD GET COPIES OF:

RSTS - - MAP - maps RSTS monitor
BASIC " " " " RTS
INIT " " " " INIT
OVER - - MAP overlay

THESE ARE OUTPUT TO DISK ANYWAY

```
THE SYSTEM GENERATION  PROCESS  INCLUDES                     (2.7.13)
THE  ASSEMBLY  OF  THE SYSTEM TABLES AND
THE TERMINAL SERVICE  SOFTWARE.   IT  IS
POSSIBLE  TO  HAVE LISTINGS OF THESE TWO
TAILORED MODULES  PRINTED  ON  THE  LINE
PRINTER.   WOULD YOU LIKE TO PRINT THESE
LISTINGS (YES OR NO).

LISTINGS ?                        ##NO##

THREE  CARD  READERS  ARE  AVAILABLE  ON
RSTS/E  SYSTEMS,  THE  CR11 PUNCHED CARD
READER, THE CM11 MARKED CARD READER, AND
THE CD11 HIGH-SPEED PUNCHED CARD READER.
IF THIS SYSTEM  DOES  NOT  HAVE  A  CARD
READER,  ANSWER  'NONE'; OTHERWISE, DOES
IT HAVE A 'CR11', A 'CM11' OR A 'CD11'.

CARD READER ?                     **CR** CD11

THE CARDS READ BY RSTS/E  WILL  NORMALLY                     (2.7.14)
BE INTERPRETED AS BEING IN DEC 029 CODE.
IF THIS INSTALLATION USES CARDS  IN  DEC
026  CODE OR FROM A 1401, YOU MAY SELECT
THE APPROPRIATE SPECIAL DECODING  TABLE.
IF YOU DO NOT WANT THE SPECIAL DECODING,
ANSWER "029"; OTHERWISE,  SHOULD  IT  BE
"026" OR "1401" CODE :

CARD DECODE ?                     ##029## 1401

IS THERE A HIGH-SPEED PAPER TAPE  READER
(YES OR NO).

P.T.READER ?                      **Y **

IS THERE A HIGH-SPEED PAPER  TAPE  PUNCH
(YES OR NO).

P.T.PUNCH ?                       **Y **

THE DQ11  NPR SYNCHRONOUS LINE INTERFACE
IS NOT SUPPORTED UNDER RSTS/E.  HOWEVER,
TO DETERMINE THE CORRECT FLOATING DEVICE
ADDRESSES  FOR OTHER SUPPORTED HARDWARE,
WE NEED TO KNOW HOW MANY DQ11'S ARE CON-
NECTED TO THIS SYSTEM (0 TO 16) :

DQ11'S ?                          **00**
```

ONLY GET THESE IF YOU HAVE 132 CHAR LINE PRINTER - THESE WOULD TAKE AN HOUR AND ARE NOT USUALLY NEEDED.

WATCH DOG TIMER – IF SYSTEM IS GOING DOWN – IT OUTPUTS PAPER TAPE & IMAGE OF UP TRANSMISSIONS FOR NEWSPAPER –

```
ONE DP11 OR DU11 SYNCHRONOUS LINE INTER-
FACE  IS REQUIRED FOR  OPERATION  OF THE
RSTS/E 2780 PACKAGE (IF INCLUDED BELOW).
ADDITIONAL DP11  AND DU11 INTERFACES MAY
BE  CONNECTED  TO THE  UNIBUS  FOR OTHER
PURPOSES.  PLEASE  SPECIFY THE NUMBER OF
THESE DEVICES  ON THIS SYSTEM (INCLUDING
ANY TO BE USED BY THE 2780 PACKAGE) :

DP11'S ?                            **02**
```

RELATES TO 2780 PKG

```
DU11'S ?                            **01**

THE  RSTS/E 2780  PACKAGE  EMULATES  THE
OPERATION  OF  THE IBM 2780 MODEL 1 DATA
TRANSMISSION TERMINAL.  THE PACKAGE PER-
MITS  THE RSTS/E  SYSTEM  TO COMMUNICATE
WITH ANY IBM SYSTEM  WHICH SUPPORTS  THE
DEVICE,OR WITH ANOTHER RSTS 2780 SYSTEM.
THE  2780 PACKAGE  IS SOLD AND SUPPORTED
BY THE DEC  COMMUNICATIONS GROUP.  IT IS
NOT INCLUDED IN THE STANDARD RSTS/E KIT.
DO YOU HAVE THE 2780 PACKAGE AND WISH TO
INCLUDE IT IN THIS SYSTEM (YES OR NO) :
```

(2.7.15)

```
2780 ?                              ##NO## YES

THE 2780 PACKAGE REQUIRES EITHER A  DP11
OR  DU11  SYNCHRONOUS LINE INTERFACE AND
THE  KG11-A  (COMMUNICATIONS  ARITHMETIC
OPTION).  WHICH TYPE OF SYNCHRONOUS LINE
INTERFACE WILL  YOUR  2780  PACKAGE  USE
(DP11 OR DU11) :

2780 INTERFACE ?                    ##DU## DP11

THE FOLLOWING DEVICES ARE NOT  SUPPORTED
BY  THE  RSTS/E SYSTEM SOFTWARE ALTHOUGH
THEY MAY BE ATTACHED TO THE  UNIBUS  FOR
OTHER PURPOSES.
```

(2.7.16)

```
DM11-A  SINGLE-SPEED MULTIPLEXER
DN11-DA AUTOMATIC CALLING UNIT INTERFACE
DR11A,C GENERAL DEVICE INTERFACES
PA611R  TYPESET READER
PA611P  TYPESET PUNCH
DT03-FP PROGRAMMABLE BUS SWITCH
DX11    IBM 360/370 INTERFACE
GT40    GRAPHIC SYSTEM
LPS11   LABORATORY PERIPHERAL SYSTEM
KW11W   WATCHDOG TIMER

ARE ANY OF THESE  NON-SUPPORTED  DEVICES
CONNECTED TO THIS SYSTEM (YES OR NO).

NON-SUPPORTED DEVICES ?             ##Y ## YES
```

```
ENTER THE NUMBER OF DM11-A  SINGLE SPEED
MULTIPLEXERS ON THIS SYSTEM (0 TO 16) :

DM11-A'S ?                      **00**

HOW MANY DN11-DA  AUTOMATIC CALLING UNIT
LINE  INTERFACES ARE  CONNECTED  TO THIS
SYSTEM (0 TO 64) :

DN11-DA'S ?                     **02**

ENTER  THE  TOTAL  NUMBER  OF DR11-A AND
DR11-C GENERAL DEVICE INTERFACES ON THIS
SYSTEM (0 TO 32) :

DR11-A,C'S ?                    **00**

HOW  MANY  PA611R  TYPESET  READERS  ARE
INCLUDED IN THIS CONFIGURATION (0 TO 16)

PA611R'S ?                      **00**

HOW MANY PA611P TYPESET PUNCHES ARE CON-
NECTED TO THIS SYSTEM (0 TO 16) :

PA611P'S ?                      **00**

ENTER THE NUMBER OF DT03-FP PROGRAMMABLE
BUS SWITCHES ON YOUR PDP-11 (0 TO 8) :

DT03-FP'S ?                     **00**

HOW MANY DX11 IBM 360/370 INTERFACES ARE
ATTACHED TO THIS SYSTEM (0 TO 4) :

DX11'S ?                        **00**

ANSWER  "0"  IF YOU HAVE NO GT40 GRAPHIC
SYSTEM HARDWARE. ALSO ANSWER "0" IF YOUR
GT40 HAS ITS OWN PDP-11 CPU AND THE CON-
NECTION TO  THE HOST (RSTS/E) MACHINE IS
THROUGH A STANDARD  TERMINAL  INTERFACE.
ENTER "1" ONLY IF A GT40 GRAPHIC DISPLAY
PROCESSOR IS  CONNECTED DIRECTLY  TO THE
UNIBUS  OF THE  RSTS/E MACHINE  AND  ITS
INTERRUPT VECTORS ARE ASSIGNED ACCORDING
TO DEC STANDARDS FOR FLOATING VECTORS :

GT40 (1 ONLY) ?                 **00**

TYPE "1" IF AN LPS LAB PERIPHERAL SYSTEM
IS ATTACHED TO  THIS  SYSTEM.  OTHERWISE
ENTER "0".

LPS (1 ONLY) ?                  **00**

ENTER "1" IF A  KW11W  WATCHDOG TIMER IS
INCLUDED  IN YOUR CONFIGURATION.  OTHER-
ENTER "0".

KW11W (1 ONLY) ?                **00**
```

```
THE REMAINING QUESTIONS  DEAL  WITH  THE
NAME,  CAPACITY, AND  FEATURES  OF  THIS
RSTS/E  SYSTEM  PROVIDED  AT  THE SYSTEM
MANAGER'S OPTION.

RSTS/E SYSTEMS IDENTIFY THEMSELVES  WITH
A 14 CHARACTER INSTALLATION NAME.  ENTER
UP TO 14 CHARACTERS:    --------------

THE INSTALLATION NAME :          ##SYSTEM TEST## SYSTEM #880
```

(2.7.17)

```
WITH SUFFICIENT HARDWARE RSTS/E CAN HAN-
DLE UP TO 63 SIMULTANEOUS JOBS.  WHAT IS
THE MAXIMUM NUMBER  OF  JOBS (JOBMAX) TO
BE USED AT THIS INSTALLATION (1 TO 63) :

MAXIMUM JOBS ?                   ##20## 32
```

YOU COULDN'T EXCEED THIS W/O DOING ANOTHER SYSGEN.

(2.7.18)

```
SMALL BUFFERS ARE 16 WORD BLOCKS IN MON-
ITOR  STORAGE THAT ARE DYNAMICALLY ALLO-
CATED AS INPUT AND OUTPUT BUFFERS,  FILE
PARAMETER  BLOCKS,  ETC.  STORAGE MAY BE
ALLOCATED FOR  30 TO 999 SMALL  BUFFERS.
THE  RECOMMENDED  NUMBER  IS AT LEAST 10
FOR EACH  POSSIBLE  JOB.  HOW MANY SMALL
BUFFERS WOULD  THIS INSTALLATION LIKE TO
HAVE (30 TO 999) :

SMALL BUFFERS ?                  ##200## 320
```

THERE IS A LATER QUESTION WHERE YOU CAN CUT IT DOWN.

EACH WOULD ADD 5 WORDS.

RELATES TO FIP BUFFERING

SEE 2.7.18

(2.7.19)

```
BIG BUFFERS ARE 256 WORD BLOCKS IN MONI-
TOR  STORAGE  THAT ARE DYNAMICALLY ALLO-
CATED FOR INPUT AND OUTPUT  TO  DECTAPE.
STORAGE MAY BE ALLOCATED FOR FROM 1 TO 8
BIG BUFFERS.  THE RECOMMENDED NUMBER  IS
ONE  FOR EACH DECTAPE DRIVE THAT WILL BE
SIMULTANEOUSLY  IN  USE.  HOW  MANY  BIG
BUFFERS  WOULD THIS INSTALLATION LIKE TO
HAVE (1 TO 8).
```

BIG BUFFERS ARE 1 BLOCK LONG

```
BIG BUFFERS ?                    ##01##
```

(2.7.20)

```
INTER-JOB COMMUNICATION IS  ACCOMPLISHED
THROUGH THE USE OF SEND AND RECEIVE SYS-
TEM FUNCTION CALLS. SEND/RECEIVE CODE IS
NORMALLY  NON-RESIDENT BUT A SMALL TABLE
OF ACTIVE RECEIVERS  IS  RESIDENT.  THIS
TABLE  CONSISTS  OF  FIVE WORDS FOR EACH
RECEIVER ALLOWED TO BE  ACTIVE  SIMULTA-
NEOUSLY  (4 TO 8 RECEIVERS  ARE  USUALLY
SUFFICIENT).  PLEASE ENTER THE NUMBER OF
ACTIVE RECEIVERS ALLOWED AT YOUR INSTAL-
LATION (0 TO JOBMAX) :

RECEIVERS ?                      ##08## 6
```

RECEIVERS FOR:

QUEMAN

BATCH

SPOOL (FOR EACH LP)

2780

(2.7.21)

```
RSTS/E SYSTEMS CAN ATTEMPT TO  AUTOMATI-
CALLY  RECOVER FROM A POWER FAILURE.  DO
YOU WANT THIS CAPABILITY IMPLEMENTED  IN
THIS SYSTEM (YES OR NO) :

POWER FAIL ?                     ##Y ##
```

```
AN OPTIONAL FEATURE OF THE RSTS/E SYSTEM                   (2.7.22)
ALLOWS  THE FILE  PROCESSOR (FIP) TO USE
FREE SMALL OR BIG BUFFERS  TO STORE DISK
DIRECTORY INFORMATION. THIS IMPROVES THE
SPEED OF DIRECTORY  PROCESSING BY AVOID-
ING THE NECESSITY OF  REREADING COMMONLY
USED DIRECTORY INFORMATION.  DO YOU WISH
TO INCLUDE THIS OPTION (YES OR NO) :

FIP BUFFERING ?                    ##Y ##

IF THERE IS SUFFICIENT MEMORY  AVAILABLE                   (2.7.23)
ON THIS SYSTEM, THE MOST FREQUENTLY USED
NON-RESIDENT  DISK  HANDLING CODE CAN BE
MADE RESIDENT IN ORDER TO PROVIDE BETTER
SYSTEM  PERFORMANCE.  SHOULD  THIS  DISK
PROCESSING BE DONE BY RESIDENT CODE (YES
OR NO).

RESIDENT DISK HANDLING ?           **NO** YES
```

YES IF 756KW

```
THE DISPATCH CODE AND TABLES FOR THE FIP
SYSTEM FUNCTION CALLS  (SYS CODE 6)  ARE
NORMALLY  NON-RESIDENT.  OVERALL  SYSTEM
PERFORMANCE  CAN  BE  IMPROVED BY MAKING
THIS CODE RESIDENT.  DO YOU WANT THE SYS
CALL DISPATCH CODE RESIDENT (YES OR NO)

RESIDENT SYS CALL DISPATCH ?       ##NO##

THE ROUTINES WHICH  IMPLEMENT  THE SEND
AND  RECEIVE  SYSTEM FUNCTION CALLS ARE
ALSO NON-RESIDENT. IF YOUR APPLICATIONS
REQUIRE  INTER-JOB  COMMUNICATION,  YOU
MAY WANT SEND/RECEIVE RESIDENT.  PLEASE
ANSWER YES OR NO :

RESIDENT SEND/RECEIVE ?            ##NO##

NON-RESIDENT CODE IS USED TO  LIST  DISK
DIRECTORIES  (CATALOG  COMMAND).  IF YOU
WANT THE DIRECTORY LISTER RESIDENT, TYPE
"YES", OTHERWISE ANSWER "NO".

RESIDENT DIRECTORY LISTER ?        ##NO##
```

(2.7.24)

```
THE CONCISE COMMAND LANGUAGE OPTION  EN-
ABLES  EACH INSTALLATION TO DEFINE A SET
OF KEYBOARD COMMANDS WHICH CALL AND PASS
ONE COMMAND LINE TO COMMONLY USED SYSTEM
PROGRAMS. DO YOU WISH TO INCLUDE THE CCL
OPTION IN THIS SYSTEM (YES OR NO) :

CCL ?                             ##Y ## YES

RSTS/E KITS INCLUDE ONE  STANDARD  TABLE
OF CCL COMMANDS. THESE ARE LISTED IN THE
SYSTEM MANAGER'S GUIDE.  IF THIS SET  IS
ADEQUATE FOR YOUR NEEDS,  ANSWER  'YES';
OTHERWISE, ANSWER 'NEW' TO  CREATE  YOUR
OWN UNIQUE CCL COMMANDS, OR 'ADD' TO ADD
YOUR  OWN  UNIQUE  CCL  COMMANDS  TO THE
STANDARD TABLE :

STANDARD CCL TABLE ?              ##Y ## ADD

TO CREATE A CCL ENTRY,  TYPE THE PROGRAM
NAME AND THE CCL COMMAND (SEPARATED BY A
COMMA). THE PROGRAM NAME IS  1-6  ALPHA-
NUMERIC CHARACTERS;  THE CCL COMMAND MAY
BE UP TO 14 ALPHANUMERIC CHARACTERS.  UP
TO  20  CCL ENTRIES MAY BE DEFINED. TYPE
"/E" TO END YOUR CCL TABLE. TYPE "/R" TO
RESTART THE SERIES OF CCL QUESTIONS :

<PROGRAM>,<COMMAND> ?             ##??## INVENT,ONHAND

<PROGRAM>,<COMMAND> ?             ##??## INVENT,ONORDER

<PROGRAM>,<COMMAND> ?             ##??## INVENT,BACKLOG

<PROGRAM>,<COMMAND> ?             ##??## ORDER,ORDER

<PROGRAM>,<COMMAND> ?             ##??## ORDER,CANCEL

<PROGRAM>,<COMMAND> ?             ##??## ORDER,INQUIRE

<PROGRAM>,<COMMAND> ?             ##??## /END

CCL LISTING ON LP0: ?             ##NO##
```

2.7.24
STANDARD CCL'S

```
FLOATING POINT NUMBERS  ARE  REPRESENTED     (2.7.25)
INTERNALLY  AS  TWO 16-BIT WORDS, GIVING
SEVEN SIGNIFICANT DIGITS. IT IS POSSIBLE
TO MAINTAIN 17 SIGNIFICANT DIGITS BY US-
ING 4 WORDS PER NUMBER.  THE  FOUR  WORD
MATH  PACKAGES  ALSO  INCLUDE THE SCALED
ARITHMETIC FEATURE.  WOULD  THIS INSTAL-
LATION PREFER TO USE 2 OR 4 WORD MATH :

MATH PRECISION ?                 ##2 ## 4

IT IS POSSIBLE TO SAVE SPACE IN THE  BA-     (2.7.26)
SIC-PLUS SYSTEM BY OMITTING THE EXTENDED
FUNCTIONS SIN, COS, SQR, LOG,  ETC.,  IF
THEY  ARE NOT NEEDED.  DOES THIS INSTAL-
LATION NEED TO COMPUTE  THESE  FUNCTIONS
(YES OR NO).

FUNCTIONS ?                      ##Y ##

THE RSTS/E SYSTEM NORMALLY  REPORTS  THE
TIME  OF  DAY  AS  "AM"  OR  "PM".  IT IS
POSSIBLE TO HAVE 24-HOUR  TIME  REPORTED
INSTEAD  (COMMON  FOR EUROPEAN AND MILI-
TARY   INSTALLATIONS).   DO   YOU   WANT
"AM/PM" OR "24-HOUR" TIME REPORTED.

TIME FORMAT ?                    ##24## AM

SPECIAL OUTPUT FORMATTING  CAN  BE  DONE     (2.7.27)
USING   THE   "PRINT  USING"  STATEMENT.
WOULD THIS  INSTALLATION  LIKE  TO  HAVE
THIS OPTIONAL FEATURE (YES OR NO).

PRINT USING ?                    ##Y ##

BASIC-PLUS PERMITS THE USER  TO  OPERATE     (2.7.28)
ON  AN ENTIRE MATRIX USING JUST A SINGLE
STATEMENT.   THESE  STATEMENTS  ARE  THE
"MAT"  STATEMENTS.  WOULD THIS INSTALLA-
TION LIKE TO HAVE THIS OPTIONAL FEATURE.

MATRICES ?                       ##NO## YES

THE STAND-ALONE  PROGRAM  USED  TO  COPY     (2.7.29)
DISKS/TAPES,  ROLLIN, CAN BE INCLUDED IN
THE RSTS/E SYSTEM IMAGE.  IT CAN THEN BE
LOADED  FROM  THE  SYSTEM DISK USING THE
'LOAD' OPTION.  WOULD THIS  INSTALLATION
LIKE TO INCLUDE ROLLIN (YES OR NO).

ROLLIN ?                         ##Y ##
```

Rollin can be alone in CORE itself —

Checkpoint 2   SEE - p. 33

```
SYSGEN:IF YOU HAVE ANY SPECIAL REQUIREMENTS WHICH REQUIRE
SYSGEN:EDITING EITHER THE CONFIGURATION FILE (CONFIG.MAC)
SYSGEN:OR THE BATCH  GENERATION  FILE (SYSGEN.BAT), ABORT
SYSGEN:NOW BY TYPING "CONTROL/C" AND THEN "TE". RESUME AT
SYSGEN:THIS  POINT BY TYPING  "BATCH SYSGN2".  OTHERWISE,
SYSGEN:TYPE "CO" TO CONTINUE WITH SYSTEM GENERATION :

A050 000000
$CO
```

IF you have some non-standard device, requiring CONFIG.MAP, it would be changed at this point —

```
SYSGEN:CHECKPOINT TWO: IF ANY PROBLEMS DEVELOP DURING THE     (2.5.2)
SYSGEN:BATCH GENERATION PROCESS WHICH FOLLOWS, SYSGEN MAY
SYSGEN:BE ABORTED BY TYPING "CONTROL/C" AND "TE". RESTART
SYSGEN:FROM THIS POINT BY TYPING "BATCH SYSGN2".

$CH SYSGEN.BAT                                                (2.8)
$RUN PIP
$RUN MACRO
$RUN MACRO
$RUN PIP
```

THESE ARE PRINTED AND RUN DURING

```
SYSGEN:REMOVE DEC-11-ORSPA-D-XXX FROM MAGTAPE UNIT 0 TEMPORARILY
SYSGEN:MOUNT  DEC-11-ORCOA-C-MC9 OR MC7 (2780 PACKAGE) ON UNIT 0
SYSGEN:WITH NO "WRITE RING" AND SET TO "ONLINE".
SYSGEN:TYPE "CO" WHEN READY

A050 000000
$CO

$RUN PIP

SYSGEN:REMOVE  DEC-11-ORCOA-C-MC9 OR MC7 FROM MAGTAPE DRIVE 0 AND
SYSGEN:REMOUNT DEC-11-ORSPA-D-XXX ON MAGTAPE DRIVE 0
SYSGEN:WITH NO "WRITE RING" AND SET TO "ONLINE".
SYSGEN:TYPE "CO" WHEN READY

A050 000000
$CO

$RUN PIP
$RUN MACRO
$RUN EDIT
$RUN MACRO
$RUN PIP
$RUN PIP
$RUN LINK

$RUN PIP
$RUN LINK

$RUN PIP
$RUN LINK

$RUN PIP
$RUN CILUS

SYSGEN:REMOVE DEC-11-ORSPA-D-XXX AND MOUNT A NEW MAGTAPE
SYSGEN:WITH "WRITE RING" ON UNIT 0. SET TO "ON LINE" AND
SYSGEN:TYPE "CO" WHEN READY

A050 000000
$CO
```

RETURN TO STREAM BY TYPING BATCH SYSGEN2

```
$RUN PIP

SYSGEN:WHEN SYSLOD IDENTIFIES ITSELF,
SYSGEN:MOUNT A NEW DISK ON UNIT RP UNIT 0,
SYSGEN:READY AND WRITE ENABLE THE DRIVE.
SYSGEN:RESPOND AS FOLLOWS TO THE PROMPTING:
SYSGEN:CONSOLE FILL COUNT = <RETURN>
SYSGEN:DATE:                <DD-MMM-YY>
SYSGEN:DIALOGUE?            <RETURN>
SYSGEN:#DP:/NS:512:25/FO/TO:4/HO/BO<MT:RSTS.LCL
$RUN CILUS

SYSLOD V08-01A
CONSOLE FILE COUNT=
DATE: 20-DEC-74
DIALOGUE?

#DP:/NS:512:25/FO/TO:4/HO/BO<MT:RSTS.LCL

STARTING FORMAT PASS

END FORMAT PASS
SYSLOD COMPLETE



RSTS V06A-02 SYSTEM #880

OPTION:
```

(Chapter 3)

FORMAT DISK

REPOSITION BOOTSTRAP CODE TO AVOID OVERWRITING

device clustersize in words

cluster where CIL is located (block 249)

hook

BOOT AFTER loaded

clustering

WRITES CIL ON DISK FROM MT

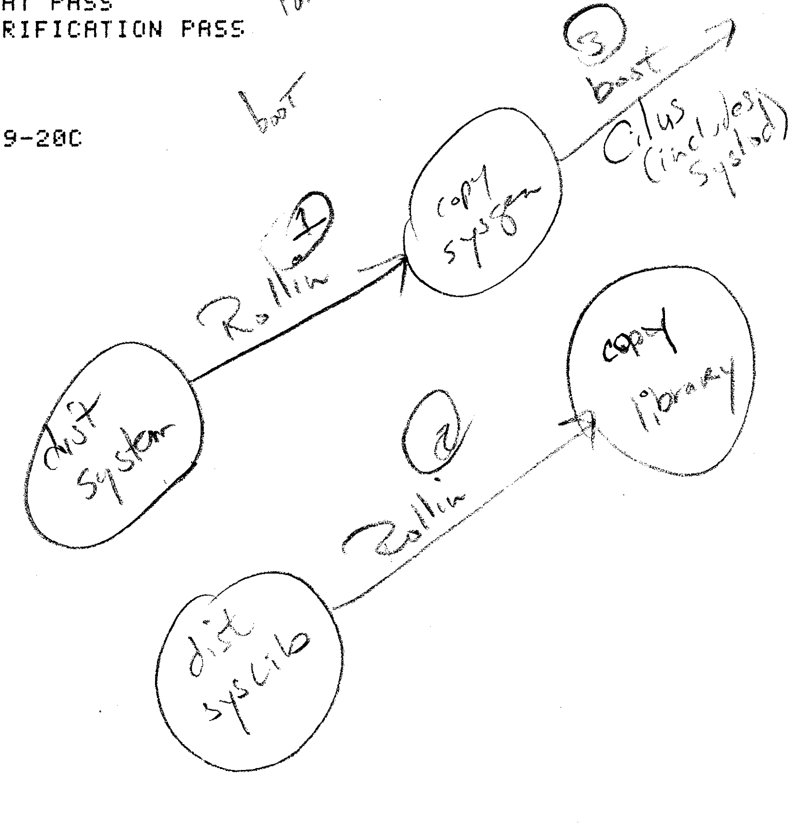
TREAD FROM DISK TO CORE, BRING UP SYSTEM

2.6.2 Disk Cartridge Software Using Short Form Questions with Automatic Answers

Bootstrap ~~the~~ system generation disk cartridge using the hardware loader.

ROLLIN TO COPY DISKS

```
DOS/BATCH V9-20C                                          (2.3.1)
DATE: 20-DEC-74        boot a DOS system                  (2.3.2)
TIME: 10:00
DIALOGUE? CR

$LO 1,1                sign on to DOS acct
DATE:-20-DEC-74
TIME:-10:00:20

$RUN CILUS ——— pgm of DOS symbol
CILUS V08-06A

#ROLLIN.CIL/BO   boot Rollin

ROLLIN V07
DF1     DP0
DK1:/FO<DK0:/VE     copy disk

STARTING RK FORMAT PASS       Format + verify
END RK FORMAT PASS
STARTING VERIFICATION PASS

#DK1:/FO<DK0:/VE     ~~copy~~ verify

STARTING RK FORMAT PASS       FORMAT
END RK FORMAT PASS
STARTING VERIFICATION PASS

#/BOOT:DP:            boot

DOS/BATCH V9-20C
DATE:
```



START HERE

THIS PAGE FORMATS DISK WHICH WILL BECOME SYSGEN DISK

BOOTSTRAP ADDRESSES

DECTAPE 773030

CASSETTE 773230

RK DISK 773010

PAPERTAPE 773312

(2.3.3)

```
DOS/BATCH V9-20C
DATE: 20-DEC-74
TIME: 10:00
DIALOGUE?

$LO 1,1
DATE:-20-DEC-74
TIME:-10:00:20

$RUN CILUS
CILUS V08-06A

#SYSLOD.CIL/BO

SYSLOD V08-01A
CONSOLE FILL COUNT=4
DATE: 20-DEC-74
DIALOGUE?

#DB0:/FO

STARTING FORMAT PASS

END FORMAT PASS
#DP0:/FO

STARTING FORMAT PASS

END FORMAT PASS
#DK0:/FO

STARTING FORMAT PASS

END FORMAT PASS
#
```

1st copy disk (dm [illegible])

Ensure that all disks are mounted and ready before answering the DIALOGUE question.

2. Ld adrs

1. Toggle bootstrap address

3. enable

4. start

```
DOS/BATCH V9-20C                                                   (2.4.1)
DATE: 20-DEC-74
TIME: 10:00
DIALOGUE? YES
DO YOU WANT TO RESET CONSOLE FILL COUNT? YES
FILL COUNT=4
ARE ANY DEVICES DOWN? NO
DO YOU WANT TO CHANGE LINE PRINTER? YES
LS11? NO
HOW MANY COLUMNS ? 132
LOWER CASE? NO
OVERPRINT? NO
DO YOU WANT TO CHANGE CARD READER DEFAULTS? NO
HAVE YOU GOT RK02 DISK? NO

$LO 1,1                                                            (2.4.2)

DATE:-20-DEC-74
TIME:-10:00:20

$BATCH SYSGEN                                                      (2.4.4)
```

run sysgen

```
$JOB SYSGEN[1,1]
TIME:-10:00:30
$CH SYSGN3

SYSGEN:CHECKPOINT ONE:  DURING THE CONFIGURATION DIALOGUE          (2.5.2)
SYSGEN:WHICH FOLLOWS, YOU  MAY  ABORT  SYSGEN  BY  TYPING
SYSGEN:"CONTROL/C" AND THEN "TE". RESTART FROM THE BEGIN-
SYSGEN:NING OF THE DIALOGUE BY TYPING "BATCH SYSGN1".
$RUN SYSGEN
```

GO TO 2-24

```
**13-JUN-75**

BEGINNING OF RSTS/E  SYSTEM  GENERATION.

QUESTIONS COME IN LONG AND SHORT  FORMS.                      (2.5.1)
IF  YOU  ARE  FAMILIAR WITH THEM, ANSWER
"S" FOR SHORT; OTHERWISE, ANSWER "L" FOR
LONG FORM.  APPEND "/A" TO ENABLE  AUTO-
MATIC ANSWERS. FORM ?   S/A

MEDIA ? RK  RP

PDP-11/70 ?                   **NO**

FPP (11/45) ?                 **NO**                          (2.7.1)

FIS (11/40) ?                 **NO**

MAX MEMORY SIZE (K WORDS) ?   **64** 124

CLOCK ?                       **P ** C                        (2.7.2)

KW11P INTERRUPT RATE ?        ##100##                         (2.7.4)

CONSOLE TYPE ?                ##LA30S.## VT05B                (2.7.5)

KL11,LC11,DL11A,DL11B'S ?     **02**                          (2.7.5.1)

DL11C, DL11D'S ?              **00**

DC11'S ?                      **00**

DL11E'S ?                     **01**

DJ11'S ?                      **00**

DH11'S ?                      **01**                          (2.7.5.2)

THERE  ARE  01 DM11-BB COMPATIBLE  MODEM
CONTROL MULTIPLEXERS ON THIS SYSTEM. YOU
HAVE ACCOUNTED FOR 00 SO FAR.  NOTE: ALL
DH11-AD'S INCLUDE DM11-BB MODEM CONTROL.

DH11 UNIT 00 TYPE ?           ##AA##

DOES DH11 00 INCLUDE A DM11-BB? ##??## YES

DH11 UNIT 00 LINES ENABLED ?  ##16##

PSEUDO KEYBOARDS ?            ##04##                          (2.7.6)

2741'S ?                      ##NO## DH                       (2.7.7)

2741 CODE(S) ?                ##CORR.## C360                  (2.7.8)

MULTI-TTY SERVICE ?           ##Y ##                          (2.7.9)
```

```
RS03'S ?                        **00**                              (2.7.10)

RS04'S ?                        **00**

RF/RS11'S ?                     **01**

RP04'S ?                        **00**

THE  RJP04  DISK  SYSTEM  CONSISTS OF AN
RH11 CONTROLLER AND UP TO EIGHT  OF  THE
RP04  40 MILLION WORD  MOVING-HEAD  DISK
PACK DRIVES.  HOW  MANY  RP04 DRIVES ARE
INCLUDED  IN THIS CONFIGURATION (0 TO 8)
NOTE:  IF YOU  HAVE  TWO  OR  MORE  RP04
DRIVES BUT PREFER  TO CONSERVE MEMORY BY
USING THE NON  OVERLAPPED  SEEK  DRIVER,
APPEND "/NO" TO THE NUMBER OF DRIVES :

RP04'S ?                        **00**
                                  02
RP02/RP03'S ?                   **01**

RK03/RK05'S ?                   ##08## 2

SYSTEM DISK ?                   **RP**

RX01'S ?                         ##??##  0

TU16'S ?                        **00**                              (2.7.11)

TU10'S ?                        ##08## 2                            (2.7.19)

DECTAPES ?                      ##08## 4

PRINTERS ?                      **01**

LP0: TYPE ?                     **LP**

LP0: WIDTH ?                    **132**

LP0: LOWER CASE ?               ##NO## YES

LP0 FOR SYSGEN ?                **Y **                              (2.7.12)

LISTINGS ?                      ##NO##                              (2.7.13)

CARD READER ?                   **CR**

CARD DECODE ?                   ##029## 026                         (2.7.14)

P. T. READER ?                  **Y **

P. T. PUNCH ?                   **Y **
```

```
DQ11'S ?                        **00**

THE DQ11  NPR SYNCHRONOUS LINE INTERFACE
IS NOT SUPPORTED UNDER RSTS/E.  HOWEVER,
TO DETERMINE THE CORRECT FLOATING DEVICE
ADDRESSES  FOR OTHER SUPPORTED HARDWARE,
WE NEED TO KNOW HOW MANY DQ11'S ARE CON-
NECTED TO THIS SYSTEM (0 TO 16) :

DQ11'S ?                        **00**

DP11'S ?                        **02**

DU11'S ?                        **01**

2780 ?                          ##NO##

THE  RSTS/E 2780  PACKAGE  EMULATES  THE                        (2.7.15)
OPERATION  OF  THE IBM 2780 MODEL 1 DATA
TRANSMISSION TERMINAL.  THE PACKAGE PER-
MITS  THE RSTS/E  SYSTEM  TO COMMUNICATE
WITH ANY IBM SYSTEM  WHICH SUPPORTS  THE
DEVICE,OR WITH ANOTHER RSTS 2780 SYSTEM.
THE  2780 PACKAGE  IS SOLD AND SUPPORTED
BY THE DEC  COMMUNICATIONS GROUP.  IT IS
NOT INCLUDED IN THE STANDARD RSTS/E KIT.
DO YOU HAVE THE 2780 PACKAGE AND WISH TO
INCLUDE IT IN THIS SYSTEM (YES OR NO) :

2780 ?                          ##NO## YES

2780 INTERFACE ?                ##DU## DP11

NON-SUPPORTED DEVICES ?         ##Y ## YES                      (2.7.16)

DM11-A'S ?                      **00**

DN11-DA'S ?                     **00**

DR11-A,C'S ?                    **00**

PA611R'S ?                      **00**

PA611P'S ?                      **00**

DT03-FP'S ?                     **00**

DX11'S ?                        **00**

GT40 (1 ONLY) ?                 **00**

LPS (1 ONLY) ?                  **01**

KW11W (1 ONLY) ?                **00**
```

```
THE INSTALLATION NAME :          ##SYSTEM TEST## SYSTEM #1136

MAXIMUM JOBS ?                   ##20## 24                       (2.7.17)

SMALL BUFFERS ?                  ##200## 240                     (2.7.18)

BIG BUFFERS ?                    ##01##                          (2.7.19)

RECEIVERS ?                      ##08## 6                        (2.7.20)

POWER FAIL ?                     ##Y ##                          (2.7.21)

FIP BUFFERING ?                  ##Y ##                          (2.7.22)

AN OPTIONAL FEATURE OF THE RSTS/E SYSTEM
ALLOWS  THE FILE  PROCESSOR (FIP) TO USE
FREE SMALL OR BIG BUFFERS  TO STORE DISK
DIRECTORY INFORMATION. THIS IMPROVES THE
SPEED OF DIRECTORY  PROCESSING BY AVOID-
ING THE NECESSITY OF  REREADING COMMONLY
USED DIRECTORY INFORMATION.  DO YOU WISH
TO INCLUDE THIS OPTION (YES OR NO) :

FIP BUFFERING ?                  ##Y ##

RESIDENT DISK HANDLING ?         **NO** YES                      (2.7.23)

RESIDENT SYS CALL DISPATCH ?     ##NO##

RESIDENT SEND/RECEIVE ?          ##NO##

RESIDENT DIRECTORY LISTER ?      ##NO##

CCL ?                            ##Y ## YES                      (2.7.24)

STANDARD CCL TABLE ?             ##Y ## YES

CCL LISTING ON LP0: ?            ##NO##

MATH PRECISION ?                 ##2 ## 4                        (2.7.25)

FUNCTIONS ?                      ##Y ##                          (2.7.26)

TIME FORMAT ?                    ##24## AM

PRINT USING ?                    ##Y ##                          (2.7.27)

MATRICES ?                       ##NO## YES                      (2.7.28)

ROLLIN ?                         ##Y ##                          (2.7.29)
```

```
SYSGEN:IF YOU HAVE ANY SPECIAL REQUIREMENTS WHICH REQUIRE
SYSGEN:EDITING EITHER THE CONFIGURATION FILE (CONFIG.MAC)
SYSGEN:OR THE BATCH  GENERATION  FILE (SYSGEN.BAT), ABORT
SYSGEN:NOW BY TYPING "CONTROL/C" AND THEN "TE". RESUME AT
SYSGEN:THIS  POINT BY TYPING  "BATCH SYSGN2".  OTHERWISE,
SYSGEN:TYPE "CO" TO CONTINUE WITH SYSTEM GENERATION :
A050 000000
```

A0 ~~[illegible]~~

```
$CO

SYSGEN:CHECKPOINT TWO: IF ANY PROBLEMS DEVELOP DURING THE        (2.5.2)
SYSGEN:BATCH GENERATION PROCESS WHICH FOLLOWS, SYSGEN MAY
SYSGEN:BE ABORTED BY TYPING "CONTROL/C" AND "TE". RESTART
SYSGEN:FROM THIS POINT BY TYPING "BATCH SYSGN2".
$CH SYSGEN.BAT

S257 000002

SYSGEN:MOUNT DEC-11-ORCOA-C-HC DISK (2780 PACKAGE) ON RK DRIVE 1. (2.8)
SYSGEN:READY AND WRITE PROTECT THE DRIVE.
SYSGEN:TYPE "CO" WHEN READY
A050 000000

$CO

$RUN MACRO
$RUN MACRO
$RUN PIP

SYSGEN:REMOVE DEC-11-ORCOA-C-HC FROM RK DRIVE 1

$RUN MACRO
$RUN LINK
$RUN LINK
$RUN LINK
$RUN PIP
$RUN CILUS
```

build CILS copy of
on
DISK DN

```
SYSGEN:MOUNT A FORMATTED RK CARTRIDGE ON DRIVE 1
SYSGEN:SET TO "RUN" AND WRITE ENABLE
SYSGEN:TYPE 'CO' WHEN READY TO LOAD THE NEW SYSTEM
A050 000000

$CO

$RUN CILUS
```

WRITE LICK ON CARTRIDGE

```
SYSGEN:REMOVE   DEC-11-ORSPA-D-HC  FROM  RK DRIVE 0
SYSGEN:MOVE THE RSTS/E SYSTEM DISK FROM  RK DRIVE 1
SYSGEN:TO DRIVE 0. BOOT NORMALLY TO BRING UP RSTS/E
$FI
TIME:-11:04:20
```

Boots now

```
DOS/BATCH V09-20C
$

RSTS V06A-02 SYSTEM #1136                                        (Chapter 3)

OPTION:
```

2.6.3 DECtape Software Using Short Form Questions

Bootstrap the DECtape using the hardware loader.

```
SYSLOD V08-01A                                                    (2.2.2)
CONSOLE FILL COUNT=0                                              (2.2.3)
DATE: 20-DEC-74
DIALOGUE?                                                         (2.2.3.1)

#DF:MONLIB.CIL/CO:0/HO/BO<DT0:MONLIB.LCL

SYSLOD COMPLETE

ANSWER WITH CARRET OR 'Y'CARRET:- IS YOUR LINE FREQUENCY 50 HERTZ?
DO YOU WANT TO DISABLE DIALOGUE FOREVER? NO

DOS/BATCH V9-20C
DATE: 20-DEC-74                                                   (2.4.1)
TIME: 12:30
DIALOGUE? YES
DO YOU WANT TO RESET CONSOLE FILL COUNT? YES
FILL COUNT=0
ARE ANY DEVICES DOWN? NO
DO YOU WANT TO CHANGE LINE PRINTER? YES
LS11? YES
OVERPRINT? NO
HAVE YOU GOT RK02 DISK? NO

$LO 1,1                                                           (2.4.2)

DATE:-20-DEC-74
TIME:-12:30:25

$RUN DT0:PIP                                                      (2.4.3)

PIP V10-02

#SY:<DT1:SYSGEN

#^C
.KI

$BATCH SYSGEN                                                     (2.4.4)

$JOB SYSGEN[1,1]
TIME:-12:31:46
$RUN DT0:PIP
$RUN LINK

$RUN PIP
$CH SYSGN3
```

```
SYSGEN:CHECKPOINT ONE:  DURING THE CONFIGURATION DIALOGUE        (2.5.2)
SYSGEN:WHICH FOLLOWS, YOU  MAY  ABORT  SYSGEN  BY  TYPING
SYSGEN:"CONTROL/C" AND THEN "TE". RESTART FROM THE BEGIN-
SYSGEN:NING OF THE DIALOGUE BY TYPING "BATCH SYSGN1".
$RUN SYSGEN

**20-DEC-74**

BEGINNING OF RSTS/E  SYSTEM  GENERATION.

QUESTIONS COME IN LONG AND SHORT  FORMS.                         (2.5.1)
IF  YOU  ARE  FAMILIAR WITH THEM, ANSWER
"S" FOR SHORT; OTHERWISE, ANSWER "L" FOR
LONG FORM.  APPEND "/A" TO ENABLE  AUTO-
MATIC ANSWERS. FORM ?   S

MEDIA ? DT
PDP-11/70 ? NO                                                   (2.7.1)

FPP (11/45) ? YES

MAX MEMORY SIZE (K WORDS) ? 96

CLOCK ? L                                                        (2.7.2)

AC FREQ ? 60                                                     (2.7.3)

CONSOLE TYPE ? LA30                                              (2.7.5)

KL11,LC11,DL11A,DL11B'S ? 4                                      (2.7.5.1)

DL11C, DL11D'S ? 0

DC11'S ? 0

DL11E'S ? 4

DJ11'S ? 0

DH11'S ? 0                                                       (2.7.5.2)

PSEUDO KEYBOARDS ? 4                                             (2.7.16)

2741'S ? NO                                                      (2.7.7)

MULTI-TTY SERVICE ? YES                                          (2.7.9)
```

```
RS03'S ?

THE RJS04/RJS03  DISK SYSTEM CONSISTS OF          (2.7.10)
AN RH11 DEVICE CONTROLLER  AND  FROM ONE
TO EIGHT RS03 (256K) OR RS04 (512K WORD)
FIXED-HEAD DISK DRIVES.  RSTS/E SUPPORTS
ANY  COMBINATION OF RS03'S AND RS04'S UP
TO A TOTAL OF FOUR DRIVES  FOR  SWAPPING
AND  NON-RESIDENT CODE (ADDITIONAL UNITS
MAY BE CONNECTED TO THE  RH11, HOWEVER).
PLEASE ENTER THE NUMBER OF EACH  TYPE OF
DRIVE (0 TO 4) :

RS03'S ? 0

RS04'S ? 0

RF/RS11'S ? 2

RP04'S ? 0

RP02/RP03'S ? 0

RK03/RK05'S ? 2

SYSTEM DISK ? RF

RX01'S ? 0

TU16'S ? 0                                        (2.7.11)

TU10'S ? 0

DECTAPES ? 2

PRINTERS ? 1

LP0: TYPE ? LS

LP0 FOR SYSGEN ?                                  (2.7.12)

IF THERE IS A LINE PRINTER AVAILABLE FOR
USE DURING SYSGEN (MUST BE LP0), IT WILL
BE USED TO PRINT THE  SYSTEM  LOAD  MAPS
AND, OPTIONALLY, TO PRINT ASSEMBLY LIST-
INGS OF THE SYSTEM TABLES  AND  TERMINAL
SERVICE  MODULES.  IF YOU HAVE A PRINTER
AND WANT THE  ASSEMBLY  LISTINGS  AND/OR
LOAD MAPS  PRINTED DURING SYSGEN, ANSWER
"YES", OTHERWISE ANSWER "NO".

LP0 FOR SYSGEN ? YES
                                                  (2.7.13)
LISTINGS ? NO

CARD READER ? NO

P.T.READER ? NO
```

```
DQ11'S ? 0

DP11'S ?

ONE DP11 OR DU11 SYNCHRONOUS LINE INTER-
FACE  IS REQUIRED FOR  OPERATION  OF THE
RSTS/E 2780 PACKAGE (IF INCLUDED BELOW).
ADDITIONAL DP11  AND DU11 INTERFACES MAY
BE  CONNECTED  TO THE  UNIBUS  FOR OTHER
PURPOSES.  PLEASE  SPECIFY THE NUMBER OF
THESE DEVICES  ON THIS SYSTEM (INCLUDING
ANY TO BE USED BY THE 2780 PACKAGE) :

DP11'S ? 1

DU11'S ? 2

2780 ? YES                                                      (2.7.15)

2780 INTERFACE ?

THE 2780 PACKAGE REQUIRES EITHER A  DP11
OR  DU11  SYNCHRONOUS LINE INTERFACE AND
THE  KG11-A  (COMMUNICATIONS  ARITHMETIC
OPTION).  WHICH TYPE OF SYNCHRONOUS LINE
INTERFACE WILL  YOUR  2780  PACKAGE  USE
(DP11 OR DU11) :

2780 INTERFACE ? DU11

NON-SUPPORTED DEVICES ? YES                                     (2.7.16)

DM11-A'S ? 0

DN11-DA'S ? 0

DR11-A,C'S ? 0

PA611R'S ? 0

PA611P'S ? 0

DT03-FP'S ? 0

DX11'S ? 0

GT40 (1 ONLY) ? 0

LPS (1 ONLY) ? 1

KW11W (1 ONLY) ? 0
```

```
THE INSTALLATION NAME : SYSTEM #372
MAXIMUM JOBS ? 24                                              (2.7.17)
SMALL BUFFERS ? 240                                            (2.7.18)
BIG BUFFERS ? 1                                                (2.7.19)
RECEIVERS ? 6                                                  (2.7.20)
POWER FAIL ? YES                                               (2.7.21)
FIP BUFFERING ? YES                                            (2.7.22)
RESIDENT DISK HANDLING ? YES                                   (2.7.23)
RESIDENT SYS CALL DISPATCH ? NO
RESIDENT SEND/RECEIVE ? NO
RESIDENT DIRECTORY LISTER ? NO
CCL ? YES                                                      (2.7.24)
STANDARD CCL TABLE ? NEW
<PROGRAM>,<COMMAND> ? PIP,PIP
<PROGRAM>,<COMMAND> ? QUE,QUE
<PROGRAM>,<COMMAND> ? PIP,HELP
<PROGRAM>,<COMMAND> ? TTYSET,SET
<PROGRAM>,<COMMAND> ? /E
CCL LISTING ON LP0: ? NO
MATH PRECISION ? 4                                             (2.7.25)
FUNCTIONS ? Y                                                  (2.7.26)
TIME FORMAT ? AM
PRINT USING ? YES                                              (2.7.27)
MATRICES ? NO                                                  (2.7.28)
ROLLIN ? YES                                                   (2.7.29)
```

```
SYSGEN:IF YOU HAVE ANY SPECIAL REQUIREMENTS WHICH REQUIRE
SYSGEN:EDITING EITHER THE CONFIGURATION FILE (CONFIG.MAC)
SYSGEN:OR THE BATCH  GENERATION  FILE (SYSGEN.BAT), ABORT
SYSGEN:NOW BY TYPING "CONTROL/C" AND THEN "TE". RESUME AT
SYSGEN:THIS  POINT BY TYPING  "BATCH SYSGN2".  OTHERWISE,
SYSGEN:TYPE "CO" TO CONTINUE WITH SYSTEM GENERATION :

A050 000000
$CO

SYSGEN:CHECKPOINT TWO: IF ANY PROBLEMS DEVELOP DURING THE       (2.5.2)
SYSGEN:BATCH GENERATION PROCESS WHICH FOLLOWS, SYSGEN MAY
SYSGEN:BE ABORTED BY TYPING "CONTROL/C" AND "TE". RESTART
SYSGEN:FROM THIS POINT BY TYPING "BATCH SYSGN2".
$CH SYSGEN.BAT

SYSGEN:REMOVE DEC-11-ORSBA-D-UC1 FROM DECTAPE DRIVE 0           (2.8)
SYSGEN:REMOVE DEC-11-ORSBA-D-UC2 FROM DECTAPE DRIVE 1
SYSGEN:MOUNT  DEC-11-ORSEA-D-UC1  ON  DECTAPE DRIVE 0
SYSGEN:MOUNT  DEC-11-ORCOA-C-UC   ON  DECTAPE DRIVE 1 (2780 PACKAGE)
SYSGEN:SET TO "REMOTE" AND "WRITE LOCKED"
SYSGEN:TYPE "CO" WHEN READY

A050 000000
$CO

$RUN PIP
$RUN MACRO
$RUN MACRO
$RUN PIP
$RUN PIP
$RUN MACRO
$RUN EDIT
$RUN MACRO
$RUN PIP

SYSGEN:REMOVE DEC-11-ORSEA-D-UC1 FROM DECTAPE DRIVE 0
SYSGEN:REMOVE DEC-11-ORCOA-C-UC  FROM DECTAPE DRIVE 1
SYSGEN:MOUNT  DEC-11-ORSEA-D-UC2  ON  DECTAPE DRIVE 0
SYSGEN:MOUNT  DEC-11-ORSEA-D-UC3  ON  DECTAPE DRIVE 1
SYSGEN:SET BOTH TO "REMOTE" AND "WRITE LOCKED"
SYSGEN:TYPE "CO" WHEN READY

A050 000000
$CO

$RUN PIP
$RUN LINK

$RUN PIP
$RUN PIP
$RUN LINK

$RUN PIP
$RUN PIP
$RUN LINK

$RUN PIP
$RUN PIP
```

```
SYSGEN:REMOVE DEC-11-ORSEA-D-UC2 FROM DECTAPE DRIVE 0
SYSGEN:REMOVE DEC-11-ORSEA-D-UC3 FROM DECTAPE DRIVE 1
SYSGEN:MOUNT A NEW FORMATTED DECTAPE ON UNIT 0
SYSGEN:SET TO "REMOTE" AND "WRITE ENABLED"
SYSGEN:TYPE "CO" WHEN READY

A050 000000
$CO

$RUN PIP
$RUN CILUS
$RUN PIP

SYSGEN:WHEN SYSLOD IDENTIFIES ITSELF,
SYSGEN:RESPOND AS FOLLOWS TO THE PROMPTING:
SYSGEN:CONSOLE FILL COUNT = <RETURN>
SYSGEN:DATE:               <DD-MMM-YY>
SYSGEN:DIALOGUE?           <RETURN>
SYSGEN:#DF:/NS:256:49/TO:4/HO/BO<DT:RSTS.LCL

$RUN CILUS

SYSLOD V08-01A
CONSOLE FILE COUNT=
DATE: 20-DEC-74
DIALOGUE?

#DF:/NS:256:49/TO:4/HO/BO<DT:RSTS.LCL

SYSLOD COMPLETE



RSTS V06A-02 SYSTEM #372

OPTION:
```

(Chapter 3)

2.7 CONFIGURATION QUESTION CONSIDERATIONS

The questions printed during the SYSGEN dialogue concern the hardware configuration parameters and software options. Those parameters and options requiring more explanation than available in the long form of the question are explained in this section. The explanations appear in the order in which the program prints the related questions. Appendix D of this manual lists the memory requirements for most hardware and software available.

2.7.1 Central Processor Unit and Options

The first three hardware questions concern the PDP-11/70 hardware and the type of floating point processor (if any). The response to the PDP-11/70 question determines the type of memory parity error handling code to be assembled into the Tables (TBL) module. (The PDP-11/70 parity memory is not compatible with the parity memory used on PDP-11/40 and PDP-11/45 processors.)

The answers to the FPP, FIS, MATH PRECISION, and FUNCTION questions determine which mathematical package is included in the BASIC-PLUS Run Time System. See Section 2.7.25 for a description of the different mathematical packages.

The response to the MAX MEMORY SIZE question determines the size of a system table (1 word for each 1K section of memory). To avoid regenerating the system, any memory to be added later should be accounted for in this response.

2.7.2 Clocks

The RSTS/E system can operate with one of two types of system clocks. The KW11-L Line Time Clock divides time into intervals based on the line frequency of the power source - either 50 Hz or 60 Hz. (The designation Hertz (Hz) is the international standard of measurement for cycles per second.) The KW11-P Programmable Real Time Clock can also employ the line frequency of the power source. However, the KW11-P has a crystal-controlled oscillator which can be used as the system time base independent of the power source frequency. The crystal operation KW11-P is beneficial in areas where the line frequency of the power source is not constant.

The answer to the CLOCK question determines which type of clock RSTS/E will use and, for the KW11-P, which mode of operation (line frequency or crystal) is desired. If line frequency is used for either clock, SYSGEN prints the AC FREQ question and omits the KW11-P INTERRUPT RATE question. If the crystal mode of operation is indicated, SYSGEN omits the AC FREQ question but prints the KW11-P INTERRUPT RATE question.

2.7.3 AC Power Frequency

The PDP-11 computer requires alternating current (AC) power input and is able to run on 60 Hz frequency, as is standard in the United States, or on 50 Hz frequency, as is standard in many other countries. If the computer operates with the KW11-P clock at line frequency or with the KW11-L clock, SYSGEN prints the AC FREQ question. The answer to the question must be either 50 or 60, depending on the standard frequency of the power input.

2.7.4 KW11-P Interrupt Rate

The KW11-P 10 KHz crystal-controlled oscillator allows time to be divided to a desired interrupt rate given as times per second. 100 is the recommended interrupt rate for all RSTS/E systems.

2.7.5 Terminals and Terminal Interfaces

2.7.5.1 Keyboard Numbers and Interfaces - The RSTS/E system is designed to handle a maximum of 64 terminals. Each terminal is assigned a keyboard number ranging from 0 to 63. The console terminal is given the keyboard number 0 on all RSTS/E systems and is referenced by the device designator KB0:. SYSGEN requires the user to enter the type of console terminal so that the RSTS/E terminal service and the terminal drivers in the initialization code and in ROLLIN can add the correct number of fill characters.

The assignment of a keyboard number, other than that of the console terminal, is determined by the type of line interface to which the terminal is attached. The local installation can have any combination of local and remote line interfaces as long as the total number of terminal lines and pseudo keyboards does not exceed 63, not including the console terminal. The assignment is important because the system manager must specify the terminal speeds allowed on each of the variable speed lines in the system based on the keyboard number. (See the description of the TTYSET.SPD file in Section 7.4.1 of the RSTS/E System Manager's Guide.)

The order in which RSTS/E assigns keyboard numbers is as follows: the system console terminal; all KL11, LC11, DL11A, and DL11B lines; DL11C and DL11D lines; PK (pseudo keyboards); DC11 (remote dial); DL11E (remote dial); and DH11 lines in increasing order of unit number and increasing order of configured (enabled) lines within each unit.

2.7.5.2 Interface Considerations - The answers to the configuration questions concerning the number of each type of terminal interface must accurately reflect the hardware configuration. For example, DL11C and DL11D interfaces are similar in construction and operation to the DL11A and DL11B interfaces. However, the interfaces have different ranges of UNIBUS addresses. Because of this similarity, improper configuration of the terminal interfaces at both the hardware and software level is a common problem for new installations. It is recommended that the system manager rely on the automatic answer facility to verify the terminal interface configuration.

If the number of DH11 multiplexers specified is non-zero, SYSGEN prints several questions for each unit. The first question requires that the type of DH11 be specified. The second question is printed only when the type of DH11 is capable of employing a separate DM11-BB modem control. The final question requests the number of lines to enable on the DH11 unit.

The DH11-AA and DH11-AC multiplexers are identical except for power requirements. The DH11-AA and DH11-AC can be wired for mixtures of automatic answer, local EIA, and local 20-milliampere lines. Both types require the DM11-BB modem control multiplexer if automatic answer data sets are to be connected to either unit. Because the DM11-BB modem control multiplexer is optional on the DH11-AA and DH11-AC, SYSGEN inquires about its presence.

The DH11-AB multiplexer is identical to the DH11-AA but is wired for telegraph and telex lines. Modem control is meaningless for telegraph service and SYSGEN does not print the question concerning the DM11-BB. RSTS/E does not provide specific support for telegraph or telex line disciplines. The DH11-AB is treated as a DH11-AA without modem control.

The DH11-AD and DH11-AE multiplexers are functionally identical to a DH11-AA unit. The DH11-AD multiplexer includes modem control on all 16 lines and therefore implements 16 EIA/CCITT (RS232-C) compatible lines with full data set control. This modem control is functionally identical to the DM11-BB. The DH11-AE multiplexer is wired for 16 EIA local lines without modem control. Neither unit allows the mixture of line types as the DH11-AA and DH11-AC units do. For DH11-AD and DH11-AE multiplexers, SYSGEN does not print the question concerning the DM11-BB.

The answers concerning the type of DH11 and the presence of a DM11-BB must accurately reflect the hardware ordering of the devices. The DH11 and the DM11-BB both have floating vectors; the DH11 is also a floating address device. The presence or absence of floating address and vector devices can affect the correct address and vector assignments of other hardware. The SYSGEN unit numbering assumes increasing UNIBUS addresses. For example, if DH11 unit 00 has base address xxxxxx depending on other floating address devices present, unit 01 has the address assignment xxxxxx+20.[1]

DM11-BB units, like DH11 units, are numbered according to increasing UNIBUS addresses. DM11-BB addressing begins at 770500 for unit 0, is followed by unit 1 at 770510, and continues in increments of 10 (octal). DM11-BB interrupt vectors are assigned in the floating vector space in the order of increasing unit numbers.

For RSTS/E to operate properly, the lowest numbered DM11-BB unit must be connected to the lowest numbered DH11 unit which is to provide modem control. The next higher DM11-BB unit must be connected to the next higher DH11 unit which provides modem control. The sequence continues for any remaining DM11-BB units. For example, DM11-BB unit 0 can be connected to DH11 unit 2 provided that DH11 units 0 and 1 do not include modem control. By the same reasoning, connecting DM11-BB unit 2 to DH11 unit 0 is always illegal.

---

[1] For more information on floating address and vector assignment, see Appendix G of this manual.

If automatic answers are enabled, the SYSGEN program counts the number of DM11-BB modem control multiplexers found on the system. SYSGEN can not determine which DM11-BB is attached to which DH11. Therefore, the indeterminate answer (**??**) is given for the DM11-BB question. The system manager must determine from the hardware configuration which DH11 units include a DM11-BB and he must report the presence or absence of the unit in response to the DM11-BB question. If automatic answers are enabled, typing either the RETURN or LINE FEED key in response to the DM11-BB question causes SYSGEN to print the number found. SYSGEN verifies that the number found equals the number reported present. If the two numbers are not equal, SYSGEN prints a question allowing the system manager to reconsider his answers. If he does reconsider, SYSGEN restarts the terminal-related questions. Otherwise, SYSGEN uses the number of DM11-BB units reported present (not the number found) and continues the dialogue.

The last question for each DH11 unit is the number of lines to enable.[1] SYSGEN permits a total of 63 single line interfaces, enabled DH11 lines, and pseudo keyboards. (The console keyboard is always enabled.) If more than 63 lines are configured, SYSGEN repeats all the terminal related questions.

Each line of a DH11 need not be enabled. Therefore, RSTS/E can be configured for less than the maximum number of 16 lines physically present on each DH11. Less than 16 lines are enabled when memory must be conserved or when the limit of 63 lines must not be exceeded. For example, if four DH11 units are present, at most 63 of the 64 possible lines can be enabled. If N lines (where N is less than 16) are enabled for a DH11, SYSGEN does not enable lines numbered N through 15 on the DH11. Lines not enabled will not be available for use and will be treated as nonexistent. Thus, when 10 lines are enabled, lines 0 through 9 will be operational and lines 10 through 15 on that DH11 cannot be used without regenerating the system.

If possible, all DH11 lines should be enabled during the system generation so that unused lines can be enabled in the future without regenerating the system. The SYSGEN questions do not specify which enabled DH11 lines are actually to be used or which lines are to have modem control enabled. The SYSGEN questions are supplemented by the SETKEY initialization option which allows any keyboard line to be temporarily disabled and which must be used to enable modem control.

---

[1] The SETKEY initialization option also enables DH11 lines but only those lines configured (enabled) at SYSGEN time. Refer to Section 3.5 for operational details.

2.7.6 Pseudo Keyboards

Any number of pseudo keyboards (up to 63) can be configured into the system. Each copy of the BATCH system program requires one pseudo keyboard to run user jobs. If the installation plans to run several copies of BATCH simultaneously, at least that number of pseudo keyboards must be specified. One additional pseudo keyboard is necessary for each application to be implemented using the pseudo keyboard feature described in Section 4.5 of the RSTS/E Programming Manual.

2.7.7 2741 Terminals

RSTS/E supports 2741 compatible terminals connected to DL11D, DL11E, and DC11 single line interfaces or a DH11 multiplexer. 2741 terminals normally use the RS232 EIA standard connection; hence, they can be connected locally to the computer through null modems to any of the above named interfaces or can be connected through data sets or acoustic couplers for operation over telephone lines.

2741 support can be included for single line interfaces only (DL11D, DL11E or DC11); for DH11 multiplexer lines only; or for both single line interfaces and DH11 multiplexer lines.

2.7.8 2741 Code and Keyboard Arrangements

RSTS/E supports four code and keyboard arrangements as shown in Section 5.8 of the RSTS-11 System User's Guide. The system manager must define the code or codes he wants his system to support so that they are assembled into the terminal service module. He can include any combination of one to four of the four supported codes.

If more than one code is configured, a small routine is included in the terminal service to enable a user at a 2741 terminal to change codes. The first code named in response to the configuration question is assigned as the system default. For example, if both Correspondence and EBCD 2741 terminals are to be supported but EBCD is the preferred default, type EBCD, CORR in response to the 2741 CODE question.

2.7.9 Multiple Terminal Service

The multiple terminal service option allows one BASIC-PLUS program to simultaneously interact with several terminals on one I/O channel.

With the multiple terminal service on the system, a BASIC-PLUS program can control several keyboards by establishing a master keyboard on a single channel and by reserving various other keyboards as slave terminals. To perform input or output, the program executes standard Record I/O statements on that single channel and utilizes special software options to determine the unique keyboard being serviced. Additionally, in absence of keyboard input, the system can automatically stall program execution and make the program eligible to run when keyboard input is pending from a particular keyboard or any keyboard controlled by the program. Thus, a single program can interactively service low volume keyboard input and output associated with several stations. This procedure eliminates the need to run separate copies of the same program at each terminal when several terminals must perform a similar function. The option is explained in Section 4.1 of the RSTS/E Programming Manual.

2.7.10 Disk Devices

Disks in the RSTS/E system operate in either the public or private structure. The disk which contains the system accounts and executable code of RSTS/E is called the system disk and is the first disk of the public structure. All other disks in the system are referred to collectively as non-system disks.

The most practical use of the system disk on the RSTS/E system is as a removable disk as opposed to a fixed head disk. If the system disk is either an RK05 or RK03 disk cartridge or an RP02, RP03 or RP04 disk pack, it can be removed from the computer area when the system is not operating and kept in a safe place, thereby reducing the chances of inadvertent or malicious destruction. To preserve the contents of a fixed head disk, a copy must be transferred to a secondary medium each time the system is shut down.

Optimum performance is obtained if the system is configured with a moving head disk (removable) and an auxiliary swapping disk (fixed head). With such a configuration, the swapping of user jobs into and out of memory is faster and more efficient. Disk accessing operations on the moving head system device can then be confined to manipulating user files and directories while the faster fixed head device takes on the burden of moving user jobs into and out of memory. In such a case, the auxiliary swapping disk acts as a logical extension of the system disk while the system is operating but contains no valuable system data when the system is not operating. At the start of time sharing operations, the initialization code creates the necessary files on the auxiliary swapping device.

If a large amount of public storage space is required on the system, a single RP02, RP03 or RP04 disk pack drive is preferable to multiple RK05 or RK03 DECpack disk drives. The use of multiple devices in the public structure increases the overhead required in file operations, since, each time a file is created within the public structure, the directories of each public device must be searched to ensure that a file of the same name does not already exist. An alternative method of structuring the disks to reduce system overhead in configurations using multiple drives is to use private disks rather than public disks. In systems having several types of disk devices, the decision is left to the system manager as to which device is to be used as the system disk and which devices are used in the public and private disk structure.

The RK11, RP11-C, and RH11 controllers allow several drives to perform seek operations simultaneously. Since the controller is not busy during seek operations, data transfers can be overlapped with seeks in progress on other drives. On systems with two or more drives on the same controller, SYSGEN normally configures an overlapped seek driver to accelerate processing on the related disk subsystem.

Since the overlapped seek drivers require more memory than the non-overlapped drivers, SYSGEN allows the system manager to include the non-overlapped version in place of the standard overlapped driver. The advantages in throughput afforded by the overlapped driver normally outweigh the disadvantages of the additional memory required. Therefore, the overlapped drivers are recommended for all systems with multiple drives on the same controller. To have SYSGEN substitute a non-overlapped seek driver, append /NO to the number specified in response to the questions concerning the number of drives.

A standard address and vector have been defined only for one floppy disk system (two drives). If the RX11 floppy disk is on the system and the standard address and vector assignments are used, simply type the number of drives in response to the RX01 question. SYSGEN then uses the standard address and vector for RX11 unit 0 (drives 0 and 1). If non-standard address and vector assignments are to be used, append the switches /A:x and /V:y to the number of drives. Give the non-standard

16-bit address in place of x and the non-standard vector in place of y in the switches. The following sample shows the procedure.

```
RX01's            ##??##    2/A:177170/V:264
```

2.7.11 Peripheral Devices

The use of peripheral devices in the RSTS/E system reduces the burden of storage requirements on the disk devices and provides a convenient medium for file archives. Program and data files that are not frequently used can be stored on magnetic tape (DECtape or magtape), paper tape (high speed, fan-folded or Teletype), and cards (marked or punched) and accessed readily when required. DECtape and magtape provide a large capacity storage medium for critical file information.

Production of hard copy output on the RSTS/E system can be improved by the use of multiple line printers, each having distinct operating characteristics which must be specified during system generation. A hardware modification can be made to a line printer to inhibit the form advance (automatic page ejection) every 66 lines and thus allow the user flexible control over vertical formatting. This feature is made available to users by the MODE option in the OPEN statement as described in Section 3.2 of the RSTS/E Programming Manual.

2.7.12 LP0: for SYSGEN

During system generation, four files (load maps) containing memory allocation information are created. For magtape and DECtape distribution media, the files are written to the resultant LICIL tape which contains the RSTS/E system. For disk cartridge distribution media, the files are retained on the system generation disk. The load maps are standard ASCII text files with filenames RSTS.MAP (monitor), OVR.MAP (overlay code), INIT.MAP (initialization code), and BASIC.MAP (Run Time System).

Since the load maps contain information essential to problem diagnosis, they should be printed at some time. If line printer unit 0 is available for use, SYSGEN can print the load maps during system generation whether or not a line printer is configured into the RSTS/E system. If line printer unit 0 is not available or if the user prefers to perform the printing at a later time using the PIP program, he can omit the system generation printing. The load maps are ~~retained on~~ the LICIL tape and on the system generation disk.

To print the load maps during system generation when the line printer is available, type YES in response to the LP0: FOR SYSGEN question. As a result, the four load maps are automatically printed later in the system generation process. Printing takes about 10 minutes on a 300 line per minute printer. For magtape and DECtape distribution media, a directory of the LICIL tape is also printed at unit 0 rather than at the console terminal. When the line printer is available, SYSGEN prints the question concerning listings described in Section 2.7.13.

To omit printing the load maps during system generation, type NO in response to the LP0: FOR SYSGEN question. As a result, the load maps are not printed but the files are available for later printing as described above. For magtape and DECtape distribution media, the directory of the LICIL tape is printed at the console terminal rather than at the line printer. In addition, SYSGEN does not print the question concerning listings described in Section 2.7.13.

2.7.13 Listings

During the system generation process, the system tables and terminal service modules are assembled. An additional assembly is performed if concise command language entries are created as described in Section 2.7.24.

Since each of the assemblies is unique to the installation, the assembly listings provide information valuable for documentation and maintenance purposes. If the LP0: FOR SYSGEN question is answered with YES, SYSGEN prints the LISTINGS question to allow the listings to be printed during the system generation process. If the answer to the LP0: FOR SYSGEN question is NO, SYSGEN does not print the LISTINGS query. In certain circumstances, the listings can be printed at a later time.

The ability to print the listings at a later time depends on whether or not the listing files TBL.LST (system tables) and TTY.LST (terminal service modules) are retained. If the system generation disk is an RF type, the files are not retained because they can not physically fit on the disk. Therefore, the listings must be printed during system generation or they can never be printed. If the system generation disk is either an RK cartridge or RP pack, the files are retained on that disk. If magtape media is used, the files are retained on the LICIL tape created during system generation. The files can be printed from the LICIL magtape by using the PIP program under either DOS or RSTS/E. The files do not fit on one DECtape and are therefore not retained on the LICIL DECtape.

To print the listings at system generation time, simply type YES in response to the LISTING and CCL LISTING questions. As a result, the listings are automatically printed later in the system generation process. The listings are quite lengthy and take approximately 30 minutes to print on a 300 line per minute printer. To omit printing the listings, type NO to the LISTING question.

2.7.14 Card Codes

If the RSTS/E system has a card reader, the user must configure one of three card codes. These card codes are presented for reference in Appendix D.3 of the BASIC-PLUS Language Manual. The standard card code is DEC029 code.

2.7.15 2780 Package

The 2780 capability on RSTS/E allows a user to communicate with IBM 360 and 370 and other DIGITAL PDP-11 computers having the 2780 Data Terminal capability described in the RSTS/E 2780 User's Guide. The RSTS/2780 software is not included in the standard RSTS/E distribution media but is sold and supported as a separate product. If the 2780 capability is to be included in the system, the 2780 software must be available on the same medium as the RSTS/E distribution media. Messages printed during the standard system generation tell when to mount the tape or disk containing the RSTS/2780 software. The RSTS/E computer must have the KG11A communications arithmetic unit and either a DP11 or a DU11 synchronous line interface unit. If answers to the SYSGEN questions indicate that neither a DP11 nor DU11 interface is present, the question concerning the 2780 package is omitted.

2.7.16 Non-Supported Devices

Devices not supported by RSTS/E may be attached to the computer on which time sharing operations are conducted. Although the system cannot address these devices, their existence must be known so that proper floating address and floating vector assignments can be made. If a positive answer is given to the non-supported device question, SYSGEN prints each device name and requires the user to type the number present on the system.

2.7.17 Maximum Number of Jobs

With sufficient hardware, RSTS/E can handle up to 63 simultaneous jobs. The maximum number of jobs must be specified at system generation time since this parameter determines the size of several monitor tables. The number can be adjusted downward when the JOB OR SWAP MAX CHANGES query is answered as described in Chapter 3. The maximum number of jobs can not be increased above the configured maximum unless the system code is regenerated.

The maximum number of jobs which can be run efficiently depends on the memory space available and the number and types of disks on the system. Memory space requirements are defined in Appendix D. To calculate the effect of disk devices, consult the material in Chapter 3 concerning swapping.

Jobs on the system are numbered sequentially from one to the maximum number the system can handle. Jobs include both those attached and detached.

2.7.18 Small Buffers

The RSTS/E system handles transfer requests and file processing requests by means of intermediate memory storage, called small buffers. These buffers are considered a system resource, and a sufficient number must be configured at system generation time. If an insufficient number is configured, jobs running on the system can become stalled waiting until enough buffers are freed by jobs currently claiming their use.

Small buffers are 16-word blocks residing in the Monitor part of memory. The number needed by a system at any one time depends upon the dynamic requirements of the jobs on the system. For efficient system operation, it is recommended that at least 10 small buffers be allocated for each possible job. Thus, on a 32-user system, 320 small buffers should be available. (An indicator of good system performance is that the number of free small buffers, as reported by the SYSTAT system program, never drops below ten. Refer to the description of SYSTAT in Section 7.2 of the RSTS/E System Manager's Guide.

On systems configured for 16 or more jobs, 10 small buffers for each job is usually an adequate number. However, on systems configured for fewer than 16 jobs, it may be advantageous to include more than 10 small buffers per job. For example, each active terminal requires 4 or 5 small buffers for performing input and output operations. A system having 8 terminals therefore needs between 32 and 40 small buffers if all terminals are to be simultaneously active. Each active job requires three small buffers. Thus, if the same system required that ten jobs be able to run simultaneously, 30 more small buffers would be needed. For each job that will run detached from all terminals, subtract the four small buffers required for terminal I/O. A running total on the 8-terminal, 10-job system is 62 small buffers for these two simple processing requirements.

Next, in the sample system, consider what kind of processing is necessary. One small buffer must be added for each open file on the system. If each program running on the system opens two disk files, 20 more small buffers must be added. If all the active programs open the maximum number of files simultaneously, 120 small buffers must be available. (BASIC-PLUS allows 12 open channels per user program.) In an average system, the two file situation is much more likely, so the sample system requires 20 more small buffers for a total of 82.

The system requires small buffers for certain transient operations. One small buffer is used for each disk transfer queued by the monitor. Programs such as QUEMAN, QUE, ERRCPY, RJ2780, BATCH, and SPOOL communicate through the SEND/RECEIVE system function call which claims one small buffer for each message queued for a receiving job. Normally the buffers used for these operations are employed for very short time periods (fractions of a second) but a reasonable number of buffers should be available. On the 10 job, 8 terminal system, a reasonable number is approximately 20 more small buffers.

Line printers on the system exhaust as many as one third of the available small buffers. A lower number of available small buffers places a larger burden on the system. For example, if 30 small buffers are available for use by the printer driver, the system can have 900 characters buffered for output to the line printer. Assume that a line printer is running at 300 lines per minute (5 lines per second) and that an average line is 90 characters. Such a line printer empties the buffers in two seconds. A spooling program for that line printer would have to be swapped into memory every 2 seconds to keep the line printer running at full speed. (For a line printer running at 1200 lines per minute, a swap operation would be necessary every one half second.)

The total requirement on the 10 job, 8 terminal system is nearly 140 small buffers for an average system load if the transient requirement for 20 is added and one third of the subtotal is added to handle the line printer running full speed (300 lines per minute). Thus, the guideline of 10 small buffers per job is too low on such a small system. Moreover, if small buffers were subtracted from 140 to account for idle terminals and detached jobs and to allow for some slow down in line printer operations, the guideline is still inadequate. On such a system, between 12 and 14 small buffers per job is a better approximation. For larger systems having 16 or more jobs, ten small buffers per job is usually a good approximation. Except for occasions of heavy keyboard and line printer activity, enough free small buffers will be available to maintain good system throughput.

2.7.19 Big Buffers

Big buffers are 256-word blocks of monitor memory used for DECtape and 2780 operations. Since the SYSGEN program automatically configures four big buffers for 2780 software, big buffers need be configured only if DECtape is on the system. Consequently, SYSGEN prints the BIG BUFFER question only if DECtape is on the system.

On systems without 2780 software but with DECtape devices, one big buffer is normally provided for each DECtape drive. However, one big buffer per drive is not a definite requirement since one big buffer can accommodate any number of DECtape drives for non-simultaneous operations. Experience indicates that, unless DECtape usage is heavy, two big buffers are sufficient even for four drives. Three big buffers are recommended for six drives and four big buffers are recommended for eight DECtape drives.

On systems with both RSTS/2780 software and DECtape drives, one big buffer is sufficient for eight DECtape units. The four big buffers automatically included for RSTS/2780 software are available for DECtape operations when the RSTS/2780 software is not operating. In any case, the SYSGEN program does not allow less than one big buffer when DECtape is present on the system. This fact applies whether or not the 2780 software is present.

2.7.20 Receivers

A receiver is a job capable of performing interjob communication using the SEND/RECEIVE system function call described in Section 7.2.12 of the RSTS/E Programming Manual. The QUEMAN job and each SPOOL and BATCH job running on the system performs as a receiver. Enough receivers must be configured for such jobs and for additional programs which declare themselves receivers. For example, on a system running one SPOOL program and one BATCH program, configure at least three receivers (QUEMAN, SPOOL, and BATCH).

2.7.21 Power Fail Recovery Code

RSTS/E systems can attempt to recover from a momentary power failure by performing an automatic restart procedure. A momentary power failure is defined in Section 5.3 of the PDP-11/70 Processor Handbook and in Section 2.7.1 of both the PDP-11/40 Processor Handbook and PDP-11/45 Processor Handbook.

2.7.22 File Processor (FIP) Buffering and Disk Caching

The optional FIP buffering module accelerates file processing on the RSTS/E system. The module reduces the number of accesses to the disk by maintaining more than one disk directory block in memory.

Normally, RSTS/E employs only one permanently allocated 256-word buffer to store disk directory information. File processing is often delayed because, at any given time, the directory block which the file processor needs is not the one currently in its buffer.

The FIP buffering module claims, from the normal buffer pool, free small or big buffers and stores additional directory blocks in the buffers. Frequently used blocks are thus kept in memory to avoid accesses to the disk. If any of the buffers are required for their standard uses, the system releases them from the FIP buffering module. The number of buffers configured for other system operations need not be increased when FIP buffering is included on the system.

The system manager can enhance FIP buffering on RSTS/E by allocating memory solely for the module. FIP buffering is like cache memory operation wherein faster access semi-conductor memory holds frequently used code or data to avoid accesses to slower core memory. FIP buffering is a disk caching mechanism which maintains frequently used disk directory blocks in memory to avoid accesses to the slower access disk. The CACHE command in the TABLE OPTION portion of the DEFAULT and START options can reserve a fixed amount of memory for FIP buffering.

Memory reserved for disk caching is exhausted by the FIP buffering module before small buffers are employed. The reserved memory thus prevents degradation of performance on a heavily loaded system when small buffers are most likely to be claimed by user jobs and are most needed for FIP buffering. (See Section 3.6.3.7 for more information on disk caching and the CACHE command.)

The FIP buffering module increases the size of the monitor and is therefore an option on all RSTS/E systems. (See Appendix D for information on the size of optional modules.)

2.7.23 Resident Code

Certain portions of the overlay code can be configured resident in memory during time sharing rather than stored on the disk. Performance is thereby improved on systems where that code has heavy usage.

The most important code in terms of system performance is the disk handling code. This block of code includes: the system routines for disk file creation; OPEN, CLOSE, and RENAME routines; some routines needed to process the RUN command; general routines for opening other devices, deassigning devices, fetching error messages, looking up files by name; and for logout. It is recommended that systems with more than 56K words of memory have disk handling code resident.

The SYS call dispatch code is used whenever a program executes a SYS call to the file processor (FIP). The code is called to dispatch to other non-resident or resident portions of monitor code. It is recommended that the SYS call dispatch code be made resident on any system having more than 64K words of memory and using FIP calls for many programs.

The SEND/RECEIVE code provides inter-job communications for such programs as QUEMAN, BATCH, SPOOL, and RJ2780. SEND/RECEIVE code is entered through the SYS call dispatch code. It is not recommended that this code be configured resident unless system operations include frequent usage of programs executing the SEND/RECEIVE SYS call and the related SYS call dispatch code is also made resident.

The directory lister code is executed for CATALOG command operations and gathers information concerning disk directories. It is small but is less important than the disk handling, SYS call dispatch, and SEND/RECEIVE code. Since the system program DIRECT is more efficient than the CATALOG command, it is recommended that DIRECT be used wherever possible. Hence, little reason exists to make this code resident.

2.7.24 Concise Command Language (CCL)

The SYSGEN program allows the user to include or omit the CCL capability described in the introductory material of Chapter 4 in the RSTS-11 System User's Guide. This capability provides a set of keyboard commands which run programs stored in the system library. If the CCL capability is included, one of three options is possible: accept a standard set of commands, define a new set of commands, or add new commands to the standard set.

The standard set of CCL commands involves the following programs and consists of the related commands.

DIRECT	DIR
EDIT	EDIT
EDIT	CREATE
PIP	HELP
PIP	PIP
QUE	QUE
SYSTAT	SYS

TTYSET	SET
UMOUNT	MOUNT
UMOUNT	DISMOUNT
COBOL	COBOL
COBOL	CBL
SORT11	SORT
SORT11	SRT

The programs listed are automatically included in the system library and are coded so that the standard CCL commands function properly.

SYSGEN allows a unique set of commands to be defined by either adding to the standard set or creating a completely new set. The programs defined to run by CCL commands must conform to standards described in Chapter 8 of the RSTS/E Programming Manual. The system manager can add 6 new commands or can create a new set containing as many as 20 commands. To define the commands, simply type the program name and the command separated by a comma. To change the standard set of CCL commands, a new set must be entered which includes any desired commands from the standard set and any additional commands required. To restrict usage of CCL commands on the system, adjust the protection code of the related program.

The precedence of CCL commands is above that of RSTS/E commands and BASIC-PLUS immediate mode statements. As a result, the system manager can control the use of a command or an immediate mode statement on his system. For example, he could specify a command BYE which performs certain operations before allowing a user to log off the system. As another example, he could specify a command PRINT which performs operations different from those of the BASIC-PLUS immediate mode PRINT statement. Such a CCL command has no effect on a BASIC-PLUS statement preceded by a line number since a line numbered statement has a higher precedence than the CCL command. Care must be taken not to unintentionally override system commands. For example, if DEL is defined as a CCL command, the BASIC-PLUS DELETE command will be overridden.

The same program can be run by more than one CCL command. A single inventory control program might be designed to handle single line requests concerning items on hand, items on order, and orders backlogged. The sample printout of Section 2.6.1 shows this type of facility.

If new commands are added to the standard set or a new set is created, the resulting CCL table must be assembled later in the system generation process. As a result, SYSGEN prints the CCL LISTING ON LP0: question if a line printer is available for use during SYSGEN as discussed in Section 2.7.10. If the standard set of CCL commands is accepted, no assembly is necessary, no listing is produced, and SYSGEN does not print the question.

Since the CCL listing provides useful system documentation, it is important that the listing file be printed. The CCL listing file USRCCL.LST is preserved on disk and tape in the same manner as the system tables and terminal service listing files described in Section 2.7.13. Similarly, the file can be printed at a later time unless the system generation disk is an RF11 device. In the case of an RF11, the CCL listing must be printed during the system generation process or it cannot be printed at all.

If YES is the response to the CCL LISTING ON LP0: query, the listing file is printed during the system generation process. This listing is fairly short and takes about two minutes to print on a 300 line per minute printer.

2.7.25 Floating Point Precision and Scaled Arithmetic

The system manager can select either single precision (2-word) or double precision (4-word) floating point format for the type of numeric format to be used on his system. These floating point formats are described in Appendix E.1 of the BASIC-PLUS Language Manual.

Answers supplied to the FPP(PDP-11/70), FIS(11/40), FPP(11/45), MATH PRECISION, and FUNCTIONS questions determine which mathematical software package is selected from the ten standard packages included in the RSTS/E software. The following list describes the 2-word math packages.

MA2	Without FIS or FPP; with extended functions
MA2X	Without FIS or FPP; without extended functions
MA2I	11/40 FIS; with extended functions
MA2IX	11/40 FIS: without extended functions
MA2F	11/70 and 11/45 FPP; with extended functions
MA2FX	11/70 and 11/45 FPP; without extended functions

The following list describes the 4-word math packages.

MA4	Without FIS or FPP; with extended functions
MA4X	Without FIS or FPP; without extended functions
MA4F	11/70 and 11/45 FPP; with extended functions
MA4FX	11/70 and 11/45 FPP; without extended functions

The most critical difference among packages is the implementation of floating point operations. The PDP-11/70 and PDP-11/45 Floating Point Processor (FPP) provides both 2-word and 4-word floating point instructions in hardware. The PDP-11/40 Floating Instruction Set (FIS) does not provide 4-word floating point instructions in hardware. Therefore, on PDP-11/40 computers with or without FIS, RSTS/E must use the slower software packages (MA4 or MA4X) to perform 4-word floating point operations.

The scaled arithmetic feature is standard only on systems with 4-word floating point format. The feature is described in Section 6.8 of the *BASIC-PLUS Language Manual* and the SCALE command is described in Section 2.8 of the *RSTS-11 System User's Guide*. Scaled arithmetic avoids problems such as loss of precision normally associated with floating point calculations. The feature is very useful for calculating sums (such as money) which cannot be manipulated easily as integer quantities.

2.7.26 Mathematical Functions

The size of the BASIC-PLUS language code can be reduced by omitting certain mathematical functions including SIN, COS, TAN, ATN, SQR, EXP, LOG, and LOG10. These functions are described in Section 3.7 of the *BASIC-PLUS Language Manual* and summarized in Table 3-1 of that section.

2.7.27 PRINT USING Option

The PRINT USING optional feature allows BASIC-PLUS programs to perform special formatting of output as described in Section 10.3.3 of the *BASIC-PLUS Language Manual*. The size of BASIC-PLUS can be reduced by omitting this option.

2.7.28 Matrix Manipulation

BASIC-PLUS can operate on an entire matrix using single statements called MAT statements as described in Chapter 7 of the BASIC-PLUS Language Manual. The system manager must configure this optional feature if he wants to include the matrix manipulation capability.

2.7.29 Stand Alone Programs

The stand alone program ROLLIN can be included in the RSTS/E CIL and can be loaded into memory using the LOAD option as described in Chapter 3. ROLLIN is described in the library document entitled PDP-11 ROLLIN Utility Program which is included in the RSTS/E distribution software.

2.8 LOADING THE CIL ONTO THE RSTS/E SYSTEM DISK

After the configuration question concerning ROLLIN is answered, the SYSGEN program completes building the configuration file and the second batch command file. Subsequently, SYSGEN prints the second checkpoint message. Next, the system generation monitor executes commands in the second batch command file and generates the RSTS/E linked core image library (LICIL) and any listings necessary.

During the generation, messages are printed indicating which devices to mount and how to proceed. The entire process takes between one and three hours depending upon the devices used and the types of listings requested. If either magtape or DECtape distribution media is employed, the user must mount a new tape to which the RSTS/E LICIL is written. Instructions are printed to mount the RSTS/E system disk on unit 0 and to type the exact command to transfer the LICIL to the disk. If disk cartridge distribution media is used, instructions are printed to mount a new cartridge or pack to be used as the RSTS/E system disk.

After the CIL is written on the RSTS/E system disk and the RSTS/E code is loaded into memory for the first time, the PATCH, DSKINT, REFRESH, SETKEY, DEFAULT, and START options must be run as described in Chapter 3 and the system library must be built according to procedures described in Chapter 4.

2.9 SYSLOD AND CILUS COMMAND STRINGS

This section describes the procedures and command strings employed in loading the RSTS/E Linked Core Image Library (LICIL) onto the RSTS/E system disk. All necessary instructions to perform the load operation are printed during the system generation procedure. If the standard procedure is followed, this section need not be referenced. However, the command strings documented here are useful if the load operation is not performed as part of the system generation procedure or if the newly created LICIL must replace an old RSTS/E Core Image Library (CIL) on an existing system disk.

Loading the CIL is the final step of the system generation process. For magtape and DECtape distribution media, the newly created Linked Core Image Library (LICIL) is written to tape along with a copy of the SYSLOD program, the batch and configuration files, system load maps, and assembly listing files (magtape only). The SYSGEN program prints the commands necessary to load the CIL onto the RSTS/E system disk with SYSLOD. For disk cartridge distribution media, the LICIL resides on the system generation disk along with the batch and configuration files, system load maps, and the listing files. The command to load the CIL is never printed because the batch stream performs the load operation with the CILUS program. However, for all distribution media, the CIL need not be loaded immediately since the standard procedure can be halted just before the load operation. If the LICIL tape or system generation disk is preserved, the device can be bootstrapped and either SYSLOD or CILUS can be used to load the CIL onto the RSTS/E system disk.

Because the load operation can be performed using either a blank or an existing system disk, two general operations are described. Section 2.9.1 presents the procedures to load a new RSTS/E system onto a disk which contains no user files to be preserved. Section 2.9.2 provides guidelines for replacing an existing RSTS/E system on a system disk containing system and user files which must be preserved.

2.9.1 Loading the RSTS/E CIL onto a Blank System Disk

The procedure and command strings detailed in this section apply only to loading a CIL onto a blank system disk. It is possible to overwrite an existing CIL on a system disk which contains system and user files without destroying the file structure. This latter procedure is described in Section 2.9.2. The procedures below destroy any existing file structure on the disk being initialized as the RSTS/E system disk.

2.9.1.1 DECtape and Magtape Procedures Using SYSLOD - The tape created during system generation contains a copy of the stand-alone program SYSLOD. SYSLOD is loaded from DECtape using one of the hardware bootstrap loaders described in Appendix A. When SYSLOD is bootstrapped into memory, it will identify itself by printing the following lines.

```
SYSLOD V08-01A
CONSOLE FILL COUNT=                Type the RETURN key
DATE:  11-JAN-75                   Type the date in format shown
DIALOGUE?                          Type the RETURN key

#
```

Before the DIALOGUE query is answered, the disk to be used as the RSTS/E system disk should be mounted and write-enabled. In the case of an RK, RP02, RP03, or RP04 system disk, an RK cartridge or an RP pack must be mounted on drive unit 0. No special action is required for an RF system disk. SYSLOD does not recognize any device which is not mounted and ready when the DIALOGUE query is answered. The RETURN key is sufficient response to the DIALOGUE query. SYSLOD responds by printing the pound sign (#) when it is ready to accept a command. A single command string is sufficient to create and load the CIL onto the system disk and bootstrap the RSTS/E Initialization code into memory. The exact command which must be entered depends on the type of system disk. If any error messages are printed by SYSLOD, consult Appendix E for the proper procedure to follow. The following command strings are used for the several types of system disks.

```
#DF:/NS:256:49/TO:4/HO/BO<DT0:RSTS.LCL   (RF11 System Disk < DECtape)
#DF:/NS:256:49/TO:4/HO/BO<MT0:RSTS.LCL   (RF11 System Disk < TU10 Magtape)
#DF:/NS:256:49/TO:4/HO/BO<MM0:RSTS.LCL   (RF System Disk   < TU16 Magtape)
```

```
#DK0:/NS:256:49/FO/TO:4/HO/BO<DT0:RSTS.LCL (RK11 System Disk < DECtape)
#DK0:/NS:256:49/FO/TO:4/HO/BO<MT0:RSTS.LCL (RK11 System Disk < TU10 Magtape)
#DK0:/NS:256:49/FO/TO:4/HO/BO<MM0:RSTS.LCL (RK11 System Disk < TU16 Magtape)

#DP0:/NS:512:25/FO/TO:4/HO/BO<DT0:RSTS.LCL (RP02 or RP03 System Disk < DECtape)
#DP0:/NS:512:25/FO/TO:4/HO/BO<MT0:RSTS.LCL (RP02 or RP03 System Disk < TU10 Magtape)
#DP0:/NS:512:25/FO/TO:4/HO/BO<MM0:RSTS.LCL (RP02 or RP03 System Disk < TU16 Magtape)

#DB0:/NS:1024:13/FO/TO:4/HO/BO<DT0:RSTS.LCL (RP04 System Disk < DECtape)
#DB0:/NS:1024:13/FO/TO:4/HO/BO<MT0:RSTS.LCL (RP04 System Disk < TU10 Magtape)
#DB0:/NS:1024:13/FO/TO:4/HO/BO<MM0:RSTS.LCL (RP04 System Disk < TU16 Magtape)
```

Messages are printed at the console terminal when SYSLOD starts and ends formatting. If any other messages appear, consult Appendix E for possible recovery procedures. Upon completion of the load operation, the RSTS/E initialization code is bootstrapped into memory signalled by the printing of the OPTION query.

To load the CIL onto a non-zero unit, include the unit number in the output device designator (that is, DKn:, DPn:, or DBn:) and omit the /BO switch from the command string. After loading the CIL, SYSLOD reprints the # character. Although SYSLOD will load onto a non-zero unit, the pack or cartridge must be bootstrapped once from unit 0 to install a new bootstrap. Halt the processor and move the system disk to unit 0. Bootstrap the device using the hardware loader to load the initialization code into memory.

After the OPTION query appears, proceed to Chapter 3 to initialize the system disk. If the OPTION query fails to appear, refer to the consistency errors described in Appendix B for possible recovery.

2.9.1.2 Disk Cartridge Procedures Using CILUS - When the disk cartridge distribution medium is used, the DOS program CILUS is used to load the RSTS/E CIL onto the system disk. CILUS does not format an RK cartridge, RP02, RP03, or RP04 pack. Refer to Section 2.3.3 for procedures to load SYSLOD and format disks.

Mount the copy of the System Generation DECpack used for the system generation on RK unit 0. Write enable the drive and bootstrap the cartridge to load the DOS/BATCH monitor. Refer to Appendix A for DECpack bootstrap procedures. When the DOS monitor identifies itself, proceed as shown below to run CILUS.

```
DOS/BATCH V9-20C
DATE:  11-JAN-75                Type the date in format shown
TIME:  16:15                    Type the time in format shown
DIALOGUE?                       Type the RETURN key
$LO 1,1
DATE:  11-JAN-75
TIME:  16:15

$RUN CILUS

CILUS V08-06A

#
```

CILUS prints the pound sign (#) when it is ready to accept a command. The CILUS command string used to create and load the RSTS/E CIL depends on the type of system disk. If the RSTS/E system is configured for an RP02, RP03 or RP04 system disk, mount a newly formatted pack on unit 0 and write-enable the drive. If the system is configured for an RK system disk, mount a newly formatted RK cartridge on RK unit 1. No special action is required for an RF system disk. One of the CILUS commands shown below is then used to load the CIL.

```
#DF:/NS:256:49/TO:4/HO/BO<DK0:RSTS.LCL/LO   (RF11 System Disk < RK cartridge)

#DK1:/NS:256:49/TO:4/HO<DK0:RSTS.LCL/LO     (RK11 System Disk < RK cartridge)
 DP1:                  <DP0
#DP0:/NS:512:25/TO:4/HO/BO<DK0:RSTS.LCL/LO (RP02 or RP03 System Disk < RK cartridge)

#DB0:/NS:1024:13/TO:4/HO/BO<DK0:RSTS.LCL/LO (RP04 System Disk < RK cartridge)
```

If the system disk is an RF11, RP02, RP03, or RP04, the CILUS command loads the CIL and then bootstraps the RSTS/E initialization code into memory. The INIT code prints the system name followed by the OPTION query. The system manager should proceed to initialize the system disk as described in Chapter 3.

The command to load the CIL onto the RK cartridge mounted on unit 1 does not bootstrap the RSTS/E Initialization code. When the load operation is complete, CILUS reprints the pound sign (#) and waits for another command. The system manager exits from CILUS and terminates the DOS monitor as shown below.

```
#^C                                CONTROL/C exit from CILUS
.KI                                "KILL" required by DOS
$FI                                FInish for an orderly exit
TIME:  16:16:16
DOS/BATCH V9-20C

$
```

Halt the processor by moving the HALT/ENABLE switch to the HALT position. Dismount both cartridges and move the RSTS/E system disk to RK unit 0. When the disk is ready, write-enable the drive and bootstrap the disk cartridge (see Appendix A) to load the RSTS/E initialization code into memory. Proceed to Chapter 3 to initialize the new system disk.

To load the CIL onto a non-zero unit, include the unit number in the output device designator (that is, DKn:, DPn:, or DBn:) and omit the /BO switch from the command string. After loading the CIL, CILUS reprints the # character and awaits another command. Although CILUS will load onto a non-zero unit, the pack or cartridge must be bootstrapped once from unit 0 to install a new bootstrap. Halt the processor and move the system disk to unit 0. Bootstrap the device using the hardware loader to load the initialization code into memory. Proceed to Chapter 3 to initialize the system disk.

2.9.2 Replacing the RSTS/E System Code

It is possible to replace the RSTS/E CIL on the system disk without destroying the file structure. This capability is important when a new system is generated to add or change hardware support or software features. The SYSLOD or CILUS command strings used for this purpose are similar to those in Section 2.9.1 but several precautions should be taken to ensure a successful replacement. Careful adherence to these procedures is critical to avoid destroying the existing file structures (system and user files) on the system disk.

It is impossible to determine the exact size of the new CIL until it is loaded onto a disk. The first step in replacing the CIL, therefore, is to load the new monitor onto a scratch disk using the standard system generation procedures or the SYSLOD or CILUS commands described in Sections 2.9.1.1 and 2.9.1.2. The scratch disk must be initialized using the DSKINT initialization option (see Section 3.3) to create the RSTS.CIL file. When a disk is initialized, pattern checking for bad blocks is normally done. However, requesting zero patterns causes DSKINT to bypass the pattern checks, saves time, and is a reasonable shortcut for the scratch disk.

The REFRESH initialization option (Section 3.4) is used after DSKINT to check the required size of the new RSTS.CIL file. REFRESH can be stopped by typing CONTROL/C after the file status table is printed. Since the scratch disk is used only to determine the size of the new RSTS.CIL file, it is of no further use after these operations are performed. The loading of the new system onto a scratch disk and the DSKINT and REFRESH operations take a total of about 10 minutes.

The next step is to determine the size of the CIL to be replaced. Simply obtain a directory of the system files account [0,1] under time sharing or print the file status table using REFRESH (see Section 3.4.3.2). The important item is the current size of the RSTS.CIL file on the system disk. The REFRESH procedures of Section 3.4 recommend that this file be made larger than the required size when the system disk is initially built. If the system manager planned for a future replacement of the system code, the current size of the old RSTS.CIL file will probably be larger than the required size of the new RSTS.CIL file.

If the old CIL is larger, the replacement can proceed as described in subsequent paragraphs. Otherwise, the system manager can attempt to increase the size of the old RSTS.CIL file using REFRESH before attempting the replacement. If the REFRESH is successful, he can proceed. If REFRESH cannot increase the size of the old RSTS.CIL file to at least the required size of the new RSTS.CIL file, the replacement cannot be performed. All library and user files must be transferred to another disk or external medium and the system disk must be initialized (destroying the existing file structure).

Assuming the RSTS.CIL file on the system disk is large enough to accommodate the new CIL, the next step is to transfer all library and user files from the system disk to a secondary storage medium. This is a time consuming but important precaution since a typographical error or a hardware malfunction while replacing the old CIL could be disastrous. The transfer is performed under time sharing using the old system. The next two sections present the SYSLOD and CILUS command strings which replace the old CIL.

2.9.2.1 DECtape and Magtape Procedures Using SYSLOD - The tape created during system generation contains a copy of the stand-alone program SYSLOD. SYSLOD is loaded from DECtape or magtape using one of the hardware bootstrap loaders described in Appendix A. When SYSLOD is bootstrapped into memory, it identifies itself by printing the following lines.

```
SYSLOD V08-01A
CONSOLE FILL COUNT=          Type the RETURN key
DATE: 11-JAN-75              Type the date in format shown
DIALOGUE?

#
```

Before the DIALOGUE query is answered, the old RSTS/E system disk should be mounted and write enabled. In the case of an RK, RP02, RP03, or RP04 system disk, the cartridge or pack must be mounted on unit 0. SYSLOD does not recognize any device which is not mounted and ready when the DIALOGUE query is answered. The RETURN key is sufficient response to the DIALOGUE query. SYSLOD responds by printing the pound sign (#) when it is ready to accept a command. The following SYSLOD commands replace an old CIL on the various types of system disks.

tape to disk

```
#DF:/NS:256:49/TO:4/HO/BO<DT0:RSTS.LCL    (RF11 System Disk < DECtape)
#DF:/NS:256:49/TO:4/HO/BO<MT0:RSTS.LCL    (RF11 System Disk < TU10 Magtape)
#DF:/NS:256:49/TO:4/HO/BO<MM0:RSTS.LCL    (RF11 System Disk < TU16 Magtape)

#DK0:/NS:256:49/TO:4/HO/BO<DT0:RSTS.LCL   (RK11 System Disk < DECtape)
#DK0:/NS:256:49/TO:4/HO/BO<MT0:RSTS.LCL   (RK11 System Disk < TU10 Magtape)
#DK0:/NS:256:49/TO:4/HO/BO<MM0:RSTS.LCL   (RK11 System Disk < TU16 Magtape)

#DP0:/NS:512:25/TO:4/HO/BO<DT0:RSTS.LCL   (RP02 or RP03 System Disk < DECtape)
#DP0:/NS:512:25/TO:4/HO/BO<MT0:RSTS.LCL   (RP02 or RP03 System Disk < TU10 Magtape)
#DP0:/NS:512:25/TO:4/HO/BO<MM0:RSTS.LCL   (RP02 or RP03 System Disk < TU16 Magtape)

#DB0:/NS:1024:13/TO:4/HO/BO<DT0:RSTS.LCL  (RP04 System Disk < DECtape)
#DB0:/NS:1024:13/TO:4/HO/BO<MT0:RSTS.LCL  (RP04 System Disk < TU10 Magtape)
#DB0:/NS:1024:13/TO:4/HO/BO<MM0:RSTS.LCL  (RP04 System Disk < TU16 Magtape)
```

The only difference between these SYSLOD commands for replacing a CIL and those used for a blank system disk is the absence of the format switch (/FO) for RK and RP02, RP03 or RP04 system disks. Formatting destroys everything on a disk and cannot be done if the file structure is to be preserved.

Upon completion of load operation, the RSTS/E system disk is bootstrapped, the initialization code is loaded into memory, and the OPTION query is printed.

To load the CIL onto a non-zero unit, include the unit number in the output device designator (that is, DKn:, DPn:, or DBn:) and omit the /BO switch from the command string. After loading the CIL, SYSLOD reprints the # character. Although SYSLOD will load onto a non-zero unit, the pack or cartridge must be bootstrapped once from unit 0 to install a new bootstrap. Halt the processor and move the system disk to unit 0. Bootstrap the device using the hardware loader to load the initialization code into memory. The system manager must reinstall all published patches using the PATCH option (see Section 3.2), set keyboard defaults with the SETKEY option (Section 3.5) and establish default start up conditions with the DEFAULT option (Section 3.6) before time sharing operations can resume with the new system. The DSKINT initialization option is not used in this case since initializing a disk destroys any existing file structures. The REFRESH option can be used to verify or change any of the system files but is not specifically required after replacing the CIL.

2.9.2.2 DECpack Procedures Using CILUS - When disk cartridge distribution media is used, the DOS program CILUS loads the new CIL onto the old system disk. Since CILUS does not format disks, the CILUS commands are the same as for loading a CIL onto a blank system disk. The CILUS procedures are repeated below for continuity.

Mount the copy of the system generation disk cartridge used for the system generation on RK unit 0. Write-enable the drive and bootstrap the cartridge to load the DOS/BATCH Monitor. Refer to Appendix A for bootstrap procedures. When the DOS Monitor identifies itself, proceed as shown below to run CILUS.

```
DOS/BATCH V9-20C
DATE:  11-JAN-75                 Type the date in format shown
TIME:  12:30                     Type the time in format shown
DIALOGUE?                        Type the RETURN key

$LO 1,1

DATE:  11-JAN-75
TIME:  12:30

$RUN CILUS

CILUS V08-06A

#
```

CILUS prints the pound sign (#) when it is ready to accept a command. If the RSTS/E system is configured for an RP02, RP03 or RP04 system disk, mount the old system disk on unit 0 and write-enable the drive. If the system is configured for an RK system disk, mount the old RK system disk cartridge on RK unit 1. No special action is required for an RF system disk. One of the CILUS commands shown below is then used to replace the old CIL.

disk to disk

```
#DF0:/NS:256:49/TO:4/HO/BO<DK0:RSTS.LCL/LO (RF11 System Disk < RK Cartridge)
#DK1:/NS:256:49/TO:4/HO<DK0:RSTS.LCL/LO    (RK11 System Disk < RK Cartridge)
#DP0:/NS:512:25/TO:4/HO/BO<DK0:RSTS.LCL/LO (RP02 or RP03 System Disk < RK Cartri
#DB0:/NS:1024:13/TO:4/HO/BO<DK0:RSTS.LCL/LO (RP04 System Disk < RK Cartridge)
```

If the system disk is an RF11, RP02, RP03, or RP04, the CILUS command loads the CIL, bootstraps the system disk, and loads the RSTS/E Initialization code into memory. The INIT code prints the system name followed by the OPTION query.

The command to replace the CIL on an RK system disk mounted on unit 1 does not bootstrap the RSTS/E Initialization code. When the load operation is complete, CILUS reprints the pound sign (#) and waits for another command. The system manager should exit from CILUS and terminate the DOS Monitor as shown below.

```
#^C                                    CONTROL/C exit from CILUS
.KI                                    "KILL" required by DOS

$FI                                    FInish for an orderly exit
TIME:  12:35:22

DOS/BATCH V9-20C

$
```

Halt the processor by moving the HALT/ENABLE switch to the HALT position. Dismount both cartridges and move the RSTS/E system disk to RK unit 0. When the disk is ready, write enable the drive and bootstrap the disk cartridge (see Appendix A) to load the RSTS/E Initialization code into memory.

To load the CIL onto a non-zero unit, include the unit number in the output device designator (that is, DKn:, DPn:, or DBn:) and omit the /BO switch from the command string. After loading the CIL, CILUS prints the # character and awaits another command. Although CILUS will load onto a non-zero unit, the pack or cartridge must be bootstrapped once from unit 0 to install a new bootstrap. Halt the processor and move the system disk to unit 0. Bootstrap the device using the hardware loader to load the initialization code into memory.

When the Initialization code prints the OPTION query, the system manager must reinstall all published patches using the PATCH option of the Initialization code (see Section 3.2), set keyboard defaults with the SETKEY option (Section 3.5) and establish default start up conditions using the DEFAULT option (Section 3.6). The DSKINT option is not used since initializing a disk destroys any existing file structure. The REFRESH option (Section 3.4) may be used to verify or change the system files but is not specifically required after replacement of a CIL.

CHAPTER 3

INITIALIZATION OPTIONS

The RSTS/E Initialization (INIT) code is a collection of routines used to create the file structures, system files, and start up conditions required for normal operation of the RSTS/E system. The INIT code is essentially one large stand-alone program with many functions. Immediately after a system generation, several options must be used before the RSTS/E system can be brought up for time sharing. Thereafter, the initialization code provides the mechanism for altering critical system files and parameters as installation requirements change. INIT includes routines which ensure the integrity of the system disk file structure and provide many checks on the hardware configuration. Options are provided which enable the system to function even when certain hardware elements are inoperative. Finally, the initialization code is responsible for loading the RSTS/E Monitor and BASIC-PLUS language code into memory for normal time sharing.

This chapter contains detailed information on all the initialization options and is essential to the proper and efficient use of system features. The system manager should fully understand the material presented here before any attempt is made to operate the RSTS/E system.

3.1 OVERVIEW OF INITIALIZATION OPTIONS

Whenever the RSTS/E system disk is bootstrapped, the initialization code is loaded into memory. The bootstrap operation is performed by a hardware ROM bootstrap or by software. In the system generation example of Sections 2.6.1 and 2.6.3, the SYSLOD program performed the bootstrap operation after the new system was loaded onto the RSTS/E system disk. In normal operations, the hardware bootstrap is used. When the INIT code is loaded, it prints a header line with the installation name followed by the OPTION query. The system manager must then select one of the valid initialization options listed in Table 3-1.

Table 3-1
Initialization (Start Up) Options

Long Form of Operator Response	Short Form of Operator Response	Meaning
PATCH	PA	Alter the RSTS/E System Code to correct problems.
DSKINT	DS	Initialize and optionally format a disk.
REFRESH	RE	Create or rebuild the system files in account [0,1] on the system disk.
SETKEY	SE	Set keyboard defaults for disabling lines and for enabling DH11 lines as local or with modem control.
DEFAULT	DE	Establish or change start up conditions.
START	ST or LINE FEED key	Start time sharing operations.
UNISYS ONLY 1 TERMINAL AVAILABLE	UN	Diagnostic aid used in conjunction with the START option to bypass the enabling of all terminal interfaces except the console interface.
BOOT	BO	Bootstrap a device.
LOAD	LO	Load a stand-alone program from the RSTS/E CIL.
ASR33 VT05B LA30S (FILL OPTIONS)	AS VT LA	Set the fill factor of the console terminal to that of the device specified.

Initialization Options

The initialization options can run from a system disk mounted on any disk unit. The new V06A-02 system disk must be bootstrapped once from unit 0 to install a RSTS/E bootstrap routine that passes the correct unit information to the initialization routines. Thereafter, the system disk may be bootstrapped from any drive. Because the BASIC-PLUS program INIT.BAC depends on unit numbers to mount non-system disks, control files have to be modified to run RSTS/E from a non-zero unit.

After a system generation is performed and the Core Image Library (CIL) is loaded onto the RSTS/E system disk, the system manager must execute the PATCH, DSKINT,[1] REFRESH,[1] DEFAULT, and START options to prepare for time sharing.

The PATCH option provides a mechanism by which the RSTS/E system code can be easily altered as problems are discovered and corrected. Patches are published in the RSTS/E Installation Notes and in monthly software publications.

The DSKINT option creates the minimal RSTS/E file structure on all RSTS/E system disks. In addition, DSKINT formats a disk and performs pattern checks to detect bad disk blocks.

REFRESH creates the system files for normal RSTS/E system operation. REFRESH also provides the capability for altering system files as installation requirements change.

The SETKEY option establishes defaults for disabling keyboard lines and enabling DH11 lines as either local or with modem control. The option supplements the SYSGEN program question concerning the number of DH11 lines to enable. SETKEY is necessary for enabling modem control on DH11 lines.

The DEFAULT option establishes start up parameters such as the maximum number of jobs which can be run and the maximum size for each job. DEFAULT allows the system manager to determine the most efficient use of memory on his system. Suboptions of DEFAULT permit locating the BASIC-PLUS language code in high-speed semiconductor memory and making certain failing sections of memory unavailable.

---

[1] DSKINT is not used if a new RSTS/E CIL replaces an old system on a system disk where the file structure is to be preserved (see Section 2.9). Furthermore, since system and user file structures are not disturbed by the replacement, REFRESH is not required.

Initialization Options

START brings the RSTS/E up to a full running state. It also permits temporary changes to the default start-up conditions mentioned above. START is responsible for loading the monitor tables, enabling terminal interfaces, loading the RSTS/E Monitor and BASIC-PLUS into memory, and starting time sharing.

UNISYS is a diagnostic aid used in conjunction with the START option. Problems are often encountered with new systems due to errors in the hardware or software terminal interface configuration. The UNISYS option helps to isolate the problem to the terminal interface configuration by allowing the system to be brought up for time sharing without enabling any terminal interface except for the console device.

The BOOT option emulates the operation of the hardware bootstrap loaders. BOOT bootstraps a program or system from disk or tape devices.

LOAD permits loading and executing stand-alone programs from the RSTS/E Core Image Library (CIL) created during system generation. SYSGEN provides a way to include the stand-alone program ROLLIN (a tape and disk copy utility) in the CIL.

Finally, the fill options VT05, ASR33 and LA30S condition the keyboard driver in the INIT code to insert the proper number of fill characters when printing messages on the console terminal. These options are used only when the actual console device is something other than the console terminal specified at system generation time.

If only the RETURN key is typed in response to the OPTION query, the following message is printed.

```
OPTION:
PLEASE ENTER ONE OF THE VALID RSTS SYSTEM INITIALIZATION
OPTIONS OR TYPE 'HELP' FOR A HELP MESSAGE:
```

If the user types other than a valid response or types HELP, the system prints the following information and reprints the OPTION query.

Initialization Options

```
OPTION: HELP
THE VALID RSTS V05 INITIALIZATION OPTIONS ARE:
        BOOT    BOOTSTRAP A DEVICE
        LOAD    LOAD AND EXECUTE A PROGRAM FROM THIS CIL
        DEFAULT ESTABLISH OR CHANGE STARTUP DEFAULTS
        START   START TIMESHARING
        SETKEY  SET KEYBOARD DEFAULTS
        DSKINT  INITIALIZE DISK TO RSTS FILE STRUCTURE
        REFRESH REFRESH YOUR SYSTEM DISK
ONLY THE FIRST TWO CHARACTERS NEED BE TYPED.

OPTION:
```

When a valid option name is entered, the system proceeds according to the option specified. Typing the CONTROL key and C key combination (CTRL/C) when any of the initialization options is asking for a response immediately interrupts execution of the option and returns the user to the OPTION query.

The following sections explain each of the initialization options in detail.

3.2 PATCH OPTION - PATCHING THE RSTS/E SYSTEM

The RSTS/E Initialization code PATCH option provides a convenient means for altering the RSTS/E system code as errors are found and corrections are published. When a RSTS/E system generation is performed, all patches are installed immediately after the Core Image File (CIL) is loaded onto the system disk. This is necessary since patches may affect the initialization code used to build required file structures, create the system files, and set up tables used during normal timesharing. Patches are published in the RSTS/E Installation Notes if problems are uncovered after a "code freeze," but before a new release is available from Digital's Software Distribution Center. Thereafter, patches are published in the monthly Digital Software News or Software Dispatch.

The PATCH option makes permanent changes to the RSTS/E CIL on the system disk. The CIL is made up of several modules including INIT (the initialization code) RSTS (the resident monitor and device drivers), OVR (Overlay Code), ERR (error messages), and BASIC (the BASIC-PLUS compiler and Run Time System). Any of these modules may be altered using the PATCH option.

Patches take many different forms. Some are in-place patches to one or more words in one or more modules. Others require patch space in the affected modules. Patch space is always included in the Monitor, BASIC-PLUS, and the Initialization code. The overlay code can be patched using free space in overlay segments or monitor patch space. In some cases, patches affect fixed addresses and are straightforward; however in most cases it is necessary to refer to the system load maps to find the addresses of affected sections. Published patches describe the procedures required to make the alteration correctly.

The PATCH option is called by typing PATCH or simply PA in response to the initialization code OPTION query. PATCH replies by asking for a MODULE NAME (one of the five listed above), a BASE ADDRESS, and an OFFSET ADDRESS. The module name determines the CIL module to be changed. The response "BASIC" indicates that the patch applies to the BASIC-PLUS compiler or Run Time System. The base address further determines the actual locations to be patched. For example, the base address for the Print Using section of BASIC-PLUS is found in the BASIC load map and might be entered as the response to the BASE ADDRESS query. Finally, the offset address is the first location to be changed relative to the specified base. For example, a PRINT-USING patch may begin at an offset of 100 octal bytes from the beginning of Print Using. After these items are entered, PATCH prints the old contents of the specified location and opens the word for change. PATCH opens and changes successive locations depending on the user responses.

Details for the use of the patch option are included in the two examples presented below. The first demonstrates some of the complexities of patching. The second, a simple INIT patch, emphasizes the need for using the BOOT option (Section 3.9) after any patch is made to the Initialization code.

3.2.1 Patching the RSTS/E Monitor - EXAMPLE

The following example describes the use of the PATCH option to alter the RSTS/E monitor. (Note: When the patch is made, the monitor resides on disk and not in memory.) The Initialization Code is in memory and the PATCH option allows changes to the monitor disk image before it is loaded into memory for normal timesharing.

```
OPTION:  PATCH                                  EXAMPLE ONLY:
MODULE NAME ? RSTS                              NOT A REAL PATCH
BASE ADDRESS ? [NAME]
OFFSET ADDRESS ? 120
MODULE    BASE   OFFSET  OLD      NEW?
RSTS     [NAME] 000120 104760 ? 004737
RSTS     [NAME] 000122 103364 ? [PATCH]+20     OCTAL ADDITION.
RSTS     [NAME] 000124 005062 ? 102637
RSTS     [NAME] 000126 012762 ? [JBSTAT]-2     OCTAL SUBTRACTION.
RSTS     [NAME] 000130 004737 ? <LF>           NO CHANGE.
RSTS     [NAME] 000132 104726 ? 104730
RSTS     [NAME] 000134 010423 ? ↑C             CONTROL/C EXIT.

OPTION:  PATCH
MODULE NAME ? RSTS
BASE ADDRESS ? [PATCH]                          FROM RSTS LOAD MAP
OFFSET ADDRESS ? 20
MODULE    BASE   OFFSET  OLD      NEW?
RSTS    [PATCH] 000020 000000 ? 010203
RSTS    [PATCH] 000022 000000 ? 011104
RSTS    [PATCH] 000024 000000 ? 000207
RSTS    [PATCH] 000026 000000 ? ↑C             CONTROL/C EXIT

OPTION:
```

All numbers printed by the PATCH option and all numeric responses are octal. In the example, the notation [NAME] indicates an address that must be found in a load map or a quantity that must be computed. PATCH does not perform any arithmetic; hence, expression of the form [NAME] + 20 must be manually calculated using 2's complement arithmetic. (If unfamiliar with the octal representation of binary numbers or with 2's complement arithmetic, consult a Software Support Representative.) As PATCH opens successive locations, it prints the current or old location contents and waits for new data to be entered as an octal word. A carriage return (CR) is used to enter the new data. PATCH then sequences to the next location. A line feed <LF> with no new data causes PATCH to sequence to the next location without altering the current location. PATCH continues to open successive locations until the CONTROL/C combination is typed. CTRL/C returns to the initialization code OPTION query.

Note that changes are made immediately upon typing the carriage return key. If an error is made it becomes necessary to reenter the PATCH option to correct the mistake. Printing the old contents of a location provides a check for proper placement of a patch. If the old contents of any location shown in a published patch are not identical to those printed by the PATCH option, all locations should be restored to their old contents. This may indicate an error in the use of load maps or an error in the published patch itself. Finally, a complete patch may be double checked by reentering the PATCH option and using the line feed key to examine successive locations.

3.2.2 Patching the Initialization Code - EXAMPLE

Patching the Initialization Code is usually simpler since INIT is the same for all systems. There is usually no need to refer to a load map unless the value of a global parameter is needed for the patch. Patches are made to the CIL on disk and not in core. This is an important distinction when patching INIT since the in-core copy (which is running) is not changed by the PATCH option. It is necessary to use the BOOT option (described fully in Section 3.8) to load the altered INIT code into memory. The example below illustrates the procedure for making an INIT patch.

```
OPTION:  PATCH                           EXAMPLE ONLY -
MODULE NAME ? INIT                       NOT A REAL PATCH
BASE ADDRESS ? 67472
OFFSET ADDRESS ? 4724
MODULE   BASE   OFFSET  OLD    NEW?
INIT    067472 004724 100200 ? 104200 SINGLE WORD PATCH.
INIT    067472 004726 XXXXXX ? <LF>   OLD CONTENTS VARIABLE.
INIT    067472 004730 005766 ? <LF>   PRINTED FOR VERIFICATION ONLY.
INIT    067472 004732 001000 ? ↑C     CONTROL/C EXIT.

OPTION:  BOOT                    BOOT REQUIRED TO LOAD ALTERED
                                 INIT CODE INTO MEMORY

BOOT DEVICE ? <LF>               LINE FEED BOOTS THE SYSTEM DISK

RSTS V06A-02 TEST SYSTEM

OPTION:
```

3.3 DSKINT OPTION - DISK INITIALIZATION

The DSKINT option initializes all disks (system, public, and private) to be used on the RSTS/E system. DSKINT writes the minimal RSTS/E file structure on the disk. Hence, all disks, except auxiliary swapping disks, must be initialized prior to use. DSKINT destroys any existing file structure on a disk and should not be used on any RSTS/E disk if the file structure is to be preserved.

The minimal file structure written on all disks consists of the following elements:

(1) The MFD account [1,1] for the disk,

(2) The UFD for the system account [0,1] on the disk,

(3) Two files for the system account [0,1] of the disk: BADB.SYS and SATT.SYS, and

(4) A public or private disk designation for the device.

The initialization of a system disk differs slightly from the initialization of non-system disks. When the RSTS/E system disk is initialized, the Core Image Library (CIL) is preserved through the DSKINT process. Several additional elements are also included in the minimal file structure. These include:

(1) The system file RSTS.CIL under the system files account [0,1] which maps the Core Image Library (CIL).

(2) The entry in the MFD for the system library account [1,2]

Initialization Options

The REFRESH option described in Section 3.4 creates the other required system files on the system disk. The BUILD process described in Chapter 4 installs all the system programs under the library account [1,2] on the system disk. Finally the REACT system program enters user accounts into the MFD on all disks.

In addition to initializing the disk with a minimal file structure, the DSKINT routines allow the user to format the disk and to specify the number of patterns used to check for bad blocks. The system manager must format any non-system disk before it is used on the RSTS/E system. Formatting the disk causes all the necessary timing and sense marks to be written on the disk and erases any extraneous information from the disk. (Formatting applies only to RK and RP type removable disks.) The system disk which contains the RSTS/E CIL should not be formatted after the CIL is loaded by SYSLOD or CILUS since the formatting operation would destroy the CIL.

The DSKINT option allows the system manager to choose from one to eight possible patterns to check for bad blocks on a disk. A bad block is a portion of the disk to which information is written but from which that same information cannot be read. It is recommended that the system manager specify a minimum of two patterns; he must specify at least one pattern.

The specification of a larger number of patterns increases the probability that all bad blocks are detected during the initialization of the disk and decreases the possibility of loss of valuable system information which is later written on the disk. The possibility of loss is less since clusters containing detected bad blocks are allocated to a file called BADB.SYS stored under account [0,1] on the disk. These clusters are never allocated for any other purpose.

The minimal file structure written by the DSKINT routines enables the disk to be used under either the public or the private structure of disks in the RSTS/E system. The system manager must, however, designate to the DSKINT routines whether the disk is to be public or private. Public and private disks are discussed in Section 9.1.2 of the BASIC-PLUS Language Manual. Structurally, the only difference between a public and a private disk is that a bit (termed the private bit) is set in the MFD of a private disk. Operationally, however, an important distinction is made by the RSTS/E system between a private and a public disk.

During time sharing operations all disks within the public structure must be readied, write enabled, and logically mounted. The system treats the entire public structure as one general, default reference. A user's file could reside on any one of the physical devices within the public structure and, therefore, all devices in the public structure must be available to ensure successful file access.

In addition, if a user creates a file, the directories of each public disk are searched to ensure that a file of the same name does not already exist. The system manager makes a public disk available by implementing the guidelines in Section 3.1 of the RSTS/E System Manager's Guide for mounting disks automatically at start up time.

Disks within the private structure, however, need not always be present during time sharing operations. Access to files on a private disk requires that the user refer specifically to a distinct physical device. To create or access a file on a private disk, only the directory of that device need be searched. As a result, private packs can be interchanged during time sharing operations if the proper disk management procedures are followed. (Refer to Section 7.1.2 of the RSTS/E System Manager's Guide for the discussion of disk management procedures.)

The system manager must decide how much of his disk resources to commit to the public structure or to the private structure. Disks are designated public or private only when they are initialized. To change the designation, any useful contents of the disk must be preserved elsewhere and the disk must be reinitialized. If the disk is initialized as a private disk, the system manager must enter user account information in the MFD of the disk with the REACT system program, so that files can be created on the disk. (REACT is described in Section 4.1 of the RSTS/E System Manager's Guide.

3.3.1 Using DSKINT

When the RSTS/E initialization routines are executed, they print the system installation name followed by the OPTION query. The system manager types DSKINT or simply DS to request the disk initialization option and then types the current date and time in response to two successive queries. The following sample dialog illustrates the procedure.

```
OPTION: DSKINT
DD-MMM-YY? 25/JAN/75
ILLEGAL DATE ENTERED; PLEASE TRY AGAIN
DD-MMM-YY? 25-JAN-75
HH:MM? 12:15

DISK?
```

Subsequent DSKINT queries request the user to enter the disk type and unit, pack identification and pack cluster size, and the passwords and cluster sizes for the MFD and library accounts. The DSKINT queries are presented for reference in Table 3-2.

If the routines determine that a response is incorrect or unacceptable, the query is reprinted. The queries are printed in a short form to save time. If the system manager types the RETURN key in response to any of the queries, DSKINT prints a longer, explanatory form of the query or, in some cases, the acceptable responses.

During the formatting of removable disks, messages are printed to indicate the start and the end of the format pass. The time between the messages depends on the size and type of disk.

If bad blocks are uncovered during a pattern test, a table is printed indicating the sector and cluster number (in decimal) in which a bad block exists and the contents of the disk hardware register (in octal) at the time of the error. If an excessive number of errors are encountered, a fatal error message is printed. In such a case, it is advised that another disk be used.

The completion of disk initialization is signalled by printing the OPTION query. The system manager then builds the system files on the initialized system disk using REFRESH as described in Section 3.4. The examples contained in the two following sections illustrate the use of DSKINT for system and non-system disks.

Table 3-2

DSKINT Queries

Query	Description of Response
DD-MMM-YY?	Type the current date in the exact format shown. For example, 3-FEB-75.
HH:MM?	Type the current, 24-hour time, which is used to start the Monitor clock, in the exact format shown. For example, 13:52.
DISK?	Type two characters which indicate the type of the disk being initialized. Acceptable entries are RF, RS, RK, RP (RP03), or RB (RP04).
UNIT?	For RS, RK, RP and RB type disks. Type the physical unit number on which the disk, DECpack, or disk pack resides. Acceptable entries are 0 through 7, inclusive.
PLATTERS?	For RF type disks. Type the number of RS11 disks (platters) connected to the RF11 controller. Acceptable entries are 1 through 8, inclusive.
PACK ID?	Type up to six alphanumeric characters which are used when logically mounting or dismounting the device. (See the description of the MOUNT and DISMOUNT commands in Section 6.3.2.1.)
PACK CLUSTER SIZE?	Type the decimal number of 256-word blocks which each cluster allocated on the disk will contain. Clusters are described in Section 5.4.3. Acceptable pack cluster sizes are 1, 2, 4, 8, or 16 for RF, RS, and RK disks. For RP03 disks, acceptable values are 2, 4, 8, or 16. For RP04 disks, acceptable values are 4, 8, or 16.
MFD PASSWORD?	Type up to six alphanumeric characters which become the password of account [1,1] on the device.
MFD CLUSTER SIZE?	Type the decimal number of 256-word blocks which a cluster allocated to the MFD will contain. Must be equal to or greater than the pack cluster size. The number of user accounts which can be created is approximately 108* MFD cluster size.

(continued on next page)

Table 3-2 (Cont.)
DSKINT Queries

Query	Description of Response
PUB, PRI, OR SYS?	Type PUB to designate the disk as public or type PRI to designate the disk as private. Type SYS to initialize a system disk. CAUTION: Initializing a system disk as PUB or PRI destroys the CIL.
LIBRARY PASSWORD?	Query is printed only when DSKINTing a system disk. Enter up to 6 alphanumeric characters which become the password of the library account [1,2].
LIBRARY UFD CLUSTER SIZE?	Printed only when DSKINTing a system disk. Type the decimal number of 256-word blocks allocated for each of the seven possible UFD clusters for the library account [1,2]. The number of files which can be stored under any account is approximately 72 * UFD cluster size. Library UFD cluster size must be greater or equal to pack cluster size.
FORMAT?	For public and private disks only. Type Y to write hardware timing and sensing data on the disk or type N to omit same. If Y is typed, messages are printed indicating the start and end of the format pass. All new disk cartridges and packs must be formatted. System disks should never be formatted after the CIL has been loaded.
PATTERNS?	Type a decimal number between 1 and 8 to choose the number of patterns used to check for bad blocks. Type the RETURN key only to print the time required to execute each pattern according to device type.
PROCEED (Y OR N)?	Type Y to proceed with the execution of the initialization. Type N to abort initialization and return to the OPTION query. This query is printed to allow the system manager to double check the responses to the queries.

3.3.2 Initializing the System Disk

After the stand-alone program SYSLOD writes the CIL onto the RSTS/E system disk and bootstraps the initialization code into memory, the DSKINT option is used to initialize the newly formatted and otherwise blank system disk. Initialization of a newly created system disk is a unique process and is described solely in this section.

CAUTION

> DSKINT destroys any existing file structure on a disk and must not be used on any RSTS/E disk if the file structure is to be preserved. If SYSLOD or CILUS is used to write a new RSTS/E CIL onto an old system disk and the existing file structure is to be preserved, the DSKINT option is not used.

When the SYSLOD or CILUS run terminates, a new system disk contains only a contiguous CIL which is loaded beginning at sector 49 (for RF or RK type disks) or sector 50 for RP02 and RP03 disks and sector 52 for RP04 type disks. The CIL extends for as many blocks as needed to contain the RSTS/E code. Sector 0 of the system disk contains a bootstrap record and sectors 1 through 48 or 49 are reserved for directories and other system files. The pure code of the CIL must be incorporated into a structure which enables the RSTS/E system to access and manipulate its contents. The DSKINT initialization option creates the minimal RSTS/E file structure including a file which maps the CIL on the system disk (RSTS.CIL).

Under normal circumstances, initialization destroys all data on a disk. However, the CIL is preserved if it was bootstrapped into memory. The system disk may be bootstrapped from SYSLOD or CILUS (/BO switch) or by using the hardware bootstrap. The bootstrapping operation provides pointers to the initialization code which are used to determine the size and location of the CIL. With this information the CIL (but not user files) is preserved through the DSKINT process.

The following is a sample dialog, each line of which is denoted by a letter and explained in the subsequent text.

Initialization Options

```
OPTION: DSKINT
DISK ? RP                                       (line a)
UNIT ? 0                                        (line b)
PACK ID ? SYSPAK                                (line c)
PACK CLUSTER SIZE ? 2                           (line d)
MFD PASSWORD ? SYSMFD                           (line e)
MFD CLUSTER SIZE ? 4                            (line f)
PUB, PRI, OR SYS ? SYS                          (line g)
LIBRARY PASSWORD ? SYSLIB                       (line h)
LIBRARY UFD CLUSTER SIZE ? 4                    (line i)
PATTERNS ? 2                                    (line j)
PROCEED Y OR N ? Y                              (line k)
PATTERN # 2
PATTERN # 1

OPTION:
```

this will also be password for acct (0,1)

MFD is for acct (1,1)

acct (1,2) is on Sys pack

bad block checking

Answer '8'

RK 8 minutes

print out bad blocks found

will not re-write over C/L ?

At line a, the system manager types a two-letter designator of the disk that was bootstrapped; valid designators are RF, RK, RP, or RB. The query at line b for an RK, RP, or RB system disk requires the unit number of the device on which the system disk is mounted. For an RF system disk, the query UNIT is replaced by the query PLATTERS, to which the system manager responds by typing the number of RS11 disks (platters) connected to the RF11 controller.

The system manager types from one to six alphanumeric characters in response to the query at line c. The characters typed are called the pack identification or pack label and are used internally by RSTS/E as the password of system account [0,1].

The system manager must specify, at line d, the pack cluster size for the system device. Pack cluster sizes for an RF11 disk or an RK05 or RK03 cartridge can be 1, 2, 4, 8, or 16, or, for an RP02 or RP03 disk pack, 2, 4, 8, or 16. For an RP04 disk, pack cluster sizes are 4, 8, or 16. In general, larger pack cluster sizes permit faster access to data stored on the disk at the expense of possible wasted disk space. Smaller pack cluster sizes permit more efficient allocation of disk storage space at the expense of more frequent access to retrieval information (pointers to file data) stored in user file directories. The latter situation is improved by clustering individual files and user file directories (UFDs) at some cluster size greater than the pack cluster size. Therefore, the minimal cluster size is recommended for almost all installations.

The system manager types from one to six alphanumeric characters in response to the query printed at line e. The characters are used as the password of the MFD account [1,1] on the system device. The MFD password should be kept secret, since irresponsible access to the Master File Directory or for that matter, to any privileged account, can destroy system software.

At line f, the DSKINT routines request the cluster size of the MFD account [1,1]. With the restriction that no cluster size be less than the pack cluster size, the MFD cluster size can be 1, 2, 4, 8, or 16. The DSKINT routines prevent the use of any unacceptable values. The cluster size of the MFD on the system disk determines the maximum number of user accounts that the system manager can create. It is recommended that the MFD cluster size be the same as the pack cluster size unless the system requires a larger number of user accounts than that value allows. The maximum number of user accounts allowed is approximated by multiplying the MFD cluster size by 108.

The system manager must type SYS in response to the query shown at line g to specify initialization of a system disk. The queries at lines h and i concern the password and UFD cluster size of the system library account [1,2]. The password can be up to six alphanumeric characters.

It is recommended that the system manager set the UFD cluster size of the account [1,2] to a minimum value of 4. A larger cluster size may be set if the system manager intends to increase the system library files above those supplied by DIGITAL and those required to build the system library. (The total number of files created under account [1,2] as a result of following the procedures described in Chapter 4 is approximately 60.) The maximum number of files allowed by any account is approximated by multiplying the UFD cluster size by 72.

The query at line j requests the number of patterns (between 1 and 8) to be used when the disk is tested for bad blocks.

The query at line k is printed as a safety measure to allow the system manager to make final checks and to ensure that he desires to proceed. If N is typed, the DSKINT routines are not executed. The initialization routines return to the OPTION query. If Y is typed, DSKINT checks for bad blocks as shown in the example.

3.3.3 Initializing a Non-System Disk

DSKINT is used to initialize, format, and pattern check non-system disks on the RSTS/E system. System disk initialization is described in Section 3.3.2. The examples below apply only to non-system disks.

The following example initializes an RK cartridge as a private disk:

```
OPTION: DSKINT
DD-MMM-YY? 22-JUL-74
HH:MM? 11:30

DISK ? RK
UNIT ? 1
PACK ID ? MYPACK
PACK CLUSTER SIZE ? 1
MFD PASSWORD ? MYMFD
MFD CLUSTER SIZE ? 1
PUB, PRI, SYS ? PRI
FORMAT ? YES
PATTERNS ? 4
PROCEED (Y OR N) ? Y

STARTING RK FORMAT PASS
END RK FORMAT PASS

PATTERN # 4
PATTERN # 3
PATTERN # 2
PATTERN # 1


OPTION:
```

3.3.4 Using DSKINT to Pattern Check a Swapping Disk

The DSKINT option may be used to perform a basic reliability test on RF11, RS03, and RS04 disks configured as swapping disks on the RSTS/E system. Swapping disks do not require the minimal file structure created by DSKINT. Using DSKINT for this purpose writes the minimal file structure on the swapping device but is useless for, and is overwritten, under normal time sharing operations.

Initialization Options

During initial hardware installation or if a hardware malfunction is suspected, DSKINT can supplement the tests performed by standard diagnostic methods. DSKINT does not duplicate all the tests performed by diagnostic programs and should be used only in addition to standard hardware maintenance procedures.

The example below depicts pattern tests on RS03 drive unit 2. Note that DSKINT does not ask whether the drive is an RS03 or RS04. The initialization code reads the drive type registers for RS03 and RS04 units to determine the type and size of the device (RS03 = 1024 256-Word segments, RS04 = 2048 256-Word segments).

```
OPTION: DSKINT
DD-MMM-YY? 22-JUL-74
HH:MM? 12:00

DISK ? RS
UNIT ? 2
PACK ID ? TEST
PACK CLUSTER SIZE ? 1
MFD PASSWORD ? TEST
MFD CLUSTER SIZE ? 1
PUB, PRI, OR SYS ? PUB
PATTERNS ? 8
PROCEED (Y OR N) ? Y

PATTERN # 8
PATTERN # 7
PATTERN # 6
PATTERN # 5
PATTERN # 4
PATTERN # 3
PATTERN # 2
PATTERN # 1


OPTION:
```

If bad blocks are detected on a swapping device (denoted by a table of bad blocks being printed), the swapping disk should normally not be used. Call DEC field service to locate the hardware problem. As a temporary measure, the swap files can be moved to the system disk or several swap files can be used to avoid the malfunctioning portion of the swapping device. Swap files are discussed in Section 3.4.2.4.

3.4 REFRESH OPTION - CREATING THE SYSTEM FILES

Normal RSTS/E system operation requires certain system files under the system files account [0,1]. The DSKINT option creates the system files account and two system files (BADB.SYS and SATT.SYS) required on all disks. When DSKINT is used to initialize a system disk, a third system file (RSTS.CIL) is also created to map the RSTS/E Core Image Library. The remaining system files are created by the REFRESH option of the Initialization code.

Table 3-3 lists the system files with a brief description of each. An overview of the REFRESH option is included below to emphasize the importance of using the option to full advantage when the system disk is first built. Detailed discussions of each system file is then presented to provide background information for proper use of the option. Finally, the details of using REFRESH are presented with several examples.

Table 3-3

Contents of System Account [0,1]

File Name	Location	Status	Description
BADB.SYS	Each disk device in system except swapping disks.	Required	Created by the initialization option DSKINT; all clusters containing bad disk blocks detected during disk initialization are allocated to this file to prevent possible loss of data.
RSTS.CIL	System disk only	Required	RSTS/E Core Image Library containing resident and non-resident (overlay) code, BASIC-PLUS language code, error messages, initialization code, and standalone programs such as ROLLIN.
SATT.SYS	Each disk device in system except swapping disks	Required	Storage allocation table. This file is a bit map used to control allocation of disk space.
SWAP0.SYS	System or swapping disk only	Required	Files used to store user job images not in core.
SWAP1.SYS SWAP2.SYS SWAP3.SYS	System or swapping disk only	Optional	Auxiliary swap files used to store additional user job images not in core.
OVR.SYS	System or swapping disk only	Optional	Copy of non-resident (overlay) code from CIL (RSTS.CIL).
ERR.SYS	System or swapping disk only	Optional	Copy of error messages from CIL (RSTS.CIL) which can be accessed and modified by system manager.
BUFF.SYS	System or swapping disk only	Required for DECtape	File to retain DECtape directories during processing.
CRASH.SYS	System disk only	Optional	File used to store exact image of valuable contents of core following a system crash.

3.4.1 REFRESH Option Overview

The REFRESH option allows the system manager to create, modify, reposition, and, in some cases, delete system files stored under the system files account [0,1]. The system files reside on the system disk or on auxiliary swapping disks. (In either case, these files are catalogued by the directory for account [0,1] on the system disk.) The REFRESH routines assume that the system disk contains at least the minimum RSTS/E file structure as written during the initialization of the system disk using the DSKINT option. The REFRESH operation verifies that the entire disk structure is intact, but only the files stored under account [0,1] are manipulated. Thus, it is possible to refresh a system disk which contains many files and accounts without destroying the existing files.

The REFRESH option must be used to initially create the required system files. This is done immediately after the system disk is initialized with DSKINT and all patches have been installed. Proper use of REFRESH when the system disk is created ensures that changes can be made to the system files at a later date.

Before REFRESH is used, the system manager must understand the principles in the construction of the system disk. The logical structure of the system disk is such that a linear string of unallocated space exists on the disk prior to loading the CIL. When the RSTS/E CIL is written on the system disk, it occupies the lower physical portion of the disk. Initializing the system disk writes a minimum RSTS/E file structure on the disk and assimilates the pure code of the CIL into the minimum file structure.

If the hardware includes ~~one~~ or more swapping disks, these devices are considered logical extensions of the system disk. Figure 3-1 shows how disk sectors are numbered for several possible system and swapping disk configurations. Swapping disks do not require a file structure or directory information to be on the disk. All such information is included in the directories on the system disk. To allocate space for system files on the swapping devices, the initialization code temporarily extends the system disk storage allocation table (SATT.SYS) to include the space on the configured swapping devices. Only the REFRESH option manipulates swapping disk space. When REFRESH completes manipulation of the system files, the allocation table is updated on the system disk. Once written back

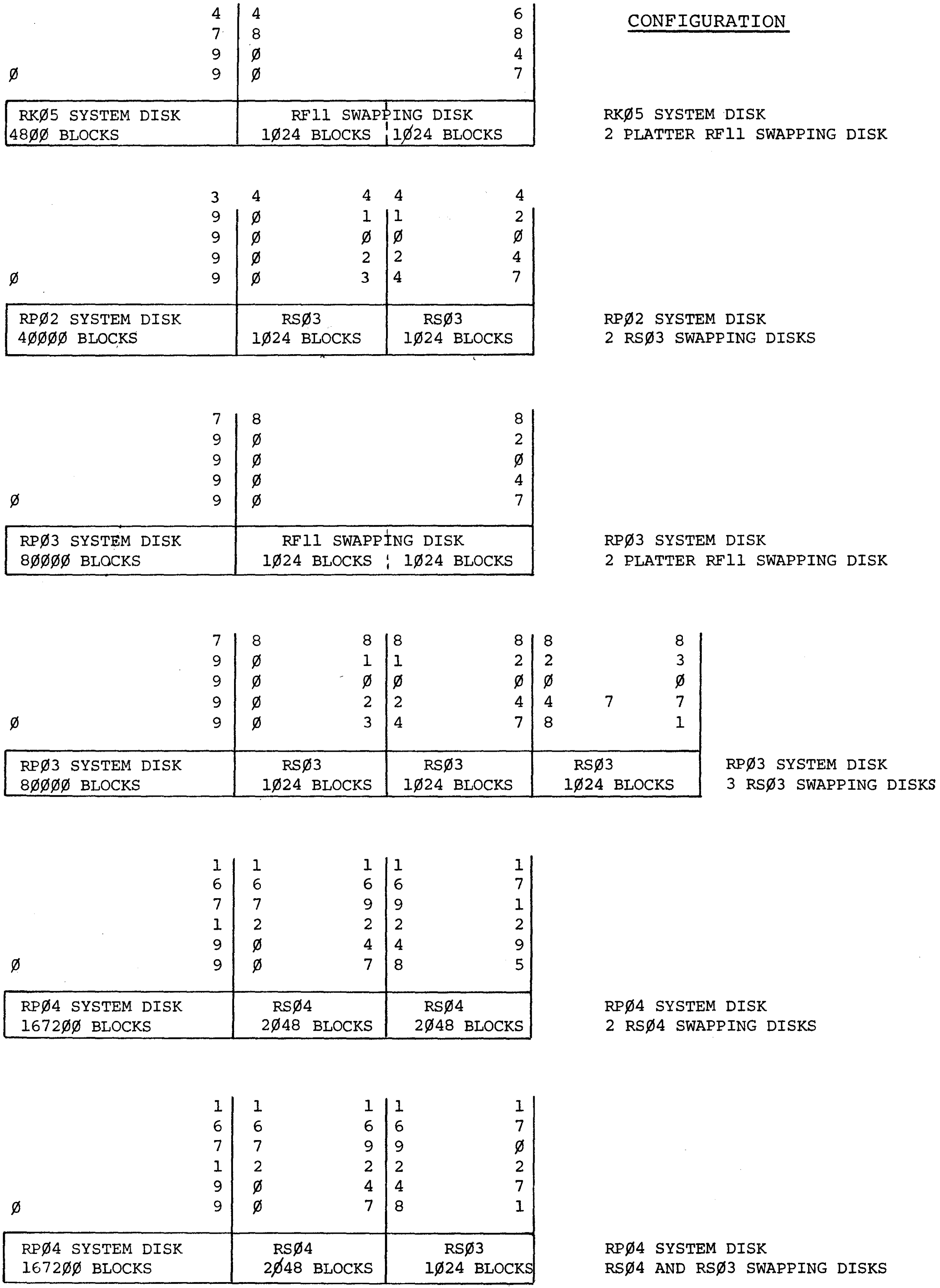


Figure 3-1
Disk Sector Numbers

out to the system disk, the allocation table does not map any space on the swapping disks regardless of whether the space is free or allocated. REFRESH allocates space on swapping devices based on calculated sizes of configured swapping disks. The physical devices are never accessed during REFRESH. Hence the swapping disks need not be physically present for normal REFRESH operation.

The size of RC11 and RF11 swapping disks vary depending on the number of platters. Each RC platter is 64K words or 256 segments; Each RF platter is 256K words or 1024 segments. RF11 and RC11 disks are considered one unit regardless of the number of platters. Hardware permits data transfers to continue across these boundaries. The total space available is the sum of the platters sizes. RSTS/E supports the hardware maximum or eight RF platters or four RC platters.

RS03 and RS04 fixed-head disks are treated as separate units. The system files must fit on a unit and may not cross unit boundaries. Each RS03 is 256K words or 1024 segments; each RS04 is 512K words or 2048 segments. If a system contains both RS03s and RS04s, all RS04s logically follow the system disk and all RS03s follow the RS04s. Physical unit numbers need not be known to use REFRESH. Physical-to-logical unit mapping is done at Start up time. These distinctions are important when trying to locate system files on particular RS units. RSTS/E supports only four RS disk units for swapping but additional units may be attached to the RH11 controller for use by other operating systems.

If the REFRESH operation proceeds without the system manager's requirements, only required system files are created on the system disk. These files are allocated the minimum required storage space in disk sectors immediately following the space occupied by the CIL. A typical representation appears as follows:

Physical
Sector

0	1 48	49	289		
Boot-strap	MFD[1,1] UFD[0,1] UFD[1,2] SATT.SYS	RSTS/E CIL (RSTS.CIL)	BUFF.SYS	SWAP0.SYS	Unallocated Space

Such distribution of the required system files is seldom optimal. For example, the default size of SWAP0.SYS can handle only the configured number of jobs at the default size of 8K words. Most RSTS/E systems will run BASIC-PLUS programs larger than 8K. Furthermore, REFRESH locates all the system files on the system disk, any swapping disks are not used.

At the time the system manager builds the system disk, he has available all the contiguous unallocated storage space. If he does not use this file space, he may be limited by the existing structures at a later time, since time sharing operations cause storage space on the system disk to be allocated randomly on an as-needed basis. The free contiguous space at the front portion of the system disk can be allocated to user and system program and data files, thus limiting and restricting later structural changes to the system files in account [0,1] on the system disk.

A few examples can show the significance of fully using REFRESH. The RSTS.CIL file is created at the minimum required size during the DSKINT process. (The minimum required size is actually rounded up to span an integral number of disk clusters, as determined by the pack cluster size.) The contiguous storage space at the lower portion of the system disk immediately following the RSTS.CIL file is allocated to whatever other system files are created by REFRESH. During time sharing operations, subsequent contiguous storage space is allocated to user and system program and data files. Future expansion of storage space occupied by CIL becomes restricted.

If the system manager wishes to augment the system at some future date with either new hardware or software features, the size of the CIL will necessarily increase by an undetermined amount. The probability of successfully incorporating a newly generated CIL into an existing file structure is greatly improved if the system manager increases the size spanned by the CIL when the system files are initially built on the system disk. The REFRESH option provides this capability. Thus, to allow for future changes, the system manager specifies from 20 to 40 extra blocks for the RSTS.CIL file. The extra contiguous space is then allocated to the CIL file at the lower portion of the disk, allowing for possible future, successful incorporation of a new CIL without destroying system and user files on the system disk.

Initialization Options

For certain system files, the position of those files on the disk influences the efficiency of the system operation. For systems having a moving head system disk without a swapping disk, the system manager can position certain system files in the center portion of the disk. This appreciably reduces the average amount of SEEK time since the moving heads are not operating back and forth from opposite ends of the disk. Again, allocation of oversize files when the disk is built ensures that future growth can be accommodated. An attempt to position the files on the disk later at an optimal location becomes increasingly difficult as more of the contiguous file space is allocated during time sharing operations.

The system manager has the capability to structure the system files by responses to the many queries printed by REFRESH. The response to each file-related query determines how and where each system file is accessed and positioned. REFRESH queries are described in Section 3.4.3.3. The next section provides detailed information on each of the system files.

3.4.2 System Files

The system manager must know the purpose of each system file (Table 3-3) to use REFRESH properly. This section contains more detailed information on each of the system files noting the operations performed by REFRESH.

3.4.2.1 BADB.SYS - Bad Block File

The bad block file permanently allocates bad disk segments and makes them unavailable for use by system and user files. The bad block file is required on all disks (except swapping disks) used under RSTS/E. The file length can be zero if there are no known bad blocks. BADB.SYS is created during the pattern checks performed by the DSKINT option. A bad block is defined as a disk segment to which data can be written but the same data cannot be retrieved. Disks are usually not guaranteed to be error free. Hence, it is important to perform the pattern tests when the disk is initialized. If bad blocks are found, the clusters (equal to the pack cluster size) containing the bad blocks are allocated to the bad block file. REFRESH only prints a list of bad blocks: the capability for adding blocks to the bad block file has not yet been implemented.

3.4.2.2 RSTS.CIL - Core Image Library

The RSTS.CIL file contains the initialization code, monitor code, BASIC-PLUS compiler and Run Time System, overlay code, error messages, and usually the stand-alone program ROLLIN. This file exists only on the system disk. The CIL is loaded during system generation by either the DOS/BATCH program CILUS or the stand-alone program SYSLOD. When the system disk is initialized using DSKINT the CIL is preserved and mapped into the RSTS.CIL file.

The RSTS.CIL file cannot be moved by REFRESH and will always be located on the lower physical portion of the system disk. DSKINT creates the RSTS.CIL file at its minimum required size in the range of 250-350 disk segments of 256 words each. The size of the CIL depends on the monitor and BASIC-PLUS options included. As stated earlier, the size of the RSTS.CIL file can be increased by REFRESH to allow replacement of the CIL at a later time without disturbing the existing file structure on the system disk. The system manager performs new system generations as the installation requirements change. Hence, the CIL file should be made large enough to cover any foreseeable additions to the system. At the time of the initial system generation, increasing the size of RSTS.CIL by 20 to 40 segments over the required size allows for replacement by a substantially larger monitor. If the contiguous file space immediately following the RSTS.CIL file is allocated to user files under normal time sharing, then it may not be possible to increase the size of the CIL at a later time.

3.4.2.3 SATT.SYS - Storage Allocation Table

The storage allocation table is a bit map used to keep track of disk space. Each bit in the map represents one disk cluster of n 256-word segments where n is the pack cluster size. A one bit represents an allocated (in use) cluster and a zero bit represents a free cluster. The SATT.SYS file is required on all disks (except swapping disks) and is created by DSKINT. The disk space mapped by the SATT.SYS file on each disk includes only the actual space on that disk. The size of the file varies depending on disk size and the pack cluster size. The maximum size is 10 segments for all disk devices currently supported under RSTS/E.

REFRESH will permit repositioning of the SATT.SYS file on a moving head system disk. This capability is primarily used when no swapping disk exists and the swap files and other system files must also be located on the moving head device. Grouping all system files near the center of the disk tends to reduce average seek time for the moving heads.

3.4.2.4 SWAPn.SYS - SWAP Files

During normal timesharing operations, user job images are frequently moved (swapped) to disk in order that other jobs can be loaded into memory and executed. The REFRESH option is used to create the required swap areas or swap files according to installation requirements. Under RSTS/E there are four possible swap files called SWAP0.SYS, SWAP1.SYS, SWAP2.SYS, and SWAP3.SYS. These system files may be located on the system disk, on auxiliary swapping disks, or on combinations of the system and swapping disks. Guidelines for creating an efficient swapping structure and the RSTS/E swap algorithm are described below. The material presented here should be fully understood before REFRESH is used to create the swap files.

SWAP0.SYS is the only required swap file. However, other swap files may be needed depending on the hardware configuration and the installation requirements. The sizes of existing swap files limit the values of JOBMAX and SWPMAX which can be specified when using the DEFAULT and START initialization options. JOBMAX is the maximum number of jobs which can be run simultaneously. JOBMAX must always be less than, or equal to the maximum number of jobs configured during system generation. SWPMAX may be set to any number between 8 and 28 K) words.[1] For the purpose of calculating the size of swap files, no provision is made for limiting certain jobs or even certain numbers of jobs to less than the system wide SWPMAX. Thus if any job is allowed to expand to 16K, the amount of swap space reserved for all jobs must be 16K each.

Each of the swap files holds an integral number of user job images determined by the size of the file and the desired SWPMAX. The following equations describe the relations between size of the swap files, SWPMAX, and JOBMAX.

$$\text{JOBS for SWAPn.SYS} = \frac{\text{Size of SWAPn.SYS in segments}}{(\text{Desired SWPMAX}(\underset{w}{\text{K}}) * 4)} \quad \text{(truncated)}$$

If more than one swap file is created, the total number of jobs which can be run simultaneously is equal to the sum of jobs accommodated by each swap file.

$$\text{JOBMAX} = \sum_{n=0}^{3} \text{JOBS for SWAPn.SYS} \leq \text{Configured number of jobs.}$$

---

[1] The BASIC-PLUS language code limits the maximum size of BASIC-PLUS programs to 16K words.

The RSTS/E monitor uses four swap bit maps (one for each of the four possible swap files) to control allocation of swap space. Each bit map contains 4 words or 64 bits and each bit represents a swap "slot" of size SWPMAX. In the bit maps, a zero bit represents an allocated slot and a one bit represents a free slot. Bit 0 (out of 64) in each map is always 0 (allocated). Thus each swap file and its corresponding map can handle the theoretical maximum of 63 jobs, or up to 63 jobs could be spread over 1 to 4 swap files (see Figure 3-2).

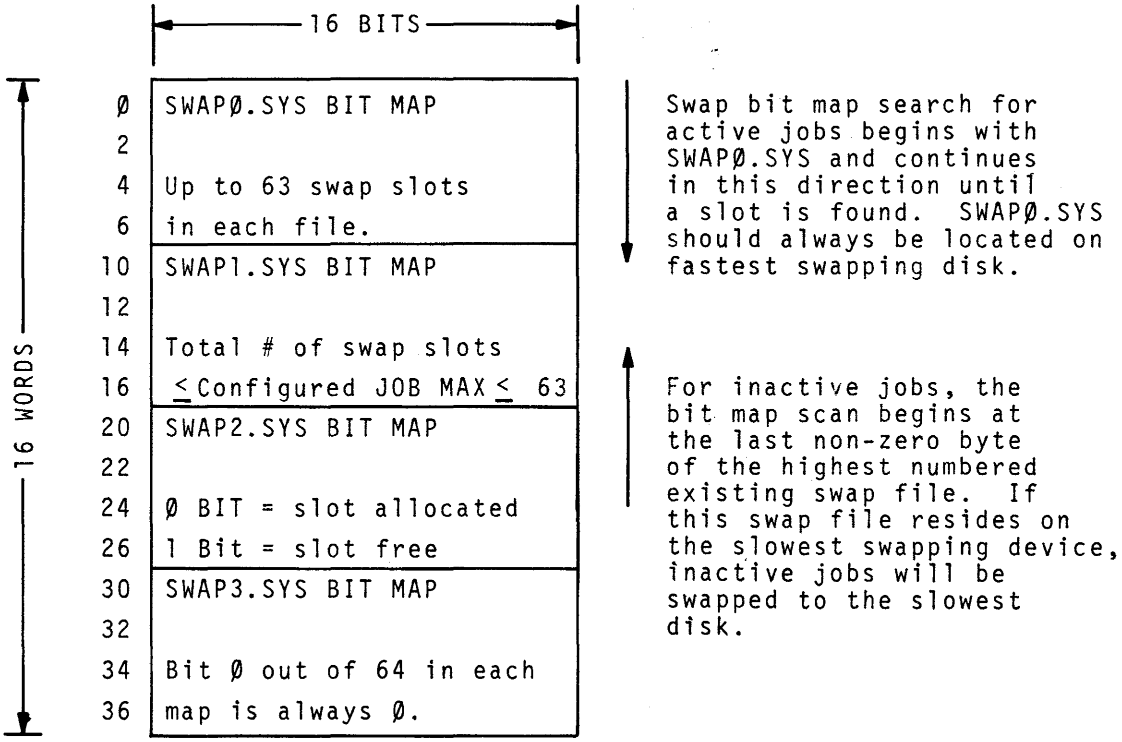
Swap slots are dynamically allocated when a job is swapped out to disk. The bit map is scanned forward for actively running jobs. Hence, if SWAP0.SYS is located on the fastest swapping device, active jobs will be swapped to the fastest medium. If a job is inactive, the bit map scan is reversed. Hence, inactive jobs will be swapped to the higher numbered swap files.

This mechanism is used for several reasons. First, to handle multi-unit swap structures such as multiple RS03 and/or RS04 fixed head disks. RS03s and RS04s are addressed by unit in the same way that RP03 or RK05 drives are referenced by unit. A data transfer may not continue across a physical unit boundary. The RF11 and RC11 fixed head disks are addressed as one unit regardless of the number of platters. For these devices, a data transfer may continue across platter boundaries. Secondly, this method allows systems which only occasionally run many jobs, to operate with less fast swapping space. Jobs which would overflow swapping space on the swapping disk(s) are simply swapped into a file on the system disk.

To demonstrate proper and efficient use of the swap files, several systems are considered. The first system requires only a single swap file to satisfy system requirements. The second case is when two swap files are used. The third is a large system that must use three swap files.

For the first example, consider a system that runs 15 jobs with the maximum size of 16K words. The required swap space in disk segments is calculated as

SWAP SIZE = JOBMAX * SWPMAX (in K) * 4 (SEGMENTS PER K)



Swap bit map search for active jobs begins with SWAP0.SYS and continues in this direction until a slot is found. SWAP0.SYS should always be located on fastest swapping disk.

For inactive jobs, the bit map scan begins at the last non-zero byte of the highest numbered existing swap file. If this swap file resides on the slowest swapping device, inactive jobs will be swapped to the slowest disk.

Figure 3-2. SWAP Bit Maps

In this case 15 jobs at 16K requires 960 disk segments. Also assume that the hardware configuration includes a single platter RF11 disk (1024 segments) as the swapping device. Note that the same numbers apply for a single RS03 unit which is also 1024 segments. Since the required swap space is available on the swapping device, a single swap file is sufficient to meet installation requirements. The REFRESH option is used to create SWAP0.SYS on the swapping disk at size equal to 960 segments.

It is also advantageous to locate other system files including OVR.SYS (overlay code), ERR.SYS (error messages), and BUFF.SYS (DECtape buffers) on the swapping device to speed up access to these files. In the example above, 64 disk segments remained on the swapping disk. This is more than enough space to hold the overlay code, error messages, and any required DECtape buffers. These system files would be created and located on the swapping disk using the REFRESH option.

If the installation requirements for the system described above are increased to 20 jobs at 16K, the amount of swap space required would increase to 1280 disk segments. With the physical limit of 1024 segments on the RF11 disk, a second swap file must be used to accommodate the overflow. The system files OVR.SYS, ERR.SYS, and BUFF.SYS could be moved to the system disk to free up space for one additional job on the swapping disk. Fast access to these files is more important than one extra job on the swapping disk, however. The previous structure will be left intact and a second swap file added to handle the five additional jobs. SWAP1.SYS would be created on the system disk at size equal to 320 segments. Note that this arrangement will have little effect on system performance since inactive jobs are swapped on the slower system disk and active jobs are swapped on the faster device.

For an example of the third system, consider a very large system used to service several elementary schools. During the day, the system must service up to 48 student jobs. At non-peak hours the system is also used for large administrative programs. The installation requires a JOBMAX of 48 and SWPMAX of 16K due to the administrative processing performed.

The hardware configuration includes two RS03 fixed-head disks (1024 segments each) for swapping. Three swap files are required to handle the peak load of 48 jobs at 16K. SWAP0.SYS is created on the first RS03 to handle 16 jobs (1024 segments). SWAP0.SYS completely fills the first RS03 unit. SWAP1.SYS resides on the second RS03, but handles only 15 jobs (960 segments) to leave room for the other system files. Finally SWAP2.SYS is located on the system disk and handles the remaining 17 jobs (1088 segments).

Note that if the swapping device is two-platter RF11 disk rather than the two RS03 units, only one swap file is needed to cover both RF11 platters. As mentioned above, data transfers may cross platter boundaries on the RF11 but may not cross RS03 or RS04 unit boundaries. Hence, it was necessary to create two distinct swap files to use both RS03s. The BASE query described in Section 3.4.3.3 provides the mechanism by which the system files are located on specific RS03 or RS04 units.

If the hardware configuration does not include a swapping disk, a single swap file (SWAP0.SYS) on the system disk is normally used for all jobs. Several swap files are used only if the required contiguous file space is not available to handle all jobs in one swap file.

The REFRESH option blindly creates the swap files ensuring only that they fit on the configured system disk and/or swapping disk(s). The DEFAULT and START options described in Sections 3.5 and 3.6 ensure proper correlation between JOBMAX, SWPMAX, and the sizes of the swap files created by REFRESH. The START option, furthermore, loads the monitor swap bit maps for use during normal timesharing.

3.4.2.5 OVR.SYS - Overlay Code

The non-resident (overlay) code in the RSTS/E system may be accessed directly from the Core Image Library file RSTS.CIL or it may be accessed as a separate file OVR.SYS. If the system disk is a fixed-head disk, there is no need to create a separate copy of the non-resident code. However, if the system disk is a moving-head device and an auxiliary swapping disk is available on the system, performance is increased by having the non-resident code as a separate file on the fast swapping disk.

REFRESH allows the OVR.SYS file to be created and positioned wherever desired. If a swapping disk is not included in the hardware configuration but the system files are being positioned near the center of a moving head disk (to reduce average seek time), OVR.SYS could be created and positioned with the other system files.

The size of the overlay code is the same for all systems. During system generation some of the overlay code may be made resident to increase system performance. The size of OVR.SYS is not reduced by making parts of the overlay code resident. If OVR.SYS is created, REFRESH allows its size to be greater than the required size. This provides a means to reserve contiguous file space on the system disk.

If the OVR.SYS file is created, the system will access the overlay code from this file. During REFRESH, the file contains no useful information and is not accessed in any way. The overlay code is transferred from the CIL to the OVR.SYS file when the system is brought up for normal time sharing with the START option.

3.4.2.6 ERR.SYS - Error Messages

Like the non-resident code, the RSTS/E system error messages may exist as a separate file, ERR.SYS, or they may be accessed directly from the RSTS.CIL file. If the system manager wishes to modify the error messages or have access to them on line, the ERR.SYS file must be created. Otherwise, the choice is based on fast access to the error messages and reducing the accesses to the system disk.

If the system disk is a fixed-head disk, there is normally no need to create the separate file except to allow modifications. If the system disk is a moving-head device and an auxiliary swapping disk is available on the system, the error messages can be retrieved faster if they exist as a separate file on the fast swapping device. The separate file on the swapping disk also reduces the burden on the system disk and helps in a small way to speed user file processing on the system disk. As in the case of the overlay code, REFRESH permits the ERR.SYS file to be created and positioned wherever desired. If a swapping disk is not included in the hardware configuration but the system files are being positioned near the center of a moving head disk, ERR.SYS could be created and positioned with the other system files.

The size of ERR.SYS is 2K words (8 blocks) for all systems. REFRESH allows the size of ERR.SYS to be greater than this required size to reserve contiguous file space. If the ERR.SYS file is created, the system reads the error messages from the file rather than the CIL. During REFRESH, the ERR.SYS file contains no useful information and is not accessed in any way. The error messages are transferred from the RSTS.CIL file to the ERR.SYS file when the system is brought up for normal timesharing with the START option.

3.4.2.7 BUFF.SYS - DECtape Buffers

DECtape processing in the RSTS/E system requires three 256-Word blocks of buffer storage for each drive unit in the system configuration. Thus, one TU56 Dual DECtape transport requires six blocks of buffer storage space. The file BUFF.SYS provides the required buffer space for DECtape operations. BUFF.SYS is required only on systems configured for DECtape.

The required size of BUFF.SYS is fixed by the number of configured DECtape units. REFRESH allows the size of BUFF.SYS to be increased to any number of blocks greater than the required size of three blocks per unit. This facility should be used if additional DECtape transports are added in the future. BUFF.SYS may be located on the swapping disk if one exists or may be positioned anywhere desired on a moving head system disk. On systems without DECtape, BUFF.SYS may be created to reserve contiguous file space on the system disk.

3.4.2.8 CRASH.SYS - Crash Dump File

The crash dump file CRASH.SYS is always optional but, if present, must reside on the system disk. The CRASH.SYS file must exist if the crash dump facility is to be enabled at the start of time sharing operations when using the START option or when establishing default start-up conditions using the DEFAULT option.

If the RSTS/E error handling code detects an error from which it cannot recover, the system is reloaded. In this controlled-crash sequence, the contents of the read/write area of monitor memory is dumped to the CRASH.SYS file. For example, if a memory parity error occurs in the area of memory occupied by the monitor or the BASIC-PLUS Run Time System, reloading the system is the safest recovery. This is done by the parity error handling code in the monitor. Since the dump preserves the state of the system at the time of the crash, it contains useful diagnostic information. Parity errors, for example, are logged by the error logging routines in the monitor. There is, however, a finite delay between the time an error is logged and the time this information can be picked up by the ERRCPY program. If the error causes a crash or if a crash occurs for any reason before ERRCPY can save the error information, this data (and probably the cause of the crash) will be lost if the dump was not performed. Hence, it is important that CRASH.SYS exist and the crash dump facility is enabled.

The size of CRASH.SYS depends on the size of the read/write portion of the monitor. The RSTS/E Monitor includes many tables and data areas which change during normal operation. This information resides in the lowest portion of memory (lowest physical addresses). The rest of the Monitor is system code (instructions) which is never changed under time sharing. Similarly, the BASIC-PLUS compiler and Run Time System is pure code which never changes. This "read only" code need not be dumped at the time of a crash. The size of the read/write area depends on the hardware and software configuration at each site. Hence, the size of CRASH.SYS is fixed at system generation time. The size is not the same on all systems but is normally 8 to 16K words (32 to 64 blocks).

REFRESH provides only the option to create or delete the CRASH.SYS file. If created, the size of CRASH.SYS will be the required size as discussed above. Furthermore, it will reside on the system disk wherever REFRESH can find sufficient contiguous file space.

3.4.3 Using REFRESH

The REFRESH option is invoked by typing REFRESH or simply RE in response to the initialization OPTION query. REFRESH responds by asking for the date and time followed by a series of questions. The complete REFRESH example on the following pages is included without comment to show the general flow of the REFRESH option. This example is repeated with detailed comments in Section 3.4.4.1. The sections below explain each of the REFRESH queries and the File Status Table.

```
OPTION: REFRESH
DD-MMM-YY? 22-JUL-74
HH:MM? 12:30

OLD ? NO

FILE                                    CURRENT  REQUIRED   START    START
NAME          REQUIRED? EXIST STATUS     SIZE      SIZE    CLUSTER  SECTOR

BADB   .SYS       YES    SYS     OK         0         0
RSTS   .CIL       YES    SYS    D/C       274       274       24       50
SATT   .SYS       YES    SYS    D/C        10        10        2        6
SWAP0  .SYS       YES    NO     CRE         0      1280
SWAP1  .SYS        NO    NO      OK         0         0
SWAP2  .SYS        NO    NO      OK         0         0
SWAP3  .SYS        NO    NO      OK         0         0
OVR    .SYS        NO    NO      OK         0        28
ERR    .SYS        NO    NO      OK         0         8
BUFF   .SYS       YES    NO     CRE         0        12
CRASH  .SYS        NO    NO      OK         0        37

BADS ? LIST

THERE ARE NO BAD BLOCKS

BADS ?  <LF>
```

<LF> denotes typing the LINE FEED key

```
RSTS.CIL CHANGES ? YES

SIZE ? 300

SATT.SYS CHANGES ? YES

BASE ? 40000

SWAP0.SYS CHANGES ? YES

SIZE ? 1024

DISK ? SWP

BASE ? <LF>

SWAP1.SYS CHANGES ? YES

SIZE ? 960
```

```
DISK ? SWP

BASE ? 81024

SWAP2.SYS CHANGES ? YES

SIZE ? 1088

DISK ? SYS

BASE ? 40000

SWAP3.SYS CHANGES ? NO

OVR.SYS CHANGES ? YES

CIL ? NO

SIZE ? <LF>

DISK ? SWP

BASE ? <LF>

ERR.SYS CHANGES ? YES

CIL ? NO

SIZE ? <LF>

DISK ? SWP

BASE ? <LF>

BUFF.SYS CHANGES ? YES

SIZE ? <LF>

DISK ? SWP

BASE ? <LF>

CRASH ? YES

FILE                                   CURRENT  REQUIRED   START    START
NAME        REQUIRED? EXIST  STATUS      SIZE     SIZE    CLUSTER   SECTOR

BADB   . SYS     YES     SYS     OK         0        0
RSTS   . CIL     YES     SYS     OK       300      274       24       50
SATT   . SYS     YES     SYS     OK        10       10    19999    40000
SWAP0  . SYS     YES     SWP     OK      1024     1280    39999    80000
SWAP1  . SYS      NO     SWP     OK       960        0    40511    81024
SWAP2  . SYS      NO     SYS     OK      1088        0    20004    40010
SWAP3  . SYS      NO      NO     OK         0        0
OVR    . SYS      NO     SWP     OK        28       28    40991    81984
ERR    . SYS      NO     SWP     OK         8        8    41005    82012
BUFF   . SYS     YES     SWP     OK        12       12    41009    82020
CRASH  . SYS      NO     SYS     OK        38       37      174      350


OPTION:
```

<LF> denotes
typing the LINE
FEED key

3.4.3.1 Selecting the Short- or Long-Form of REFRESH

Immediately after the date and time are entered, REFRESH prints the OLD query. The response determines the type of REFRESH operation performed. When a new system disk is built, there are two basic alternatives. The system manager may let REFRESH do all the work and accept the default structure of the system files (short form). Alternatively, he may enter specifications for each system file (long form). One of these two options is selected by the response to the OLD query. Valid responses are listed in Table 3-4.

If the short form is selected, REFRESH creates the system files without further interaction. The answers to all file related queries are extracted from the existing file structure and configuration parameters. All required files are created at their required sizes and reside on the system disk. Optional files are not created at all. This procedure is not recommended since it seldom provides an optimal structure for the system files. The short form of REFRESH is primarily used on test systems where optimal structures are not required. The ability of REFRESH to extract answers to file related queries is helpful when REFRESH is used to alter the system files as installation requirements change. The long form of REFRESH is used but file related queries can be answered with the LINE FEED key to leave a current file specification unchanged.

Table 3-4

Valid Responses to OLD Query of REFRESH

Response	Meaning
YES or OLD	The short form of REFRESH. File characteristics are extracted from existing file structure and configuration parameters. Only nonexistent required files are created. Not recommended.
Y, O, or LINE FEED key	Same as YES but the first File Status Table is not printed.
NO	The long form of REFRESH. Individual file characteristics will be entered. REFRESH subsequently prints queries for each system file. Recommended.
N	Same as NO except the first File Status Table is not printed.
Anything Else	Repeats the OLD query.

The answer to the OLD query also determines whether the File Status Table is printed before the REFRESH operation is performed. The table is always printed before the option terminates to show the current structure of the system files. Normally, both tables are useful for reference. If several passes through REFRESH are needed due to some oversight, the table printed at the end of the previous pass is sufficient reference. Time can be saved by not printing the table on a subsequent try.

3.4.3.2 File Status Table

A File Status Table is printed twice during REFRESH. It is printed before any REFRESH action so that the system manager can see the current status of each system file and any REFRESH action required. The table is also printed at the end of the REFRESH operation to reflect any changes made. The status information given in the printout is summarized in Table 3-5.

The entry in the REQUIRED column indicates whether the file in that row of the table is necessary for operation of the RSTS/E system. The files BADB.SYS (bad block file) and SATT.SYS (storage allocation table) are required on all RSTS/E disks. The RSTS.CIL file is required on the system disk since it contains the permanent copy of the system code. One swap file (SWAP0.SYS) is required while the others are optional. Separate copies of the overlay code and error messages may exist apart from the CIL as files OVR.SYS and ERR.SYS. These two files are not considered required since the information they contain may be accessed directly from the RSTS.CIL file. BUFF.SYS (DECtape buffers) is only required on systems configured for DECtape. Finally, the crash dump file (CRASH.SYS) is recommended for all systems but is not required.

The EXIST column merely indicates whether the file exists on the system disk (SYS), on the swapping disk or disks (SWP), or does not exist at all (NO). The entries in the STATUS column indicate to the system manager, the action which REFRESH must take independent of any changes he may specify. The STATUS column indicates that no REFRESH action is required (OK), the file must be created (CRE), or it must be deleted and then re-created (D/C). If any REFRESH action is required or requested on any of the system files, the other existing system files are also deleted and recreated. This is done to free up as much contiguous file space as possible before any attempt is made to restructure the system files.

Entries in the CURRENT SIZE and REQUIRED SIZE columns of the file status table display the decimal number of 256-word blocks occupied by and necessary for each system file. The values shown in the REQUIRED SIZE column depends on either the fixed system requirement for the particular file or the configuration of the system. For example, ERR.SYS is always 8 blocks (2K words) for all systems whereas the required size of the RSTS.CIL file depends on the hardware and

Table 3-5

System Files Status Table Entries

Status Table Entry Heading	Possible Entries	Meaning
FILE NAME	All system files in account [0,1]	See Section 3.4.2 for descriptions of system files in account [0,1].
REQUIRED?	YES	The file is currently necessary on the local installation system.
	NO	The file is not needed to operate the system.
EXIST	NO	The file does not currently exist.
	SYS	The file currently exists on the system disk.
	SWP	The file currently exists on the swapping disk.
STATUS	OK	No REFRESH action is required.
	CRE	File is marked for creation.
	DEL	File is marked for deletion.
	D/C	File is marked for deletion and re-creation.
CURRENT SIZE	n	The decimal number of 256-word blocks the file occupies on the disk.
REQUIRED SIZE	n	The decimal number of 256-word blocks the file minimally requires.
START CLUSTER	n	The decimal number of the first logical cluster occupied by the file.
START SECTOR	n	The decimal number of the physical sector occupied by the first 256-word block of the system file. Disk sectors are numbered from 0 to (disk size -1)

software configuration at each site. REFRESH normally does not permit the system manager to create any required file at a size smaller than the REQUIRED SIZE. The one exception is the SWAP0.SYS file. The REQUIRED SIZE shown for SWAP0.SYS handles the configured number of jobs at a maximum size (SWPMAX) of 8K. Since the actual number of jobs may be anything less than or equal to the configured maximum, the size of SWAP0.SYS may be less than, equal to, or greater than the stated REQUIRED SIZE. The absolute minimum size allowed for SWAP0.SYS is 32 blocks (1 job at 8K). For files such as OVR.SYS and ERR.SYS which are not explicitly required for correct system operation, the REQUIRED SIZE column indicates the number of blocks needed if the file is created.

The values in the CURRENT SIZE and REQUIRED SIZE columns may differ for several reasons. The most obvious is a system file which does not yet exist. The CURRENT SIZE will be zero while the REQUIRED SIZE will reflect the fixed system or configuration requirement. Secondly, the system manager may change the size of certain system files to meet installation needs. One such possibility with SWAP0.SYS was mentioned above. The CURRENT SIZE of SWAP0.SYS may be less than, equal to, or greater than the required size. Furthermore, if the system manager chooses to plan for future expansion, the CURRENT SIZE of system files such as RSTS.CIL and BUFF.SYS may be greater than the REQUIRED SIZE. Finally, system files occupy space to the nearest cluster boundary. Thus, if the REQUIRED SIZE is not an integral number of clusters at the system disk pack cluster size, REFRESH automatically rounds up to an integral number of clusters when the file is created. This round up is reflected in the CURRENT SIZE of the file. The sizes of the system files can be changed by response to the SIZE query printed by REFRESH. In all cases, REFRESH will prevent changes to the system file which are explicitly illegal.

The START CLUSTER and START SECTOR columns indicate, respectively, the logical and physical beginning address of the system file in question. The START CLUSTER column is a diagnostic aid to the designers. It has little practical value to the system manager. The START SECTOR column indicates the actual physical location of the file. Each of the system files occupies a contiguous area on the system disk or swapping disk(s). Disk sectors (256-word blocks) are numbered from 0 to the size of the disk minus one. An RK05 DECpack contains 4800 sectors or blocks numbered 0 to 4799. An RP02 contains 40000 sectors numbered 0 to 39999. An RP03 pack contains 80000 sectors numbered 0 to 79999. RP04 sectors are numbered 0 to 167199. Since swapping disks are considered logical extensions of the system disk, swapping disk sectors are numbered starting at the end of the system disk (e.g., 4800 and up for an RK05 system disk, 40000 and up for an RP02 system disk, 80000 and up for an RP03 system disk and 167200 and up for an RP04 system disk). Thus, by looking at the START SECTOR column, it is possible to know exactly where the system files reside. The BASE query of REFRESH permits locating certain system files at any desired start sector subject to availability of contiguous file space.

3.4.3.3 REFRESH Queries - All of the file related queries are listed for reference in Table 3-6. Additional comments are included below.

The <filename> CHANGES query allows an existing system file to be left unchanged by use of the NO, OLD, or LINE FEED response. When necessary to accommodate other system files, REFRESH repositions the system files even if no change is requested. The size of a file and the disk used are not changed, however. The situation is different if a required file does not yet exist. This situation arises when a new system disk is built. REFRESH creates a required file at the required size on the system disk if the CHANGES query is answered with the NO, OLD, or LINE FEED response.

Table 3-6

REFRESH File-Related Queries

File-Related Query	Applicable Files	Possible Responses	Meaning
BADS?	BADB.SYS	LIST	Print the start sector numbers of clusters which contain bad blocks.
		ADD	ADD option not yet implemented.
<filename> CHANGES?	RSTS.CIL SATT.SYS SWAP0.SYS SWAP1.SYS SWAP2.SYS SWAP3.SYS OVR.SYS ERR.SYS BUFF.SYS	YES	Causes more queries to be printed each of which is directly related to the possible changes that can be made to the specific system file. (See individual file descriptions in Section 3.4.2.)
		NO, OLD or LINE FEED key	Causes the answers to the related filename changes queries to be extracted from the existing file structure or from system configuration parameters.

(continued on next page)

Table 3-6 (Cont.)
REFRESH File-Related Queries

File-Related Query	Applicable Files	Possible Responses	Meaning
SIZE?	RSTS.CIL SWAP0.SYS SWAP1.SYS SWAP2.SYS SWAP3.SYS OVR.SYS ERR.SYS BUFF.SYS	OLD or LINE FEED key	If the file exists, retain the current size. If the file is being created, create at the required size.
		n	An integer decimal number which specifies the size in 256-word blocks. The value n must be greater than or equal to the required size except for SWAP0.SYS which may be any size greater than 32 blocks.
DISK?	SWAP0.SYS SWAP1.SYS SWAP2.SYS SWAP3.SYS OVR.SYS ERR.SYS BUFF.SYS	OLD or LINE FEED key	The file is written to the disk on which it previously resided. If file is newly created, place it on the system disk.
		SYS	The file is created on the system disk.
		SWP	The file is created on the swapping disk.
BASE?	SATT.SYS SWAP0.SYS SWAP1.SYS SWAP2.SYS SWAP3.SYS OVR.SYS ERR.SYS BUFF.SYS	OLD or LINE FEED key	If the file existed on the disk, causes the file to be placed at the base physical sector number used previously. If the file is new, the base cluster value is the default of 0 (anywhere on the disk).
		n	An integer decimal number which specifies the base sector at which the file begins on the disk device. Sectors are numbered from 0 to the capacity of the disk (in 256-word blocks) minus 1. See Figure 3-1.
		0	If the number 0 is specified, the file is placed anywhere on the disk.

(continued on next page)

Table 3-6 (Cont.)
REFRESH File-Related Queries

File-Related Query	Applicable Files	Possible Responses	Meaning
CIL?	OVR.SYS ERR.SYS	YES	Indicates that the system accesses directly the relevant code in the CIL file (RSTS.CIL) on the system disk. Any existing copy of the file is deleted.
		NO	Indicates that the system manager desires a separate copy of the relevant code to be created. The file is created and the contents of the CIL code is copied to the file at start up time.
		OLD or LINE FEED key	Retain the file if it previously existed; otherwise, the code in the CIL is accessed directly.
DELETE?	BUFF.SYS SWAP1.SYS SWAP2.SYS SWAP3.SYS	YES	Causes file to be marked for deletion.
		NO or LINE FEED key	Causes more queries to be printed: SIZE?, DISK?, and BASE?.
CRASH?	CRASH.SYS	YES	Causes the file CRASH.SYS to be created.
		NO	Causes the file CRASH.SYS to be deleted.

The SIZE query is printed for several fixed-length system files. This provides a way to reserve contiguous file space on the system disk to allow for future expansion of the system files and provide for a swapping disk failure. For example, on a system with a single-platter RF11 (or RS03) swapping disk, a swap file normally resides on this swapping device. In the event of a swapping disk hardware failure, the system can continue to operate normally if this swap file is moved to the system disk. REFRESH moves the swap file if there is sufficient contiguous space available on the system disk. This space will probably not be available unless corrective action for such failures are considered when the system disk is initially built. By locating one of the system files such as BUFF.SYS (whether required or not) on the system disk and making its size equal to 1024 blocks (size of the RF11) plus the required size of BUFF.SYS, the system can operate normally even if the swapping disk is down. REFRESH is used to move all files from the swapping disk to the system disk and BUFF.SYS is reduced to the required size. The files which previously resided on the swapping disk merely fill in the previously unused area of the BUFF.SYS file. The reverse procedure is performed when the swapping disk is again operational. The system files are moved back to the swapping disk and the size of BUFF.SYS is increased to 1024 blocks plus the required size. The RSTS.CIL file (which must reside on the system disk) can be enlarged for the same reasons. This has the added advantage of allowing a larger system (CIL) to replace the current CIL when a new system generation is performed.

The DISK query is printed only if a swapping disk(s) is included in the hardware configuration. If there is no swapping disk, all the system files must reside on the system disk.

The BASE query allows the system files to be positioned anywhere desired on the system or swapping disks. If a sector number is entered, it is considered a preferred location for the file rather than an absolute requirement. The disk allocation routines use this number as a place to start looking for contiguous file space. Any location between the specified BASE and the physical end of the disk is considered a legal start sector for the file. REFRESH works in two passes with regard to these preferred locations. If on the first pass, sufficient space is not available between the specified BASE and the end of the disk for any of the system files, a second try is automatically performed. Before the second try, all the preferred locations for all the system files are discarded. Thus, this second pass attempts to find sufficient space anywhere on the system disk or swapping disk(s). The particular disk used for a particular file is determined by the DISK query. REFRESH never moves a file from the swapping disk to the system disk (or vice-versa) unless told to do so by the appropriate response to the DISK query.

The DELETE query is printed only for optional files. For instance, if the BUFF.SYS file is created but is not required, the system manager has the option to delete the file using the REFRESH option.

3.4.3.4 REFRESH Action Message

After all file-related questions are answered, the REFRESH routines attempt to build the system file structure, as specified. If the REFRESH operation is successful, the File Status Table is reprinted indicating the new structure of the system files. The initialization routines then return to the OPTION query. The system manager should proceed to Section 3.5 and follow the procedures to establish the default start up conditions.

If the REFRESH routines are not able to make the requested changes, the following message is printed.

```
UNABLE TO CREATE REQUIRED FILES - REBUILDING DISK
```

There is a short pause while REFRESH reconstructs the original system files. The message

```
DISK REBUILT
```

appears after REFRESH recreates the original files and the File Status Table is printed. The table reflects the old file structure.

If the REFRESH operation is unsuccessful, the system manager must consider what caused the failure. The most likely reason for an unsuccessful REFRESH is an attempt to allocate more blocks to a file (or files) than there are blocks on the disk. (An RK05 DECpack has 4800 blocks; an RP02 has 40000 blocks; an RP03 has 80000 blocks; an RP04 has 167200 blocks; each RS03 unit and RF11 platter has 1024 blocks; and an RS04 has 2048 blocks). The system manager should repeat the REFRESH procedure.

3.4.4 REFRESH Examples

Two REFRESH examples are included in this section. The examples are intended to show some of the possible philosophies which can be used with REFRESH. In the first example, the system files are structured to allow for future expansion and even a hardware failure in the swapping disks. The second example is a small system where disk space is of primary importance. A reasonable structure of system files is created but no provision is made for future expansion.

3.4.4.1 RP03 System Disk with two RS03 Swapping Disks

The first example is a typical REFRESH for a large system. The hardware configuration includes an RP03 system disk and two RS03 fixed-head swapping disks. The only other relevant hardware included is four DECtape units. The system is configured for a maximum of 40 jobs and the desired swap maximum is 16K. The REFRESH proceeds as shown on the following pages.

After the date and time are entered, the OLD query is printed at line a. The NO response is used to request the long form of REFRESH. The system manager enters his requirements for each of the system files. The File Status Table is then printed indicating that only the BADB.SYS, RSTS.CIL, and SATT.SYS files currently exist. These three files are created by the DSKINT option. A request to LIST bad blocks at line b results in the message, THERE ARE NO BAD BLOCKS. If DSKINT had found any bad blocks on the RP03 pack, REFRESH prints the start sector numbers of the clusters containing bad blocks. When the BADS query is repeated at line c, the LINE FEED response is used to continue.

Changes are made to the RSTS.CIL file beginning at line d. The only operation permitted by REFRESH is to increase the size of the CIL file. The required size in this case is 274 blocks. The size is increased to 300 blocks to allow for future replacement of the CIL.

The only change which can be made to the SATT.SYS file (line e) is to position the file on the moving head RP03 disk. The Storage Allocation Table is referenced frequently under time sharing. It is therefore located near the center of the disk. The SWAP2.SYS file is also located near the center of the disk. The effect is to create a

```
OPTION: REFRESH
DD-MMM-YY? 22-JUL-74
HH:MM? 12:30

OLD ? NO                                                    (line a)

FILE                              CURRENT  REQUIRED   START    START
NAME       REQUIRED? EXIST STATUS   SIZE     SIZE    CLUSTER  SECTOR

BADB  .SYS     YES    SYS     OK         0        0
RSTS  .CIL     YES    SYS    D/C       274      274        24       50
SATT  .SYS     YES    SYS    D/C        10       10         2        6
SWAP0 .SYS     YES     NO    CRE         0     1280
SWAP1 .SYS      NO     NO     OK         0        0
SWAP2 .SYS      NO     NO     OK         0        0
SWAP3 .SYS      NO     NO     OK         0        0
OVR   .SYS      NO     NO     OK         0       28
ERR   .SYS      NO     NO     OK         0        8
BUFF  .SYS     YES     NO    CRE         0       12
CRASH .SYS      NO     NO     OK         0       37


BADS ? LIST                                                 (line b)

THERE ARE NO BAD BLOCKS                 <LF> denotes
                                        typing the LINE
BADS ?<LF>                              FEED key            (line c)

RSTS.CIL CHANGES ? YES                                      (line d)

SIZE ? 300

SATT.SYS CHANGES ? YES                                      (line e)

BASE ? 40000

SWAP0.SYS CHANGES ? YES                                     (line f)

SIZE ? 1024

DISK ? SWP

BASE ? <LF>

SWAP1.SYS CHANGES ? YES                                     (line g)

SIZE ? 960

DISK ? SWP

BASE ? 81024

SWAP2.SYS CHANGES ? YES                                     (line h)

SIZE ? 1088

DISK ? SYS

BASE ? 40000
```

```
SWAP3.SYS CHANGES ? <LF>                        <LF> denotes        (line i)
                                                typing the LINE
OVR.SYS CHANGES ? YES                           FEED key            (Line j)
       LEAVE IT IN CIL?
CIL ? NO        THIS   MAKES A COPY (IN A MORE
                                     ACCESSIBLE
SIZE ? <LF>                          LOCATION)

DISK ? SWP

BASE ? <LF>

ERR.SYS CHANGES ? YES                                               (line k)

CIL ? NO

SIZE ? <LF>

DISK ? SWP

BASE ? <LF>
                             FOR (DECTAPE) LARGE   BUFFERS
BUFF.SYS CHANGES ? YES                                              (line l)

SIZE ? <LF>

DISK ? SWP

BASE ? <LF>
                DO YOU  WANT  A  CRASH.OST SYS FILE?
CRASH ? YES                                                         (line m)
        'YES!'
FILE                                   CURRENT  REQUIRED   START    START
NAME        REQUIRED? EXIST STATUS       SIZE     SIZE    CLUSTER  SECTOR

BADB  .SYS     YES     SYS     OK          0        0
RSTS  .CIL     YES     SYS     OK        300      274        24       50
SATT  .SYS     YES     SYS     OK         10       10     19999    40000
SWAP0 .SYS     YES     SWP     OK       1024     1280     39999    80000
SWAP1 .SYS      NO     SWP     OK        960        0     40511    81024
SWAP2 .SYS      NO     SYS     OK       1088        0     20004    40010
SWAP3 .SYS      NO      NO     OK          0        0
OVR   .SYS      NO     SWP     OK         28       28     40991    81984
ERR   .SYS      NO     SWP     OK          8        8     41005    82012
BUFF  .SYS     YES     SWP     OK         12       12     41009    82020
CRASH .SYS      NO     SYS     OK         38       37       174      350


OPTION:                                                             (line n)
```

quiescent or "home" position for the heads near the center of the disk. The seek distance to access user file data on either side of center is thereby reduced. This will have more effect as user files and directories are scattered over the full disk surface.

The questions for SWAP0.SYS begin at line f. Both RS03 drives are available for swap files and other system files. SWAP0.SYS is located on the first RS03 unit. This choice is an arbitrary one in this case since both swapping disks are equally fast and are equal in size. The size of SWAP0.SYS is set to 1024 blocks which is equal to the capacity of the drive. SWAP0.SYS handles 16 jobs at 16K words each.

The second swap file, SWAP1.SYS (line g), is located on the second RS03. The response to the BASE query shown in the example forces REFRESH to try the allocation beginning at sector 81024 which is the first sector of the second RS03 (see Figure 3-1). It is not necessary to specify a BASE since all the space on the first RS03 has already been assigned. The second RS03 is used for SWAP1.SYS even if BASE were answered with the LINE FEED key. The size of SWAP1.SYS is set at 960 blocks to support 15 jobs at 16K. This leaves room on the swapping disk for other system files as discussed below.

Beginning at line h SWAP2.SYS is created on the system disk. SWAP2.SYS is purposely larger than necessary to allow for future expansion to 48 jobs. Expansion to 48 jobs requires a new SYSGEN but a new CIL can replace the current CIL without affecting the file structures on this system disk. With the size of SWAP2.SYS set to 1088 blocks, this file handles 17 jobs at 16K words. The size of SWAP2.SYS also provides for recovery from a total failure of the swapping disk subsystem (controller and/or both drives). In the event of a failure, SWAP0.SYS moves to the system disk to replace SWAP2.SYS. SWAP1.SYS and SWAP2.SYS can be deleted if there is insufficient contiguous space to accommodate these files. Even if no other contiguous space is available, the space reserved in SWAP2.SYS is enough to handle 16 jobs at 16K with 64 blocks left over for the other system files. SWAP2.SYS is located near the center of the disk, physically adjacent to the SATT.SYS file.

The SWAP3.SYS CHANGES query is answered with the LINE FEED response (line i) since a fourth swap file is not needed. Since SWAP3.SYS does not exist, REFRESH does not create it.

The overlay code (OVR.SYS), error messages (ERR.SYS), and the DECtape buffers (BUFF.SYS) are all located on the second RS03. The CIL query for both OVR.SYS and ERR.SYS is answered NO to cause these files to be created. The SIZE queries for OVR.SYS, ERR.SYS, and BUFF.SYS are all answered with the LINE FEED key to allow REFRESH to create these files at their required sizes. There is no need to allocate extra space to these files when they reside on a swapping disk since unallocated space is never used for user files. The DISK queries are answered with the SWP response to locate all three files on the swapping disk. Finally, the BASE queries are also answered with LINE FEED to allow REFRESH to position these files wherever space is available on the swapping disk.

The CRASH.SYS file is created by the YES response to the CRASH query at line m. If the crash dump file is not created at this time, the system manager can not enable the crash dump facility using the DEFAULT or START option and valuable diagnostic information is lost in the event of a system crash.

The CRASH query ends the REFRESH interaction. There is a pause while REFRESH builds the system file structure as specified. The File Status Table is then reprinted to reflect the new structure. Note in the example that REFRESH positions all the files, as requested. The only other notable thing in the table is the expansion of the CRASH.SYS file to 38 blocks from the required size of 37 blocks. The pack cluster size of the RP03 system disk is the minimum pack cluster size of 2 blocks. The size of CRASH.SYS is rounded up to an integral number of clusters (38 blocks is 19 clusters with a pack cluster size of 2).

The printing of the OPTION query at line n signals the end of the REFRESH operation. The system manager then proceeds to the DEFAULT option (described in Section 3.5) to establish start up conditions for his system.

3.4.4.2 RK05 System Disk Without an Auxiliary Swapping Disk

The second example is a REFRESH for a smaller system which includes an RK system disk but no swapping disk. The system is configured for 16 jobs. Two DECtape drives are also included in the hardware configuration. The REFRESH operation is shown and explained below.

```
OPTION: REFRESH
DD-MMM-YY? 22-JUL-74
HH:MM? 18:30

OLD ? NO                                                       (line a)
```

SHOULD FILES BE LEFT STATIC SO OLD LISTING WOULD REMAIN CURRENT? RESPOND 'NO' FIRST TIME

```
FILE                                     CURRENT  REQUIRED  START    START
NAME        REQUIRED? EXIST STATUS       SIZE      SIZE    CLUSTER  SECTOR

BADB  .SYS      YES     SYS      OK         0         0
RSTS  .CIL      YES     SYS     D/C       264       264       48       49
SATT  .SYS      YES     SYS     D/C         2         2        4        5
SWAP0 .SYS      YES      NO     CRE         0       512
SWAP1 .SYS       NO      NO      OK         0         0
SWAP2 .SYS       NO      NO      OK         0         0
SWAP3 .SYS       NO      NO      OK         0         0
OVR   .SYS       NO      NO      OK         0        28
ERR   .SYS       NO      NO      OK         0         8
BUFF  .SYS      YES      NO     CRE         0         6
CRASH .SYS       NO      NO      OK         0        28
```

clusters counted from '0' up

sectors counted from '1' up

```
BADS ? <LF>                                   <LF> denotes       (line b)
                                              typing the LINE
RSTS.CIL CHANGES ? NO                         FEED key           (line c)

SATT.SYS CHANGES ? YES                                           (line d)

BASE ? 1600

SWAP0.SYS CHANGES ? YES                                          (line e)

SIZE ? 1024

BASE ? 1600

SWAP1.SYS CHANGES ? <LF>

SWAP2.SYS CHANGES ? <LF>

SWAP3.SYS CHANGES ? <LF>
```

YES - SAVE ROOM

every time you write to disk this is changed - accessed frequently - so put it at middle of disk

```
OVR.SYS CHANGES ? YES                                          (line f)

CIL ? NO                             <LF> denotes
                                     typing the LINE
SIZE ? <LF>                          FEED key

BASE ? 1600

ERR.SYS CHANGES ? YES                                          (line g)

CIL ? NO

SIZE ? <LF>

BASE ? 1600

BUFF.SYS CHANGES ? YES                                         (line h)

SIZE ? <LF>

BASE ? 1600

CRASH ? YES                                                    (line i)

FILE                                  CURRENT  REQUIRED   START    START
NAME       REQUIRED? EXIST STATUS      SIZE      SIZE    CLUSTER  SECTOR

BADB  .SYS     YES    SYS    OK          0         0
RSTS  .CIL     YES    SYS    OK        264       264        48       49
SATT  .SYS     YES    SYS    OK          2         2      1599     1600
SWAP0 .SYS     YES    SYS    OK       1024       512      1601     1602
SWAP1 .SYS     NO     NO     OK          0         0
SWAP2 .SYS     NO     NO     OK          0         0
SWAP3 .SYS     NO     NO     OK          0         0
OVR   .SYS     NO     SYS    OK         28        28      2625     2626
ERR   .SYS     NO     SYS    OK          8         8      2653     2654
BUFF  .SYS     YES    SYS    OK          6         6      2661     2662
CRASH .SYS     NO     SYS    OK         28        28        12       13


OPTION:
```

This REFRESH is simpler than the previous example since there are fewer possibilities when no swapping disk exists. The REFRESH operation shown also has slightly different goals. On systems with RK disks, disk space for user files is usually in short supply. In this example, no disk space is reserved for future expansion. All of the system files (except SWAP0.SYS) are created at their required sizes. Since the RK system disk is small in comparison to an RP03 (4800 blocks vs. 80000 blocks), it is not difficult to backup and delete all library and user files to gain space for future expansion. On systems with many RK drives, a completely different philosophy can be used. The system disk may be totally dedicated to system and library files. All unused space can be reserved by creating a dummy file under time sharing which completely fills all space not required for system or library files. The library can be easily saved and the dummy file deleted during time sharing to allow restructuring or expansion of the system files at a later date.

As shown in the example, the OLD query is answered NO at line a to request the long form of REFRESH. As in the previous example, the File Status Table indicates that only the BADB.SYS, RSTS.CIL, and SATT.SYS files exist before the initial use of REFRESH. The BADS query at line b is answered with the LINE FEED key since the status table indicates there are no bad blocks (size of BADB.SYS is zero). At line c, no change in the RSTS.CIL file is requested. The choice is to conserve disk space rather than plan for future expansion. Hence, the size of the CIL file is not increased to allow for replacement of the system at a later time.

The movable system files are all located near the center of the disk to reduce average seek time for the moving heads. At line d a change to SATT.SYS is requested and the file is positioned at sector 1600. The center of an RK cartridge disk is actually sector 2400. The system files span a range of sectors on each side of center.

Beginning at line e, a single swap file, SWAP0.SYS is created to support 16 jobs at 16K and is positioned adjacent to the SATT.SYS file. Note that the DISK query is not printed in this case since the system is not configured for a swapping disk. The auxiliary swap files are not needed so the CHANGE queries for the other swap files are answered with the LINE FEED key.

Beginning at line f, the overlay code (OVR.SYS) and error message (ERR.SYS) files are created so that they can be positioned with the other system files. If disk space is extremely critical, this data can be accessed from the RSTS.CIL file. The DECtape buffer file is required and is created at the required size by the responses shown at line h. Finally, the CRASH.SYS file is created despite the space constraint since the crash dump facility cannot be enabled and valuable error logging information could be lost if the CRASH.SYS file did not exist.

The File Status Table is printed to reflect the new file structure and REFRESH terminates by returning to the OPTION query. The system manager proceeds to the DEFAULT option to establish start up conditions.

3.5 SETKEY - SETTING KEYBOARD DEFAULTS

The initialization option SETKEY sets default conditions for keyboard lines on the RSTS/E system. The option allows the user to enable and disable individual lines and to specify whether a DH11 multiplexor line is to be enabled with or without modem control.

The SETKEY option is required only if modem control must be enabled on a DH11-AA or DH11-AC multiplexor line (with DM11-BB modem control hardware) or on a DH11-AD line. SETKEY can optionally be used to disable any terminal interface or DH11 line which should not be used. At start up time, the initialization code enables all terminal interfaces (if the hardware is present) except those lines specifically disabled with SETKEY. DH11 lines wired for modem control are enabled as local lines unless modem control is specifically enabled with the SETKEY option. Modem control is always enabled on DL11E and DC11 single line interfaces unless the interface is disabled with SETKEY.

The default conditions established by SETKEY remain in effect until altered by SETKEY. Each time the system starts up, the initialization code enables terminals according to the SETKEY default values and the presence or absence of hardware.

To execute the SETKEY option, type SETKEY or SE in response to the OPTION: message printed by the initialization code. In response to SE, the query COMMAND is printed and one of the responses shown in Table 3-7 can be typed. The following dialog demonstrates the sequence.

```
OPTION: SETKEY
COMMAND?
```

To terminate SETKEY, type the EXIT command or the LINE FEED key and the OPTION: message is printed again.

Table 3-7

Responses to the SETKEY COMMAND Query

Response	Short Form	Meaning
DISABLE	DI	Disable a keyboard.
ENABLE	EN	Enable a keyboard on a single line interface or enable a keyboard as local on a DH11 multiplexer interface.
LIST	LI	Print a list of current default conditions for all keyboards configured on the system.
LOCAL	LO	Enable a keyboard as local on a DH11 multiplexer interface or enable a keyboard on a single line interface.
MODEM	MO	Enable a keyboard with modem control on a DH11-AA or DH11-AC multiplexer interface with DM11-BB modem hardware or on a DH11-AD multiplexer interface.
EXIT	EX	Exit to the OPTION: message.
Type the LINE FEED key		Exit to the OPTION: message.
Type the RETURN key		Print a list of legal commands and an accompanying description of action.

If either DISABLE, ENABLE, LOCAL, or MODEM is entered in response to the COMMAND query, the prompting message KB NUMBER is printed. Typing the unit number of the keyboard causes the related default condition to be set for the specified keyboard. The system continues printing the prompting message until the LINE FEED key is typed. The following sample dialog shows the procedure.

```
COMMAND?  DISABLE
KB NUMBER? 1
KB NUMBER? 3
KB NUMBER? <LF>             <LF> denotes typing the LINE FEED key

COMMAND?  EXIT

OPTION:
```

Typing DISABLE causes the system to print the prompting message. The disable condition is set when the keyboard unit number is entered. Typing the LINE FEED key returns to the COMMAND query. The EXIT command returns to the OPTION: message.

The LIST command causes the system to print a list containing keyboard unit designators, addresses, and current default settings. Each entry has the following general format.

```
                     address
KBn:                    or                 IS     default setting
                 address/address
```

The designation n gives the unit number of the keyboard. The address is either a single line interface address or the DH11 multiplexer address. If two addresses are present and separated by a slant character, the first address is that of the DH11 multiplexer and the second is that of the DM11-BB modem control. The default setting can be either DISABLED, ENABLED, or ENABLED WITH MODEM.

Because pseudo keyboards are not physical devices, they do not have device registers in the peripheral address space. The address printed by SETKEY for pseudo keyboards is the address of a word in memory which serves as a pseudo device register. This word allows pseudo keyboards to be treated as physical devices by the terminal service code. An address less than 160000(octal) indicates a pseudo keyboard.

The following sample dialog demonstrates the usage of the SETKEY commands.

```
OPTION:  SETKEY

COMMAND? DISABLE
KB NUMBER?
ENTER KEYBOARD NUMBER OR  <LF>  TO EXIT?  5
KB NUMBER? 13
KB NUMBER? 1
KB NUMBER?

COMMAND? MODEM
KB NUMBER? 20
KB NUMBER?
```

```
COMMAND?  LIST
KB0: (177560) IS ENABLED
KB1: (175610) IS DISABLED
KB2: (175620) IS ENABLED
KB3: (032424) IS ENABLED
KB4: (032430) IS ENABLED
KB5: (032434) IS DISABLED
KB6: (032440) IS ENABLED
KB7: (160020/170500) IS ENABLED
KB8: (160020/170500) IS ENABLED
KB9: (160020/170500) IS ENABLED
KB10: (160020/170500) IS ENABLED
KB11: (160020/170500) IS ENABLED
KB12: (160020/170500) IS ENABLED
KB13: (160020/170500) IS DISABLED
KB14: (160020/170500) IS ENABLED
KB15: (160020/170500) IS ENABLED
KB16: (160020/170500) IS ENABLED
KB17: (160020/170500) IS ENABLED
KB18: (160020/170500) IS ENABLED
KB19: (160020/170500) IS ENABLED
KB20: (160020/170500) IS ENABLED WITH MODEM
KB21: (160020/170500) IS ENABLED
KB22: (160020/170500) IS ENABLED

COMMAND?
```

In the example above, the DISABLE command disabled keyboards 1, 5, and 13. The MODEM command enabled modem control on one DH11 multiplexer line (keyboard 20). The LIST command prints a listing of conditions for all keyboard units on the system. The listing shows the following conditions. Keyboard units 0 through 3 are single line interfaces with discrete addresses in the I/O address space. Keyboard unit 1 is shown disabled as a result of the DISABLE command. Keyboard units 3 through 6 are pseudo keyboard lines, unit 5 of which is disabled as a result of the DISABLE command. Keyboard units 7 through 22 are DH11 multiplexer lines. The two addresses shown with each line are the UNIBUS addresses of the DH11 multiplexer and the DM11-BB modem, respectively. The listing shows that all DH11 lines are enabled and that keyboard unit 20 is enabled with modem control as a result of the MODEM command.

3.6 DEFAULT OPTION - SETTING TIME SHARING DEFAULTS

The DEFAULT initialization option is used to establish certain default start up conditions for the RSTS/E system. These conditions include:

a) The maximum number of jobs the system allows during time sharing operations.

b) The maximum storage space that a job is allowed to occupy in the swapping files (and, therefore, the size to which a job is allowed to expand during time sharing operations).

c) The allocation of the BASIC-PLUS Compiler and Run Time System to an area of memory.

d) The assignment of certain portions of memory as usable or not usable, either by the system or user jobs.

e) The enabling or disabling of the crash dump facility.

The system manager is required to establish the initial default start-up conditions. The DEFAULT option is used immediately after the REFRESH operation is completed. During the first use of DEFAULT, the system manager may accept the system-assigned defaults or change them to meet the installation needs. The DEFAULT option may also be used at any later time to change the start up conditions as installation requirements change.

This section describes the start up conditions and the use of the DEFAULT option in detail. The options available with DEFAULT are also available when the START option is used to begin time sharing. It is possible to set job and swap maximums, to alter the memory allocation, and to enable or disable the crash dump facility, using either the DEFAULT or START options. There is an important distinction, however; the start up conditions established with DEFAULT are permanent. Parameters are saved in the RSTS.CIL file on the system disk so that they can be used each time the system is brought up. The DEFAULT start up conditions may be overridden for one time sharing session by making changes with the START option. Thus, changes to the start up conditions made with the START option are temporary. The DEFAULT parameters are used if the system is restarted. The permanent DEFAULT start up conditions can only be changed by a subsequent use of the DEFAULT option.

3.6.1 Using the DEFAULT Option

The system manager requests the DEFAULT option by typing DEFAULT (or DE) in response to the initialization code OPTION query. The DEFAULT routines print a series of three queries to determine which start up conditions are to be changed. A brief description of the DEFAULT queries is provided in Table 3-8. Valid responses and additional related options are described in subsequent sections.

Table 3-8

DEFAULT and START Option Queries

Query	Meaning	References
JOB OR SWAP MAX CHANGES?	Allows system manager to change (or establish) either the number of jobs allowed to run during time sharing (JOB MAX) or the length of the job swap area in the swapping files (SWAP MAX).	See Section 3.6.2 for the operational description and Table 3-9 for the possible responses.
ANY MEMORY ALLOCATION CHANGES?	Allows system manager to locate the Run Time System in an area of memory and to prohibit or allow use of certain portions of memory by the system.	See Section 3.6.3 for the operational description and Table 3-11 for the possible responses.
CRASH DUMP?	Allows system manager to either enable or disable the crash dump facility.	See Section 3.6.4 for the operational description and Table 3-14 for the possible responses.
MAGTAPE LABELLING DEFAULT (NONE):	Sets magtape labelling default to either DOS or ANSI format.	See Section 3.6.5 and Table 3-15.

The following sample dialogue illustrates the DEFAULT option. In this particular example, old DEFAULT start up conditions are being printed for reference. When DEFAULT is used for the first time on a new system, the message NO DEFAULTS ARE CURRENTLY SET precedes the dialogue and the Memory Allocation Table is printed automatically.

```
OPTION:  DEFAULT

YOU CURRENTLY HAVE:  JOB MAX = 22, SWAP MAX = 16K

JOB MAX OR SWAP MAX CHANGES ?

ANY MEMORY ALLOCATION CHANGES ? YES

TABLE OPTION ? LIST

MEMORY ALLOCATION BREAKDOWN:

0000000 - 0117777(  20K) : EXEC
0120000 - 0207777(  14K) : BASIC
0210000 - 0437777(  38K) : USER
0440000 -    END         : NXM → non-existent memory

TABLE OPTION? EXIT

YOU CURRENTLY HAVE: CRASH DUMP ENABLED

CRASH DUMP ? OLD

MAGTAPE LABELLING DEFAULT (DOS) ? OLD

RSTS V06A-02 TEST SYSTEM

OPTION:
```

The DEFAULT code is the most critical of the RSTS/E initialization routines. Prior to using this option, the KT11 memory management unit, physical memory above 28K, and the memory parity registers (if any) have not been used. DEFAULT enables memory mapping and scans the 124K of physical address space to determine the size and location of all available memory. DEFAULT also enables memory parity checking to determine the association between parity registers and the parity memory banks. If any of this hardware is not operating properly, it is very likely that DEFAULT will fail. If a crash occurs during execution of the DEFAULT option, the components mentioned should be carefully checked with diagnostics.

3.6.2 JOB and SWAP Maximums

The DEFAULT and START options allow the system manager to establish limits on the number of jobs which can be run during time sharing and the maximum size to which each job can expand. The JOB MAX and SWAP MAX established with the DEFAULT option are in effect each time the system is brought up unless specifically changed at start up time. If JOB MAX or SWAP MAX are changed using the START option, the change is in effect only for that one time sharing session. Normally JOB MAX and SWAP MAX are set with the DEFAULT option and are left unchanged when the system is brought up.

With either option, the system prints the current values of JOB MAX and SWAP MAX and then asks if any changes are desired. The sequence appears as shown below.

```
OPTION: DEFAULT

YOU CURRENTLY HAVE: JOB MAX = 32, SWAP MAX = 16K.

JOB MAX OR SWAP MAX CHANGES ?
```

When the DEFAULT option is used for the first time, the value of JOB MAX is the maximum number of jobs configured at system generation time. The value of SWAP MAX is 8K. These system defaults will probably be changed since few installations are satisfied with the 8K SWAP MAX. The system manager can request a change by the appropriate response to the JOB MAX or SWAP MAX CHANGES query. The possible responses are shown in Table 3-9.

Table 3-9

Possible Responses to JOB OR SWAP MAX CHANGES Query

Response(s)	Meaning
NO N OLD O LINE FEED key	Causes the currently established default values to remain unchanged and the second in the series of three queries, ANY MEMORY ALLOCATION CHANGES?, to be printed.
RETURN key	Causes an explanatory form of the query to be printed, after which the system manager types his response.
CONTROL key and C combination	Causes interrupt of execution and printing of the OPTION query.
YES	Indicates system manager wants to change either JOB MAX or SWAP MAX and causes additional related queries to be presented.

If a change is requested, additional queries ask the system manager to specify a new value for JOB MAX and/or SWAP MAX. These two variables are dependent on each other and on the sizes of the swap files created by the REFRESH option. Recall from the discussion of the swap files (see Section 3.4.2.4) that each file holds an integral number of jobs defined by

$$\text{JOBS for Swap File n} = \frac{\text{Size of SWAPn.SYS in segments}}{\text{SWPMAX*4}} \text{ (truncated)}$$

Jobs X swp x4 = Size

The sum of the jobs held by each of the existing swap files must be greater than or equal to the desired JOB MAX specified. The desired JOB MAX and SWAP MAX are entered in response to the queries shown in Table 3-10.

The initialization routines accept any value for JOB MAX between 1 and the configured maximum number of jobs. Any value of SWAP MAX between 6(K) and 28(K) is accepted. The BASIC-PLUS Run Time System does not allow any job to exceed 16K, however. The large range of values accepted for SWAP MAX is intended to provide for future additions to the RSTS/E system. After a new JOB MAX and/or SWAP MAX is entered, the DEFAULT routines print the new values and check them against the sizes of the swap files. If the existing swap files do

Table 3-10

Possible Responses to NEW JOB MAX and NEW SWAP MAX Queries

Query	Possible Responses	Meaning
NEW JOB MAX?	1 to maximum number of jobs configured at system generation time	Set the maximum number of jobs which can be run during normal time sharing operations.
NEW SWAP MAX?	6 to 28	The amount of storage space in K (1024) words to be allocated for each job in the swap file. Determines the maximum size to which each job is allowed to expand. (See text below.)
EITHER OF ABOVE QUERIES	RETURN key	Causes an explanatory message to be printed after which the query is reprinted.
	LINE FEED key, NO, or OLD	Do not change current maximum.

not accommodate the SWAP MAX and JOB MAX specified, an error message is printed and the system manager must enter new values. If the new JOB MAX and SWAP MAX are consistent with the sizes of the existing swap files, the JOB MAX OR SWAP MAX CHANGES query is reprinted. The system manager normally requests no further changes and proceeds to set up the memory allocation table described in Section 3.6.3.

The example below demonstrates the procedure for changing JOB MAX and SWAP MAX. The example assumes two swap files: SWAP0.SYS is 1024 segments long; SWAP1.SYS has 960 segments.

```
OPTION: DEFAULT

YOU CURRENTLY HAVE: JOB MAX = 32, SWAP MAX = 8K.

JOB MAX OR SWAP MAX CHANGES ? YES

NEW JOB  MAX ?

NEW SWAP MAX ? 16

YOU CURRENTLY HAVE: JOB MAX = 32, SWAP MAX = 16K.


YOUR CURRENT JOB AND/OR SWAP MAXIMUMS ARE INCONSISTENT
WITH THE SIZE(S) OF THE EXISTING SWAP FILE(S). YOU MUST,
THEREFORE, SPECIFY A NEW JOB MAX OR SWAP MAX.


NEW JOB  MAX ? 31

NEW SWAP MAX ?

YOU CURRENTLY HAVE: JOB MAX = 31, SWAP MAX = 16K.

JOB MAX OR SWAP MAX CHANGES ? NO


ANY MEMORY ALLOCATION CHANGES ?
```

Note in the example that the attempt to set JOB MAX to 32 and SWAP MAX to 16K failed because the swap files are not large enough. SWAP0.SYS (1024 segments) handles 16 jobs at 16K. SWAP1.SYS (960 segments) holds only 15 jobs at 16K. The second attempt which reduced JOB MAX to 31 while leaving SWAP MAX at 16K was successful. Note also that with the same swap files, other JOB MAX and SWAP MAX combinations can be accommodated. For example, 33 jobs can be run with a SWAP MAX of 15K.

If there are any problems arriving at the desired JOB MAX and SWAP MAX, it may be necessary to abort the DEFAULT (or START) option by typing CONTROL/C and to enter REFRESH to check the sizes of the existing swap files. After REFRESH prints the File Status Table (see Section 3.4.3.1), the system manager can determine from the sizes of the swap files what values of JOB MAX and SWAP MAX are allowed. If no changes to the swap files are desired, REFRESH can be aborted with CONTROL/C after the File Status Table is printed. The system manager can then reenter DEFAULT (or START) and enter new values for JOB MAX and/or SWAP MAX. Alternatively, a complete REFRESH can be performed to increase the size of the swap files so that the desired JOB MAX and SWAP MAX are accepted.

3.6.3. Memory Allocation

The DEFAULT option allows the system manager to allocate a specific area of memory to the BASIC-PLUS Run Time System, to make certain portions of memory either unavailable or available for use, and to list the current usage and types of memory on the system.

When DEFAULT is used to establish the start up conditions for a new system, a memory allocation table is printed as shown in the example of Section 3.6.1. The system manager may accept the system default memory allocation or make changes by appropriate responses to the TABLE OPTION query. The memory allocation table is not printed automatically after the initial use of the DEFAULT option. Instead, the system prints the query, ANY MEMORY ALLOCATION CHANGES? The system manager may leave the defaults as they stand or request a change in the memory allocation by one of the responses shown in Table 3-11.

Table 3-11

Possible Responses to ANY MEMORY ALLOCATION CHANGES Query

Response(s)	Meaning
NO OLD O or LINE FEED key	Causes currently established memory allocation assignments to remain unchanged and next query, CRASH DUMP?, to be printed.
RETURN key	Causes an explanatory form of the query to be printed, after which the user types his response.
YES	Causes printing of the query TABLE OPTION? See Table 3-12 for additional related queries.
CONTROL key and C combination	Causes interrupt of execution and the printing of the OPTION query.

If the system manager indicates that he wishes to alter the current memory allocation, the TABLE OPTION query is printed. The possible responses to the TABLE OPTION query are summarized in Table 3-12. Subsequent sections describe each of the table options in more detail.

Table 3-12

Additional Related Queries and Responses to ANY MEMORY ALLOCATION CHANGES Query

Additional Related Query	Response(s)	Result
TABLE OPTION?	RETURN key	Causes a list of the options to be printed after which the TABLE OPTION query is reprinted and user is allowed to type his response.
	LIST	Causes the current memory allocation breakdown to be printed.
	PARITY	Used as a diagnostic tool to identify various types of parity memory on the system.
	LOCATE	Used to load BASIC-PLUS at a certain available portion of memory. Causes the query NEW RUN-TIME SYSTEM ADDRESS IS? to be printed. (See description of this query below.)
	LCK for when there is bad memory	Used to prevent the system from accessing some portion of memory. Prints LOCKOUT ADDRESS IS? query. (See query described below.)
	UNLOCK	Used to allow a currently locked portion of memory to be available for a user job. Causes an additional query (UNLOCK ADDRESS IS?) to be printed. (See additional query described below.)
	RESET	Allows the initialization code to set up the memory allocation table. Unlocks all locked memory, locates RTS immediately after the monitor, and makes available for use any new memory added to the system.
	CACHE	Allows allocation of a certain portion of memory for disk caching operations. Causes additional query DISK CACHE ADDRESS RANGE to be printed. (See query described below.)
	EXIT	Causes exit from routine which changes memory allocation and prints next DEFAULT option query, CRASH DUMP?

(continued on next page)

Table 3-12 (Cont.)

Additional Related Queries and Responses to ANY MEMORY ALLOCATION CHANGES Query

Additional Related Query	Response(s)	Result
NEW RUN-TIME SYSTEM ADDRESS IS?	RETURN key	Causes an explanatory message to be printed and the query to be reprinted, after which the user types the six digit octal number, xxxxxx.
	xxxxxx	The six digit octal number representing the 18-bit address of the portion of memory at which BASIC-PLUS is loaded. (Loading is done from low memory to high memory.) The octal number can be derived from the current memory allocation breakdown.
LOCKOUT ADDRESS IS?	RETURN key	Causes explanatory message to be printed, after which the query ADDRESS OF 1K MEMORY SECTION IS? is printed.
	xxxxxxx or	The four-to-seven digit octal number which corresponds to the 22-bit address of the 1K portion of memory to be locked out. The octal number can be derived from the current memory allocation breakdown.
	xxxxxxx-xxxxxxx	Two numbers, separated by a dash, specifies a range of successive 1K portions of memory to be locked.
UNLOCK ADDRESS IS?	RETURN key	Causes explanatory message to be printed, after which the query ADDRESS OF 1K MEMORY SECTION IS? is printed.
	xxxxxxx or	The four-to-seven digit octal number which corresponds to the 22-bit address of the 1K portion of memory to be unlocked. The octal number can be derived from the current memory allocation breakdown.
	xxxxxxx-xxxxxxx	Two numbers, separated by a dash, specifies a range of successive 1K portions of memory to be unlocked.
DISK CACHE ADDRESS RANGE?	xxxxxxx-xxxxxxx	Two octal numbers, separated by a dash, specifies a range of 1K portions of memory to be reserved for disk caching operations.

Table 3-12 (Cont.)

Additional Related Queries and Responses to ANY MEMORY ALLOCATION CHANGES Query

Additional Related Query	Response(s)	Result
DISK CACHE ADDRESS RANGE? (cont.)	xxxxxxx	One octal number specifies a single 1K portion of memory to reserve.
	REMOVE	Remove the memory from disk cache and return it to user job space.

3.6.3.1 LIST Table Option

The LIST table option prints the current memory allocation breakdown on the console terminal. The breakdown provides useful information on the size of the RSTS/E Monitor, the size of the BASIC-PLUS Run Time System (compiler and interpreter), the amount of memory available for user jobs, which portions of memory (if any) are disabled or assigned to disk caching, and the total size of the memory on the machine. The table covers the full physical address space and shows the allocation or state of each range of addresses with appropriate symbols. The following example demonstrates the use of the LIST table option.

```
TABLE OPTION ? LIST

MEMORY ALLOCATION BREAKDOWN:

0000000 - 0127777 ( 22K) : EXEC
0130000 - 0237777 ( 18K) : USER
0240000 - 0247777 (  2K) : LOCKED
0250000 - 0477777 ( 38K) : USER
0500000 - 0567777 ( 14K) : BASIC
0570000 - 0677777 ( 18K) : USER
0700000 -   END          : NXM
```

Each row of the memory allocation breakdown represents a variable range within the physical address space. Included are the starting and ending octal addresses of the range, the amount of memory occupied, and the purpose for which that portion of memory is employed. The symbols used in the memory allocation table are listed and described in Table 3-13.

Table 3-13

Memory Allocation Table Symbols

Symbol	Meaning
EXEC	Occupied by RSTS/E Monitor (Executive)
BASIC	Occupied by BASIC-PLUS (Compiler and Run Time System)
USER	Available for user space
LOCKED	Memory has been made unavailable for use (locked)
CACHE	Reserved for disk caching
NXM	Memory does not exist (nonexistent) on this PDP-11

The LIST table option can be used as often as desired in both the DEFAULT and START options. If changes are made to the memory allocation table using the LOCATE, LCK (LOCK), UNLOCK, CACHE, or RESET table options described in the following sections, a copy of the final DEFAULT memory allocation breakdown should be kept with other system documentation for future reference.

3.6.3.2 PARITY Table Option

The PARITY table option is used primarily as a diagnostic tool to identify and locate the various types of parity memory on the system. The option is useful to DIGITAL Field Service personnel to verify the memory configuration when the system is installed. It is also useful to identify the physical address of MOS memory on PDP-11/45 systems. The types of memory (MOS or core) are determined by looking at the hardware parity registers. It is impossible for the software to determine the type for non-parity memories.

On a PDP-11/70 configuration, the listing of the parity configuration is as follows.

```
TABLE OPTION ? PARITY

PARITY REGISTER BREAKDOWN:

ALL MEMORY IS 11/70 PARITY MEMORY.

TABLE OPTION ?
```

The following example shows a configuration for a PDP-11/40 or PDP-11/45 system.

```
TABLE OPTION ? PARITY

PARITY REGISTER BREAKDOWN:

0000000 - 0277777 ( 48K) : 00/02
0300000 - 0377777 ( 16K) : 20(NA)
0400000 - 0477777 ( 16K) : 24(NA)
0500000 - 0577777 ( 16K) : 14
0600000 - 0677777 ( 16K) : NO
0700000 -   END          : NXM
```

The following are the meanings of the codes indicating the type of parity memory.

NO	Non-parity memory
NXM	Nonexistent memory
nn/mm	Interleaved memory
nn	Parity memory with address information
nn(NA)	Parity memory with no address information

The values nn and mm are the last two octal digits of the address of the parity register responsible for that section of memory. Up to 16 parity registers are in the UNIBUS address range 772100 to 772136. When a parity error is detected, the parity register responsible for that section of memory usually contains information on the location of the last error detected. The code NA indicates that the parity register contains no address information on the location of the error. All core memory returns address information.

One parity register controls either 24K (MM11-LP) or 32K words (MM11-UP) of parity core memory. If core memory is interleaved, two parity registers are used for either 48K (MM11-LP) or 64K words (MM11-UP). For MOS memory one parity register handles one 16K bank. MOS memory cannot be interleaved.

Interleaving means that one bank of memory responds to "even" word memory addresses while another bank responds to "odd" word addresses as shown below:

ADDRESS		PARITY REGISTER
000000	EVEN	nn
000002	ODD	mm
000004	EVEN	nn
000006	ODD	mm
000010	EVEN	nn
000012	ODD	mm

Interleaving is used because core memories are destructive read out devices. After each read from a core memory, the original data must be restored. However, once the processor receives the data requested, it can go on to other things (presumably another memory reference) during the restore cycle in the memory bank just referenced. The bank remains busy until the restore cycle completes. When memory is interleaved, the probability is low that the same bank will be referenced on the next memory cycle. Hence, interleaving allows some overlap of memory operations with a resultant speed up in program execution.

3.6.3.3 LOCATE Table Option

The LOCATE table option allows the system manager to position the BASIC-PLUS Run Time System (compiler and interpreter) anywhere in the first 124K words of memory. For example, if the hardware configuration on a PDP-11/45 computer includes 16K words of MOS memory, it is advantageous to locate BASIC-PLUS in that memory. The bulk of processing done by the RSTS/E system involves executing BASIC-PLUS programs. Timing studies indicate that 70% to 80% of the CPU time is spent executing code in BASIC-PLUS. Hence, optimal performance is realized if this code resides in the fastest memory available on the machine. If the machine has a mixture of fast and slow core memory, there may be some small advantage in locating BASIC-PLUS in fast core. On most systems without semiconductor memory, the only reason to locate BASIC would be to avoid a defective section of memory. BASIC-PLUS would normally follow immediately after the Monitor in the lowest physical memory addresses.

When the LOCATE table option is requested, the system prints one additional query, NEW RUN-TIME SYSTEM ADDRESS IS?. The system manager enters the 6-digit octal address of a 1K section of memory that is to become the lowest address of BASIC-PLUS. The initialization code checks the address to ensure that sufficient contiguous memory is available to load the RTS starting at the specified address. If any of the memory in the required address range is locked out, nonexistent, allocated to the Monitor, or extends above address 0770000(octal), an error message is printed and the TABLE OPTION query is reprinted. The system manager is free to try again. If the new BASIC-PLUS address is accepted, the TABLE OPTION query is reprinted and another option can be entered. The LIST table option can be used to verify the location of BASIC-PLUS. The procedure is shown in the example below.

```
ANY MEMORY ALLOCATION CHANGES ? YES

TABLE OPTION ? LIST

MEMORY ALLOCATION BREAKDOWN:

0000000 - 0117777 ( 20K) : EXEC
0120000 - 0207777 ( 14K) : BASIC
0210000 - 1777777 (222K) : USER
2000000 -   END          : NXM

TABLE OPTION ? LOCATE
NEW RUN-TIME SYSTEM ADDRESS IS ? 300000
```

```
TABLE OPTION ? LIST

MEMORY ALLOCATION BREAKDOWN:

0000000 - 0117777 ( 20K) : EXEC
0120000 - 0277777 ( 28K) : USER
0300000 - 0467777 ( 14K) : BASIC
0470000 - 1777777 (194K) : USER
2000000 -    END         : NXM
```

Under certain conditions, BASIC-PLUS should not be located in high-speed memory. Consider the configuration below where the memory allocated to the Monitor is shown but, for the purpose of this illustration, the location of BASIC-PLUS is not shown.

Locations 0000000 through 0117777 (20K) are RSTS Executive
Locations 0120000 through 0177777 (12K) are available core
Locations 0200000 through 0277777 (16K) are available MOS
Locations 0300000 through 0337777 (8K) are available core
Locations 0340000 upwards are nonexistent.

Assume that the size of BASIC-PLUS is 14K words. If BASIC-PLUS is located at address 200000, the remaining user memory is divided into one 12K section below and one 10K section above. Because user job images are loaded into contiguous physical memory, no program larger than 12K can run with this memory configuration. Therefore, it is better to locate BASIC-PLUS at the high end of available memory starting at address 250000 and leave 22K words of contiguous memory for user jobs between the Monitor and BASIC-PLUS. This configuration allows 16K programs to be run at the expense of not using the high speed MOS memory to full advantage. Only 6K words of MOS are used for the RTS while the other 10K words are part of the 22K words of user memory. Similar situations force each memory configuration to be considered individually.

3.6.3.4 LOCK(LCK) Table Option

The LOCK table option locks out certain portions of memory. Locking memory prevents the RSTS/E system from using that memory during time sharing operations. This capability is valuable if a certain section of memory is defective and disrupting RSTS/E system operation. Providing sufficient user memory remains after the malfunctioning sections are locked, the system can be started for normal time sharing.

The system responds to LCK by printing a related query LOCKOUT ADDRESS?. The system manager can then type a 4- to 7-digit octal number to specify the 22-bit address of the 1K section of memory to be locked. If two numbers, separated by a dash, are entered, a range of successive 1K sections is locked. The 1K section corresponding to the first address and successive 1K sections between and including the second address are locked. After either a valid address or a range of addresses is entered, the TABLE OPTION query is reprinted and another option may be entered. The LIST option can be used to verify that the desired sections were locked out. The following example illustrates the procedure.

```
ANY MEMORY ALLOCATION CHANGES ? YES

TABLE OPTION ? LIST

MEMORY ALLOCATION BREAKDOWN:

0000000 - 0117777 ( 20K) : EXEC
0120000 - 0507777 ( 62K) : USER
0510000 - 0517777 ( 14K) : BASIC
0600000 -   END          : NXM

TABLE OPTION ? LCK
LOCKOUT ADDRESS IS? 240000-254000

TABLE OPTION ? LIST

MEMORY ALLOCATION BREAKDOWN:

0000000 - 0117777 ( 20K) : EXEC
0120000 - 0237777 ( 20K) : USER
0240000 - 0253777 (  4K) : LOCKED
0254000 - 0507777 ( 38K) : USER
0510000 - 0577777 ( 14K) : BASIC
0600000 -   END          : NXM

TABLE OPTION ?
```

Certain restrictions apply to locking out memory. Memory in use by the Monitor cannot be locked because the Monitor can not be relocated to another section of memory. Similarly, memory currently in use by BASIC-PLUS cannot be locked until it is relocated to a usable area of memory. If an attempt is made to lock out a portion of memory already in use by the Monitor or BASIC-PLUS, the error message PART OF THAT AREA IS ALREADY IN USE is printed. Finally, to remove a full bank of memory from use, the 4, 8, 16, or 32 contiguous 1K sections which comprise the hardware bank must all be locked. If the bad memory bank is interleaved, an amount of memory equal to twice its length should be locked.

Certain types of memory failures affect only one word. Other types affect larger sections or even the full hardware bank. Any memory failure must be carefully analyzed to determine which 1K section must be locked. After locking memory, study the resultant memory configuration to ensure that sufficient contiguous user memory is available to run the installation's programs. (See related discussion in Section 3.6.3.3.)

RSTS/E can continue running with certain types of memory failures. On systems with parity memory, parity errors are logged by the error handling routines in the monitor. The recovery procedures depend on the usage of the offending section of memory at the time of the error. Four cases need to be considered.

The first case is a parity error in a section of memory allocated to the Monitor, BASIC-PLUS, or the disk caching. Since continued system operation would be risky, the error is logged, a crash dump is taken (if the crash dump facility is enabled), and the system is reloaded. When the system restarts, the ERRCRS program is used to extract the error log information from the CRASH.SYS file and ERRDIS is used to print the location of the error. ERRDIS provides sufficient information to identify a 1K (Core) or 4K (MOS) section which should be locked out. If the error occurred in BASIC-PLUS or disk caching memory, these items can be relocated and the sections locked. If the error was in Monitor memory, the hardware must be repaired. In either case, a hard failure may prevent the system from running at all and, hence, neither ERRCRS nor ERRDIS can be run. The only recourse in such cases is to run memory diagnostics to locate the problem. If the failure was transient, the system manager can shut the system down to lock out memory or can continue time sharing. At some point, memory diagnostics should be run to reproduce the failure, if possible.

The second case differs slightly from the first case. If a parity error occurs in user memory, the parity error handling routines determine whether a single user is affected (resides in the malfunctioning section) or more than one user is affected. If more than one user is affected, the system is reloaded as described above. If a single user is affected, that user is aborted with the error message MEMORY PARITY FAILURE. The error is logged, the section is automatically locked out from future use (if the error was reproducible), and the system continues running. The bad section of memory should be permanently locked out (until repaired) using the LCK table option the next time the system is started.

The third case involves multiple parity errors which occur in rapid succession. RSTS/E halts at location 54 (56 is displayed in the address lights) if a second parity error occurs while the first parity error is being processed. Memory diagnostics must be used to locate the malfunctioning memory.

The fourth case concerns a parity error on a PDP-11/70. If the PDP-11/70 hardware cache malfunctions and causes a parity error, the hardware refetches the needed word from main memory and causes a warning parity trap to occur. RSTS/E logs this type of error, and, if the error occurs twice within one minute and is reproducible, the half of the cache causing the malfunction is disabled. RSTS/E continues running but with degraded performance. No message is printed but two successive errors within the same minute in the ERRDIS printout should alert the user. Examination of the control register data reported by ERRDIS shows which half of the cache is disabled.

On systems without parity memory, no way exists for the software to detect or locate a memory failure. Programs may get wrong results, memory management errors may occur, or any number of random problems may be observed. At worst, the system may crash with misleading clues to the cause.

3.6.3.5 UNLOCK Table Option

The UNLOCK table option is used to free a previously locked portion of memory for use by BASIC-PLUS or by user jobs. Typing UNLOCK in response to the TABLE OPTION query causes the system to print the related query UNLOCK ADDRESS IS?. The system manager can then enter one address, or two addresses separated by a dash, to unlock either a single 1K section or a range of successive 1K sections of memory. The procedure is the same as that described for LOCK. If the memory specified is not locked, the error message PART OF THAT AREA WAS NOT LOCKED is printed followed by the TABLE OPTION query.

3.6.3.6 RESET Table Option

The RESET option allows the initialization code to set up the memory allocation table. All memory which was locked is unlocked. BASIC-PLUS is moved to follow immediately after the Monitor in low physical memory. All memory reserved for disk caching is released to user space. Any new memory added to the system is made available for use. In fact, the RESET table option must be used whenever additional new memory is added. The initialization code recognizes and does not attempt to use memory which has been removed. However, the initialization code does not alter the memory allocation table to include previously nonexistent memory unless told to do so.

RESET can be used with either the DEFAULT or START option. It usually does not make sense to use RESET at startup time, however. The types of memory allocation changes described above are most often permanent changes. RESET should only be used, therefore, with the DEFAULT option. The effect of RESET is demonstrated in the example below.

```
ANY MEMORY ALLOCATION CHANGES ? YES

TABLE OPTION ? LIST

MEMORY ALLOCATION BREAKDOWN:

0000000 - 0117777 ( 20K) : EXEC
0120000 - 0237777 ( 20K) : USER
0240000 - 0253777 (  4K) : LOCKED
0254000 - 0407777 ( 22K) : USER
0410000 - 0477777 ( 14K) : BASIC
0500000 -   END          : NXM

TABLE OPTION ? RESET

TABLE OPTION ? LIST

MEMORY ALLOCATION BREAKDOWN:

0000000 - 0117777 ( 20K) : EXEC
0120000 - 0207777 ( 14K) : BASIC
0210000 - 0577777 ( 62K) : USER
0600000 -   END          : NXM

TABLE OPTION ?
```

3.6.3.7 CACHE Table Option

The CACHE table option allows the system manager to reserve 1K portions of memory for disk caching by the FIP buffering module.[1] Any memory reserved for disk caching is employed solely by the FIP buffering module. The cache memory maintains a constant amount of FIP buffering on his system because all of the memory reserved for disk caching must be claimed before small buffers are used by the FIP buffering module.

When the CACHE command is typed, an additional query DISK CACHE ADDRESS RANGE? is printed. A 4- to 7-digit octal number must be entered to specify the 22-bit address of the 1K section of memory to be reserved for disk caching. If two numbers, separated by a dash, are entered, a range of successive 1K sections is reserved. The range consists of the 1K sections between the first address and including the 1K section specified by the second address.

The lowest 1K section must start above the lowest 28K words of memory (addresses 0000000 through 0157777(octal)). The range of memory must be available as user space and not reserved for other usage. There is no limit on the address of the upper 1K section. However, the FIP buffering module can only make use of the first 175K words of the range.

The number of directory blocks which a given range of disk caching memory holds is approximately one less than four times the number of 1K word sections in the range. If N is the number of 1K sections in the range, the number of disk blocks is given exactly by the following formula.

$$\text{minimum} \left(\frac{(1024 * N) - 32}{262}, 682 \right)$$

Any remainder from the division is truncated.

The number of directory blocks held in the small buffer pool varies depending on the demands for small buffers. A reasonable approximation is the number of free small buffers divided by 20 and the number of

---

[1] The FIP buffering module is optional and is not present on all RSTS/E systems. See Section 2.7.22 for more information on FIP buffering.

large buffers. The FIP buffering module does not use either small or large buffers if more than 8K words of memory are reserved for disk caching.

To release disk caching memory for user job area, type REMOVE in response to the DISK CACHE ADDRESS RANGE query. To reserve a portion of memory for disk caching when a portion is currently reserved, type the new range. The CACHE routines automatically release the current disk cache memory to user job space and reserve the new portion specified.

The following sequence demonstrates the use of the CACHE table option.

```
TABLE OPTION ? CACHE

DISK CACHE ADDRESS RANGE ? 700000-1100000

TABLE OPTION ? LIST

0000000 - 0123777 ( 21K) : EXEC
0124000 - 0213777 ( 14K) : BASIC
0214000 - 0603777 ( 62K) : LOCKED
0604000 - 0677777 ( 15K) : USER
0700000 - 1103777 ( 33K) : CACHE
1104000 - 2777777 (239K) : USER
3000000 -   END          : NXM

TABLE OPTION ?
```

Impact of Disk Caching on System Performance

Disk caching improves performance of system operations which involve accessing directory blocks. The following are representative examples.

File OPEN and CLOSE operations
Updating an in-core window of a file
Listing directories
Wild card file operations
Disk cleaning operations

The improvement of individual operations can be none (when the operation accesses no blocks in the cache) or as much as a 10-fold increase in speed.

Disk caching slightly degrades performance of system operations which involve computing. This degradation is caused by increased overhead in the monitor and can be as low as a fraction of 1% or as high as 12%. In general, the central processor degradation increases in direct proportion to the improvement in directory related operations.

To determine the usefulness of disk caching and the FIP buffering module, the system manager should study the characteristics of the processing to be done on the system. If the most important jobs on the system are heavily compute bound, do not make many directory references, and require good response, the CPU degradation caused by disk caching may outweigh any performance improvement. If the small buffer pool is heavily utilized (less than 50 free small buffers are typically available) and memory space is not available to increase the number of buffers, the FIP buffering module may not be useful. Additionally, if memory requirements are currently strained, disk caching can not be effectively used. In the final analysis, the increase in size caused by the FIP buffering module may make the Monitor unacceptably large.

3.6.3.8 EXIT Table Option

When all desired changes to the memory allocation table have been made, the EXIT command causes the DEFAULT (or START) routines to continue to the next query, CRASH DUMP? The crash dump facility can be enabled or disabled as described in Section 3.6.4.

3.6.4 CRASH DUMP Facility

After the memory allocation has been determined, the DEFAULT and START routines print the current status of the crash dump facility. A subsequent query allows the system manager to enable or disable crash dumps. The CRASH.SYS file created by the REFRESH option (see Section 3.4.2.8) must exist if crash dumps are to be enabled.

When the DEFAULT option is used for the first time, crash dumps are disabled even if CRASH.SYS exists. Whenever crash dumps are disabled but the CRASH.SYS file exists, the following message is printed, followed by the CRASH DUMP query.

```
YOU CURRENTLY HAVE CRASH DUMP DISABLED.

CRASH DUMP?
```

If the crash dump facility is enabled and CRASH.SYS exists, the message reflects the current condition.

```
YOU CURRENTLY HAVE CRASH DUMP ENABLED.

CRASH DUMP?
```

If the CRASH.SYS file does not exist, the following message is printed regardless of the current state of the crash dump facility.

```
BUT, SINCE THE CRASH DUMP FILE DOESN'T EXIST, CRASH
DUMP MUST BE DISABLED.
```

The initialization routines automatically disable the crash dump facility if the CRASH.SYS file does not exist. If the system manager wishes to enable the crash dump facility, he must use the REFRESH option to create the CRASH.SYS file. Once the file is created, the crash dump facility can be enabled permanently, using the DEFAULT option, or temporarily, using the START option. If the crash dump facility is disabled automatically, the DEFAULT routine saves the specified default parameters in the RSTS.CIL file on the system disk and returns to the OPTION query.

When the CRASH DUMP query is printed by either the DEFAULT or START option, the responses shown in Table 3-14 are valid.

Table 3-14

Possible Responses to CRASH DUMP Query

<table>
<tr><th>Response(s)</th><th colspan="3">Meaning</th></tr>
<tr><td rowspan="5">LINE FEED key
OLD
O</td><td colspan="3">The default response. If typed, the system checks for the existence of the CRASH.SYS file and establishes the result shown below based on the current condition of the crash dump facility and the existence of CRASH.SYS.</td></tr>
<tr><td>Current Condition</td><td>CRASH.SYS Exists?</td><td>Result</td></tr>
<tr><td>Enabled</td><td>Yes</td><td>Remains enabled</td></tr>
<tr><td>Enabled</td><td>No</td><td>Automatically disabled</td></tr>
<tr><td>Disabled</td><td>N/A</td><td>Remains disabled</td></tr>
<tr><td>YES
Y</td><td colspan="3">Enables the crash dump facility.</td></tr>
<tr><td>NO
N</td><td colspan="3">Disables the crash dump facility.</td></tr>
<tr><td>RETURN key</td><td colspan="3">Causes an explanatory form of the query to be printed, after which the user types his response.</td></tr>
<tr><td>CONTROL key and C combination</td><td colspan="3">Returns user to OPTION query.</td></tr>
</table>

After enabling or disabling the crash dump facility, the DEFAULT routines print the next query described in Section 3.6.5. START routines continue by asking for the current date and time prior to loading the system for normal timesharing.

For more information on the crash dump facility, see Sections 2.3 and 3.1 of the RSTS/E System Manager's Guide.

3.6.5 Magtape Labelling Default

At system generation time, the DEFAULT routines print the labelling question as follows.

```
MAGTAPE LABELLING DEFAULT (NONE) :
```

Because a default is not set, the text indicates NONE as the current default. Either DOS or ANSI must be typed to set the default.

After the default is initially established, the DEFAULT routines print the current setting in the labelling question as follows.

```
MAGTAPE LABELLING DEFAULT (DOS) :
```

Table 3-15 gives the valid responses to the LABELLING query.

Table 3-15

Valid Responses to LABELLING Query

Response(s)	Meaning
LINE FEED key or OLD	Keep the current default. (Not valid if current default is NONE.)
DOS	Set the default format to DOS/BATCH-11 labelling.
ANSI	Set the default format to ANSI standard labelling.
RETURN key	Print explanatory form of query.

After the response to the labelling query is answered, the DEFAULT routines save the default parameters and return to the OPTION query.

The magtape labelling format set by DEFAULT can be reset and restored for individual jobs during time sharing. An individual job can set the default by the ASSIGN command described in Section 2.7.4 of the RSTS-11 System User's Guide. An individual program can set the default by the MODE option in the magtape OPEN statement described in Section 2.2 of the RSTS/E Programming Manual. The default for the system can be reset only by running the DEFAULT option.

3.7 START OPTION - STARTING TIME SHARING OPERATIONS

The START initialization option is used to bring the RSTS/E system up for normal time sharing. The START option is similar to the DEFAULT option in that job and swap maximums, the current memory allocation, and the condition of the crash dump facility may all be set during the START sequence. As emphasized in the previous section, any changes made to the start-up parameters override the DEFAULT parameters for only one time sharing session. The DEFAULT start-up conditions always apply unless specifically changed each time the START option is used.

The START option is requested by typing START, or ST, or simply the LINE FEED key in response to the initialization code OPTION query. The following sample dialogue shows the normal procedure for bringing up a new RSTS/E system for the first time.

```
OPTION: START

YOU CURRENTLY HAVE: JOB MAX = 31, SWAP MAX = 16K.

JOB MAX OR SWAP MAX CHANGES ?

ANY MEMORY ALLOCATION CHANGES ?

CRASH DUMP ?

DD-MMM-YY? 22-JUL-74
HH:MM? 12:18

CAN'T FIND FILE OR ACCOUNT
PROGRAM LOST - SORRY

READY
```

The first three queries are the same as those printed when the DEFAULT option is used. Job and swap maximums are described in Section 3.6.2. Memory allocation options are discussed in Section 3.6.3. The enabling or disabling of the crash dump facility is the subject of Section 3.6.4. Normally the DEFAULT start-up conditions are not changed. In the example, these queries are answered with the LINE FEED key which implies no change to the related DEFAULT start-up condition.

START continues by asking for the date and time which must be entered in the exact format shown. After date and time are accepted, there is a pause while the START code is executed. During these several seconds the start-up routines enable all configured terminal interfaces, dump the overlay code and error messages from the RSTS.CIL file to the OVR.SYS and ERR.SYS files (if they exist), set up monitor tables, turn on memory management, load the Monitor and Run Time System, and start the system clock. Finally the system attempts to execute the INIT system program (described in Section 3.1 of the RSTS/E System Manager's Guide). When a new system is brought up for the first time, INIT does not yet exist in the system library. As a result, the error messages shown in the example are printed. The system initialization routines terminate as signaled by the printing of the READY message.

When the READY message is printed, job 1 is logged into the system at the console terminal under account [1,2] and is at the BASIC-PLUS command level. The situation is the same as if the system manager had logged into the system by use of the HELLO command described in Section 2.1.2 of the RSTS-11 System User's Guide. At this point, the system manager begins to build the files in the system library on the system disk.

The system library consists of two types of files: compiled system program files and formatted ASCII text and message files. The system manager follows the procedures described in Section 4.2 to build the compiled files. After the compiled files are created, the system manager must follow the procedures described in Section 4.3 to create the ASCII text and message files.

3.7.1 Start Up Errors and Messages

The START sequence is a critical part of the system initialization. START references all configured terminal interfaces and disables any that do not respond. The KT11 memory management unit is activated to determine the size of available memory (as in DEFAULT), to load the monitor and Run Time System, and to prepare for normal time sharing. START, furthermore, loads the stack limit register, enables parity traps for all parity memory, and activates the system clock to begin time sharing. Terminal interfaces, other than that for the console terminal, and the stack limit register have not been used up to this time. Even the clock was not critical to the operation of the System Generation Monitor or the other initialization routines. If the system does not come up as shown in the preceding example, these components are suspect and should be exercised with the standard diagnostics.

The message "SWAP MAX OF XX BEING LOWERED TO YY" is printed if the available contiguous memory is not large enough to support the requested value of SWAP MAX (see Section 3.6.2). This condition can occur if the BASIC-PLUS Run Time System is located such that it subdivides the contiguous user memory. To correct this situation the DEFAULT option can be used to relocate the BASIC-PLUS Run Time System (see LOCATE and LIST memory allocation options in Sections 3.6.3.1 and 3.6.3.3). A similar situation can occur on systems with a minimum amount of memory but many peripherals or software options. The amount of user memory left over may not be sufficient to support the desired SWAP MAX. The only way to correct this situation (other than adding memory) is to perform a new system generation requesting few jobs and fewer options. In either case, the RSTS/E system operates normally with the lower SWAP MAX but there may be limitations on which system programs can be run.

The START code attempts to enable all configured terminal interfaces, DH11 terminal multiplexers, and DM11BB modem control multiplexers. If any of these configured devices are not found, a message is printed indicating that the keyboards associated with these interfaces are being disabled. The hardware or software may not be configured properly or certain interfaces may be missing or inoperative. DEC Field Service should check the hardware configuration if unexplained messages of this type are printed. In most cases the system will operate normally after the affected keyboards are disabled. However, it is impossible for the software to run with certain types of hardware configuration errors. These messages may precede a crash or signal hardware problems.

Finally, if the processor halts during the START sequence, the possibility exists that the wrong type of system clock was configured.

START also performs certain software checks to verify that the required system files exist and the DEFAULT start-up conditions have been established. Any attempt to start time sharing before the DEFAULT option is run results in the following message.

```
OPTION: START

DEFAULTS MUST BE SET PRIOR TO START UP

OPTION:
```

The system manager must execute the DEFAULT option (see Section 3.6) before he can begin timesharing.

Similarly, if the required system files do not exist, the START sequence aborts with the message below.

```
OPTION: START

CANNOT START WITH THIS SYSTEM DISK - PLEASE REFRESH

OPTION:
```

The REFRESH option must also be used to create the system files before timesharing can begin.

3.8 UNISYS OPTION

The UNISYS option provides a way to bring up the RSTS/E system for time sharing without enabling any terminal interfaces except the console (KB0) interface. UNISYS is used as a diagnostic tool in cases where the system will not come up. If an otherwise nonworking system does work with UNISYS, the hardware or the software terminal configuration is probably faulty. Responses given in the SYSGEN questions should be checked against the hardware configuration. Communications devices and terminal interfaces should be checked by DEC Field Service for proper UNIBUS addresses and interrupt vectors.

The UNISYS option merely sets a flag which is checked at start-up time. The only way to clear the flag is to reboot the RSTS/E system disk to reload the initialization code (see BOOT option in Section 3.9). UNISYS is therefore used immediately before the START option. When the flag is set, START bypasses the code which enables terminal interfaces. The option is used as shown below.

```
     OPTION: UNISYS
                                        (Return to OPTION
                                         is immediate)
     OPTION: START

    (normal START messages)
```

3.9 BOOT OPTION - BOOTSTRAPPING A DEVICE

The RSTS/E initialization code includes bootstrap routines for common disk and tape devices. The BOOT option is provided as a convenient means to simulate the hardware ROM bootstraps. BOOT is used to load another operating system into memory from its system disk and to reload an altered copy of the RSTS/E initialization code into memory after an INIT patch has been installed.

The BOOT option is run by typing BOOT or BO in response to the OPTION query. BOOT replies by asking for the device to be bootstrapped. The system verifies that the device exists before attempting the bootstrap operation. The following example demonstrates the use of the option.

```
OPTION: BOOT
BOOT DEVICE:
BOOTABLE DEVICES ARE:
        DT      DECTAPE DRIVE 0
        MT      MAGTAPE DRIVE 0
        MM      TU16 MAGTAPE DRIVE 0
        DC      RC11 DISK
        DF      RF11 DISK
        DK      RK11 DISK
        DP      RP11 DISK
        DB      RP04 DISK
PLEASE SELECT ONE OF THOSE DEVICES: DT
```

For RK, RP02, RP03, and RP04 disks, the BOOT option allows a nonzero unit to be bootstrapped. Another query is printed as shown below.

```
BOOT DEVICE:     DB
BOOT UNIT ? 1
```

Typing the LINE FEED key in response to the BOOT DEVICE query bootstraps the device on which the RSTS/E system disk is mounted. The BOOT routines determine which device has the system disk and therefore the BOOT UNIT query is not printed. The LINE FEED key is not an acceptable response to the BOOT UNIT query.

The program or secondary bootstrap for the monitor to be loaded must reside on unit 0 if the device requested is not disk.

Initialization Options

The first 64 words from block 0 on the device are read and loaded into the first 64 words of memory. Control then transfers to address 0. Any monitor or program which can be loaded using the hardware bootstraps can also be loaded using the BOOT option. The new monitor overlays the RSTS/E initialization code.

The RSTS/E system disk can be dismounted if the drive is needed for the system to be loaded. The initialization code continues running when the system disk is removed from the drive. Hence, if a different operating system is to be loaded, its system disk can replace the RSTS/E system disk.

The default device (LINE FEED response to the BOOT DEVICE query) is useful when the initialization code is patched using the PATCH option (see the example in Section 3.2.2). The PATCH option modifies the RSTS.CIL file. To load the altered INIT code into memory, the LINE FEED response is used to bootstrap the RSTS/E system disk.

3.10 LOAD OPTION - LOADING STAND-ALONE PROGRAMS

The LOAD option provides a mechanism for loading and executing stand-alone programs from the RSTS.CIL file. The DEC supplied program which currently can be included in the CIL is ROLLIN. The option is invoked by typing LOAD or LO in response to the OPTION query. The option then asks for the name of the program to load. A table of loadable programs will be printed if the RETURN key is typed in response to the LOAD PROGRAM query as shown in the example below.

```
OPTION: LOAD
LOAD PROGRAM:
  TABLE OF LOADABLE PROGRAMS:
 NAME   IDENT   LOAD    SIZE    TRANSFER
ROLLIN  V07     001000  020672  001332
PLEASE ENTER NAME OF PROGRAM TO LOAD: ROLLIN


ROLLIN V07

#
```

If the program to be loaded has an odd transfer address[1], the following message is printed.

```
THE TRANSFER ADDRESS OF THE SPECIFIED PROGRAM IS ODD.
PLEASE ENTER A NEW (EVEN) TRANSFER ADDRESS OR TYPE
CONTROL/C TO ABORT PROGRAM LOAD, NEW TRANSFER ADDRESS?
```

The PDP-11 computer traps to an error vector if an odd (byte) address is specified. Therefore, the user types an even (word) octal address at which the program is to be started or types the CONTROL key and C key combination simultaneously (CTRL/C) to abort the LOAD operation.

If the new transfer address typed is odd, the following message is printed and INIT returns to the OPTION query.

```
ILLEGAL NEW TRANSFER ADDRESS SPECIFIED
OPTION:
```

If the transfer address entered is even, the system searches for the specified program, loads it into memory, and starts execution. When execution of the stand-alone program has completed, RSTS/E can be restored by following the applicable bootstrap procedures. For ROLLIN, simply type the command /BO:dev.

---

[1] Programs assembled or linked without specifying a transfer address are given a default transfer address of 000001. This is an odd address.

3.11 FILL OPTIONS - ALTERING CONSOLE TERMINAL CHARACTERISTICS

The characteristics of the console terminal (KB0:) are determined by the response to the system generation question CONSOLE TYPE?. The response to the question establishes the device data block (DDB) for that type of terminal. If the console terminal is either an LA30 or a VT05, the response also establishes proper fill characteristics in the initialization code and in the ROLLIN module included in the RSTS/E core image library (CIL).

The console terminal characteristics can be overridden so that, while the initialization code is running, a different fill factor is used. The fill option commands ASR33, VT05B, and LA30S set the fill characteristics to enable the initialization code to be run on a console terminal other than the type specified at system generation time. These characteristics are effective only while the current copy of the initialization code is running. When the system disk is bootstrapped again or when time sharing starts, the characteristics of the console terminal revert to those set at system generation time. The fill option commands do not affect the characteristics set in the ROLLIN module at system generation time.

If the characteristics of the console terminal are to be altered for time sharing without regenerating the system, the proper macro command can be placed in the START.CTL and CRASH.CTL control files. At each system start up, the DDB for KB0: is set by TTYSET rather than determined by the system generation characteristics. For more information on control files, see Section 3.1 of the RSTS/E System Manager's Guide.

CHAPTER 4

BUILDING SYSTEM LIBRARY FILES

The system generation procedures continue with the system manager building the system library files and creating user accounts. The DSKINT initialization option creates the system library account [1,2] on the system disk. The BUILD program is used under time sharing to create the system library files. Tailored text and message files are created with PIP or EDIT and user accounts are created with REACT.

When time sharing operations start, the monitor sets up job 1 under account [1,2] and attempts to run the INIT system program. Since no programs initially exist in account [1,2], the monitor generates the CAN'T FIND FILE OR ACCOUNT error and prints the READY message. The console terminal (KB0:) is at BASIC-PLUS command level and job 1 is active as if a user had logged into the system.

With the console terminal at command level, a single command is typed to run BUILD from the medium on which the RSTS/E system library files are stored. Once started, the BUILD program runs detached, reads commands from a control file, and executes those commands at the console terminal. Upon completion, the BUILD program reattaches job 1 to the console keyboard.

The RSTS/E software contains several control files listed and described below.

BUILD.CTL	Builds a standard system library usable on all systems.
SPLER.CTL	Builds files executable only on a system with a SWAP MAX value larger than 8K words.

RJ2780.CTL	Builds a file executable only on systems having the RSTS/2780 optional software.
AUXBLD.CTL	Builds auxiliary run time system used by languages other than BASIC-PLUS.

The BUILD program must be run once for each set of files required. Section 4.1 describes the files created by each control file and Section 4.2 describes the procedures to run BUILD from the three types of media. The system programs built by these procedures are documented in Chapter 4 of the RSTS-11 System User's Guide and in the RSTS/E System Manager's Guide. The system manager can remove from the RSTS-11 System User's Guide descriptions of programs he does not wish to be run on the system. The RJ2780 program is described in the RSTS/E RJ2780 User's Guide.

After the appropriate files are built, several ASCII text and message files must be created. The procedures to do this are given in Section 4.3. The user accounts must be created on the system disk. The procedures for this activity are in Section 4.4. Section 4.5 describes the auxiliary run time system build procedures and Section 4.6 describes the COBOL build procedure.

4.1 CONTROL FILES FOR THE BUILD PROGRAM

This section describes the programs and files created by BUILD when it executes commands in each control file.

4.1.1 BUILD.CTL File

Commands in the BUILD.CTL file create a standard system library for all systems. The programs and files are listed and described in Table 4-1. Procedures to change the sample ASCII text and messages are given in Section 4.3. References provided in Table 4-1 are section numbers in either the RSTS-11 System User's Guide (SUG) or the RSTS/E System Manager's Guide (SMG).

Table 4-1
BUILD.CTL Programs and Files

Program or File Name	Protection (if other than <124>)	Description	Section Reference[1]
LOGIN.BAC	<232>	Logs users into system	SUG - 2.1, 4.1
LOGOUT.BAC	<232>	Logs user off system	SUG - 2.1, 4.2
BUILD.BAC		Builds system library files	Chapter 4
UTILTY.BAC		System utility program	SMG - 7.1
INIT.BAC		Time sharing initialization program	SMG - 3.1
SYSCAT.BAC		Creates a directory listing of accounts on a file structured device.	SMG - 4.3
PRIOR.BAC		Changes priority, run burst, and SWAP MAX value.	SMG - 3.3
TALK.BAC	<232>	Terminal communications	SMG - 7.8.2
ANALYS.BAC		Crash analysis	SMG - 6.2
SYSTAT.BAC	<232>	Status reports	SMG - 7.2 SUG - 4.3
UMOUNT.BAC	<232>	Private disk mounter and dismounter	SUG - 4.14
QUOLST.BAC	<232>	Lists disk quota and usage data for current user	SUG - 4.6
ERRCPY.BAC		Copies hardware error data to a disk file	SMG - 6.1.1
ERRDIS.BAC		Formats data saved by ERRCPY.BAC and ERRCRS.BAC	SMG - 6.1.3
ERRDI1.BAC		Chained to by ERRDIS.BAC	SMG - 6.1.3
SHUTUP.BAC		Performs standard system shut down	SMG - 3.2
PIP.BAC	<104>	Peripheral interchange program to transfer files (8K version)	SUG - 4.4

(Continued on next page)

---

[1] The abbreviation SUG refers to the RSTS-11 System User's Guide and SMG refers to the RSTS/E System Manager's Guide.

Table 4-1 (Cont.)
BUILD.CTL Programs and Files

Program or File Name	Protection (if other than <124>)	Description	Section Reference[1]
FILCOM.BAC	<104>	Compares ASCII files	SUG - 4.15
MONEY.BAC	<104>	System accounting program	SUG - 4.7 SMG - 4.2
GRIPE.BAC	<232>	User comments program	SUG - 4.8 SMG - 7.7
REACT.BAC		Account creator	SMG - 4.1
TTYSET.BAC	<232>	Terminal characteristics setter	SUG - 4.5 SMG - 7.4
PLEASE.BAC	<232>	Console terminal request printer	SMG - 7.8.1
INUSE.BAC	<232>	Terminal in use warning	SUG - 4.16
RESEQ.BAC	<104>	BASIC-PLUS line resequencer	RESEQ.DOC
CREF.BAC	<104>	BASIC-PLUS cross reference program	CREF.DOC
START.CTL	<60>	Sample system start up control file	SMG - 3.1.2
CRASH.CTL	<60>	Sample system crash recovery control file	SMG - 3.1.2
ACCT.SYS	<60>	Sample system accounts file	SMG - 4.1.3
TTYSET.SPD	<60>	Sample terminal speed control file	SMG - 7.4.1
PIP.TXT	<40>	PIP help message text file (8K version)	SUG - 4.4.2
ERRDIS.HLP	<60>	Error display program help file	SMG - 6.1.3
HELP.TXT	<40>	Sample system help message text file	SUG - 2.1.4

(Continued on next page)

---

[1] The abbreviation SUG refers to the <u>RSTS-11 System User's Guide</u> and SMG refers to the <u>RSTS/E System Manager's Guide</u>.

Table 4-1 (Cont.)
BUILD.CTL Programs and Files

Program or File Name	Protection (if other than <124>)	Description	Section Reference[1]
NOTICE.TXT	<40>	Sample system notices text file	SUG - 2.1.2, 4.1.1
RESEQ.DOC	<40>	RESEQ program document file	
CREF.DOC	<40>	CREF program document file	
EDIT.BAC	<104>	DOS/BATCH-11 compatible text editor	SUG - 4.9
EDITCH.BAC	<104>	Chained to by EDIT.BAC	SUG - 4.9
BACKUP.BAC	<232>	System back up (fail safe)	SUG - 4.10
BACKDK.BAC	<232>	Chained to by BACKUP.BAC	SUG - 4.10
ODT.BAC		Octal debugging tool	SMG - 6.3
DSKINT.BAC		Initializes disks	SMG - 7.5
DIRECT.BAC	<232>	Disk directory lister	SUG - 4.13
COPY.BAC	<104>	Copies entire tapes and disk cartridges	SUG - 4.4.7
ERRCRS		Recovers error data from CRASH.SYS file	SMG - 6.1.2
REORDR		Restructures directory blocks for optimal access	SMG - 7.6
BACKUP.TXT	<40>	Describes BACKUP program	SUG - 4.10
UTILTY.TXT	<60>	UTILTY program help message file	SMG - 7.1.1
COPY.TXT	<40>	COPY program help message file	SUG - 4.17

---

[1] The abbreviation SUG refers to the RSTS-11 System User's Guide and SMG refers to the RSTS/E System Manager's Guide.

4.1.2 SPLER.CTL Files

Commands in the SPLER.CTL file create programs which require a job area larger than 8K words. Table 4-2 describes these programs and gives an approximate job size for each. References in Table 4-2 are section numbers in either the RSTS-11 System User's Guide (SUG) or the RSTS/E System Manager's Guide (SMG).

To build these files, the system SWAP MAX value must have been set large enough when the DEFAULT initialization option was run. If the current job size is too small when BUILD creates a given program, the system generates the MAXIMUM CORE EXCEEDED error. After the BUILD program terminates, all of the SPLER.CTL files exist in the system library. Those program which generated the MAXIMUM CORE EXCEEDED error are not runnable and must be deleted to prevent further errors. If the queue management and batch facilities are not required, delete the programs QUE, SPOOL, BATCH, QUEMAN, and CHARS from the library and replace the special version of SHUTUP with the version supplied in the BUILD.CTL file set.

4.1.3 RJ2780.CTL File

Commands in the RJ2780.CTL file create the RJ2780.BAC program with a protection code of <232>. This file is for use by customers who have purchased the RSTS/E 2780 software package and have included the appropriate software in the RSTS/E system. The program is described in the RSTS/E 2780 User's Guide.

Table 4-2
SPLER.CTL Programs and Files

Program or File Name	Protection (if other than <124>)	Job Size (in words)	Description	Section Reference[1]
CHARS.BAC		8K	Creates the character generation file	SMG - App. C
QUE.BAC	<232>	10K	Creates requests for spooling programs	SUG - 4.11
QUEMAN.BAC		14K	Queue manager	SMG - 5.1
SPOOL.BAC		10K	Line printer spooling program	SMG - 5.2
BATCH.BAC		13K	Batch control spooling program	SUG - 4.12 SMG - 5.3
RUNOFF.BAC	<232>	14K	Document formatting program	RUNOFF.DOC and RUNOFF USER'S GUIDE
VT5DPY.BAC	<232>	15K	System status display on VT05	SMG - 7.3
SHUTUP.BAC		8K	Performs standard system shut down and terminates spooling operations automatically	SMG - 3.2
ERRDIS.BAC		12K	Special version of ERRDIS not requiring chain module ERRDI1. BAC	SMG - 6.1.3
START.CTL	<60>		Special version of START.CTL for systems with spooling	SMG - 3.1
CRASH.CTL	<60>		Special version of CRASH.CTL for systems with spooling	SMG - 3.1
RUNOFF.DOC	<40>		Explains use of sample file RUNOFF.RNO	See RUNOFF. BAC
RUNOFF.RNO	<40>		Sample RUNOFF text file	See RUNOFF. DOC

---

[1] The abbreviation SUG refers to the RSTS-11 System User's Guide and SMG refers to the RSTS/E System Manager's Guide.

Table 4-2 (Cont.)
SPLER.CTL Programs and Files

Program or File Name	Protection (if other than <124>)	Job Size (in words)	Description	Section Reference[1]
BATDCD.BAC		8K	Creates BATCH program command decoding file BATCH.DCD	SMG - C.2
PIP.BAC	<232>	15K	Extended Peripheral Interchange Program	SUG - 4.18
VT50PY.BAC	<232>	15K	Status display program for VT50. Created from DISPLY.BAS and VT50.DPY	SMG - 7.3
PIP.TXT	<40>		Extended PIP help message file	SUG - 4.18

---

[1] The abbreviation SUG refers to the RSTS-11 System User's Guide and SMG refers to the RSTS/E System Manager's Guide.

4.2 STARTING TIME SHARING TO BUILD THE LIBRARY

After the DEFAULT option described in Section 3.6 is executed, type START in response to the OPTION query to begin time sharing. The following sample dialog shows the procedures.

```
OPTION:  START
YOU CURRENTLY HAVE: JOB MAX = 17, SWAP MAX = 28K
JOB OR SWAP MAX CHANGES?  <LF>          <LF> denotes typing the
ANY MEMORY ALLOCATION CHANGES?  <LF>    LINE FEED key.
CRASH DUMP?  <LF>

DD-MMM-YY?  20-AUG-74
HH:MM?  8:55

CAN'T FIND FILE OR ACCOUNT
PROGRAM LOST - SORRY

READY
```

The remaining steps depend upon the medium on which the RSTS/E software is stored. For the magtape procedure, go to Section 4.2.1; for disk cartridge, go to Section 4.2.2; and for DECtape, go to Section 4.2.3.

4.2.1 Building the System Library from Magtape

Perform the following steps:

Ensure that the write enable ring is removed from the reel labelled DEC-11-ORSLA-D-MA7 (7-track) or DEC-11-ORSLA-D-MA9 (9-track), SYSTEM LIBRARY & RELIABILITY TAPE.

Mount this tape on unit 0. Ensure that no other drive is selected to 0.

Ensure that the FILE PROT indicator is on.

Position the tape at its load point. (The LD PT indicator comes on.)

Set the ON-LINE/OFF-LINE switch to its ON-LINE position. Ensure that the READY indicator comes on.

4.2.1.1 Building the Standard System Library, BUILD.CTL - Users on all systems must execute the commands from the BUILD.CTL file. Type the following command to execute the BUILD program from either TU10 or TU16 magtape.

[1,3]SY:
RUN ~~MT0~~:BUILD$

BUILD runs and prints the following lines:

```
SYSTEM BUILDER
SYSTEM BUILD?  YES
SYSTEM BUILD DEVICE?  MT0 [1,3]SY:
```

Type YES in response to the SYSTEM BUILD question and type MT0 (omit the colon) in response to the subsequent SYSTEM BUILD DEVICE question. The BUILD program begins executing commands in the BUILD.CTL file.

NOTE

> If any errors occur in reading the magtape, retry the procedure on another drive. If no other drive is available, either request Field Service to align the heads on the drive or obtain a new magtape.

The console printout of the entire procedure is shown below.

```
RUN MT0:BUILD$
SYSTEM BUILDER
SYSTEM BUILD? YES
SYSTEM BUILD DEVICE? MT0
OLD MT0:$LOGIN
COMPILE SY0:LOGIN
CHAIN 'MT0:$BUILD' 9200

READY

READY

READY

^C
HELLO

RSTS V06A-02 SYSTEM TEST  JOB 2  KB0   28-MAY-75  12:02
# 1 / 2
PASSWORD:
JOB(S) 1 ARE DETACHED UNDER THIS ACCOUNT
JOB NUMBER TO ATTACH TO?
1 OTHER USER(S) ARE LOGGED IN UNDER THIS ACCOUNT



READY

!********** BUILD.CTL - STANDARD LIBRARY PROGRAMS

RUN MT0:$PRIOR

'PRIOR' PRIORITY, BURST, MAXIMUM CHANGER

ENTER ANOTHER JOB NUMBER?

CURRENT STATISTICS ARE:
    -8 PRIORITY
     6 RUN BURST
   16K SIZE MAXIMUM
```

```
ANY CHANGES? YES
CHANGE PRIORITY? NO
CHANGE RUN BURST? NO
CHANGE SIZE MAXIMUM? YES
        CHANGE IT TO? 8

CURRENT STATISTICS ARE:
    -8 PRIORITY
     6 RUN BURST
    8K SIZE MAXIMUM

ANY CHANGES? NO

READY
OLD MT0:$BUILD
READY
COMPILE SY0:BUILD
READY
OLD MT0:$LOGOUT
READY
COMPILE SY0:LOGOUT
READY
OLD MT0:$UTILTY
READY
COMPILE SY0:UTILTY
READY
OLD MT0:$INIT
READY
COMPILE SY0:INIT
READY
OLD MT0:$SYSCAT
READY
COMPILE SY0:SYSCAT
READY
OLD MT0:$PRIOR
READY
COMPILE SY0:PRIOR
READY
OLD MT0:$TALK
READY
COMPILE SY0:TALK
READY
OLD MT0:$ANALYS
READY
COMPILE SY0:ANALYS
READY
OLD MT0:$SYSTAT
READY
COMPILE SY0:SYSTAT
READY
OLD MT0:$UMOUNT
READY
COMPILE SY0:UMOUNT
READY
OLD MT0:$QUOLST
READY
COMPILE SY0:QUOLST
READY
OLD MT0:$ERRCPY
READY
COMPILE SY0:ERRCPY
READY
```

```
OLD MT0:$ERRDIS
READY
COMPILE SY0:ERRDIS
READY
OLD MT0:$ERRDI1
READY
COMPILE SY0:ERRDI1
READY
OLD MT0:$ERRCRS
READY
COMPILE SY0:ERRCRS
READY
OLD MT0:$SHUTUP
READY
COMPILE SY0:SHUTUP
READY
OLD MT0:$EDIT
READY
COMPILE SY0:EDIT<40>
READY
OLD MT0:$EDITCH
READY
COMPILE SY0:EDITCH<40>
READY
OLD MT0:$BACKUP
READY
COMPILE SY0:BACKUP
READY
OLD MT0:$BACKDK
READY
COMPILE SY0:BACKDK
READY
OLD MT0:$ODT
READY
COMPILE SY0:ODT
READY
OLD MT0:$PIP
READY
COMPILE SY0:PIP<40>
READY
OLD MT0:$DSKINT
READY
COMPILE SY0:DSKINT
READY
OLD MT0:$DIRECT
READY
COMPILE SY0:DIRECT
READY
OLD MT0:$COPY
READY
COMPILE SY0:COPY<40>
READY
OLD MT0:$FILCOM
READY
COMPILE SY0:FILCOM<40>
READY
```

```
OLD MT0:$MONEY
READY
COMPILE SY0:MONEY<40>
READY
OLD MT0:$GRIPE
READY
COMPILE SY0:GRIPE
READY
OLD MTO:$REACT
READY
COMPILE SY0:REACT
READY
OLD MT0:$TTYSET
READY
COMPILE SY0:TTYSET
READY
OLD MT0:$PLEASE
READY
COMPILE SY0:PLEASE
READY
OLD MT0:$INUSE
READY
COMPILE SY0:INUSE
READY
OLD MT0:$REORDR
READY
COMPILE SY0:REORDR
READY
OLD MT0:$RESEQ
READY
COMPILE SY0:RESEQ<40>
READY
OLD MT0:$CREF
READY
COMPILE SY0:CREF<40>
READY

RUN PIP
PIP - RSTS V06A-02 SYSTEM TEST
#SY0:NOTICE.TXT<40><MT0:NOTICE.TXT$/FA
#SY0:HELP  .TXT<40><MT0:HELP  .TXT$/FA
#SY0:START .CTL    <MT0:START .CTL$/FA
#SY0:CRASH .CTL    <MT0:CRASH .CTL$/FA
#SY0:ACCT  .SYS    <MT0:ACCT  .SYS$/FA
#SY0:TTYSET.SPD    <MT0:TTYSET.SPD$/FA
#SY0:PIP   .TXT<40><MT0:PIP   .TXT$/FA
#SY0:ERRDIS.HLP    <MT0:ERRDIS.HLP$/FA
#SY0:RESEQ .DOC<40><MT0:RESEQ .DOC$/FA
#SY0:CREF  .DOC<40><MT0:CREF  .DOC$/FA
#SY0:BACKUP.TXT<40><MT0:BACKUP.TXT$/FA
#SY0:UTILTY.TXT    <MT0:UTILTY.TXT$/FA
#SY0:COPY  .TXT<40><MT0:COPY  .TXT$/FA
```

```
#=BACKUP.BAC<232>/RE
#=BACKDK.BAC<232>/RE
#=DIRECT.BAC<232>/RE
#=TALK  .BAC<232>/RE
#=SYSTAT.BAC<232>/RE
#=UMOUNT.BAC<232>/RE
#=QUOLST.BAC<232>/RE
#=GRIPE .BAC<232>/RE
#=TTYSET.BAC<232>/RE
#=PLEASE.BAC<232>/RE
#=INUSE .BAC<232>/RE
#=LOGIN .BAC<232>/RE
#=LOGOUT.BAC<232>/RE
#^C

READY

RUN UTILTY
SYSTEM UTILITY PROGRAM 'UTILTY V06-03'
? LOGINS
? EXIT

READY

^C

READY

HELLO

RSTS V06A-02 SYSTEM TEST  JOB 2  [1,2]  KB0  28-MAY-75  12:11
JOB(S) 1 ARE DETACHED UNDER THIS ACCOUNT
JOB NUMBER TO ATTACH TO? 1
ATTACHING TO JOB 1

BUILD COMPLETE

READY
```

Execution of the commands in the BUILD.CTL file takes approximately 10 minutes. The tape rewinds several times at the beginning of the procedure but does not rewind after the statement OLD MT0:LOGOUT$. Multiple copies of some programs are on the tape to reduce the number of rewinds required.

The end of execution is signalled by the messages BUILD COMPLETE and READY. Do not rewind or move the tape in any way. If the user is not including files from the SPLER.CTL and RJ2780.CTL file sets on his system, he need not build any more files and can proceed to Section 4.3 to create the ASCII text and message files. Proceed to Section 4.2.1.2 to build the spooling programs and files or skip to Section 4.2.1.3 to build the RJ2780 program.

4.2.1.2 Building the Large Programs, SPLER.CTL - Building the programs from the SPLER.CTL file requires a user job area larger than 8K words as described in Section 4.1.2. To continue building the library, leave the magtape at its current position and type the following command.

```
RUN $BUILD
```

If the magtape is moved, the system rewinds the magtape before searching for the first file (SPLER.CTL). The above command runs the BUILD program from the system disk and leaves the tape at its current position.

The program prints the following two lines.

```
SYSTEM BUILDER
SYSTEM BUILD? NO
```

Type NO in answer to the SYSTEM BUILD question. BUILD continues with the following questions.

AUXILIARY BUILD DEVICE? ~~MT0~~ [1,3]SY:
CONTROL FILE IS? SPLER.CTL$ SPLER.CTL

Type MT0 (omit the colon) as the device and type SPLER.CTL$ to specify the correct control file. BUILD begins executing the commands in the file SPLER.CTL. The complete console printout of the entire procedure is shown below.

```
RUN BUILD
SYSTEM BUILDER
SYSTEM BUILD? NO
AUXILIARY BUILD DEVICE? MT0
CONTROL FILE IS? SPLER.CTL
^C
HELLO

RSTS V06A-02 SYSTEM TEST  JOB 2  KB0  28-MAY-75  12:11
# 1 / 2
PASSWORD:
JOB(S) 1 ARE DETACHED UNDER THIS ACCOUNT
JOB NUMBER TO ATTACH TO?
1 OTHER USER(S) ARE LOGGED IN UNDER THIS ACCOUNT

READY
```

```
!*********** SPLER.CTL - LARGE PROGRAM BUILD
OLD MT0:PIPEXT
READY
COMPILE SY0:PIP
READY
OLD MT0:$CHARS
READY
COMPILE SY0:CHARS
READY
RUN CHARS
READY
OLD MT0:$QUE
READY
COMPILE SY0:QUE
READY
OLD MT0:$QUEMAN
READY
COMPILE SY0:QUEMAN
READY
OLD MT0:$SPOOL
READY
COMPILE SY0:SPOOL
READY
OLD MT0:$BATDCD
READY
COMPILE SY0:BATDCD
READY
RUN BATDCD
BATCH DECODE SET-UP - V06A-03
READY
OLD MT0:$BATCH
READY
COMPILE SY0:BATCH
READY
OLD MT0:$RUNOFF
READY
COMPILE SY0:$RUNOFF
READY
OLD MT0:$DISPLY
READY
SAVE SY0:DISPLY
READY
APPEND MT0:$VT05  .DPY
READY
COMPILE SY0:VT5DPY
READY
OLD DISPLY
READY
APPEND MT0:$VT50  .DPY
READY
COMPILE SY0:VT50PY
READY
UNSAVE DISPLY.BAS
READY
```

```
OLD MT0:$SHUTUP.SPL
READY
COMPILE SY0:SHUTUP
READY
OLD MTO:$ERRDIS.SPL
READY
COMPILE SY0:ERRDIS
READY
UNSAVE ERRDI1.BAC
READY
RUN $PIP
PIP V06A-15 - RSTS V06A-02 SYSTEM TEST
#SY0:START  .CTL     <  MT0:$START1.CTL
#SY0:CRASH  .CTL     <  MT0:$CRASH1.CTL
#SY0:PIP    .TXT<40><  MT0:$PIPEXT.TXT.CTL/FA
#SY0:RUNOFF.DOC<40><  MT0:$RUNOFF.DOC
#SY0:RUNOFF.RNO<40><  MTO:$RUNOFF.RNO
#=PIP    .BAC<232>/RE
#=QUE    .BAC<232>/RE
#=RUNOFF.BAC<232>/RE
#=VT5DPY.BAC<232>/RE
#=VT50PY.BAC<232>/RE
#^C

READY

HELLO

RSTS V06A-02 SYSTEM TEST  JOB 2· [1,2]  KB0  28-MAY-75  12:19
JOB(S) 1 ARE DETACHED UNDER THIS ACCOUNT
JOB NUMBER TO ATTACH TO?  1
ATTACHING TO JOB 1

BUILD COMPLETE

READY
```

Completion of the procedure is signalled by the BUILD COMPLETE message. To build the RJ2780 program, proceed to Section 4.2.1.3; otherwise, proceed to Section 4.3 to create the ASCII text and message files.

4.2.1.3 Building the RJ2780 Program, RJ2780.CTL - To build the RJ2780 program, run BUILD from disk again and specify the RJ2780.CTL file as shown in the following sample printout. This procedure can be performed only if the system includes the RSTS/2780 software and the job area is at least 10K words.

```
RUN BUILD
SYSTEM BUILDER
SYSTEM BUILD? NO
AUXILIARY BUILD DEVICE? MT0
CONTROL FILE IS? RJ2780.CTL
^C
HELLO

RSTS V06A-02 SYSTEM TEST  JOB 2  KB0  28-MAY-75  12:19
# 1 / 2
PASSWORD:
JOB(S) 1 ARE DETACHED UNDER THIS ACCOUNT
JOB NUMBER TO ATTACH TO?
1 OTHER USER(S) ARE LOGGED IN UNDER THIS ACCOUNT


READY

!********** RJ2780.CTL - RJ2780 ONLY

OLD MT0:$RJ2780
READY
COMPILE SY0:RJ2780
READY
RUN $PIP
PIP V06A-15 RSTS V06A-02 SYSTEM TEST
#=RJ2780.BAC<232>/RE
#^C

READY

HELLO

RSTS V06A-02  SYSTEM TEST  JOB 1  [1,2]  KB0  28-MAY-75  12:20
JOB(S) 1 ARE DETACHED UNDER THIS ACCOUNT
JOB NUMBER TO ATTACH TO?  1
ATTACHING TO JOB 1

BUILD COMPLETE

READY
```

The SPLER build procedure takes approximately 15 minutes. The tape does not rewind.

The RJ2780 procedure takes approximately two minutes.

Completion of the procedure is signalled by the BUILD COMPLETE message. Proceed to Section 4.3 to create the ASCII text and message files.

4.2.2 Building the System Library from Disk Cartridge

Perform the following steps.

Mount the copy of the system library disk cartridge labelled DEC-11-ORSLA-D-HA SYSTEM LIBRARY AND RELIABILITY DECPACK on a free unit.

Place the LOAD/RUN switch on the drive to its RUN position and ensure that the READY indicator comes on.

Depress the WR PROT switch so that the WR PROT indicator goes off.

At this point, the user must type in the following BASIC-PLUS program to logically mount the disk cartridge on the system. First, type the NEW MOUNT command to clear memory, after which the system prints READY. After entering the program, type the RUNNH command to execute it.

NOTE

> When entering the statement at line number 50, replace <UNIT NUMBER> with the number of the drive being used. For example, if drive 1 is used, type the following.
>
> `50 M%(25%) = 1`

The entire procedure is as follows.

```
NEW MOUNT

READY

10      DIM M%(30%)
20      CHANGE SYS(CHR$(6%)+CHR$(-10%)+"SYSLIB") TO M%
        ! GET THE PACK ID IN RADIX-50
30      M%(0%)=26% : M%(1%)=6%    : M%(2%)=3%
40      M%(3%)=0%  : M%(23%)=68% : M%(24%)=75%
50      M%(25%)=<UNIT NUMBER>
        ! REPLACE <UNIT NUMBER> WITH ACTUAL UNIT ON
        ! WHICH DISK IS RUNNING.
60      M%(26%)=255%
        ! SET UP FOR THE MOUNT SYS FUNCTION
70      CHANGE M% TO M$
80      M$=SYS(M$)
        ! MOUNT THE DISK
90 END

RUNNH

READY
```

If any errors occur, perform the following recovery procedures.

a) Ensure that the drive unit is READY and is write enabled.

b) Type the LISTNH command and check each line of the program to ensure each is exactly correct. For more information on inputting and editing BASIC-PLUS programs, see Sections 2.2.2 through 2.2.5 of the *RSTS-11 System User's Guide*.

If no errors are found by performing steps a and b above, retry the entire procedure several times. If possible, try different drive units and replace line 50 with the proper unit number. If the error persists, the drive heads may require alignment.

4.2.2.1 *Building the Standard System Library, BUILD.CTL* - The BUILD.CTL file must be executed on all systems. For the purposes of example, drive unit 1 is used in all commands. Wherever DK1 appears in the examples, substitute the device designator of the drive unit on which the disk cartridge is mounted. Type the following command to run the BUILD program from the proper disk cartridge.

```
RUN DK1: BUILD$
```

BUILD prints the following lines:

```
SYSTEM BUILDER
SYSTEM BUILD?  YES
SYSTEM BUILD DEVICE?  DK1
```

Type YES in answer to the SYSTEM BUILD question and type the proper device designator (omit the colon) to specify the drive unit in response to the SYSTEM BUILD DEVICE question. The unit number must be included since DK alone refers to the system device and DK0 refers to unit 0 of the RK drives. The BUILD program begins executing commands in the BUILD.CTL file. The printout for the program and the guidelines for completion time are the same as that given in Section 4.2.1.1 except that MT0 is replaced by the disk drive designator. Upon completion of the procedure, proceed to Section 4.2.2.2 to build spooling library programs, to Section 4.2.2.3 to build the RJ2780 program, or to Section 4.3 to create the ASCII text and message files.

4.2.2.2 Building the Large Programs, SPLER.CTL - Building the programs from the SPLER.CTL file requires a user job area larger than 8K words as described in Section 4.1.2. To continue building the library, type the following command.

```
RUN $BUILD
```

The program prints the following lines.

```
SYSTEM BUILDER
SYSTEM BUILD?  NO
```

Type NO in answer to the SYSTEM BUILD question, after which BUILD prints, in turn, the following questions.

```
AUXILIARY BUILD DEVICE?  DK1
CONTROL FILE IS?  SPLER.CTL$
```

Type the proper device designator (omit the colon) and type SPLER.CTL$ to specify the correct control file. BUILD begins executing commands in the file and prints messages similar to those given in Section 4.2.1.2.

When the procedure is completed, proceed to Section 4.2.2.3 to build the RJ2780 program or to Section 4.3 to create the ASCII text and message files.

4.2.2.3 Building the RJ2780 Program, RJ2780.CTL - To build the RJ2780 program, run BUILD again, specify the proper auxiliary device, and specify RJ2780.CTL as the control file. This procedure must be performed if the system includes the RSTS/2780 software. The job area must be at least 10K words. The entire procedure is similar to that shown in the sample printout of Section 4.2.1.3 except that the proper device designator replaces MT0. Upon completion, proceed to Section 4.3 to create the ASCII text and message files.

4.2.3 Building the System Library from DECtape

Perform the following steps:

Mount the DECtape reel labelled DEC-11-ORSCA-D-UA1, SYSTEM LIBRARY TAPE PART 1, on drive unit 0.

Mount DECtape reel labelled DEC-11-ORSCA-D-UA2, SYSTEM LIBRARY TAPE PART 2, on drive unit 1.

Set the WRITE ENABLE/WRITE LOCK switch on drive 0 and drive 1 to its WRITE LOCK position.

Set the REMOTE/OFF/LOCAL switch on drive 0 and drive 1 to its REMOTE position.

The tape mounted on unit 0 contains BUILD.CTL, the control file for the first half of the standard library build (PART I). The tape mounted on unit 1 contains a second BUILD.CTL file for the second half of the standard library build (PART II). Ensure that the correct tapes are mounted before proceeding.

4.2.3.1 Building the Standard System Library, BUILD.CTL - The BUILD.CTL file must be executed on all systems. Type the following command to execute the BUILD program from DECtape.

```
RUN DT0:BUILD
```

BUILD runs from the DECtape and prints the following lines:

```
SYSTEM BUILDER
SYSTEM BUILD? YES
SYSTEM BUILD DEVICE?  DT0
```

Type YES in response to the SYSTEM BUILD question and type DT0 (omit the colon) in response to the subsequent SYSTEM BUILD DEVICE question. The BUILD program begins executing commands from the BUILD.CTL file. The entire procedure takes approximately 20 minutes. The printout for the program is similar to that shown for magtape in Section 4.2.1.1 up to the processing of the BACKDK program.

Upon completion of the procedure, BUILD prints the BUILD COMPLETE and READY messages.

Building System Library Files

Rerun the BUILD program to build the remaining files for the standard system library.

```
RUN BUILD
SYSTEM BUILDER
SYSTEM BUILD?  NO
```

Type NO in response to the SYSTEM BUILD question and proceed as follows.

```
AUXILIARY BUILD DEVICE?  DT1
CONTROL FILE IS?  BUILD.CTL
```

Type DT1 as the device designator and type to specify the BUILD.CTL control file. BUILD runs and prints messages similar to those shown in Section 4.2.1.2 for magtape from the processing of the ODT program. The entire procedure takes approximately 15 minutes.

When the BUILD COMPLETE and READY messages appear, dismount the tapes from units 0 and 1 and store them in a safe place. Proceed to Section 4.2.3.2 to build large programs, to Section 4.2.3.3 to build the RJ2780 program, or to Section 4.3 to create ASCII text and message files.

4.2.3.2 Building the Large Programs, SPLER.CTL - Building the programs from the SPLER.CTL file requires a user job area larger than 8K words as described in Section 4.1.2. To continue the build procedure, perform the following steps:

Mount the DECtape reel labelled DEC-11-ORSCA-D-UA3, SYSTEM LIBRARY TAPE - LARGE PROGRAMS, on drive unit 0.

Set the WRITE ENABLE/WRITE LOCK switch on drive 0 to its WRITE LOCK position.

Set the REMOTE/OFF/LOCAL switch on drive 0 to its REMOTE position.

Type the following command to start the procedure.

```
RUN $BUILD
```

BUILD prints the following lines:

```
SYSTEM BUILDER
SYSTEM BUILD?  NO
```

Type NO in response to the SYSTEM BUILD question, after which BUILD prints, in turn, the following questions.

```
AUXILIARY BUILD DEVICE?  DT0
CONTROL FILE IS?  SPLER.CTL
```

Type DT0 (omit the colon) as the device designator and type SPLER.CTL to specify the correct control file. BUILD runs and prints messages similar to those shown in Section 4.2.1.2 for magtape. The entire procedure takes approximately 25 minutes.

When the BUILD COMPLETE and READY messages appear, dismount the tape from unit 0 and store it in a safe place. Proceed to Section 4.2.3.3 to build the RJ2780 program or to Section 4.3 to create the ASCII text and message files.

4.2.3.3 Building the RJ2780 Program, RJ2780.CTL - Note that the RJ2780 program must be installed only if the system includes the RSTS/2780 software. The job area must be at least 10K words. Perform the following steps:

Mount the DECtape reel labelled DEC-11-ORSCA-D-UA4, SYSTEM LIBRARY TAPE - SPECIAL PROGRAMS, on drive unit 0.

Set the WRITE ENABLE/WRITE LOCK switch on drive 0 to its WRITE LOCK position.

Set the REMOTE/OFF/LOCAL switch on drive 0 to its REMOTE position.

Execute the BUILD program again as described in Section 4.2.3.2 but type RJ2780.CTL to specify the control file. The complete procedure is summarized in the following sample printout.

```
RUN $BUILD
SYSTEM BUILD
SYSTEM BUILDER? NO
AUXILIARY BUILD DEVICE? DT0
CONTROL FILE IS? RJ2780.CTL
```

BUILD runs and prints messages similar to those shown for magtape in Section 4.2.1.3. The procedure takes only a few minutes. When the BUILD COMPLETE and READY messages appear, dismount the tape from unit 0 and store it in a safe place. Proceed to Section 4.3 to create the ASCII text and message files.

4.3 CREATING THE ASCII TEXT AND MESSAGE FILES

After the BUILD program terminates, remove the tape or disk cartridge containing the system library files and ensure that it is stored in a safe place with the remainder of the RSTS/E kit.

> NOTE
>
> For disk cartridge distribution, run the UTILTY program and execute the DISMOUNT command to dismount the system library cartridge.

At this point in the process of building the system library, run the PIP system program and create the required ASCII text files. (For information concerning the use of PIP, refer to Section 4.4 of the RSTS-11 System User's Guide.) The subsequent sections describe the procedures to follow when building the ASCII text files and contain references to further descriptions of the use and contents of the files. Editing conventions for the use of the RUBOUT key and CTRL/U as described in Section 2.2.3 of the RSTS-11 System User's Guide apply when using the PIP system program.

4.3.1 System Message File NOTICE.TXT

The file NOTICE.TXT is printed when a user is successfully logged into the system as described in Section 2.1.2 of the RSTS-11 System User's Guide. In this manner, the system manager can relay to the user information concerning operation of the installation or changes or additions to the system. Prior to creating his own NOTICE.TXT file, the system manager can print the sample file in the library on the console printer.

```
RUN PIP
PIP - RSTS  V05C-01  SYSTEM #880
#KB:/FA<NOTICE.TXT
WELCOME TO RSTS/E TIME SHARING
#
```

A tailored NOTICE.TXT file can be created and can replace the sample file by imitating the following sample procedure.

```
#SY0:NOTICE.TXT<40><KB:/FA
WELCOME TO RSTS/E SYSTEM #880
HOURS OF OPERATION ARE:

     MON-FRI   0800 TO 1700 HOURS
     SAT-SUN   ON REQUEST

TRANSMIT USER COMMENTS VIA THE GRIPE
SYSTEM PROGRAM
↑Z
#
```

The ↑Z at the end of the text file is necessary. It is the result of typing the CTRL/Z[1] combination and signals the end of the ASCII text, closes the file properly and returns control to PIP as indicated by the # character being printed again. The system manager can update the file as needed.

4.3.2 System Help File HELP.TXT

The file HELP.TXT is printed when a user types HELP at a logged-out terminal. The user can print the sample file stored in the system library by typing the following command string. (The example assumes that PIP is still running from the previous procedures.)

```
#KB:/FA<HELP.TXT
TO GET ON-LINE AND USE RSTS/E, FOLLOW THE
INSTRUCTIONS FOUND IN CHAPTER 2 OF THE
RSTS-11 SYSTEM USER'S GUIDE.
#
```

If the text is satisfactory, but the system manager wishes to add some local information, he can use the append feature of PIP, delete the old file and rename the new file.

```
#SY0:NEW.TXT<HELP.TXT,KB:/FA
CONTACT MR. JONES FOR A PROJECT-PROGRAMMER
NUMBER AND PASSWORD.
↑Z
#HELP.TXT/DE
#SY0:HELP.TXT<40>=NEW.TXT/RE
#
```

---

[1] See the description of control characters in Chapter 3 in the RSTS-11 System User's Guide and of CTRL/Z in Section 9.5 of the BASIC-PLUS Language Manual.

The sample HELP.TXT file can be replaced by using a command string similar to the one shown in Section 4.3.1.

4.3.3 Control Files START.CTL and CRASH.CTL

Two control files are required by the INIT system program whenever the RSTS/E system is initialized for time sharing. The use and content of both of these files are explained in Section 3.1 of the RSTS/E System Manager's Guide. The procedures described here show how to print the sample files and how to replace the sample files with versions created to suit the needs of the local installation.

It is essential that these files contain the information required to properly initialize the system for time sharing operations. Before the system manager attempts to modify or replace the contents of the example files, it is highly recommended that he thoroughly familiarize himself with the concepts presented in the RSTS/E System Manager's Guide.

```
#KB:/FA<START.CTL, CRASH.CTL
    (Both files are printed.)
#SY0:START.CTL<KB:/FA
    (Type new version.)
↑Z
#SY0:CRASH.CTL<KB:/FA/UP
    (Type new version.)
↑Z
#
```

As a result of following the above procedures, the sample files in the system library are replaced by the versions entered.

4.3.4 Terminal Speed Characteristics File TTYSET.SPD

The system manager must create the terminal speed characteristics file TTYSET.SPD. Its use and contents are explained in Section 7.4.1 of the RSTS/E System Manager's Guide. An entry in TTYSET.SPD must be created for each local or remote line which allows a variable baud rate. The following sample dialogue shows how to print the sample file in the library and how to replace the sample file with the new version.

```
#KB:/FA<TTYSET.SPD
     (The sample file is printed.)
#SY0:TTYSET.SPD<KB:/FA
     (Type new version)
↑Z
#
```

As a result of following the above procedure, the sample file in the system library is replaced by the new version.

4.3.5 Standard Account File ACCT.SYS

As a convenience, the REACT system program is designed to create a large number of user accounts automatically. To accomplish this, the system manager must create the ACCT.SYS file in the system library. The contents of the ACCT.SYS file are explained in Section 4.1.3 of the RSTS/E System Manager's Guide. A sample of ACCT.SYS is created by the BUILD program. This sample file can be printed and replaced by performing the following procedures. (It is suggested that the system manager make entries in ACCT.SYS to create a privileged and a non-privileged account for himself.)

```
#KB:/FA<ACCT.SYS
     (The sample file is printed.)
#SY0:ACCT.SYS<KB:/FA
     (Type new version.)
↑Z
#↑Z

READY
```

The new version of ACCT.SYS replaces the sample ACCT.SYS in the system library and the PIP system program run is terminated. Proceed to Section 4.4 to create the user accounts.

4.4 CREATING USER ACCOUNTS

The system manager can run REACT and use the STANDARD function to create the user accounts in the MFD of the system disk. The explanation of REACT is given in Section 4.1 of the RSTS/E System Manager's Guide. The following procedures describe the method used to create the user accounts from information in the ACCT.SYS file.

```
RUN REACT
'REACT' SYSTEM ACCOUNT MANAGER
FUNCTION?  STANDARD
ALL ACCOUNTS IN $ACCT.SYS ARE NOW ENTERED
FUNCTION?  ↑Z
READY
```

REACT prints an identification message and a request for a function. Type STANDARD or S and the REACT program creates user accounts from information in the ACCT.SYS file. When the standard function is completed, an advisory message is printed and the FUNCTION request reprinted. Type CTRL/Z (↑Z) to terminate REACT.

At this point, the system disk is built and the RSTS/E system is fully operational. It is suggested that the system manager shut down the system as described in Section 3.2 of the RSTS/E System Manager's Guide and restart it to test the new START.CTL file.

4.5 BUILDING THE AUXILIARY RUN TIME SYSTEM RTSLIB

The BUILD program must be run to create the auxiliary run time system RTSLIB and install the task builder (TKB). The task builder is used to link language processors other than BASIC-PLUS. RTSLIB is required for operation of both TKB and the language processors.

4.5.1 Running BUILD - AUXBLD.CTL

The control file AUXBLD.CTL must be used with the BUILD program. Table 4-3 describes the files created by executing the AUXBLD.CTL file.

Table 4-3
AUXBLD.CTL Programs and Files

Program or File Name (with account)	Protection (if other than <124>)	Job Size (in words)	Description
RTSLIB.RTS [0,1]	<60>	4K	Auxiliary run time system
TKB.TSK [1,2]	<232>	18K	Program which creates the auxiliary language and program modules.
TKB.BAC [1,2]	<232>	2K	Program which runs the module TKB.TSK.
LNGBLD.BAC [1,2]		4K	Program which builds the language and program modules

Before beginning the build procedures, obtain a hard copy of the memory allocation breakdown. (Use the LIST TABLE OPTION of either the DEFAULT or the START initialization option to generate the copy.) This memory breakdown is used to verify the optimum load address of the auxiliary run time system and to ensure that language and program modules have sufficient user memory space.

After the build procedures, the proper run time system commands of the UTILTY system program must be added to the START.CTL and CRASH.CTL files. Read the material in Section 7.1.4 of the RSTS/E System Manager's Guide concerning the run time system commands.

The auxiliary run time system can be built from either magtape, disk cartridge, or DECtape. It is assumed that the procedures described in Sections 4.2, 4.3, and 4.4 have been completed and that the user has not altered his system library account. It is also assumed that the terminal at which the build procedures are executed is logged into the system under the system library account [1,2]. The following sections describe the procedures for each device.

4.5.1.1 Magtape Procedures - Perform the following steps.

Ensure that the write enable ring is removed from the reel labelled SYSTEM LIBRARY AND RELIABILITY, DEC-11-ORSLA-D-MA7 (for 7-track drives) or DEC-11-ORSLA-D-MA9 (for 9-track drives).

Mount this tape on a free drive. Ensure that no other user is assigned the unit and that no other drive has the same unit number.

Position the tape at load point. (The LD PT indicator comes on.)

Ensure that the FILE PROT indicator is on. (If the FILE PROT indicator is off, dismount the reel and remove the write enable ring.)

Set the ON-LINE/OFF-LINE switch to its ON-LINE position. Ensure that the READY indicator comes on.

While logged into the system under account [1,2], run the BUILD program from the system disk as follows.

```
RUN BUILD
SYSTEM BUILDER
SYSTEM BUILD? NO
AUXILIARY BUILD DEVICE? MT0
CONTROL FILE IS? AUXBLD.CTL
```

Replace 0 in the build device designation with the unit number on which the tape is mounted. After the control file name AUXBLD.CTL$ is typed, BUILD runs and creates the necessary files as is shown in the following sample printout.

```
RUN BUILD
SYSTEM BUILDER
SYSTEM BUILD? NO
AUXILIARY BUILD DEVICE? MT0
CONTROL FILE IS? AUXBLD.CTL
^C
HELLO

RSTS V06A-02 SYSTEM TEST  JOB 2  KB0  28-MAY-75  12:20
# 1 / 2
PASSWORD:
JOB(S) 1 ARE DETACHED UNDER THIS ACCOUNT
JOB NUMBER TO ATTACH TO?
1 OTHER USER(S) ARE LOGGED IN UNDER THIS ACCOUNT
```

```
READY

! ********** AUXBLD.CTL - SET UP THE AUXILIARY RUN-TIME SYSTEM
OLD MT0:$TKB
READY
COMPILE SY0:TKB
READY
OLD MT0:$LNGBLD
READY
COMPILE SY0:LNGBLD
READY
RUN PIP
PIP V06A-15  RSTS V06A-02  SYSTEM TEST
#SY0:RTSLIB.RTS[0,1]<MT0:$RTSLIB.RTS/CO/CL:16
#SY0:TKB   .TSK     <MT0:$TKB   .TSK/CO/CL:16
#SY0:TKB   .BAC<232>/RE
#SY0:TKB   .TSK<232>/RE
#^C

READY


RUN MT0:$INSTAL
RUN-TIME SYSTEM NAME? RTSLIB
RTSLIB IS NOW TEMPORARILY INSTALLED AT 108K

READY
```

The RUN-TIME SYSTEM NAME question shown above is answered automatically by the BUILD program. The INSTAL program calculates a best-fit location to load RTSLIB. A message is printed giving the calculated load address. (The 108 in the sample is replaced by the 1K section number.) The dialogue continues as follows:

```
^C

READY

HELLO

RSTS V06A-02 SYSTEM TEST  JOB 2  [1,2]  KB0  28-MAY-75  12:26
JOB(S) 1 ARE DETACHED UNDER THIS ACCOUNT
JOB NUMBER TO ATTACH TO? 1
ATTACHING TO JOB 1

BUILD COMPLETE

READY
```

Execution of the commands in the AUXBLD.CTL file takes approximately 5 minutes. The end of the procedure is signalled by the messages BUILD COMPLETE and READY. Dismount the tape and store it in a safe place. Proceed to Section 4.5.2 to place the proper commands in the start up and crash control files.

4.5.1.2 Disk Cartridge Procedures - Perform the following steps.

Mount the copy of the disk cartridge labelled SYSTEM LIBRARY AND RELIABILITY DECPACK, order number DEC-11-ORSLA-D-HA, on a free unit.

Place the LOAD/RUN switch on the drive to its RUN position and ensure that the READY indicator comes on.

Ensure that the WR PROT indicator is off.

Use the identification label SYSLIB to logically mount the disk with UTILTY.

While logged into the system under account [1,2], run the BUILD program from the system disk as follows.

```
RUN $BUILD
SYSTEM BUILDER
SYSTEM BUILD? NO
AUXILIARY BUILD DEVICE? DK0
CONTROL FILE IS? AUXBLD.CTL$
```

Replace 0 in the build device designation with the unit number on which the disk is mounted. After the control file name AUXBLD.CTL$ is typed, BUILD runs and creates the necessary files as is shown in the sample printout of Section 4.5.1.1. The build procedure continues as shown in Section 4.5.1.1.

Execution of the commands in the AUXBLD.CTL file takes approximately 3 minutes. The end of the procedure is signalled by the messages BUILD COMPLETE and READY. Dismount the disk cartridge with the DISMOUNT command of UTILTY, remove the cartridge from the drive and store it in a safe place. Proceed to Section 4.5.2 to place the proper commands in the start up and crash control files.

4.5.1.3 DECtape Procedures - Perform the following steps.

Mount the DECtape reel labelled SPECIAL PROGRAMS, order number DEC-11-ORSCA-D-UA4, on a free drive. Ensure that no other drive is using the same unit number.

Set the WRITE ENABLE/WRITE LOCK switch on the drive to its WRITE LOCK position.

Set the REMOTE/OFF/LOCAL switch on the drive to its REMOTE position.

While logged into the system under account [1,2], run the BUILD program from the system disk as follows.

```
RUN $BUILD
SYSTEM BUILDER
SYSTEM BUILD? NO
AUXILIARY BUILD DEVICE? DT0
CONTROL FILE IS? AUXBLD.CTL$
```

Replace 0 in the build device designation with the unit number on which the tape is mounted. After the control file named AUXBLD.CTL$ is typed, BUILD runs and creates the necessary files as is shown in the sample printout of Section 4.5.1.1 The build procedure continues as shown in Section 4.5.1.1.

Execution of the commands in the AUXBLD.CTL file takes approximately 10 minutes. The end of the procedure is signalled by the messages BUILD COMPLETE and READY. Dismount the tape and store it in a safe place. Proceed to Section 4.5.2 to place the proper commands in the start up and crash control files.

4.5.2 Tailoring the Control Files for RTSLIB

After the BUILD program executes the commands in the AUXBLD.CTL file and terminates, the auxiliary run time system RTSLIB resides at a temporary location calculated by the INSTAL program.

NOTE

> The INSTAL program is provided on the tape to load the auxiliary run time system during the build procedure and must not be used under any other circumstances.

If the temporary location chosen by INSTAL is not suitable for operations at the local installation, a new location can be specified by the LOAD command of UTILTY. The following sample dialog shows the procedure.

```
RUN $UTILTY
SYSTEM UTILITY PROGRAM 'UTILTY V06A-03'
? LOAD RTSLIB/ADDR:xxx
? EXIT

READY
```

Replace xxx in the /ADDR: option with the appropriate 1K memory section number at which RTSLIB is to be loaded. The 1K sections are numbered starting at 0. After the load address has been fixed, place the following sequence of commands in the START.CTL and CRASH.CTL control files so that the auxiliary run time system can be used during time sharing.

```
FORCE KB1: RUN $UTILTY
FORCE KB1: ADD RTSLIB
FORCE KB1: LOAD RTSLIB/ADDR:xxx
FORCE KB1: UNLOAD RTSLIB
FORCE KB1: EXIT
```

The commands can be forced to keyboard unit 0. If unit 0 is used, however, ensure that the commands are executed before the FORCE KB0: RUN $ERRCPY command which detaches from the console terminal and leaves the console terminal logged out. Replace xxx in the /ADDR: option with the appropriate 1K memory section number at which RTSLIB is to be loaded. (See Section 3.1.2.4 of the RSTS/E System Manager's Guide for a sample control file. If the run time system commands are not executed at the start of time sharing, RTSLIB will not be in the monitor's table of run time systems. An attempt to run the language or the programs which depend on RTSLIB generates the NO RUN TIME SYSTEM error message.) If the load address is changed in the future (because memory is locked out or BASIC-PLUS is relocated), the commands in the control files must also be changed.

To ensure that the TKB program is properly installed on the system, type the following commands.

```
RUN $TKB
TKB>^Z

READY
```

The prompting message TKB> indicates that the program operates properly.

4.6 BUILDING THE COBOL COMPILER AND SORT11 PROGRAM

Before building COBOL, ensure that the following requirements are satisfied.

1. The auxiliary run time system RTSLIB must be on the system.
2. The default swap maximum must be at least as large as the size of the COBOL compiler (specified during the build procedure below).
3. The size of the COBOL compiler must not be greater than the available contiguous user space on the system.
4. The total of the COBOL compiler size and the RTSLIB size (fixed at 4K words) must not be greater than the available user space on the system.

Section 4.5 describes the build procedures for RTSLIB. The current default swap maximum can be found by executing the DEFAULT initialization option described in Section 3.5. The available user space can be found by using the LIST TABLE OPTION of either DEFAULT or START.

4.6.1 Building COBOL and SORT11

To build COBOL, the LNGBLD program is run from the system disk. The COBOL compiler and the SORT11 program are distributed as one package. The distribution medium is used as the build device. LNGBLD creates the files described in Table 4-4.

Table 4-4
LNGBLD Programs and Files

Program or File Name (with account)	Protection (if other than <124>)	Job Size (in words)	Description
nnnnnn.TSK [1,2]	<232>	?	Default is COBOL.TSK, the COBOL compiler. Otherwise, nnnnnn is a user-specified name.
nnnnnn.BAC [1,2]	<232>	4K	Default is COBOL.BAC, the program which loads the PDP-11 COBOL compiler COBOL.TSK. If other than COBOL.TSK is created, nnnnnn.BAC is created to load the module nnnnnn.TSK.
SORT11.TSK [1,2]	<232>	8K	The PDP-11 Sort program SORT11.

Table 4-4 (Cont.)
LNGBLD Programs and Files

Program or File Name (with account)	Protection (if other than <124>)	Job Size (in words)	Description
SORT11.BAC [1,2]	< 232>	4K	Program which loads the PDP-11 Sort program SORT11.
REFRMT.OBJ [1,2]	<104>	4K	Reads a program coded in the terminal format and converts it to 80-column conventional format.
COBRG.OBJ [1,2]	<104>	8K	Produces printed reports from data files.

The LNGBLD program creates an account on the system disk. Upon completion of the build procedure, the account is zeroed and deleted.

The LNGBLD procedures allow the size and name of the COBOL compiler to be altered. This facility enables systems to have both a large and small COBOL compiler at the same time. (LNGBLD creates only one compiler at a time.) The default compiler, COBOL.TSK, is run by means of the standard CCL commands COBOL and CBL. Other versions of the compiler are run by typing the RUN command with the dollar sign ($) and the user-specified compiler name. Thus, if a compiler named CBL28 is created, the command RUN $CBL28 loads the alternate COBOL compiler named CBL28.TSK from the system library account.

The following sections describe the procedure to build COBOL and SORT11 from each distribution medium.

4.6.1.1 Magtape Procedure - Perform the following steps.

Ensure that the write enable ring is removed from the reel labelled RSTS/E COBOL V01A, order number DEC-11-LCBSA-B-MB7 (for 7-track drives) or DEC-11-LCBSA-B-MB9 (for 9-track drives).

Mount this tape on a free unit and ensure that no other drive is set to that unit.

Ensure that the FILE PROT indicator is on.

Position the tape at load point. (The LD PT indicator comes on.)

Set the ON-LINE/OFF-LINE switch to its ON-LINE position. Ensure that the READY indicator comes on.

While logged into the system under account [1,2], run the LNGBLD program from the system disk as follows.

```
RUN LNGBLD
LNGBLD  V06A-05    BUILD AN AUXILIARY LANGUAGE
WHAT LANGUAGE TO TO BE BUILT? COBOL
FROM WHAT DEVICE(S) WILL THE FILES COME? MT0:
```

NOTE

If multiple devices are necessary, type each device designator separated by a comma.

After the device designator(s) is typed, the LNGBLD program prints the following messages.

```
ACCOUNT (1,6) WILL BE CREATED AND USED FOR THE BUILD
THIS ACCOUNT WILL BE DELETED AFTER THE BUILD
^C
```

NOTE

The CTRL/C combination may appear before the ACCOUNT WILL BE CREATED and WILL BE DELETED messages finish printing. This action does not affect the proper result of the program.

```
HELLO 1/6;A

READY

ASSIGN MT0:.DOS

READY

RUN PIP
PIP V06A-15  RSTS  V06A-02  SYSTEM TEST
#SY:=MT0:[128,128]/CO
#^Z

READY

DEASSIGN

READY
!********** BUILD THE COBOL LANGUAGE PROCESSOR

RUN CBLBLD
DEFAULTS FOR THE COBOL BUILD ARE:
        NAME            COBOL
        SIZE            18K WORDS
        SAVE MAP?       NO
ANY CHANGES?
```

At this point in the procedure, either YES or NO must be typed. If NO is typed, the program builds COBOL with the defaults listed in the printout. If YES is typed, the program prints a series of three questions which allow the defaults to be changed.

The first question concerns the size of the COBOL job image. Typing NO establishes the default size and causes the second question to be printed. Typing YES causes the program to print an additional question concerning the size. The size can be between 18K and 28K words provided that the available user job space is at least equal to the sum of the COBOL size and the size of RTSLIB. If enough user job space is not available and a user attempts to run COBOL, the system prints the MAXIMUM CORE EXCEEDED error message. A larger size for COBOL increases execution efficiency because the number of overlays is reduced. A smaller size requires less memory and swapping storage.

The second question allows the name of the COBOL compiler to be changed. Typing NO to the question establishes the default name. Typing YES causes the program to print an additional question in response to which a name between one and six characters can be typed. The name specified is used for the COBOL module and for the program which loads the COBOL module.

The third question concerns saving the load map file. Typing NO results in the map being deleted. Typing YES saves the map under account [1,2] in a file nnnnnn.MAP where nnnnnn is the one to six characters assigned to the COBOL module name. (If YES is typed, a message giving the file name and account is printed on the next line for documentation purposes.)

The dialogue continues as shown in the following sample printout.

```
ANY CHANGES? YES
        CHANGE SIZE? YES
                CHANGE IT TO? 24
        CHANGE NAME? NO
        SAVE MAP? YES
        THE MAP WILL BE SAVED AS COBOL.MAP ON ACCOUNT [1,2].

READY

RUN $TKB

TKB>@CBRSTS
TKB -- *DIAG*-22 P0A

TKB -- *DIAG*-22 WFL0

TKB -- *DIAG*-22 MSGR

TKB -- *DIAG*-22 LOAD

TKB -- *DIAG*-22 RFCSGB

TKB>@SRTRST
TKB -- *DIAG*-21 SORTP  FA.ENB

TKB -- *DIAG*-21 SORTP  FA.POS

TKB -- *DIAG*-21 SORTP  FA.RWD

TKB -- *DIAG*-21 SORTP  FA.ENB

TKB -- *DIAG*-21 SORTM  FA.POS

TKB -- *DIAG*-21 SORTM  FA.RWD

TKB -- *DIAG*-22 SORTS1

TKB>^Z
```

```
READY
OLD COBOL
READY
10      DATA    COBOL,$COBOL.TSK,RTSLIB,-1
COMPILE SY0:$COBOL
READY
OLD SORT11
READY
COMPILE SY0:$SORT11
READY
NAME "$COBOL.BAC" AS "$COBOL.BAC<232>"
READY
NAME "$COBOL.TSK" AS "$COBOL.TSK<232>"
READY
NAME "$SORT11.TSK" AS "$SORT11.TSK<232>"
READY
NAME "$SORT11.BAC" AS "$SORT11.BAC<232>"
READY

!****RUN A COBOL TEST PROGRAM ****
!
RUN $COBOL

CBL>XTST01=XTST01
CBL -- 4    ERRORS
CBL -- LOADING
COMAPARE THE FOLLOWING TWO ROWS:
A     ABCD A
A     ABCD A

COMPARE THE FOLLOWING TWO ROWS:
AAE1212221B1B1K01201202201B01B01K
AAE1212221B1B1K01201202201B01B01K

BUILD COMPLETE - ZEROING ACCOUNT (1,6) - DELETING ACCOUNT (1,6)
READY
```

Completion of the build is signalled by the BUILD COMPLETE and READY messages. The procedure takes between 10 and 15 minutes. Proceed to Section 4.6.2 for guidelines on testing the results of the build procedures for COBOL and SORT11.

4.6.1.2 Disk Cartridge Procedure - Perform the following steps.

Mount the disk cartridge labelled RSTS/E COBOL V01A, order number DEC-11-LCBSA-B-HB, on a free drive.

Place the LOAD/RUN switch on the drive to its RUN position and ensure that the READY light comes on.

Ensure that the WR PROT indicator is on.

While logged into the system under account [1,2], run the LNGBLD program from the system disk as follows.

```
RUN $LNGBLD
LNGBLD V06A-05 BUILD AN AUXILIARY LANGUAGE
WHAT LANGUAGE IS TO BE BUILT? COBOL
FROM WHAT DEVICE(S) WILL THE FILES COME? DK0:
```

After the device designator is typed, LNGBLD prints messages similar to those in Section 4.6.1.1. Follow the guidelines given in Section 4.6.1.1 to answer the ANY CHANGES question.

Completion of the build is signalled by the BUILD COMPLETE and READY messages. The procedure takes approximately ten minutes. Proceed to Section 4.6.2 for guidelines on testing the results of the build procedures.

4.6.1.3 DECtape Procedure - Perform the following steps.

Mount the DECtape reels labelled RSTS/E COBOL V01A, order numbers DEC-11-LCBSA-B-UB1 and -UB2, on free drives.

Set the WRITE ENABLE/WRITE LOCK switches on the drives to WRITE LOCK.

Set the REMOTE/OFF/LOCAL switches on the drives to REMOTE.

While logged into the system under account [1,2], run the LNGBLD program from the system disk as follows.

```
RUN $LNGBLD
LNGBLD    V06A-05 BUILD AN AUXILIARY LANGUAGE
WHAT LANGUAGE IS TO BE BUILT? COBOL
FROM WHAT DEVICE(S) WILL THE FILES COME? DT0:,DT1:
```

Building System Library Files

After the device designator is typed, LNGBLD prints messages similar to those in Section 4.6.1.1. Follow the guidelines given in Section 4.6.1.1 to answer the ANY CHANGES query.

Completion of the build is signalled by the BUILD COMPLETE and READY messages. The procedure takes approximately 20 minutes. Proceed to Section 4.6.2 for guidelines on testing the results of the build procedures.

4.6.2 Testing the Results of the Build Procedures

It is advisable to test the start up control file used at the local installation. Run the SHUTUP program and, when SHUTUP terminates, press the CPU console CONT switch to bootstrap the system disk. Use the START option and, when the initialization program runs, specify the standard control file which contains the run time system control commands. After start up, the system should be able to run COBOL and SORT11. The results of the start up can be tested by typing the proper command to run COBOL as shown below.

```
RUN $COBOL

CBL>
```

RSTS/E loads the auxiliary run time system and runs the COBOL compiler which prints its prompting indicator. If the NO RUN TIME SYSTEM message occurs, check the control file to ensure that the proper commands are present and try the procedure again. (Refer to Section 4.5.2 for the proper run time system commands.) Type the CTRL/Z combination to terminate COBOL and type the command to run SORT11.

```
CBL>^Z

READY

RUN $SORT11

SORT>^Z

READY
```

For more information on running COBOL and SORT11, see the PDP-11 COBOL User's Guide and the PDP-11 SORT User's Guide.

APPENDIX A

HARDWARE BOOTSTRAP PROCEDURES

Bootstrapping a device involves using the central processor unit (CPU) console switches to access and initiate a hardware loader. The hardware loader contains machine instructions for reading a special record from the device. The record, called a bootstrap record, is transferred into memory and executes a specially designed software program. For the bootstrap operation to succeed, the device accessed must be on line and ready; the medium accessed must contain a proper bootstrap record; and the console terminal must be on line.

The console switches and their usage are described in Chapter 11 of the *PDP-11/70 Processor Handbook* and in Chapter 8 of the *PDP-11/45 Processor Handbook* and the *PDP-11/40 Processor Handbook*. The bootstrap procedure to use depends upon the type of hardware bootstrap device on the system. Table A-1 summarizes the addresses needed to bootstrap a device. The detailed procedures to bootstrap a device are presented according to the types of hardware bootstrap devices available.

Table A-1
Summary of Hardware Bootstrap Addresses

Device to Bootstrap	Bootstrap Type			
	BM873-YA	BM873-YB[1]	MR11-DB	BM792-YB[2]
RF11 disk	773000	773136	773100	777462
RK11 disk cartridge unit 0	773010	773030	773110	777406
RP02 or RP03 disk pack unit 0	773100	773350	773154	776716
RP04 disk pack unit 0	-	773320	-	-
RK11 disk (unit specified in SR)	-	773032	-	-
RP02 or RP03 disk (unit specified in SR)	-	773352	-	-
RP04 disk (unit specified in SR)	-	773322	-	-
TM11/TU10 magtape	773050	773110	773136	See footnote 3
TM02/TU16 magtape	-	773150	-	-
TC11/TU56 DECtape	773030	773070	773120	777344

---

[1] To bootstrap a non-zero disk unit, set the address in the Switch Register, depress the LOAD ADRS switch, set the unit number in the Switch Register, and press the START switch.

[2] For the BM792-YB loader, set the address 773100 in the Switch Register, depress the LOAD ADRS switch, set the value from the table in the Switch Register, and press the START switch.

[3] To bootstrap a magtape, use the loading routine described in Section A.4.

Hardware Bootstrap Procedures

A.1 BM873-YA PROCEDURE

If the BM873-YA Restart/Loader is on the system, perform the following steps.

Move the CPU Console ENABLE/HALT switch to its HALT position and back to its ENABLE position.

Set the CPU Switch Register to one of the following values.

773000 for RF11 disk
773010 for RK11 disk cartridge
773100 for RP02 or RP03 disk pack
773050 for TM11/TU10 magtape
773030 for TC11/TU56 DECtape

Depress the CPU LOAD ADRS switch.

Depress the CPU START switch.

A.2 BM873-YB PROCEDURE

If the BM873-YB Restart/Loader is on the system, perform the following steps.

Move the CPU Console ENABLE/HALT switch to its HALT position and back to its ENABLE position.

Set the CPU Switch Register to one of the following values.

773030	for RK11 disk cartridge
773136	for RF11 disk
773320	for RP04 disk pack
773032	for RK11 disk unit specified in the Switch Register
773322	for RP04 disk unit specified in the Switch Register
773352	for RP02 or RP03 disk unit specified in the Switch Register
773350	for RP02 or RP03 disk pack
773110	for TM11/TU10 magtape
773150	for TM02/TU16 magtape
773070	for TC11/TU56 DECtape

Depress the CPU LOAD ADRS switch.

If necessary, set the CPU Switch Register to the unit number of the disk drive being bootstrapped.

Depress the CPU START switch.

A.3 MR11-DB PROCEDURE

If the MR11-DB Bulk Storage Loader is on the system, perform the following steps.

Move the CPU Console ENABLE/HALT switch to its HALT position and back to its ENABLE position.

Set the CPU Switch Register to one of the following values.

773100 for RF11 disk
773110 for RK11 disk cartridge
773154 for RP03 disk pack
773136 for TM11/TU10 magtape
773120 for TC11/TU56 DECtape

Depress the CPU LOAD ADRS switch.

Depress the CPU START switch.

A.4 BM792-YB PROCEDURE

If the BM792-YB Hardware Loader is on the system, perform the following steps.

Move the CPU Console ENABLE/HALT switch to its HALT position and back to its ENABLE position.

Set the CPU Switch Register to 773100.

Depress the CPU LOAD ADRS switch.

Set the CPU Switch Register to one of the following values.

777462	for RF11 disk
777406	for RK11 disk cartridge
776716	for RP03 disk pack
777344	for TC11/TU56 DECtape

Depress the CPU START switch.

To bootstrap a TM11/TU10 magtape when the system has neither the BM873 nor the MR11-DB loader, the user must manually enter a load routine into memory using the CPU console Switch Register and the DEP switch.

To load the routine, perform the following steps.

Move the CPU Console ENABLE/HALT switch to its HALT position and back to its ENABLE position.

Set the CPU Switch Register to 010000.

Depress the CPU LOAD ADRS switch.

Load the following contents into memory using the Switch Register and DEP switch.

Hardware Bootstrap Procedures

Address	Contents
010000	012700
010002	172524
010004	005310
010006	012740
010010	060011
010012	105710
010014	100376
010016	005710
010020	100767
010022	012710
010024	060003
010026	105710
010030	100376
010032	005710
010034	100777
010036	005007

Set the Console Switch Register to 010000.

Depress the CPU LOAD ADRS switch.

Depress the CPU START switch.

If the system reads the tape but halts at address 010034, the magtape generated a parity error. The user can try another drive. If the system appears to take no action and halts, verify the accuracy of the routine by using the CPU Console EXAM switch. Use the Switch Register and the DEP switch to correct any erroneous contents. Rewind the tape to its load point before executing the routine again. If no recovery is successful, it will be necessary to have a DIGITAL Field Service representative check the device. If the hardware is working properly, it will be necessary to use a new magtape reel.

APPENDIX B

RSTS/E CONSISTENCY ERROR MESSAGES

During the execution of RSTS/E initialization routines, many checks are made to determine the consistency of system structures. The existing structures are compared to their definitions and references as they appear in other parts of the system. The checks must always be successful. If they are not successful, a consistency error has been detected and the system is corrupted.

The initialization code is executed in two phases: first, the CILUS phase after the system disk is bootstrapped and before the OPTION: message is printed; and, second, the option phase during which any of the initialization options can be executed.

B.1 CILUS PHASE ERRORS

If an error occurs during the CILUS phase, the initialization code prints a descriptive message and the following text.

```
FATAL RSTS SYSTEM INITIALIZATION ERROR!

THE FATAL ERROR OCCURRED DURING THE CILUS PHASE
OF SYSTEM INITIALIZATION; THERE IS NO RECOVERY.
```

The processor is halted so that, if the CPU console CONT switch is pressed, the code bootstraps the system disk again. Bootstrapping the disk, however, is usually not a worthwhile procedure because most of the errors reflect a serious problem rather than a transient hardware error. Reloading of the system disk (see Section 2.9 of the RSTS/E System Generation Manual) is recommended in some cases but regenerating the system is required in other cases.

RSTS/E Consistency Error Messages

The list below gives the descriptive messages followed by a 1-word advice (in parentheses) indicating the recovery procedure. The following legend briefly summarizes the recovery procedures.

(reboot)	Bootstrap again. If unsuccessful, reload the system disk.
(reload)	Reload the system disk (see Section 2.9). If reloading does not eliminate the error condition, regenerate the system.
(regen)	Regenerate the system and load the new CIL. (See Section 2.1.)
(hard)	A hardware adjustment or addition is required.
(spr)	File a Software Performance Report. There is no recovery.

The following are the descriptive messages.

CHECKSUM ERROR IN CIL INDEX.	(reload)
CIL LINE IS MISSING FROM CIL INDEX.	(reload)
COMD LINE IN CIL INDEX IS IN INCORRECT FORMAT.	(reload)
DEVICE BOOTED DOES NOT MATCH SYSTEM DEVICE.	(regen)
DEVICE BOOTED IS NOT A LEGAL RSTS SYSTEM DEVICE.	(regen)
DEVICE ERROR WHEN READING CIL INDEX.	(reboot)
DOUBLE OCCURRENCE OF SOME SYSTEM IMAGE.	(regen)
FORMAT ERROR IN CIL INDEX.	(reload)
ILLEGAL BLOCK SIZE IN CILUS BOOTSTRAP PARAMETERS.	(regen)
ILLEGAL SWAPPING DISK DETECTED.	(regen)
INIT ERROR - SECTION I.BOOT TOO SMALL.	(spr)
INIT WAS INCORRECTLY ASSEMBLED OR LINKED.	(spr)
*** MONITOR IS TOO BIG ***	(regen)
RSTS/E REQUIRES EIS TO RUN!	(hard)
RSTS/E REQUIRES MEMORY MANAGEMENT HARDWARE!	(hard)
TOO MANY NON-SYSTEM IMAGES IN CIL.	(regen)

If the initialization code finds that certain modules are missing, it prints one of the two sets of messages.

SYSTEM MODULE xxx IS MISSING FROM CIL. ONE OR MORE CRITICAL MODULES MISSING FROM CIL.	(regen)
RUN-TIME SYSTEM MISSING FROM CIL. ONE OR MORE MODULES MISSING FROM CIL.	(regen)

The module name replaces xxx in the message.

RSTS/E Consistency Error Messages

Under certain conditions, the initialization code prints a warning message followed by the OPTION: message. The user can operate within the restrictions described in the warning. The following are the warning messages possible.

```
WARNING ** THIS PDP-11/70 WILL RUN IN 11/45 MODE
DUE TO THE NUMBER OF UNIBUS NPR DEVICES.

WARNING ** THIS MACHINE DOES NOT HAVE THE CONFIGURED CLOCK!
RSTS/E WILL CRASH IF YOU ATTEMPT TO START TIME-SHARING.

WARNING ** THIS MACHINE DOES NOT HAVE A STACK LIMIT REGISTER!
ALTHOUGH NOT ADVISED, RSTS/E WILL RUN WITHOUT THE OPTION.

WARNING ** THIS RSTS SYSTEM, WHICH WAS NOT BUILD FOR AN 11/70,
WILL CRASH IF PARITY ERRORS OCCUR!

WARNING ** THIS RSTS SYSTEM, WHICH WAS BUILT FOR AN 11/70,
WILL CRASH IF PARITY ERRORS OCCUR!

WARNING ** DBx: IS DUAL PORTED.  PROCEED WITH CAUTION!
```

If the warning message concerning the dual ported RP04 disk is printed, ensure that the Controller Select Switch on the related drive is in the correct position.

B.2 OPTION PHASE ERRORS

If an error occurs during the option phase, the initialization code prints a descriptive message and halts. The following are the messages.

```
ATTEMPT TO ASK FOR OPTION WHEN CILUS PHASE NOT DONE.
BAD DIRECTORY DETECTED DURING CLEAN.
EXISTING SYSTEM FILE EMPTY OR NON-CONTIGUOUS.
FILE [0,1]BADB.SYS MISSING FROM SYSTEM DISK.
INIT BUG - ATTEMPT TO DELETE NONEXISTENT FILE.
INIT BUG - FAILED TO CREATE RSTS.CIL ON 2ND TRY.
INIT BUG - FILE EXISTED WHEN TRYING TO CREATE.
INIT BUG - INSUFFICIENT DIRECTORY SPACE FOR CREATE.
INIT BUG - INSUFFICIENT DISK SPACE FOR CREATE.
INIT BUG - SATT.SYS NONEXISTENT AT TIME OF WOMP.
INIT BUG - UNABLE TO REBUILD DISK.
INSUFFICIENT DIRECTORY SPACE FOR SYSTEM FILES.
INSUFFICIENT DISK SPACE FOR [0,1] DIRECTORY.
PACK CLUSTER SIZE IS NOT 1, 2, 4, 8 OR 16.
REQUIRED FILE BADB.SYS DOES NOT EXIST.
```

REQUIRED FILE SATT.SYS FILE DOES NOT EXIST.
REQUIRED LIBRARY ACCOUNT [1,2] DOESN'T EXIST.
RSTS CIL IS NOT ON A CLUSTER BOUNDARY.
SYSTEM DISK SAT SIZE NOT EQUAL TO COMPUTED SIZE.
SYSTEM FILE CONTAINS BAD BLOCKS - CANNOT REFRESH.

APPENDIX C

TABULATION OF SHORT FORM CONFIGURATION QUESTIONS

The user initiates the short form of the dialogue by typing S in response to the first question printed by the SYSGEN program. The following list shows the questions numbered in hierarchical order, the possible answers, and comments on SYSGEN actions.

Question Number	Question	Possible Answers	Comment
1	MEDIA?	MT,DT,RK,MM,RP	
2.0	PDP-11/70?	Y,N	If N, SYSGEN skips to question 2.2.
2.1	FPP (11/70)?	Y,N	SYSGEN skips to question 2.4.
2.2	FPP (11/45)?	Y,N	If Y, SYSGEN skips to question 2.4.
2.3	FIS (11/40)?	Y,N	
2.4	MAX MEMORY SIZE (K WORDS)?	32 to 1024	
3	CLOCK?	L,P,C	SYSGEN prints 4.1 if answer is L or P or prints 4.2 if answer is C.
4.1	AC FREQ?	50,60	
4.2	KW11P INTERRUPT RATE?	50,100,...1000	
5	CONSOLE TYPE?	ASR33,KSR33, ASR35,KSR35, LA30P,LA30S, VT05,VT05B, VT50,VT51, LA36	
6	KL11,LC11,DL11A, DL11B'S?	0 to 15	Do not include the console terminal.
7	DL11C,DL11D'S?	0 to 31	Do not include the console terminal.
8	DC11'S?	0 to 32	
9	DL11E'S?	0 to 31	If the sum of the answers to this question and question number 7 is

Tabulation of Short Form Configuration Questions

Question Number	Question	Possible Answers	Comment
			less than or equal to 31, SYSGEN proceeds to 10. Otherwise, SYSGEN repeats 6 through 9.
10	DJ11'S?	0 to 16	
11	DH11'S?	0 to 16	If answer is greater than 0, SYSGEN proceeds to 11.1. Otherwise, it goes to question 12.
11.1	DH11 UNIT xx TYPE?	AA,AB,AC,AD,AE	If answer is AA or AC, SYSGEN prints 11.1.1 to determine presence of DM11-BB on this DH11 unit. If answer is AB,AD, or AE, SYSGEN prints 11.2. SYSGEN repeats 11.1 for each unit.
11.1.1	DOES DH11 xx INCLUDE A DM11-BB?	Y,N	
11.2	DH11 UNIT xx LINES ENABLED?	0 to 16	SYSGEN repeats 11.2 for each unit.
12	PSEUDO KEYBOARDS?	0 to 63	If the sum of answers to 6, 7, 8, 9, and 11.2 is less than or equal to 63, then SYSGEN prints question 13. Otherwise, it repeats from 6.
13	2741'S?	NO,SL,DH or BO	No support; support single line only; support only a DH11 line; support both a single and a DH11 line.
13.1	2741 CODE(S)?	CORR,EBCD,SBCD C360	Any combination of 1 to 4 separated by commas.
14	MULTI-TTY SERVICE?	Y,N	
15	RS03'S?	0 to 4	
16	RS04'S?	0 to 4	SYSGEN prints 17 only if answers to 15 and 16 are 0. If the sum of the answers to 15 and 16 is greater than 4, SYSGEN repeats 15.

<u>Tabulation of Short Form Configuration Questions</u>

Question Number	Question	Possible Answers	Comment
17	RF/RS11'S?	0 to 8	If 0, SYSGEN prints 17.1. Otherwise, SYSGEN prints 18.
17.1	RC11/RS64'S?	0 to 4	
18	RP04'S?	0 to 8 n/NO	If answer is 2 or greater, appending /NO substitutes the non-overlapped seek driver. SYSGEN prints 19 only if answer is 0.
19	RP02/RP03'S?	0 to 8 n/NO	If answer is 2 or greater, appending /NO substitutes the non-overlapped seek driver.
19.1	RK03/RK05'S?	0 to 8 n/NO	If answer is 2 or greater, appending /NO substitutes the non-overlapped seek driver.
20	SYSTEM DISK?	RF,RK,RP,RB	
21	RX01'S?	0,2,4,6,8	
22	TU16'S?	0 to 8	If answer is 0, question 23 is printed. Otherwise, question 24 is printed.
23	TU10'S?	0 to 8	
24	DECTAPES?	0 to 8	
25	PRINTERS?	0 to 8	
25.1.1	LPn: TYPE?	LP,LS	Each line printer unit must be defined; SYSGEN repeats these questions for each line printer unit.
25.1.2	LPn: WIDTH?	80,132	
25.1.3	LPn: LOWER CASE?	Y,N	
26	LP0 FOR SYSGEN?	Y,N	If Y, SYSGEN prints 26.1. Otherwise, SYSGEN proceeds to 27.
26.1	LISTINGS?	Y,N	
27	CARD READER?	N,CR,CD,CM	If N, SYSGEN omits 27.1.
27.1	CARD DECODE?	029,026,1401	
28	P.T. READER?	Y,N	

Tabulation of Short Form Configuration Questions

Question Number	Question	Possible Answers	Comment
29	P.T. PUNCH?	Y,N	
30	DQ11'S?	0 to 16	
31	DP11'S?	0 to 32	
32	DU11'S?	0 to 32	SYSGEN omits 32.1 if answers to 31 and 32 are 0.
32.1	2780?	Y,N	If N, SYSGEN proceeds to question 33. If Y and the sum of the answers to 31 and 32 is 1, SYSGEN proceeds to question 33.
32.1.1	2780 INTERFACE?	DP,DU	Printed only if answer to 32.1 is Y and answers to both 31 and 32 are non-zero.
33	NON-SUPPORTED DEVICES?	Y,N	If Y, then SYSGEN prints 33.1 through 33.10. Otherwise, it prints 34.
33.1	DM11-A'S?	0 to 16	Single speed multiplexer.
33.2	DN11-DA'S?	0 to 64	Automatic calling unit interface.
33.3	DR11-A,C'S?	0 to 32	General device interface.
33.4	PA611R'S?	0 to 16	Typeset reader.
33.5	PA611P'S?	0 to 16	Typeset punch.
33.6	DT03-FP'S?	0 to 8	Programmable UNIBUS switch.
33.7	DX11'S?	0 to 4	IBM 360/370 interface.
33.8	GT40 (1 ONLY)?	0,1	Answer 1 only if GT40 does not have its own CPU and is not connected through a standard terminal interface.
33.9	LPS (1 ONLY)?	0,1	LAB peripheral system.
33.10	KW11W (1 ONLY)?	0,1	Watchdog timer.
34	THE INSTALLATION NAME:		14 characters or less.

Tabulation of Short Form Configuration Questions

Question Number	Question	Possible Answers	Comment
35	MAXIMUM JOBS?	1 to 63	
36	SMALL BUFFERS?	30 to 999	10 times the maximum number of jobs recommended.
37	BIG BUFFERS?	1 to 8	SYSGEN prints this question only if the answer to question 24 is greater than 0.
38	RECEIVERS?	0 to 63	From 4 to 8 is usually sufficient.
39	POWER FAIL?	Y,N	
40	FIP BUFFERING?	Y,N	
41	RESIDENT DISK HANDLING?	Y,N	
42	RESIDENT SYS CALL DISPATCH?	Y,N	
43	RESIDENT SEND/RECEIVE?	Y,N	
44	RESIDENT DIRECTORY LISTER?	Y,N	
45	CCL?	Y,N	If N, SYSGEN proceeds at 46.
45.1	STANDARD CCL TABLE?	Y,NEW,ADD	If NEW or ADD, SYSGEN prints 45.1. Otherwise, it prints 46.
45.1.1	<PROGRAM>,<COMMAND>?		User types program name and command separated by a comma. SYSGEN prints question up to 20 times or until user types /E to end or types /R to restart.
45.2	CCL LISTING ON LP0:?	Y,N	Printed only if reply to question 26 was Y and to question 45.1 was NEW or ADD.
46	MATH PRECISION?	2,4	
47	FUNCTIONS?	Y,N	
48	TIME FORMAT?	24,AM	
49	PRINT USING?	Y,N	
50	MATRICES?	Y,N	
51	ROLLIN?	Y,N	

APPENDIX D

SYSTEM MODULE SIZES

The following tabulation gives the approximate memory sizes of software modules on RSTS/E. By summing the values given, the total size of the system can be estimated.

Module[1]	Decimal Words	Comments
Monitor Code (Scheduler, FIP, ERRLOG, and other modules)	6000	All systems
Tables	-	Varies greatly
Monitor 11/45 FPP Support	37	
Monitor Options		
FIP Buffering	408	Optional
Resident Disk Handling	1388	Optional
Resident SEND/RECEIVE	207	Optional
Resident SYS Call Dispatch	226	Optional
Resident Directory Lister	242	Optional
Small Buffers	16 per buffer	Ten buffers per job recommended
Big Buffers	256 per buffer	For DECtape support
Receivers	5 per receiver	Optional
Power Fail	32	Optional
2780 Package	5600	RSTS/2780 systems only
Terminal Service		
Common	1340	All systems
KL11, LC11, DL11A, DL11B	0	16 words per keyboard unit
DL11C, DL11D	20	16 words per keyboard unit
DL11E	20	16 words per keyboard unit
DH11	300	16 words per line enabled at system generation time
Common Modem Control	150	First DM11-BB, DC11, DL11E
DL11E, DC11 Modem Control	20	
DH11 Modem Control	150	

---

[1] Certain entries share code with or require other modules. Therefore, the user must add those values to attain the total size.

System Module Sizes

Module	Decimal Words	Comments
Pseudo Keyboards	118	Optional. Add 32 words per PK unit configured.
2741 Support		Optional
Common	220	Any 2741 support
Single line	100	Single Line 2741 support
DH11	100	DH11 2741 support
Code table	128 per table	Add 35 words if more than one table; four tables maximum
Multiple Terminal Service	100	Optional
Disks		
RS Disk (RS03/RS04)	112	
RF Disk	71	
RC Disk	62	
RB Disk (non-overlapped)	202	RP04
RB Disk (overlapped)	312	
RP Disk (non-overlapped)	111	RP02 or RP03
RP Disk (overlapped)	213	
RK Disk (non-overlapped)	82	
RK Disk (overlapped)	278	
Disk Common	14	All systems
Disk Optimization	43	If any RK, RP, or RB disk
Tape		
Magtape - TU16	625	Add 16 words per drive
Magtape - TU10	640	Add 16 words per drive
DECtape	405	Add 16 words per drive
Other Peripherals		
Line Printer	397	Add 16 words per unit of any printer type
Card Reader - CD11	265	Add 16 words for DDB
Card Reader - CR11	238	Add 16 words for DDB
Card Reader - CM11	238	Add 16 words for DDB
Card Decode Table	47	026, 029, or 1401 codes
Card Reader Buffer	82	Any card reader
Paper Tape Reader	81	Add 16 words for DDB
Paper Tape Punch	54	Add 16 words for DDB
BASIC-PLUS		
Compiler and RTS	11000	All systems
CCL Common	46	Optional
Standard CCL Table	69	Optional. Add 10 words per additional CCL command

System Module Sizes

Module	Decimal Words	Comments
BASIC-PLUS (cont.)		
Math Package		
MA2	2460	See Section 2.7.25 for the description of math packages.
MA2X	1989	
MA2I	2126	
MA2IX	1655	
MA2F	1930	
MA2FX	1586	
MA4	3590	
MA4X	2844	
MA4F	2460	
MA4FX	1990	
Print Using	727	Optional
Matrices	754	Optional
Record I/O	297	All Systems

APPENDIX E
SYSLOD ERROR MESSAGES

The system loader program SYSLOD transfers a linked core image library from one device to a disk as a contiguous core image library. The user runs SYSLOD from magtape or DECtape during the system generation procedure. The user must consult this section for error messages and possible recovery procedure if errors are generated during SYSLOD execution.

Error messages issued by the SYSLOD program can be of two types: recoverable errors and non-recoverable errors.

E.1 RECOVERABLE ERRORS

The following errors are diagnosed and printed by SYSLOD. Once the error message is printed, SYSLOD restarts by identifying itself again, and printing the # (input request) character at the keyboard. The user should retry the most recent command, making the indicated corrections. Error messages for recoverable errors are preceded by one of the following:

```
CIL dev
LICIL dev          (dev represents the device mnemonic)
```

depending upon whether an error has been detected in the CIL (output) or LICIL (input) side of the most recent command string.

SYNTAX ERROR

This message is printed if the command input line contains a syntax error.

SYSLOD Error Messages

TOO MANY SWITCHES

SYSLOD does not accept switches on the input side of the command string. If too many switches are specified on the output side of the command string for SYSLOD to handle, it issues this message.

UNKNOWN SWITCH

If SYSLOD does not recognize the switch as a valid switch, it prints this message.

SWITCH ERROR

If a switch is used incorrectly, SYSLOD prints this message. Incorrect use of a switch implies specification of an argument when no argument is valid or the lack of an argument when one is required.

SWITCH CONTEXT ERROR

This message is issued when switches are specified incorrectly for their definitions.

ERROR IN SWITCH ARGUMENT

This message is issued when decimal argument for any switch is too large to be contained in 16 bits.

NONEXISTENT DISK OR DISK NOT READY

Either (1) the disk specified in a command string does not exist in the configuration, or (2) the disk exists, but is not ready.

UNKNOWN DISK NAME SPECIFIED

This message is issued when a disk name other than those listed below is specified in a command string.

DF (DF0 through DF7)
DK (DK0 through DK7)
DP (DP0 through DP7)

ERROR WHILE FORMATTING RK DISK

This message is issued whenever an error is detected while formatting an RK disk unit.

A xxx
READY

This message is issued when a problem exists with a peripheral device not in the "ready" state. Check the device and the operating instructions to determine the cause of the error and correct the error. When the problem has been rectified, type YES and the CARRIAGE RETURN key to continue.

E.2 NON-RECOVERABLE ERRORS

The SYSLOD program prints an error message at the keyboard device when a non-recoverable error is encountered during processing. These messages are listed below, along with the action to be taken (if any).

INPUT IS NOT A LICIL

The first line of the input file is not in correct format for a LICIL. The most probable cause of this error is an attempt to transfer a load module (filnam.LDA) instead of the LICIL (filnam.LCL).

END OF FILE BEFORE CIL LINE READ

This error message is issued when SYSLOD reaches end-of-file before detecting the CIL line. If the CIL is being loaded from DECtape, magtape, or disk, it is likely that part of the file has been destroyed and must be rebuilt.

BOOTSTRAP NOT IN BLOCK 0

This message occurs in Replace mode only (neither the NS nor ZE switch has been specified). In Replace mode, SYSLOD searches bootstrap parameters to find the CIL to be replaced. If the first block number of the CIL hooked to the bootstrap (location 176 of BOOT) is 0, then block 0 is not a hooked bootstrap, and this message is issued.

BOOTSTRAP NOT HOOKED TO CIL; CANNOT REPLACE

This message occurs in Replace mode only; (neither the NS nor ZE switch has been specified). If the first block indicator of the bootstrap (location 176 of BOOT) is non-zero, it must be pointing to a CIL. If the first formatted binary line of the "hooked" file is not COMD section #3, then the file is not a CIL.

BLOCK SIZE DISCREPANCY BETWEEN CILUS AND SYSLOD

This error message is issued when the NS switch is used with SYSLOD. Exactly the same parameters must be specified for SYSLOD as were specified for CILUS.

LICIL TOO BIG, NOT ENOUGH RESERVED BLOCKS

This message occurs for either of two conditions:

a) In Replace mode, the new LICIL is larger that the old.
b) In any other mode, the number of reserved blocks (BL:nnnn) is not large enough for the CIL.

1ST LINE NOT COMD SECTION #4 OR 1

After reading the CIL line, SYSLOD begins to load the LICIL. The first formatted binary line after the CIL line must be COMD section #4 or COMD section #1. After each core image is loaded, SYSLOD is set to load a new core image. If the beginning of the new core image is not COMD section #4 or #1, this error message is issued.

COMD SECTION #4 SEQUENCE ERROR

This error message occurs if the LICIL is being loaded from paper tapes, and one or more tapes are out of order.

SYSLOD Error Messages

INPUT ERROR

After a READ, the status in the buffer header has indicated that one of the following errors occurred:

a) invalid line error
b) checksum error
c) character parity error or illegal binary format
d) device parity error

LOGICAL BLOCK SIZE ERROR

This error message is issued when the logical block size specified for the NS switch is not an integral multiple of the physical block size for the disk.

END OF DISK BEFORE CIL COMPLETE

The last block number of the output disk has been written, but the CIL is not complete.

ILLEGAL EMT CALL

An EMT call was made that was not recognizable as a valid DOS/BATCH EMT. Notify your Software Support representative.

FATAL ERROR RETRY

A fatal error has aborted the current operation. It must be retried.

MT DISASTER NXM OR ILC - IRRECOVERABLE

An irrecoverable I/O error has occurred while reading or writing magtape.

IRRECOVERABLE MT PROBLEM

A persistent error has occurred while reading magtape.

LICIL FILE NOT FOUND

The specified LICIL could not be found under UIC [1,1] or UIC [200,200].

NO SPACE FOR CIL

Not enough contiguous disk space is available to create the CIL. Either (1) too many files already exist on the disk, or (2) there are too many bad blocks on the disk.

BLOCK 0 OR BLOCK 1 BAD

Either block 0 or block 1 on the disk is bad; the system cannot be generated, as both these blocks are essential to the disk file structure.

PACK IS TOO BAD

The current disk pack cannot be used, as there are too many bad blocks (BADB.SYS is full).

ESSENTIAL DISK BLOCK HAD I/O ERROR

A MFD, UFD, or bit map block could not be written without encountering I/O errors. Note that these blocks are written after the verification phase.

E.3 NOTES

If the default block size is not used, the block size must be an integral multiple of the default size for the disk.

The COMD (COMmunication Directory) contains a code number that identifies the kind of information that follows. SYSLOD expects the first formatted binary line it receives on input to be identified as code #4 (indicating that it is a LICIL).

On occasion, certain hardware problems such as magtape and RK disk head alignment, UNIBUS time out traps, and disk controller errors cause SYSLOD to halt. The system manager should immediately contact a DIGITAL field service representative.

APPENDIX F

SPECIAL SYSGEN OPTIONS

During development of the RSTS/E system, several hidden options were built into the SYSGEN program to assist DIGITAL development personnel with system generations and performance analysis. The hidden options are invoked through non-standard answers to several SYSGEN questions. Support for these features is neither expressed nor implied by this document. The SYSGEN questions of interest are listed below together with a brief explanation of the non-standard answers required to invoke the special options.

F.1 CLOCK ?

L, P or C	Normal responses to use the standard KW11L Line Time Clock (L), the KW11P Programmable Clock on line frequency (P), or the KW11P with crystal (C).
L/STAT P/STAT C/STAT	The modifier /STAT is used to include special statistics gathering code in the RSTS/E monitor. Includes code and tables to record job and disk access statistics. The statistics code is intended for performance analysis at DIGITAL and is not a supported feature of RSTS/E.

RSTS/E standardly supports the LP11 and LS11 (dot matrix) line printers. The printer device is also coded to handle the LV11 electrostatic printer/plotter in print mode only. The LV11 is not a supported device and has not been tested with the RSTS/E driver. A hidden option is used to set the line printer characteristics to those of an LV11.

F.2 LPn: TYPE?

LP or LS	Normal response for LP11 or LS11 printers.

LV	Response to set LV11 printer characteristics. LV11 can only be used in print mode. No plotting capability is provided.

APPENDIX G

ADDRESS AND VECTOR ASSIGNMENTS

The RSTS/E system assumes that all devices attached to the PDP-11 UNIBUS have been assigned addresses and vectors according to manufacturing standards. Several devices have so called "Floating Addresses." This means that the presence or absence of any floating address device will affect the assignment of addresses to other floating address devices. Similarly, several devices have "Floating Vectors." Interrupt vectors must be assigned in a specific sequence and the presence of one type of device will affect the correct assignment of interrupt vectors for other devices. There are also many standard options which have fixed addresses and vectors. This Appendix presents the algorithms for assignment of floating addresses and vectors, and lists the fixed assignments for devices supported under RSTS/E.[1]

G.1 FLOATING ADDRESSES

Currently the floating address devices include the DJ11 Multiplexor, DH11 Multiplexor, DQ11 Synchronous Line Interface, and the DU11 Synchronous Line Interface. The following ground rules apply to these devices and future floating address devices:

1. Only new devices will be assigned floating addresses. Devices now in production will keep their old addresses.
2. Future devices may float both their addresses and interrupt vectors.
3. The floating address space starts at 760010(8) and proceeds upward to 764000(8).
4. A gap in the address space (no slave SYNC) implies a device does not exist.

5. The first address of a new type device will always be on a 2↑N word boundary, where N is the integer value of (LOG$_2$M+.9999999), and M is the number of device registers.

Number of Registers In Device	Possible Boundaries
1	Any Word
2	XXXXX0, XXXXX4
3,4	XXXXX0
5,6,7,8	XXXX00, XXXX20, XXXX40, XXXX60
9 thru 16	XXXX00, XXXX40

6. A "gap" of at least one word will be left after each type of device, starting on the same boundary the device would start on. Note that the gap must be at least one word in length but may be longer than one word. Gap length is determined by the boundary on which the next device must begin.
7. Multiple devices of the same type must be addressed contiguously.

Address 760010 is reserved for the first DJ11. Since the DJ11 has four registers, additional DJ11's are assigned addresses modulo 10 (base 8) immediately following the first DJ11 (i.e. 760010, 760020, etc.). The modulo 10 (base 8) address following the last DJ11 is left empty and is known as the DJ11 gap. If there are no DJ11's, the gap is at 760010. If there is one DJ11, the gap will be at 760020. All gaps must be at least one word in length.

After all DJ11 addresses and the DJ11 gap are defined, the address for the first DH11 can be assigned. DH11's have eight registers which implies a modulo 20 (base 8) boundary. The address of the first DH11 is the first modulo 20 address following the DJ11 gap. If there are no DJ11's (DJ11 gap at 760010), the first DH11 is assigned address 760020. Similarly, if there is one DJ11, the DJ11 gap will begin at 760020 and the next available mod 20 boundary is 760040. All additional DH11's are assigned addresses modulo 20 immediately after the first DH11. The DH11 gap begins on the 20 boundary following the last DH11.

After all DH11 addresses and the DH11 gap are defined, DQ11 addresses may be assigned. Since the DQ11 has four registers, a modulo 10 boundary is required. This will be the first mod 10 boundary following the DH11 gap. On a system with one DJ11 and one DH11, the DH11 gap would be at address 760060 and the first available DQ11 address would be 760070. All additional DQ11's are assigned addresses immediately following the first DQ11. The DQ11 gap address is the mod 10 boundary following the last DQ11. Again a gap of at least one word is required.

Finally DU11 addresses can be defined in a similar manner. The DU11 has four registers and requires a modulo 10 boundary, for example, assume a system has one DJ11, one DH11, no DQ11, and at least one DU11. As mentioned above, the DH11 gap would be at 760060. Since there are no DQ11's, the DQ11 gap must be located at the first modulo 10 boundary following the DH11 gap. The DQ11 gap would, therefore, be at address 760070. The first available DU11 address would be 760100 and additional DU11 units would be assigned mod 10 addresses immediately following the first DU11.

Addresses for any future floating address devices will be assigned in a similar manner following the DU11 gap.

FLOATING ADDRESS TABLE
DJ11, DH11, DQ11, DU11

PAGE 1 OF 3

```
        DJ 0    0    0    0    0    1    1    1    1    1    2    2    2    2    2    3    3    3    3    3    4    4    4    4    4
        DH 0    1    2    3    4    0    1    2    3    4    0    1    2    3    4    0    1    2    3    4    0    1    2    3    4

DQ DU
 0  0   ---- -B-- -B-- -B-- -B-- A--- AD-- AD-- AD-- AD-- A--- AD-- AD-- AD-- AD-- A--- AF-- AF-- AF-- AF-- A--- AF-- AF-- AF-- AF--
        ---- ---- -D-- -D-- -D-- ---- ---- -F-- -F-- -F-- B--- B--- BF-- BF-- BF-- B--- B--- BH-- BH-- BH-- B--- B--- BH-- BH-- BH--
        ---- ---- ---- -F-- -F-- ---- ---- ---- -H-- -H-- ---- ---- ---- -H-- -H-- C--- C--- C--- CJ-- CJ-- C--- C--- C--- CJ-- CJ--
        ---- ---- ---- ---- -H-- ---- ---- ---- ---- -J-- ---- ---- ---- ---- -J-- ---- ---- ---- ---- -L-- D--- D--- D--- D--- DL--

 0  1   ---D -B-F -B-H -B-J -B-L A--F AD-H AD-J AD-L AD-N A--F AD-H AD-J AD-L AD-N A--H AF-J AF-L AF-N AF-P A--H AF-J AF-L AF-N AF-P
        ---- ---- -D-- -D-- -D-- ---- ---- -F-- -F-- -F-- B--- B--- BF-- BF-- BF-- B--- B--- BH-- BH-- BH-- B--- B--- BH-- BH-- BH--
        ---- ---- ---- -F-- -F-- ---- ---- ---- -H-- -H-- ---- ---- ---- -H-- -H-- C--- C--- C--- CJ-- CJ-- C--- C--- C--- CJ-- CJ--
        ---- ---- ---- ---- -H-- ---- ---- ---- ---- -J-- ---- ---- ---- ---- -J-- ---- ---- ---- ---- -L-- D--- D--- D--- D--- DL--

 0  2   ---D -B-F -B-H -B-J -B-L A--F AD-H AD-J AD-L AD-N A--F AD-H AD-J AD-L AD-N A--H AF-J AF-L AF-N AF-P A--H AF-J AF-L AF-N AF-P
        ---E ---G -D-I -D-K -D-M ---G ---I -F-K -F-M -F-O B--G B--I BF-K BF-M BF-O B--I B--K BH-M BH-O BH-Q B--I B--K BH-M BH-O BH-Q
        ---- ---- ---- -F-- -F-- ---- ---- ---- -H-- -H-- ---- ---- ---- -H-- -H-- C--- C--- C--- CJ-- CJ-- C--- C--- C--- CJ-- CJ--
        ---- ---- ---- ---- -H-- ---- ---- ---- ---- -J-- ---- ---- ---- ---- -J-- ---- ---- ---- ---- -L-- D--- D--- D--- D--- DL--

 0  3   ---D -B-F -B-H -B-J -B-L A--F AD-H AD-J AD-L AD-N A--F AD-H AD-J AD-L AD-N A--H AF-J AF-L AF-N AF-P A--H AF-J AF-L AF-N AF-P
        ---E ---G -D-I -D-K -D-M ---G ---I -F-K -F-M -F-O B--G B--I BF-K BF-M BF-O B--I B--K BH-M BH-O BH-Q B--I B--K BH-M BH-O BH-Q
        ---F ---H ---J -F-L -F-N ---H ---J ---L -H-N -H-P ---H ---J ---L -H-N -H-P C--J C--L C--N CJ-P CJ-R C--J C--L C--N CJ-P CJ-R
        ---- ---- ---- ---- -H-- ---- ---- ---- ---- -J-- ---- ---- ---- ---- -J-- ---- ---- ---- ---- -L-- D--- D--- D--- D--- DL--

 0  4   ---D -B-F -B-H -B-J -B-L A--F AD-H AD-J AD-L AD-N A--F AD-H AD-J AD-L AD-N A--H AF-J AF-L AF-N AF-P A--H AF-J AF-L AF-N AF-P
        ---E ---G -D-I -D-K -D-M ---G ---I -F-K -F-M -F-O B--G B--I BF-K BF-M BF-O B--I B--K BH-M BH-O BH-Q B--I B--K BH-M BH-O BH-Q
        ---F ---H ---J -F-L -F-N ---H ---J ---L -H-N -H-P ---H ---J ---L -H-N -H-P C--J C--L C--N CJ-P CJ-R C--J C--L C--N CJ-P CJ-R
        ---G ---I ---K ---M -H-O ---I ---K ---M ---O -J-Q ---I ---K ---M ---O -J-Q ---K ---M ---O ---Q -L-S D--K D--M D--O D--Q DL-S
```

WHAT IS THIS ?

THIS IS A SYMBOLIC TABLE OF UNIBUS ADDRESSES FOR THE FLOATING ADDRESS DEVICES DJ11,DH11, DQ11, AND DU11. THE LETTERS A THRU W CORRESPOND TO UNIBUS ADDRESSES SHOWN IN THE LIST TO THE RIGHT.

THE TABLE CONTAINS ALL POSSIBLE COMBINATIONS OF UP TO FOUR OF EACH TYPE OF DEVICE. THIS SHOULD COVER MOST OF THE PDP-11 CONFIGURATIONS. THE TABLE SHOULD NOT BE USED FOR ANY MACHINE CONFIGURED FOR MORE THAN FOUR DJ'S,DH'S,DQ'S, OR DU'S.

COLUMN HEADERS ARE THE NUMBER OF DJ11'S AND DH11'S. ROW HEADERS ARE THE NUMBER OF DQ11'S AND DU11'S.

HOW TO USE THE TABLE

DETERMINE THE NUMBER OF DJ11'S, DH11'S, DQ11'S, AND DU11'S TO BE INCLUDED IN THE HARDWARE CONFIGURATION.

FIND THE COLUMN CORRESPONDING TO THE NUMBER OF DJ11'S AND DH11'S. SCAN DOWN THIS COLUMN FOR THE ROW CORRESPONDING TO THE DESIRED NUMBER OF DQ11'S AND DU11'S.

AT THE INTERSECTION POINT IS A 4 BY 4 BLOCK OF SYMBOLS INCLUDING DASHES AND LETTERS A THRU W. THE LETTERS REPRESENT THE PROPER UNIBUS ADDRESS FOR EACH DEVICE. THE LIST TO THE RIGHT SHOWS THE ADDRESS CORRESPONDING TO EACH LETTER. THE DASHES INDICATE NON-EXISTENT DEVICES. THE 4X4 BLOCK IS INTERPRETED AS SHOWN BELOW:

```
         D D D D
         J H Q U

    1ST  A D K M

    2ND  B F - N

    3RD  - H - O

    4TH  - - - P
```

ADDRESSES

A = 760010
B = 760020
C = 760030
D = 760040
E = 760050
F = 760060
G = 760070
H = 760100
I = 760110
J = 760120
K = 760130
L = 760140
M = 760150
N = 760160
O = 760170
P = 760200
Q = 760210
R = 760220
S = 760230
T = 760240
U = 760250
V = 760260
W = 760270

EXAMPLE

CONSIDER A SYSTEM TO INCLUDE TWO DJ'S, 3 DH'S, 1 DQ, AND 4 DU'S.

THE COLUMN CORRESPONDING TO 2 DJ'S AND 3 DH'S IS FOUND AT THE CENTER OF THE TABLE. THE ROW FOR 1 DQ AND 4 DU'S IS LOCATED ON PAGE 2 OF 3. THE BLOCK AT THE INTERSECTION IS THE ONE SHOWN TO THE LEFT BELOW.

USING THE 4 X 4 BLOCK AND THE LIST OF ADDRESSES, THE CORRECT FLOATING ADDRESSES CAN BE READ IMMEDIATELY.

1ST DJ11 = A = 760010
2ND DJ11 = B = 760020
1ST DH11 = D = 760040
2ND DH11 = F = 760060
3RD DH11 = H = 760100
1ST DQ11 = K = 760130
1ST DU11 = M = 760150
2ND DU11 = N = 760160
3RD DU11 = O = 760170
4TH DU11 = P = 760200

```
                                         FLOATING ADDRESS TABLE                                   PAGE 2 OF 3
                                         DJ11, DH11, DQ11, DU11

   DJ   0    0    0    0    0    1    1    1    1    1    2    2    2    2    2    3    3    3    3    3    4    4    4    4    4
   DH   0    1    2    3    4    0    1    2    3    4    0    1    2    3    4    0    1    2    3    4    0    1    2    3    4

DQ DU
 1  0 --C- -BE- -BG- -BI- -BK- A-E- ADG- ADI- ADK- ADM- A-E- ADG- ADI- ADK- ADM- A-G- AFI- AFK- AFM- AFO- A-G- AFI- AFK- AFM- AFO-
      ---- ---- -D-- -D-- -D-- ---- ---- -F-- -F-- -F-- B--- B--- BF-- BF-- BF-- B--- B--- BH-- BH-- BH-- B--- B--- BH-- BH-- BH--
      ---- ---- ---- -F-- -F-- ---- ---- ---- -H-- -H-- ---- ---- ---- -H-- -H-- C--- C--- C--- CJ-- CJ-- C--- C--- C--- CJ-- CJ--
      ---- ---- ---- ---- -H-- ---- ---- ---- ---- -J-- ---- ---- ---- ---- -J-- ---- ---- ---- ---- -L-- D--- D--- D--- D--- DL--

 1  1 --CE -BEG -BGI -BIK -BKM A-EG ADGI ADIK ADKM ADMO A-EG ADGI ADIK ADKM ADMO A-GI AFIK AFKM AFMO AFOQ A-GI AFIK AFKM AFMO AFOQ
      ---- ---- -D-- -D-- -D-- ---- ---- -F-- -F-- -F-- B--- B--- BF-- BF-- BF-- B--- B--- BH-- BH-- BH-- B--- B--- BH-- BH-- BH--
      ---- ---- ---- -F-- -F-- ---- ---- ---- -H-- -H-- ---- ---- ---- -H-- -H-- C--- C--- C--- CJ-- CJ-- C--- C--- C--- CJ-- CJ--
      ---- ---- ---- ---- -H-- ---- ---- ---- ---- -J-- ---- ---- ---- ---- -J-- ---- ---- ---- ---- -L-- D--- D--- D--- D--- DL--

 1  2 --CE -BEG -BGI -BIK -BKM A-EG ADGI ADIK ADKM ADMO A-EG ADGI ADIK ADKM ADMO A-GI AFIK AFKM AFMO AFOQ A-GI AFIK AFKM AFMO AFOQ
      ---F ---H -D-J -D-L -D-N ---H ---J -F-L -F-N -F-P B--H B--J BF-L BF-N BF-P B--J B--L BH-N BH-P BH-R B--J B--L BH-N BH-P BH-R
      ---- ---- ---- -F-- -F-- ---- ---- ---- -H-- -H-- ---- ---- ---- -H-- -H-- C--- C--- C--- CJ-- CJ-- C--- C--- C--- CJ-- CJ--
      ---- ---- ---- ---- -H-- ---- ---- ---- ---- -J-- ---- ---- ---- ---- -J-- ---- ---- ---- ---- -L-- D--- D--- D--- D--- DL--

 1  3 --CE -BEG -BGI -BIK -BKM A-EG ADGI ADIK ADKM ADMO A-EG ADGI ADIK ADKM ADMO A-GI AFIK AFKM AFMO AFOQ A-GI AFIK AFKM AFMO AFOQ
      ---F ---H -D-J -D-L -D-N ---H ---J -F-L -F-N -F-P B--H B--J BF-L BF-N BF-P B--J B--L BH-N BH-P BH-R B--J B--L BH-N BH-P BH-R
      ---G ---I ---K -F-M -F-O ---I ---K ---M -H-O -H-Q ---I ---K ---M -H-O -H-Q C--K C--M C--O CJ-Q CJ-S C--K C--M C--O CJ-Q CJ-S
      ---- ---- ---- ---- -H-- ---- ---- ---- ---- -J-- ---- ---- ---- ---- -J-- ---- ---- ---- ---- -L-- D--- D--- D--- D--- DL--

 1  4 --CE -BEG -BGI -BIK -BKM A-EG ADGI ADIK ADKM ADMO A-EG ADGI ADIK ADKM ADMO A-GI AFIK AFKM AFMO AFOQ A-GI AFIK AFKM AFMO AFOQ
      ---F ---H -D-J -D-L -D-N ---H ---J -F-L -F-N -F-P B--H B--J BF-L BF-N BF-P B--J B--L BH-N BH-P BH-R B--J B--L BH-N BH-P BH-R
      ---G ---I ---K -F-M -F-O ---I ---K ---M -H-O -H-Q ---I ---K ---M -H-O -H-Q C--K C--M C--O CJ-Q CJ-S C--K C--M C--O CJ-Q CJ-S
      ---H ---J ---L ---N -H-P ---J ---L ---N ---P -J-R ---J ---L ---N ---P -J-R ---L ---N ---P ---R -L-T D--L D--N D--P D--R DL-T

 2  0 --C- -BE- -BG- -BI- -BK- A-E- ADG- ADI- ADK- ADM- A-E- ADG- ADI- ADK- ADM- A-G- AFI- AFK- AFM- AFO- A-G- AFI- AFK- AFM- AFO-
      --D- --F- -DH- -DJ- -DL- --F- --H- -FJ- -FL- -FN- B-F- B-H- BFJ- BFL- BFN- B-H- B-J- BHL- BHN- BHP- B-H- B-J- BHL- BHN- BHP-
      ---- ---- ---- -F-- -F-- ---- ---- ---- -H-- -H-- ---- ---- ---- -H-- -H-- C--- C--- C--- CJ-- CJ-- C--- C--- C--- CJ-- CJ--
      ---- ---- ---- ---- -H-- ---- ---- ---- ---- -J-- ---- ---- ---- ---- -J-- ---- ---- ---- ---- -L-- D--- D--- D--- D--- DL--

 2  1 --CF -BEH -BGJ -BIL -BKN A-EH ADGJ ADIL ADKN ADMP A-EH ADGJ ADIL ADKN ADMP A-GJ AFIL AFKN AFMP AFOR A-GJ AFIL AFKN AFMP AFOR
      --D- --F- -DH- -DJ- -DL- --F- --H- -FJ- -FL- -FN- B-F- B-H- BFJ- BFL- BFN- B-H- B-J- BHL- BHN- BHP- B-H- B-J- BHL- BHN- BHP-
      ---- ---- ---- -F-- -F-- ---- ---- ---- -H-- -H-- ---- ---- ---- -H-- -H-- C--- C--- C--- CJ-- CJ-- C--- C--- C--- CJ-- CJ--
      ---- ---- ---- ---- -H-- ---- ---- ---- ---- -J-- ---- ---- ---- ---- -J-- ---- ---- ---- ---- -L-- D--- D--- D--- D--- DL--

 2  2 --CF -BEH -BGJ -BIL -BKN A-EH ADGJ ADIL ADKN ADMP A-EH ADGJ ADIL ADKN ADMP A-GJ AFIL AFKN AFMP AFOR A-GJ AFIL AFKN AFMP AFOR
      --DG --FI -DHK -DJM -DLO --FI --HK -FJM -FLO -FNQ B-FI B-HK BFJM BFLO BFNQ B-HK B-JM BHLO BHNQ BHPS B-HK B-JM BHLO BHNQ BHPS
      ---- ---- ---- -F-- -F-- ---- ---- ---- -H-- -H-- ---- ---- ---- -H-- -H-- C--- C--- C--- CJ-- CJ-- C--- C--- C--- CJ-- CJ--
      ---- ---- ---- ---- -H-- ---- ---- ---- ---- -J-- ---- ---- ---- ---- -J-- ---- ---- ---- ---- -L-- D--- D--- D--- D--- DL--

 2  3 --CF -BEH -BGJ -BIL -BKN A-EH ADGJ ADIL ADKN ADMP A-EH ADGJ ADIL ADKN ADMP A-GJ AFIL AFKN AFMP AFOR A-GJ AFIL AFKN AFMP AFOR
      --DG --FI -DHK -DJM -DLO --FI --HK -FJM -FLO -FNQ B-FI B-HK BFJM BFLO BFNQ B-HK B-JM BHLO BHNQ BHPS B-HK B-JM BHLO BHNQ BHPS
      ---H ---J ---L -F-N -F-P ---J ---L ---N -H-P -H-R ---J ---L ---N -H-P -H-R C--L C--N C--P CJ-R CJ-T C--L C--N C--P CJ-R CJ-T
      ---- ---- ---- ---- -H-- ---- ---- ---- ---- -J-- ---- ---- ---- ---- -J-- ---- ---- ---- ---- -L-- D--- D--- D--- D--- DL--

 2  4 --CF -BEH -BGJ -BIL -BKN A-EH ADGJ ADIL ADKN ADMP A-EH ADGJ ADIL ADKN ADMP A-GJ AFIL AFKN AFMP AFOR A-GJ AFIL AFKN AFMP AFOR
      --DG --FI -DHK -DJM -DLO --FI --HK -FJM -FLO -FNQ R-FI B-HK BFJM BFLO BFNQ B-HK B-JM BHLO BHNQ BHPS B-HK B-JM BHLO BHNQ BHPS
      ---H ---J ---L -F-N -F-P ---J ---L ---N -H-P -H-R ---J ---L ---N -H-P -H-R C--L C--N C--P CJ-R CJ-T C--L C--N C--P CJ-R CJ-T
      ---I ---K ---M ---O -H-Q ---K ---M ---O ---Q -J-S ---K ---M ---O ---Q -J-S ---M ---O ---Q ---S -L-U D--M D--O D--Q D--S DL-U
```

```
                                   FLOATING ADDRESS TABLE                         PAGE 3 OF 3
                                   DJ11, DH11, DQ11, DU11

    DJ 0     0     0     0     0     1     1     1     1     1     2     2     2     2     2     3     3     3     3     3     4     4     4     4     4
    DH 0     1     2     3     4     0     1     2     3     4     0     1     2     3     4     0     1     2     3     4     0     1     2     3     4

DQ DU
 3  0  --C-  -BE-  -BG-  -BI-  -BK-  A-E-  ADG-  ADI-  ADK-  ADM-  A-E-  ADG-  ADI-  ADK-  ADM-  A-G-  AFI-  AFK-  AFM-  AFO-  A-G-  AFI-  AFK-  AFM-  AFO-
       --D-  --F-  -DH-  -DJ-  -DL-  --F-  --H-  -FJ-  -FL-  -FN-  B-F-  B-H-  BFJ-  BFL-  BFN-  B-H-  B-J-  BHL-  BHN-  BHP-  B-H-  B-J-  BHL-  BHN-  BHP-
       --E-  --G-  --I-  -FK-  -FM-  --G-  --I-  --K-  -HM-  -HO-  --G-  --I-  --K-  -HM-  -HO-  C-I-  C-K-  C-M-  CJO-  CJQ-  C-I-  C-K-  C-M-  CJO-  CJQ-
       ----  ----  ----  ----  -H--  ----  ----  ----  ----  -J--  ----  ----  ----  ----  -J--  ----  ----  ----  ----  -L--  D---  D---  D---  D---  DL--

 3  1  --CG  -BEI  -BGK  -BIM  -BKO  A-EI  ADGK  ADIM  ADKO  ADMQ  A-EI  ADGK  ADIM  ADKO  ADMQ  A-GK  AFIM  AFKO  AFMQ  AFOS  A-GK  AFIM  AFKO  AFMQ  AFOS
       --D-  --F-  -DH-  -DJ-  -DL-  --F-  --H-  -FJ-  -FL-  -FN-  B-F-  B-H-  BFJ-  BFL-  BFN-  B-H-  B-J-  BHL-  BHN-  BHP-  B-H-  B-J-  BHL-  BHN-  BHP-
       --E-  --G-  --I-  -FK-  -FM-  --G-  --I-  --K-  -HM-  -HO-  --G-  --I-  --K-  -HM-  -HO-  C-I-  C-K-  C-M-  CJO-  CJQ-  C-I-  C-K-  C-M-  CJO-  CJQ-
       ----  ----  ----  ----  -H--  ----  ----  ----  ----  -J--  ----  ----  ----  ----  -J--  ----  ----  ----  ----  -L--  D---  D---  D---  D---  DL--

 3  2  --CG  -BEI  -BGK  -BIM  -BKO  A-EI  ADGK  ADIM  ADKO  ADMQ  A-EI  ADGK  ADIM  ADKO  ADMQ  A-GK  AFIM  AFKO  AFMQ  AFOS  A-GK  AFIM  AFKO  AFMQ  AFOS
       --DH  --FJ  -DHL  -DJN  -DLP  --FJ  --HL  -FJN  -FLP  -FNR  B-FJ  B-HL  BFJN  BFLP  BFNR  B-HL  B-JN  BHLP  BHNR  BHPT  B-HL  B-JN  BHLP  BHNR  BHPT
       --E-  --G-  --I-  -FK-  -FM-  --G-  --I-  --K-  -HM-  -HO-  --G-  --I-  --K-  -HM-  -HO-  C-I-  C-K-  C-M-  CJO-  CJQ-  C-I-  C-K-  C-M-  CJO-  CJQ-
       ----  ----  ----  ----  -H--  ----  ----  ----  ----  -J--  ----  ----  ----  ----  -J--  ----  ----  ----  ----  -L--  D---  D---  D---  D---  DL--

 3  3  --CG  -BEI  -BGK  -BIM  -BKO  A-EI  ADGK  ADIM  ADKO  ADMQ  A-EI  ADGK  ADIM  ADKO  ADMQ  A-GK  AFIM  AFKO  AFMQ  AFOS  A-GK  AFIM  AFKO  AFMQ  AFOS
       --DH  --FJ  -DHL  -DJN  -DLP  --FJ  --HL  -FJN  -FLP  -FNR  B-FJ  B-HL  BFJN  BFLP  BFNR  B-HL  B-JN  BHLP  BHNR  BHPT  B-HL  B-JN  BHLP  BHNR  BHPT
       --EI  --GK  --IM  -FKO  -FMQ  --GK  --IM  --KO  -HMQ  -HOS  --GK  --IM  --KO  -HMQ  -HOS  C-IM  C-KO  C-MQ  CJOS  CJQU  C-IM  C-KO  C-MQ  CJOS  CJQU
       ----  ----  ----  ----  -H--  ----  ----  ----  ----  -J--  ----  ----  ----  ----  -J--  ----  ----  ----  ----  -L--  D---  D---  D---  D---  DL--

 3  4  --CG  -BEI  -BGK  -BIM  -BKO  A-EI  ADGK  ADIM  ADKO  ADMQ  A-EI  ADGK  ADIM  ADKO  ADMQ  A-GK  AFIM  AFKO  AFMQ  AFOS  A-GK  AFIM  AFKO  AFMQ  AFOS
       --DH  --FJ  -DHL  -DJN  -DLP  --FJ  --HL  -FJN  -FLP  -FNR  B-FJ  B-HL  BFJN  BFLP  BFNR  B-HL  B-JN  BHLP  BHNR  BHPT  B-HL  B-JN  BHLP  BHNR  BHPT
       --EI  --GK  --IM  -FKO  -FMQ  --GK  --IM  --KO  -HMQ  -HOS  --GK  --IM  --KO  -HMQ  -HOS  C-IM  C-KO  C-MQ  CJOS  CJQU  C-IM  C-KO  C-MQ  CJOS  CJQU
       ---J  ---L  ---N  ---P  -H-R  ---L  ---N  ---P  ---R  -J-T  ---L  ---N  ---P  ---R  -J-T  ---N  ---P  ---R  ---T  -L-V  D--N  D--P  D--R  D--T  DL-V

 4  0  --C-  -BE-  -BG-  -BI-  -BK-  A-E-  ADG-  ADI-  ADK-  ADM-  A-E-  ADG-  ADI-  ADK-  ADM-  A-G-  AFI-  AFK-  AFM-  AFO-  A-G-  AFI-  AFK-  AFM-  AFO-
       --D-  --F-  -DH-  -DJ-  -DL-  --F-  --H-  -FJ-  -FL-  -FN-  B-F-  B-H-  BFJ-  BFL-  BFN-  B-H-  B-J-  BHL-  BHN-  BHP-  B-H-  B-J-  BHL-  BHN-  BHP-
       --E-  --G-  --I-  -FK-  -FM-  --G-  --I-  --K-  -HM-  -HO-  --G-  --I-  --K-  -HM-  -HO-  C-I-  C-K-  C-M-  CJO-  CJQ-  C-I-  C-K-  C-M-  CJO-  CJQ-
       --F-  --H-  --J-  --L-  -HN-  --H-  --J-  --L-  --N-  -JP-  --H-  --J-  --L-  --N-  -JP-  --J-  --L-  --N-  --P-  -LR-  D-J-  D-L-  D-N-  D-P-  DLR-

 4  1  --CH  -BEJ  -BGL  -BIN  -BKP  A-EJ  ADGL  ADIN  ADKP  ADMR  A-EJ  ADGL  ADIN  ADKP  ADMR  A-GL  AFIN  AFKP  AFMR  AFOT  A-GL  AFIN  AFKP  AFMR  AFOT
       --D-  --F-  -DH-  -DJ-  -DL-  --F-  --H-  -FJ-  -FL-  -FN-  B-F-  B-H-  BFJ-  BFL-  BFN-  B-H-  B-J-  BHL-  BHN-  BHP-  B-H-  B-J-  BHL-  BHN-  BHP-
       --E-  --G-  --I-  -FK-  -FM-  --G-  --I-  --K-  -HM-  -HO-  --G-  --I-  --K-  -HM-  -HO-  C-I-  C-K-  C-M-  CJO-  CJQ-  C-I-  C-K-  C-M-  CJO-  CJQ-
       --F-  --H-  --J-  --L-  -HN-  --H-  --J-  --L-  --N-  -JP-  --H-  --J-  --L-  --N-  -JP-  --J-  --L-  --N-  --P-  -LR-  D-J-  D-L-  D-N-  D-P-  DLR-

 4  2  --CH  -BEJ  -BGL  -BIN  -BKP  A-EJ  ADGL  ADIN  ADKP  ADMR  A-EJ  ADGL  ADIN  ADKP  ADMR  A-GL  AFIN  AFKP  AFMR  AFOT  A-GL  AFIN  AFKP  AFMR  AFOT
       --DI  --FK  -DHM  -DJO  -DLQ  --FK  --HM  -FJO  -FLQ  -FNS  B-FK  B-HM  BFJO  BFLQ  BFNS  B-HM  B-JO  BHLQ  BHNS  BHPU  B-HM  B-JO  BHLQ  BHNS  BHPU
       --E-  --G-  --I-  -FK-  -FM-  --G-  --I-  --K-  -HM-  -HO-  --G-  --I-  --K-  -HM-  -HO-  C-I-  C-K-  C-M-  CJO-  CJQ-  C-I-  C-K-  C-M-  CJO-  CJQ-
       --F-  --H-  --J-  --L-  -HN-  --H-  --J-  --L-  --N-  -JP-  --H-  --J-  --L-  --N-  -JP-  --J-  --L-  --N-  --P-  -LR-  D-J-  D-L-  D-N-  D-P-  DLR-

 4  3  --CH  -BEJ  -BGL  -BIN  -BKP  A-EJ  ADGL  ADIN  ADKP  ADMR  A-EJ  ADGL  ADIN  ADKP  ADMR  A-GL  AFIN  AFKP  AFMR  AFOT  A-GL  AFIN  AFKP  AFMR  AFOT
       --DI  --FK  -DHM  -DJO  -DLQ  --FK  --HM  -FJO  -FLQ  -FNS  B-FK  B-HM  BFJO  BFLQ  BFNS  B-HM  B-JO  BHLQ  BHNS  BHPU  B-HM  B-JO  BHLQ  BHNS  BHPU
       --EJ  --GL  --IN  -FKP  -FMR  --GL  --IN  --KP  -HMR  -HOT  --GL  --IN  --KP  -HMR  -HOT  C-IN  C-KP  C-MR  CJOT  CJQV  C-IN  C-KP  C-MR  CJOT  CJQV
       --F-  --H-  --J-  --L-  -HN-  --H-  --J-  --L-  --N-  -JP-  --H-  --J-  --L-  --N-  -JP-  --J-  --L-  --N-  --P-  -LR-  D-J-  D-L-  D-N-  D-P-  DLR-

 4  4  --CH  -BEJ  -BGL  -BIN  -BKP  A-EJ  ADGL  ADIN  ADKP  ADMR  A-EJ  ADGL  ADIN  ADKP  ADMR  A-GL  AFIN  AFKP  AFMR  AFOT  A-GL  AFIN  AFKP  AFMR  AFOT
       --DI  --FK  -DHM  -DJO  -DLQ  --FK  --HM  -FJO  -FLQ  -FNS  B-FK  B-HM  BFJO  BFLQ  BFNS  B-HM  B-JO  BHLQ  BHNS  BHPU  B-HM  B-JO  BHLQ  BHNS  BHPU
       --EJ  --GL  --IN  -FKP  -FMR  --GL  --IN  --KP  -HMR  -HOT  --GL  --IN  --KP  -HMR  -HOT  C-IN  C-KP  C-MR  CJOT  CJQV  C-IN  C-KP  C-MR  CJOT  CJQV
       --FK  --HM  --JO  --LQ  -HNS  --HM  --JO  --LQ  --NS  -JPU  --HM  --JO  --LQ  --NS  -JPU  --JO  --LQ  --NS  --PU  -LRW  D-JO  D-LQ  D-NS  D-PU  DLRW
```

G.2 FLOATING VECTORS

Many devices have floating vectors. The vector assignment sequence is normally the same sequence as that in which the devices enter production. A vector of a new hardware option is not inserted before the vector for a device that is already in production. Gaps in the vector assignments are not required. The floating vectors begin at address 300 and proceed continuously upwards. The vector assignment sequence for current devices is defined below.

Device	First Address	Next Addr.	Vector Size	Max # Units	BR Level	RSTS/E Notes
DC11	174000	+10	10	32	BR5	
KL11,DL11A,B	176500	+10	10	16	BR4	NON-CONSOLE
DP11	174770	-10	10	32	BR5	2780 ONLY
DM11A	175000	+10	10	16	BR5	NOT SUPPORTED
DN11	175200	+10	4	16	BR4	NOT SUPPORTED
DM11BB	170500	+10	4	16	BR4	
DR11A,C	167770	-10	10[1]	32	BR5	NOT SUPPORTED
PA611 READER	172600	+ 4	4[1]	16	BR4	NOT SUPPORTED
PA611 PUNCH	172700	+ 4	4[1]	16	BR4	NOT SUPPORTED
DT11 (DT03-FP)	174200	+ 2	10[1]	8	BR7	NOT SUPPORTED
DX11	176200	+40	10[1]	4	BR4	NOT SUPPORTED
DL11C,D,E	175610	+10	10[1]	31	BR4	
DJ11	FLOAT	+10	10[1]	16	BR4	NOT SUPPORTED
DH11	FLOAT	+20	10[1]	16	BR5	
GT40	172000		20[1]		BR4	NOT SUPPORTED
LPS11	170400	+40	30[1]	14	BR5,6	NOT SUPPORTED
DQ11	FLOAT	+10	10[1]	16	BR5	NOT SUPPORTED
KW11W	172400	NA	10[1]	1		NOT SUPPORTED
DU11	FLOAT	+10	10[1]	16	BR5	2780 ONLY

Floating address and vector devices which are not supported under RSTS/E must be identified during system generation so that the system is configured for the correct addresses and vectors.

---

[1] The first vector for the first device of this type must always be on a 10(8) boundary.

G.3 FIXED ADDRESSES AND VECTORS

The table below lists the devices supported under RSTS/E which have fixed addresses and vectors.

Device	Address	Vector	BR Level	RSTS/E Notes
RC11/RC64	177440	210	BR5	UP TO 4 PLATTERS
RF11/RS11	177460	204	BR5	UP TO 8 PLATTERS
RK11/RK03/RK05	177400	220	BR5	UP TO 8 DRIVES
RP11C/RP02/RP03	176710	254	BR5	UP TO 8 DRIVES
RH11/RS03/RS04	172040	204	BR5	UP TO 4 DRIVES
RH11/RP04	176700	254	BR5	UP TO 8 DRIVES
RX11/RX01	177170	264	BR5	UP TO 8 DRIVES
TC11	177340	214	BR6	UP TO 8 DRIVES
TM11/TU10	172520	224	BR5	UP TO 8 DRIVES
RH11/TM02/TU16	172440	224	BR5	UP TO 8 DRIVES
LP11, LS11 (LP0)	177514	200	BR4	UP TO 8 PRINTERS DEPENDING ON SPEED.
(LP1)	164004	170	BR4	
(LP2)	164014	174	BR4	
(LP3)	164024	270	BR4	
(LP4)	164034	274	BR4	
(LP5)	164044	774	BR4	
(LP6)	164054	770	BR4	
(LP7)	164064	764	BR4	
CR11, CM11	177160	230	BR5	
CD11	177160	230	BR4	NEW ADDRESS[1]
CD11	(172460)	230	BR4	OLD ADDRESS[1]
KW11L	177546	100	BR6	
KW11P	172540	104	BR6	
KG11	170700	NONE	NONE	2780 ONLY
KL11, DL11A, DL11B	177560	60	BR4	CONSOLE INTERFACE

---

[1] The CD11 address was changed by CD11 ECO number 13.

INDEX

HOW TO OBTAIN SOFTWARE INFORMATION

SOFTWARE NEWSLETTERS, MAILING LIST

The Software Communications Group, located at corporate headquarters in Maynard, publishes software newsletters for the various DIGITAL products. Newsletters are published monthly, and keep the user informed about customer software problems and solutions, new software products, documentation corrections, as well as programming notes and techniques.

There are two similar levels of service:

- The Software Dispatch
- The Digital Software News

The Software Dispatch is part of the Software Maintenance Service. This service applies to the following software products:

PDP-9/15
RSX-11D
DOS/BATCH
RSTS/E
DECsystem-10

A Digital Software News for the PDP-11 and a Digital Software News for the PDP-8/12 are available to any customer who has purchased PDP-11 or PDP-8/12 software.

A collection of existing problems and solutions for a given software system is published periodically. A customer receives this publication with his initial software kit with the delivery of his system. This collection would be either a Software Dispatch Review or Software Performance Summary depending on the system ordered.

A mailing list of users who receive software newsletters is also maintained by Software Communications. Users must sign-up for the newsletter they desire. This can be done by either completing the form supplied with the Review or Summary or by writing to:

Software Communications
P.O. Box F
Maynard, Massachusetts 01754

SOFTWARE PROBLEMS

Questions or problems relating to DIGITAL's software should be reported as follows:

North and South American Submitters:

Upon completion of Software Performance Report (SPR) form remove last copy and send remainder to:

Software Communications
P.O. Box F
Maynard, Massachusetts 01754

The acknowledgement copy will be returned along with a blank SPR form upon receipt. The acknowledgement will contain a DIGITAL assigned SPR number. The SPR number or the preprinted number should be referenced in any future correspondence. Additional SPR forms may be obtained from the above address.

All International Submitters:

Upon completion of the SPR form, reserve the last copy and send the remainder to the SPR Center in the nearest DIGITAL office. SPR forms are also available from our SPR Centers.

PROGRAMS AND MANUALS

Software and manuals should be ordered by title and order number. In the United States, send orders to the nearest distribution center.

Digital Equipment Corporation
Software Distribution Center
146 Main Street
Maynard, Massachusetts 01754

Digital Equipment Corporation
Software Distribution Center
1400 Terra Bella
Mountain View, California 94043

Outside of the United States, orders should be directed to the nearest Digital Field Sales Office or representative.

USERS SOCIETY

DECUS, Digital Equipment Computers Users Society, maintains a user exchange center for user-written programs and technical application information. The Library contains approximately 1,900 programs for all DIGITAL computer lines. Executive routines, editors, debuggers, special functions, games, maintenance and various other classes of programs are available.

DECUS Program Library Catalogs are routinely updated and contain lists and abstracts of all programs according to computer line:

- PDP-8, FOCAL-8, BASIC-8, PDP-12
- PDP-7/9, 9, 15
- PDP-11, RSTS-11
- PDP-6/10, 10

Forms and information on acquiring and submitting programs to the DECUS Library may be obtained from the DECUS office.

In addition to the catalogs, DECUS also publishes the following:

DECUSCOPE	-The Society's technical newsletter, published bi-monthly, aimed at facilitating the interchange of technical information among users of DIGITAL computers and at disseminating news items concerning the Society. Circulation reached 19,000 in May, 1974.
PROCEEDINGS OF THE DIGITAL EQUIPMENT USERS SOCIETY	-Contains technical papers presented at DECUS Symposia held twice a year in the United States, once a year in Europe, Australia, and Canada.
MINUTES OF THE DECsystem-10 SESSIONS	-A report of the DECsystem-10 sessions held at the two United States DECUS Symposia.
COPY-N-Mail	-A monthly mailed communique among DECsystem-10 users.
LUG/SIG	-Mailing of Local User Group (LUG) and Special Interest Group (SIG) communique, aimed at providing closer communication among users of a specific product or application.

Further information on the DECUS Library, publications, and other DECUS activities is available from the DECUS offices listed below:

DECUS
Digital Equipment Corporation
146 Main Street
Maynard, Massachusetts 01754

DECUS EUROPE
Digital Equipment Corp. International
(Europe)
P.O. Box 340
1211 Geneva 26
Switzerland

RSTS/E
System Generation
Manual
DEC-11-ORGNA-A-D

Please cut along this line.

READER'S COMMENTS

NOTE: This form is for document comments only. Problems with software should be reported on a Software Problem Report (SPR) form (see the HOW TO OBTAIN SOFTWARE INFORMATION page).

Did you find errors in this manual? If so, specify by page.

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

Is there sufficient documentation on associated system programs required for use of the software described in this manual? If not, what material is missing and where should it be placed?

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

- ☐ Assembly language programmer
- ☐ Higher-level language programmer
- ☐ Occasional programmer (experienced)
- ☐ User with little programming experience
- ☐ Student programmer
- ☐ Non-programmer interested in computer concepts and capabilities

Name ______________________ Date ______________

Organization ______________________

Street ______________________

City ______________ State ________ Zip Code or Country ____________

If you do not require a written reply, please check here. ☐

-- **Fold Here** --

-- **Do Not Tear - Fold Here and Staple** --

FIRST CLASS
PERMIT NO. 33
MAYNARD, MASS.

BUSINESS REPLY MAIL
NO POSTAGE STAMP NECESSARY IF MAILED IN THE UNITED STATES

Postage will be paid by:

digital

Software Communications
P. O. Box F
Maynard, Massachusetts 01754