HEWLETT PACKARD

RTE-6/VM System Manager's

Reference Manual

Software Technology Divison
11000 Wolfe Road
Cupertino, CA 95014-9804

Manual Part No. 92084-90009
E0693

Printed in U.S.A. June 1993
Fifth Edition

Printing History

The Printing History below identifies the edition of this manual and any updates that are included. Periodically, update packages are distributed which contain replacement pages to be merged into the manual, including an updated copy of this printing history page. Also, the update may contain write-in instructions.

Each reprinting of this manual will incorporate all past updates; however, no new information will be added. Thus, the reprinted copy will be identical in content to prior printings of the same edition with its user-inserted update information. New editions of this manual will contain new information, as well as all updates.

To determine what manual edition and update is compatible with your current software revision code, refer to the Manual Numbering File or the Computer User's Documentation Index. (The Manual Numbering File is included with your software. It consists of an "M" followed by a five digit product number.)

First Edition	Dec 1981	
Update 1	Feb 1982	Misc. Errata
Reprint	Feb 1982	Update 1 Incorporated
Update 2	Jul 1982	Misc. Errata. Add Library Gen. Inf.
Update 3	Jan 1983	Misc. Errata. Add CSERR, CS80, 7914
Second Edition	Dec 1983	
Third Edition	Jan 1985	
Update 1	Jan 1986	
Reprint	Jan 1986	Update 1 Incorporated
Fourth Edition	Aug 1987	
Update 1	Jan 1989	Rev. 5010 (Software Rev. 5.1)
Update 2	Jul 1990	Rev. 5020 (Software Rev. 5.2)
Fifth Edition	Jun 1993	Rev. 6000 (Software Rev. 6.0)

Preface

This manual provides the System Manager with the information required to plan, generate, initialize, and maintain the HP 92084A RTE-6/VM Software System. The System Manager is assumed to have a working knowledge of RTE and should be familiar with the family of RTE-6/VM manuals. See the *Index to the Operating Systems Manuals*, part number 92084-90001 for a documentation map and descriptions of these manuals.

Examples are given throughout this manual to illustrate specific areas of interest, for example, how to fill in the disk worksheets, how to assign system logical units, etc. For a complete model system generation example, refer to the *RTE-6/VM Online Generator Reference Manual*, part number 92084-90010.

Chapter 1	Gives a description of the System Manager's responsibilities. A procedural summary is provided. Also included in the summary is the appropriate manuals required for certain system functions.
Chapter 2	Discusses evaluation of the user base.
Chapter 3	Shows how to plan the Session Account System. It talks about setting up group/user account structure, and allocating disk resources.
Chapter 4	Describes what you need for system generation and the purpose of the major steps to be followed as described in the RTE-6/VM Online Generator Manual. This chapter lists the software components and system resources required for the operating system, the File Management System, Terminal Interface, Session Account System, and Batch and Spooling System.
Chapter 5	Provides information on transferring to the new operating system.
Chapter 6	Contains the procedures required after system generation to activate the RTE-6/VM System. Included in this chapter are boot-up procedures, FMGR initialization, system tests, installation of the Command Interpreter (CI), utility installation procedures, and spool system initialization.
Chapter 7	Provides detailed information on initializing your Session Account System.
Chapter 8	Provides additional information to quide you on the operation and utilization of the Accounts Program. This chapter will help you to alter the account structure, back it up, and add more accounts into the system.
Chapter 9	Gives you the detailed information for adjusting system parameters and tables once the system is operational.
Chapter 10	Discusses the operation of the reconfigurator program used for I/O and memory reconfiguration.
Appendix A	Discusses real-time disk usages.
Appendix B	Describes the system communication area and provides detailed information on RTE system tables.
Appendix C	Provides a description of the Session Monitor Table formats.
Appendix D	Describes the Data Control Block and Directory formats.
Appendix E	Contains system table listings.

Table of Contents

Chapter 1
System Management Overview

Chapter 2
Determining User Requirements

Chapter 3
Session Account Planning

Chapter 4
System Generation Response

Chapter 5
Transferring the New Operating System

Chapter 6
System Initialization

Chapter 7
Session Monitor Initialization

Chapter 8
Maintaining the Account System

Chapter 9
Adjusting System Parameters

Chapter 10
Memory and I/O Reconfiguration

Appendix A
Real-Time Disk Usage

Appendix B
System Tables

Appendix C
Session Monitor Tables

Appendix D
DCB and Directory Formats

Appendix E
Modifying System Tables

List of Illustrations

Tables

1

System Management Overview

This chapter provides a procedural overview for planning, generating, and maintaining the HP 92084A RTE-6/VM Software System. RTE-6/VM is a powerful operating system that offers considerable flexibility in its operation and configuration. It supports program execution in background, time-shared, and batch environments. Multi-user interfaces are provided to manage concurrent user access to system resources. As the System Manager, you are responsible for system planning, generation, installation, and initialization as well as maintenance of the system after it is operational. The following steps should be performed:

1. Evaluate the system user base.
2. Select appropriate user interface: single user, MTM, or Session Monitor.
3. Plan the system architecture.
4. Generate the system, using the RTE-6/VM On-Line Generator.
5. Install and bootstrap the newly generated operating system.
6. Initialize the system and appropriate subsystems.
7. Maintain the system.

The whole process and corresponding references are shown in Figure 1-1. This chapter describes the major steps in system management.

Evaluation of User Base

You should be aware of user requirements before generating the system. The following information is determined prior to system generation:

- Who will be using the system?
- What applications will be run in the system?
- What system resources and peripherals will be required?

Further discussion of user evaluation is given in the Determining User Requirements section of this chapter.

Selecting the Multi-User Interface

RTE-6/VM provides two optional multi-user interface packages, Session Monitor (SM) and Multi-Terminal Monitor (MTM). SM or MTM (or neither) may be selected during system generation. The choice depends on user requirements and the capabilities offered by the appropriate package. Both packages allow multi-terminal access to the operating system. Throughout this manual, references will be made to SM or MTM. Either one (but not both) may be generated into the system for a multi-terminal or multi-user environment. Unless stated otherwise, this chapter applies to both interface packages.

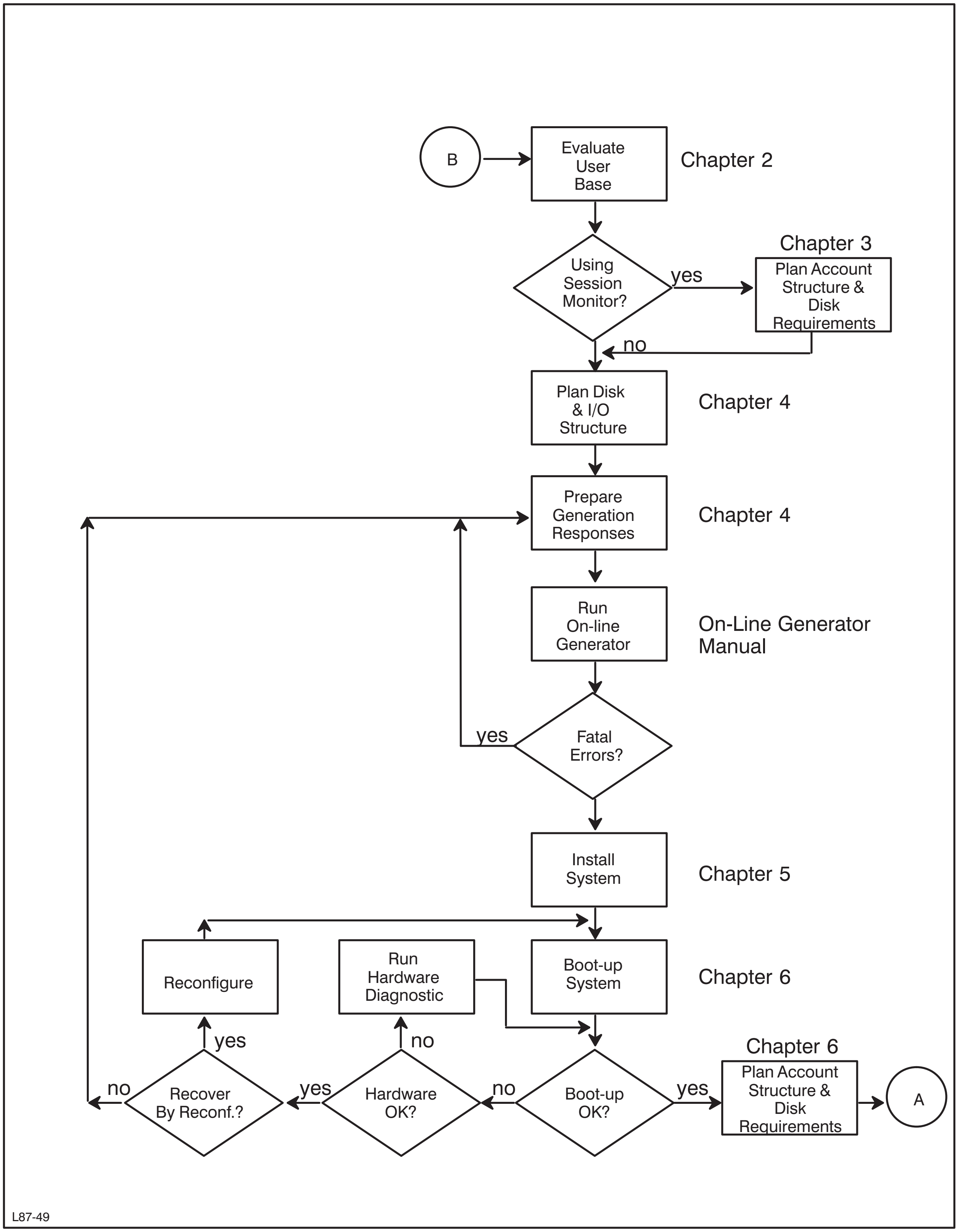


Figure 1-1. System Management Procedural Overview

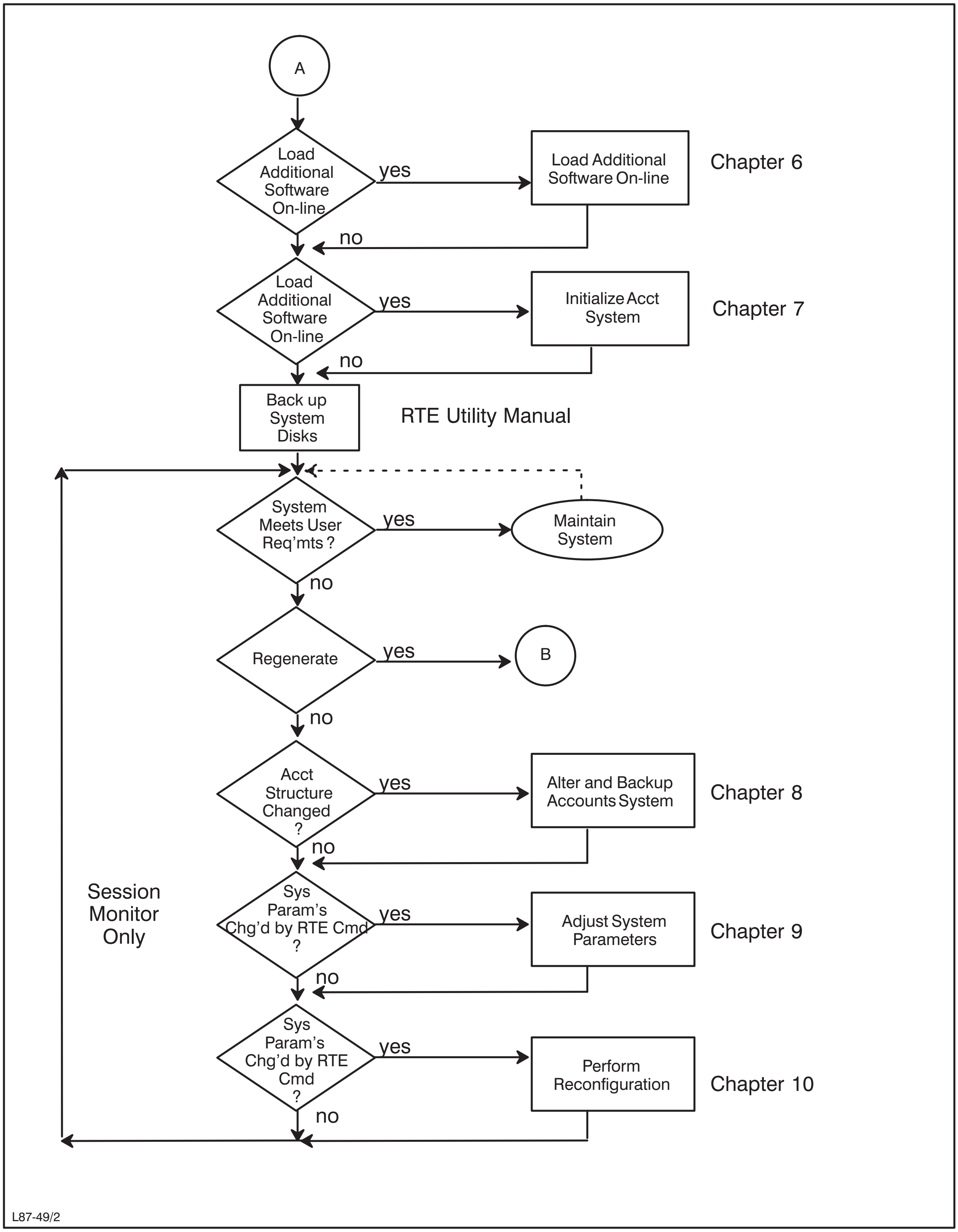


Figure 1-1. System Management Procedural Overview (Cont.)

Session Monitor

The Session Monitor facilitates multi-user system operation by providing protected file domains and controlled access to system resources and functions. Following are Session Monitor features:

- User activity on the system is defined in terms of sessions: a session user logs on, interacts with the system to perform a specific function, then logs off. A copy of the File Manager or the Command Interpreter (CI) is available as soon as log-on is successful.
- The user base is broken into two levels: groups (sets of users who share common functions, applications, and/or resources) and individual users.
- Each individual using the system must be assigned a group and user account name by the System Manager. The user must provide this information in order to log-on to the system. The account determines what resources and file cartridges can be accessed. A user may be assigned a password which must be provided in order to log-on to the system. This provides security for the account.
- A capability level is associated with each user. Each File Manager and operating system command has an associated capability level. Users must have a capability level greater than or equal to the command capability level to execute the command.
- User names, capability levels, and resource access information are stored in a system account file. This file is created using the system accounts program. The System Manager has control over the account structure and resource access.
- When the user logs on, the system builds a Session Control Block (SCB) for that user based on the account name and the terminal where the session is initiated. This Session Control Block (SCB) contains all the pertinent information about the user's session, including: cartridges mounted to that session, system resources (in terms of logical units) the user may access, the user capability level, and possibly other session information on the user and group.
- While a user is in session, the system prevents adverse interaction between sessions. This is accomplished by various means. For example: cartridges are mounted to specific users (or groups); only those users (or group members) may access them.
- Break Mode is entered when you cause an unsolicited interrupt by striking any key on the terminal. Session Monitor then reads a user command and process it, or if appropriate, send it to the operating system for processing. Only commands with capability levels less than or equal to the user's level are accepted.
- Permanent programs scheduled from File Manager are automatically copied for each user, permitting multi-terminal use of utilities and application programs.
- After a user has completed his session, he must "log-off." The system then updates the account file with the user's CPU and total session connect time and release system resources (e.g., memory for SCB) allocated for the session.

Multi-Terminal Monitor (MTM)

The Multi-Terminal Monitor (MTM) allows multi-terminal access to operating system and file manager functions. The major features are:

- Automatic scheduling of the File Manager. When your copy of the File Manager is dormant and you strike any key on the terminal, MTM schedules this copy to run from the terminal. This allows full access of the system.
- Break mode is entered when you cause an unsolicited interrupt by striking any key on his terminal. MTM then reads a user command and, if appropriate, sends it to the system for processing. You may issue virtually any system command from the terminal.
- Every user in the system has complete access to all system resources. Each user may access all file cartridge directories in the system and there is no automatic means to separate one user's file activities from another. Furthermore, every user may enter all possible system commands and accordingly adjust system parameters. Consequently, each terminal has all the power of the system console. Thus, users must agree among themselves to restrict their system activities to pre-defined domains. Since MTM allows full system access, little protection exists between users.
- The CI utility, the hierarchical file system, and related file utilities were designed to be executed within a session monitor environment. Under a non-session environment, ownership of directories, default working directories, and user-definable directory search paths are not available. In addition, CI and related utilities will only use the system console (system LU 1) for terminal input and output.

  Note that RTE and many subsystems require a hierarchical volume for updates and installation. Refer to Chapter 3 for more information on hierarchical volume requirements.

System Planning

Information obtained in the user base evaluation is used for system planning. Worksheets are filled in to prepare responses to the RTE-6/VM On-Line Generator. Steps involved in system planning include:

- Plan the session account structure and disk cartridge requirements (Session Monitor only).
- Plan disk subchannel assignments.
- Plan the computer I/O structure including setting up system LU, EQT, and interrupt table assignments.
- Allocate optional system resources such as number of classes, resource numbers, size of common, etc.
- Plan such items as RTE-6/VM I/O memory configuration including the size of partitions, and the number of partitions.

System Generation

Run the RTE-6/VM On-Line Generator to generate a system. This requires the generation chapter of this manual, the On-Line Generator Manual, and other appropriate documentation as required by your particular system.

System Installation

After the RTE-6/VM system has been generated, the RTE-6/VM SWTCH program must be run to place the generated system on the disk in the correct format. Refer to the Transferring the New Operating System Chapter in this manual for details. Backup your disk so that you always have a working operating system in case of trouble (i.e., planning errors).

System Initialization

Initialization of the system consists of the following steps:

- Initialize File Manager directories on system disk cartridges LU 2 and LU 3.
- Create the system WELCOM file.
- Bootstrap the new system. Run the reconfigurator to correct generation errors if necessary.
- Create the system message file.
- Create user welcome files (called HI and HELLO files).
- Load system and user utilities on-line if not generated into the system and make type 6 files for utility programs.
- Run a disk backup utility to save the copy of the system, using a magnetic tape unit or a cartridge tape drive if available.
- Start up appropriate subsystems, such as the spooling system.

For systems operating with the Session Monitor, perform these additional steps:

- Set up the user account structure by running the accounts (ACCTS) program.
- Initialize the appropriate system disk cartridges and CI hierarchical file volumes.
- Enable the system console as a session terminal (optional).

Maintaining the System

After the system is operational, you may alter various system definitions. For example:

- Alter or backup the Session Account System by running the accounts program.
- Save and restore disk cartridges and spare bad disk tracks using the disk backup formatting utility programs.
- Alter system parameters such as the time-slice quantum, device time-outs, and buffer limits.
- Reconfigure the system to meet new user requirements or change generation parameters.

2

Determining User Requirements

It is suggested that potential system users be interviewed to find out their needs. A sample user questionnaire is shown in Figure 2-1. This questionnaire is a guide. You should modify it to suit your specific needs. Most users do not think in terms of disk tracks, memory or disk-resident programs, or priority levels when describing their needs. The questions should be understandable and informative. You can then translate the data into information for system generation, initialization, and maintenance.

User Category

The first questionnaire section deals with the level of user sophistication. This section, applicable primarily to users of the Session Monitor, defines four levels of sophistication. The first level is that of a technician or data entry operator. Users in this group interface to the system only to the extent of operating specific programs or procedure files. No programming knowledge is necessary and little knowledge of the system is required. Users are expected to follow pre-defined procedures when dealing with the computer.

The next level of user sophistication is that of a secretary or word processor. Users at this level may require knowledge of the editor and cursory knowledge of the file system. Only limited access to system functions is needed.

The next level is for programmers. Most users of RTE fall into this category. They have knowledge of operator commands and programming calls. They are expected to use most system capabilities. This includes operation of compilers, managing data bases, manipulation of the file system, and performing network operations. However, they are not concerned with the activities of other users on the system. Furthermore, detailed system knowledge is not usually required.

Users of the highest level of sophistication include system programmers and support personnel. These users have a good working knowledge of system operation. They are capable of changing overall system operating parameters.

I. USER CATEGORY

- ☐ Technician/Data Entry Operator
- ☐ Secretary/Word Processing Operator
- ☐ General Programmer
- ☐ System Programmer/Support Personnel

II. APPLICATIONS

Subsystems ______________________________

EMA/VMA Programs

Size ________ Should EMA be shareable? Yes ☐ No ☐ With what programs? ________

Size ________ Should EMA be shareable? Yes ☐ No ☐ With what programs? ________

Size ________ Should EMA be shareable? Yes ☐ No ☐ With what programs? ________

Special Program Requirements: # of Classes ______________________________

# of SAM Required ______________________________

# of Resource Numbers ______________________________

# of Scratch Tracks ______________________________

Number of Programs Active at One Time ______________________________

Program Partition Sizes ______________________________

Real-Time Common Size ______________________________

Background Common Size ______________________________

III. PERIPHERAL RESOURCE USAGE

- ☐ Peripheral Hierarchical File System
- ☐ Using FMGR File System
- ☐ Private Cartridge Required:

CRN	SIZE	FREQ OF ACCESS
____	____	____________
____	____	____________
____	____	____________

- ☐ Common Data Base/File Access

  Users ________ ________

  ________ ________

- ☐ Line Printer Access
- ☐ Cartridge Tape Drive Access
- ☐ Magnetic Tape Unit Access
- ☐ Others: ________ ________ ________
- ☐ Special Requirements ______________________________

Figure 2-1. Sample User Evaluation Sheet

System Applications

The second section of the questionnaire deals with intended system applications. These applications are the primary source used to decide how to allocate system resources and how to set up system parameters. You should determine the following:

- Subsystems required. HP supplied subsystems, languages, utilities, and user application programs to be used on the system. When uncertain, it is recommended that as many subsystem or utilities as possible be included in the system.
- Response time requirements. Users should be queried as to their terminal and real-time response requirements. Based on their inputs, modules may get higher priority levels, generated into the system as memory-resident, or assigned to partitions. For example, in a real-time environment, response considerations may dictate that certain programs be memory-resident at all times. If this is the case, you must obtain these modules before generating the system.
- Memory requirements. If users run large application programs, partitions generated must be large enough to execute these programs. For some applications, HP supported subsystems require larger partitions for their execution (that is, compiling very large programs). Refer to the chapters on Transferring the New Operating System and System Initialization in this manual, for specific subsystem or utility memory requirements. Extended Memory Area (EMA) usage is another factor to consider. User application programs using the EMA feature require partitions of at least a certain size to be generated into the system. Therefore, users should be queried about the maximum EMA space used in application programs and whether the information in the EMA area is shared by more than one program.

Peripheral Resource Usage

The third section of the questionnaire determines peripheral resources required. The following information needs to be asked of each user (or group of users):

- Will the user be storing files or creating data bases on the system? If so, how many and how big? Does the user require disk space on a permanent or temporary basis? This gives an indication of the amount of disk space (if any) to be allocated to the user and of the disk LU (subchannel) sizes required in the system.
- Will the user's files be accessed by other users in the system? Will this user access other user's files? Which users? Does this user have files that cannot be shared? These questions are important in systems using the Session Monitor because file access can be restricted to the individual users, made available to all system users, or made available to members of a group (under FMGR only).
- Will user application programs require system scratch tracks? System scratch tracks (which are managed directly by the operating system on LU 2 and LU 3) are used in many system functions. If this usage is heavy, then a greater proportion of the system disk space should be allocated for scratch tracks when the system is initialized.
- Does the group require a special peripheral? For example, a peripheral may be necessary for one group's application, yet another group with a different application in the same system, may want to discourage its use.

- Will the users be using the screen editor (EDIT) for program development or other text processing? If so, adequate disk space should be made available to accommodate the scratch files created by EDIT. This can be done by ensuring that the users' private cartridges have enough space or by dedicating a disk volume for exclusive use of storing temporary editor scratch files. In the latter case, EDIT must create the scratch files on the disk LU specified at load time. This volume should contain the global directory /SCRATCH. Refer to the EDIT/1000 User's Guide for the editor loading information.
- Will the integrated Cartridge Tape Drive (CTD) be used? If so, reserve a buffer area on disk for the CTD. This area is referred to as the disk cache. Use of the disk cache for CTD operations greatly improves the data transfer rate.

3

Session Account Planning

This section helps you plan your session account structure and disk cartridge requirements.

To optimize the Session Monitor operations, certain steps should be taken before preparing generator responses:

- Determine intended system applications and resource requirements. This may be accomplished by interviewing users. Details are discussed in the Determining User Requirements chapter in this manual.
- Organize the user base into a hierarchy of groups and users. Groups should include sets of users with common characteristics or requirements. For example, groups might be composed of members of a project team or users performing similar functions. If desired, users can be members of more than one group. This organization should serve as the basis for the overall account structure. An account planning matrix worksheet is provided in this chapter to aid you in this process.
- Estimate the number and size of FMGR disk cartridges and CI file volumes in the system. This depends on your account structure, user application requirements, and the degree of file independence required by various users of the system. This chapter discusses the various ways disk cartridges and file volumes can be mounted in the system and how they are accessed in both session and non-session environments. You should have a thorough understanding of this information before planning your disk configuration. A requirements worksheet is provided to aid you in disk configuration planning.

The Session Environment

Session Concept

Before gaining access to the system, you must log-on by supplying the system with a user and group account name. The system then sets up a specific operating environment for you based on your private and group requirements, and the particular terminal at which you logged on.

After log on, the system will permit only those user peripheral access requests and commands allowed within the operating environment. In addition, users can access many of their peripherals with default logical unit numbers. This eliminates the need to know system logical unit assignments. For example, each user's terminal is referred to as LU 1 rather than by the actual system logical unit number assigned to it.

When finished interacting with the system, the session user logs off. The system updates its record of the cumulative CPU and connect times and clear its record of the user session from internal tables. The process of logging on, interacting with the system, and logging off is referred to as a session. With careful planning by the System Manager, each session should provide a secure, friendly, and productive software environment.

The System Manager may define the account structure such that users have varying degrees of access to system functions, files, and peripheral resources. These account definitions can be based on user applications, levels of sophistication, and other special requirements. The importance of good planning here cannot be over emphasized.

Session Control Blocks (SCB)

Every time a user logs on, the system allocates an area of memory called the Session Control Block (SCB) for that user. Session Control Blocks is created for all currently active sessions in the system. The SCB is the primary means used by the system to check user requests for validity and restrict access to system resources.

Included in each SCB is the following information:

- user command capability level.
- associated error information.
- a record of CPU usage and connect times.
- session user ID and group ID.
- the maximum number of disk cartridges that may be mounted at any one time.
- a record of all cartridges currently mounted in this session.
- Session Switch Table (SST).
- Current working directory.
- User-Definable Directory Search Path Tables.
- Primary Program name (FMGR or CI).

The complete Session Control Block format is shown in the Session Monitor Tables Appendix.

Session Switch Table (SST)

The Session Switch Table (SST) is part of the SCB and allows session users to reference peripherals associated with their operating environment via supplied session LU numbers called session LUs. When a peripheral is accessed, Session Monitor searches the SST to find the supplied session LU which points to a corresponding system LU. This allows the session user to access peripherals without requiring knowledge of system logical unit assignments. Also system logical unit numbers can be greater than 63, because several session LUs can point to the same system LU.

Every peripheral that the session user may access must be defined in his SST. This feature restricts the users access to a predefined subset of system peripherals.

Account Structure

The Session Monitor maintains two types of accounts: user accounts and group accounts. Group accounts are used to assign selected peripherals and disk cartridges to specific sets of users. User accounts provide the system with the information necessary to set up and maintain the operating environment for that user.

Every session user must be assigned at least one user account. The user account may specify which group account it is to be associated with. A user account can optionally include the resources assigned to its group account. If desired, you can assign an individual several user accounts belonging to different groups. These accounts can be structured such that the same set of private resources are retained in the user's operating environment irrespective of the group he is currently logged on with. Accounts structured in this manner are said to be linked.

The System Manager's account is treated specially by the system; it is given access to all system functions and resources.

All accounts are specified to the system in the form <user>.<group>. Where <user> and <group> are identifiers of one to ten characters in length. Within groups, the user identifier must be unique. An example account structure is shown in Figure 3-1. As can be seen from the example, the account structure is broken down into three levels: system manager, group, and user. Note that, in the diagram, Jones is a member of three groups and has three separate accounts. The System Manager has linked these accounts together so that Jones can access the same private files or peripherals from all three accounts. Note that linked accounts need not be the same user names.

In the CI file system, all files and directories can be protected from the point-of-view of the owner, members of the owner's group, and all others on the system. You determine the protection status of the files in your directories. For more information on protection, consult CI File Access Considerations in this chapter.

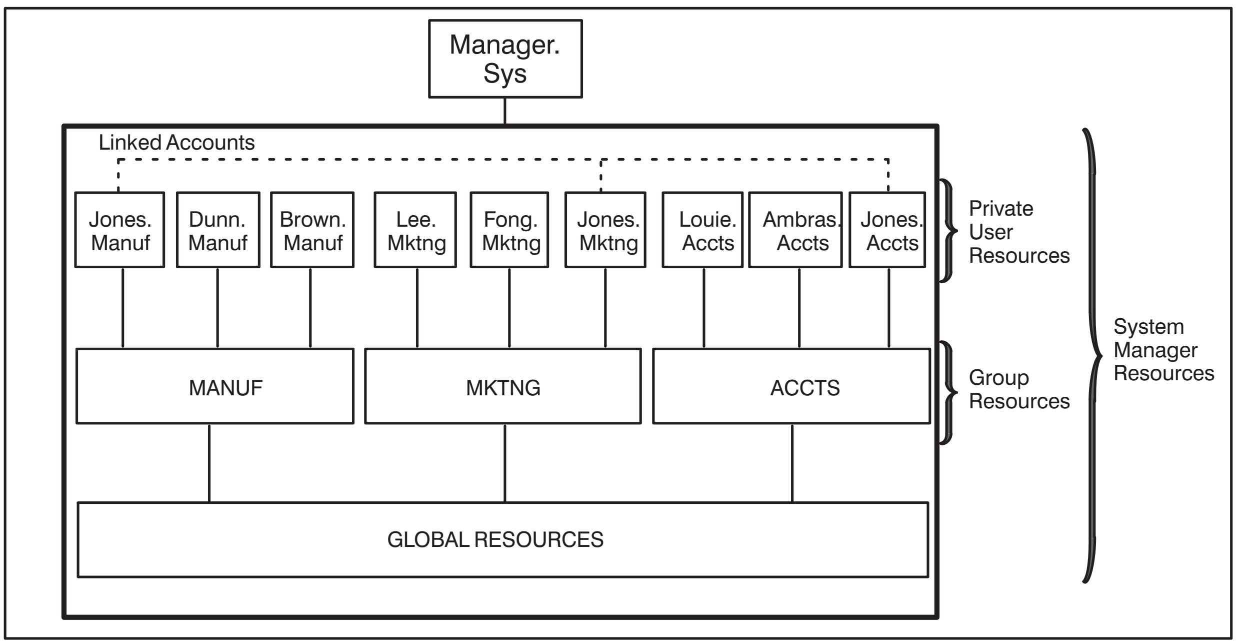


Figure 3-1. Sample Accounts Structure

Session Monitor Components

The Session Monitor consists of modules and subsystems that in total create a tailored operating environment for each user. The following components comprise the Session Monitor:

Account System
Session Account File
Break Mode Processors
Session Log-on/Log-off Processors
Operating System
File Management System
Command Interpreter (CI) File System

Account System

The account system is maintained by the accounts program (ACCTS). This program is run during system initialization by the System Manager and is also run automatically at boot-up to allocate system resources for the session monitor. It may be run at any time to maintain or examine the account structure.

Using ACCTS, you can do the following:

- Define all groups and their associated resources.
- Define all users and such items as their associated resources and capability levels.
- Specify the users primary program (FMGR or CI).
- Display the current status of the account system including all users currently logged on.
- Send messages to various users of the system.
- Shut-down and restart the entire session system.
- Save current account definitions off-line and restore them when desired.

System Initialization and Session Monitor Initialization Chapters describe the operation of the ACCTS program in detail.

Session Account File

The accounts file is created by the ACCTS program during system initialization. It contains four major sections: user account definitions, group account definitions, the Configuration Table, and system global information.

Each user account definition contains the user and group SST definitions, command capability level, log on command transfer file (HELLO file), mounted disk limit, user and group identifiers, and CPU and connect time.

The group account definitions contain additional SST definitions. These definitions are optionally placed in each group members user account definition.

In addition to user and group accounts, the account file contains global session monitor parameters and tables, including the Configuration Table.

The Configuration Table contains additional SST definitions for selected stations. In this manual, the term station refers to the session terminal and its associated peripherals. It is used interchangeably with terminal. When a user logs on, his SST will be a composite of his user account SST definitions, his group SST definitions, and his station Configuration Table SST definitions. For example, a terminal with cartridge tape units could have SST definitions equating default session LUs to their system LU assignments. Refer to the Session Monitor Initialization Chapter for details of the session account file.

Break Mode Processors

The break mode processors consist of two programs: PRMPT and RPN. These programs process break mode commands. For instance, they provide a means of interrupting program execution and examining system status, when a user's programs are running or his copy of the file manager is unavailable.

Session Log-On/Log-Off Processors

These processors operate in conjunction with the ACCTS and the break mode processors. The log-on processor, LOGON, creates active sessions in the system. It is invoked by PRMPT when no active session is operating from the terminal. LOGON prompts the user for an account name and password (if required). An attempt is made to match this name to an existing user account stored in the account file. If a match is found, a Session Control Block is created for the session, and the user's session is initiated. When a user has completed his session and logs off, the LGOFF processor updates the session account file with the CPU usage and connect time and deallocates system resources for the session.

Operating System

In the session environment, the operating system message processor MESSS executes only those commands having equal or lower capability levels than the level specified in the user's SCB. User capability level assignments are made by the System Manager when the user account is defined.

File Management System

The FMGR file system divides a disk into fixed-size cartridges that are identified with either negative LU numbers or positive cartridge reference numbers (CRN). The CRN can also be a two character string. Each cartridge has a cartridge directory containing pertinent information on all files stored on that cartridge.

All file manager commands have capability levels associated with them similar to operating system commands. You may perform only those file management commands whose capability levels are equal to or less than the level assigned to you. The file management system also restricts user access to disk cartridges. You may access only the cartridges that are specifically mounted to your session. Checks are made to restrict the number and type of cartridges that may be mounted to the session.

Command Interpreter (CI) File System

The CI file system divides the disk into large areas of free blocks. These areas are identified by LU numbers and are called disk volumes. Files in each disk volume are managed by directories and subdirectories which maintain information on the files. Table 3-1 shows a comparison summary of the two file systems.

The CI file system permits you to assign several users to one large disk volume, making the free space on the volume available to all users. Each user has a private set of hierarchical directories for file management. As listed in Table 3-1, CI provides time stamps for file creation time, last update and last access; it provides unpurge, and file names up to 16 characters long. File Manager, on the other hand, provides only one directory per disk LU, and the files on each LU must have unique names limited to six characters.

To take advantage of most of the enhancements of the hierarchical file system, your directories must be reformatted and programs must be modified. You can reformat directories by means of the FSCON utility included with RTE, or as part of a save/restore activity.

For FMGR directories that you do not reformat, CI accesses FMGR files, but without all the enhancements of the hierarchical file system.

Table 3-1. RTE-6/VM File Systems Comparison

	FMGR File System	CI File System
Filename	**1-6 characters**	**1-16 characters**
Cartridge/ directory	1-2 characters or numeric cartridge names	1-16 characters in directory names
File Security	Security code used for file protection	Protection based on directory (and file) ownership
File Types	Defines the structure of the files	File type extensions describing the contents of the files
File Mask	Mask qualifier and special characters in the file name	Mask qualifier and special characters in the file name
File Size	Extendable	Extendable
Time Stamps	None	Create, access, update times handled by the file system
Sub-directory	None	Subdirectories within directories and other subdirectories
File Recovery	None	Operator recoverable immediately after purge
Spooling	Can be done interactively and programmatically	Through FMGR (SST mapping)
Incremental Backup	None	Done in conjunction with FST or the TF utility

Planning Your Account Structure

User Accounts

Use the account planning worksheet to list all the individual users of your system. For planning convenience, you should assign a unique identifier (up to 10 characters) to each user. A sample account matrix is shown in Figure 3-2.

Users	Groups: Manuf	Mktng	Accts	
Jones	✓	✓	✓	
Dunn	✓			
Brown	✓			
Lee		✓		
Fong		✓		
Louie			✓	
Ambras			✓	

Figure 3-2. Sample Account Planning Matrix

Group Accounts

Once you have listed your system users, you should divide them into various groups. Members of a group will usually share one or more common attributes. Some of the criteria that may apply here are explained below.

EXISTING ORGANIZATION. You may find it convenient to follow an existing organizational pattern. Your account structure could reflect the actual groups in your user community.

COMMON FILES (FMGR). Users who must share files or data bases with each other can be included in the same group. FMGR disk cartridges can be associated with a group such that they may be accessed solely by members of the group. CI volumes and directories can be assigned protection in such a manner that group members have access to them.

COMMON PERIPHERALS. Groups can be formed around special peripheral access requirements. If desired, peripherals can be restricted to selected groups or users. These peripherals may be defined to the account system such that they are automatically added to the list of peripherals individual group members may access.

COMMON APPLICATIONS. You can separate users into groups based on their applications or job functions. Users performing similar tasks could then share related files and peripherals.

As a starting point in dividing your user community into groups, you might pattern the account structure after a group structure already existing in your user community. You might also want to form groups based on common peripheral or data base access requirements. Make a list of all such resources and the users requiring access to them. You should only form new groups when the list of users sharing a common resource is composed of users from two or more existing groups. If not (i.e., the list of users are all members of one existing group), add the resource to that group's domain. The information gathered here is used later on to initialize and maintain the account system.

Assign a name or an identifier, up to ten characters, to each group in your user community. This identifier must be unique. It is used by members of that group to identify themselves to the system. List each group in the diagonal group column in the account planning matrix, see Figure 3-2.

Next, indicate the members of each group. In each group column, place a check mark in all rows corresponding to the members of that particular group. Note that no restriction exists on the number of groups that a user may belong to. This may be a requirement in situations where individuals need to access resources owned by several different groups.

Disk Management

The following sections discuss how FMGR cartridges and CI file volumes are allocated and accessed. You should be familiar with this material before determining your disk cartridge and volume configuration.

FMGR Cartridge Types

Before users can access files, their associated cartridges must be mounted on the system. Cartridges may be mounted in one of four ways:

1. CARTRIDGES MAY BE MOUNTED TO INDIVIDUAL GROUPS. Cartridges mounted in this manner may be accessed by all members of the same group. For instance, group cartridges allow members of the same group to share programs, data bases, and information files.
2. CARTRIDGES MAY BE MOUNTED TO PRIVATE USERS. When a cartridge is mounted to a private user, only that user (or other users linked to his account) may access that cartridge. Private cartridges permit file security and are designed to prevent users from inadvertently accessing each others files.
3. CARTRIDGES MAY BE MOUNTED TO THE SYSTEM. These cartridges, known as system global cartridges, are accessible to all users of the system. The primary system cartridge (LU 2) and the auxiliary cartridge (LU 3) are always mounted to the system. These cartridges contain the system memory image code (LU 2) and system swap tracks, as well as an area for files. Files on LUs 2 and 3 are subject to special access restrictions (described later). You may mount additional cartridges to the system by mounting them as private cartridges to the MANAGER.SYS account. With the exception of LU 2 and LU 3, files residing on System Global cartridges may be both read from and written to by any user of the system.
4. CARTRIDGES MAY BE MOUNTED OUT OF SESSION. These cartridges can be accessed by programs not under the control of the Session Monitor. Non-session cartridges are mounted from a FMGR operating in non-session mode (e.g., from the system console). Note that in most cases LUs 2 and 3 are system type cartridges. They may, however, be made non-session cartridges (with the DC command). In this case they may be neither read nor written on by session users.

It should be emphasized that all file manager cartridges have the same format. The method cartridges are mounted determines the access restrictions imposed on them. For example, one user may mount a cartridge to his group (e.g.,:MC,−25,G), use it, then dismount it (:DC,−25,RR). Another user may mount the same cartridge to his private account (:MC,−25,P). The cartridge and its contents might remain the same; only the list of users who could access it would change. If necessary, you may permanently dedicate certain cartridges to groups and users. This depends on how your account system is set up.

Spare Cartridge Pool

In many cases, private users and groups may not need to have cartridges permanently allocated to them. They may need use of the cartridge disk space for only relatively short periods of time. The Session Monitor recognizes this need by maintaining a spare cartridge pool. This pool consists of cartridges to be allocated when users request scratch private and group cartridges (that is, :AC,crn). When dismounted from the system, the scratch cartridges are returned to the spare cartridge pool. The cartridge pool may be setup when the account system is initialized or altered.

Cartridge Mounting Considerations

A cartridge is defined to be mounted to a session when it is defined in the system cartridge list as being mounted to that session's group or private account and the cartridge LU is defined in the session's SST and SCB cartridge list.

At log-on, the following cartridges are automatically mounted to your session:

- All cartridges in the system cartridge list that are mounted to your private account or to other private accounts linked to your account.
- All cartridges in the system cartridge list that are mounted to your group account.
- All cartridges in the system cartridge list currently mounted as system global cartridges. This includes system cartridges LU 2 and LU 3, and all CI file system volumes. Note that LU 2 and LU 3 are not included if they have been changed to non-session cartridges.

The total number of private and group cartridges mounted to your session at any one time is controlled by a parameter in your account definition. This parameter, called the disk limit, is included in your SCB when you log-on. If you attempt to mount more private and group cartridges to your session than is permitted by this limit, you receive an error message. The system also warns you of this condition at log-on if more cartridges can be automatically mounted to your session than is permitted. System global cartridges are not included in your disk limit.

When a disk cartridge is mounted to a session with an MC command, the cartridge LU must be predefined in your SST. This insures that low capability users (i.e., those users not able to modify their SST with the SL command) are only able to mount cartridges defined in their account. Once the cartridge is mounted, it appears in the system cartridge list as belonging to the user's private or group account, depending on MC command parameters. Note that if a previously uninitialized cartridge is mounted in session, the MC command initializes it. The system does not allow you to mount a cartridge to your session if it is already mounted to some other group or private user account or as a non-session cartridge.

When scratch cartridges are requested with the AC command, the system first checks to see whether a cartridge with the requested CRN is already mounted to your group or private account or as a system global cartridge. If so, this cartridge is merely added to the users session. Otherwise, the spare cartridge pool is searched for an unmounted cartridge of (at least) the requested size. Cartridges are searched in the order defined when the accounts system is initialized (using ACCTS). The first cartridge that meets the size requirement will be allocated from the pool. (Note that this is not necessarily the best fit). If none can be found, an error is issued and no further action is taken. Otherwise, the cartridge is initialized according to the parameters in the AC command. All files previously stored on that cartridge are purged.

Note that cartridges mounted privately to the MANAGER.SYS account will appear in the system cartridge list as system global (S) cartridges. Their CRNs must be unique to all the cartridges in the system cartridge list.

Cartridge Dismounting Considerations Under FMGR

Cartridges in the spare cartridge pool are intended for short term use. At the end of their session, it is suggested that users back up these cartridges on magnetic tape (using WRITT or FC) and dismount them from the system. This returns the disk space to the available disk pool. Once dismounted from the system, files on a pool cartridge may not be recoverable. The cartridge is completely reinitialized (i.e., all files purged) when reallocated with an :AC command; you can recover files by mounting the cartridge by LU (that is, :MC,LU) before it is reallocated.

Dedicated cartridges are allocated to users on a longer term basis. They are assigned to users and/or groups by including the cartridge LU in their respective account definitions. Dedicated cartridges should be dismounted from the system only when they need to be physically removed or transferred to different accounts. This prevents unauthorized access and causes the system to automatically mount these cartridges to the users session at log on.

CI Hierarchical File Volumes

A volume is a self-contained section of a disk. Each volume is independent of any other volume; files or directories never cross volumes. Each physical disk drive consists of one or more volumes; volumes never cross physical drives. Each volume is identified by a logical (LU) number. Volumes are always identified by their disk LU number. The unit range of volume LU numbers is 7 to 63.

Each volume contains a unique set of information about which files are included. This information includes the names of all the global directories on the disk, as well as a table that tells which disk blocks have been allocated to files. This table is called a bit map, because the table is composed of bits rather than addresses or values.

Typically, a CI volume is considered a free pool of disk space available to all users. However, if necessary, the ownership and protection of a volume can be defined so that access to the volume is restricted to a specific user or a specific group of users.

Common operations performed are mounting a volume, dismounting a volume, and listing contents of a volume. An operation that is not commonly performed is initializing a volume, making it ready for system use.

Directory Organization

Directories are the central CI file system data structure. Directories maintain the file system state across system halts and crashes. All information pertaining to a file is kept in a directory. Directories may be included in other directories; these are considered subdirectories. Nesting of subdirectories is allowed to provide a hierarchical file system structure. At the top is a root directory that contains only unique global directories. There is one root directory per disk volume.

Mounting a disk volume makes directories on that volume accessible. All global directories on that volume are made known to the system when a volume is mounted. Global directory names must be unique in the file system. Files in a directory are not identified by disk volume.

An abbreviated form of the directory name is kept in free space in D.RTR; each global directory requires five words. For this reason, a limit of 300 global directories is recommended. The size of D.RTR can be increased to expand the limit but the space available for open flags (described below) is reduced. This limit applies only to global directories and not to subdirectories. Therefore, convert global directories into subdirectories (with the move command) if the 300 limit is reached.

Each directory consists of a doubly linked set of disk blocks containing entries for files, extents and subdirectories. Each of these entries is 32 words. In addition, 32 words at the beginning and end of each directory are used for bookkeeping information. Directories are extendible, although performance is improved when they are kept to a small number of extents; 32 words are also needed at the beginning and end of each extent of the directory.

Each file is kept as a single file entry in a single directory. The file entry contains the filename, information about the disk space used by the file, and other information such as time stamps and protection.

Purged files are flagged by a bit in the directory entry. When a file is purged, the directory entry is marked as purged, and its disk space is marked as free. The file can be unpurged until its directory entry or disk file space is allocated to a file or extent. Note that it is possible that a purged file will have its space allocated to a new file, then the new file will be purged; now the first file appears to be recoverable but actually is not. This situation is rare, but it can be confusing when it does occur.

Files are extended when necessary to hold additional data. Sequential files always have extents at least as large as the main file, but the extent size will be doubled for extent number 4, 6, 8 and so on. For example, the first 10 extents of a 1-block file will have size 1,1,1,1,2,2,4,4,8,8 blocks. This prevents having files with large numbers of extents, which slows access. Random access files always have extents the same size as the file did when it was created, but there can be missing extents. Files with missing extents are known as sparse files, and serve to conserve disk space. VMA backing store files are the only common type of sparse file because no disk space is required for records that are never accessed.

SYSTEM Directory

The SYSTEM directory should contain the system snapshot file, the NODENAMES file (for DS transparency), the system error files for MACRO/1000 and Debug/1000, and the help files for EDIT/1000.

HELP Directory

The HELP directory is designed to contain text files (type 3 or type 4 files) created as a quick operator reference. These files are named according to whatever keyword is desired in order to access them. For example, for help on the WH command, the user might typically enter ? wh from CI. This causes CI to look in directory HELP for a file named WH. If found, the contents of the file are displayed at the screen. Therefore, you can create your own help files by storing them in the HELP directory.

LIBRARIES Directory

This is the directory which is automatically accessed by the program LINK when relocating a program. The LIBRARIES directory should contain all libraries that might be required in order to accomplish a program load. At a minimum, the libraries specified at generation time to be searched whenever a program is linked should be in this directory. Other libraries may also be included here.

CATALOGS Directory

The CATALOGS directory is where certain programs keep message catalogs. FST and TF use this directory.

PROGRAMS Directory

This directory is where executable programs reside, and these programs should have the file type extension RUN. This directory is automatically searched for programs such as DL, SP, IO, and WH. For example, when DL is entered from CI, CI searches for /PROGRAMS/DL.RUN.

CMDFILES Directory

If CI fails to find a given name in the PROGRAMS directory, it looks for a directory named CMDFILES, and searches the directory for a command file by that name.

SCRATCH Directory

The SCRATCH directory is a convenient location for programs that need a work area on the disk. Use of the SCRATCH directory is not mandatory; however, if the working directories are write protected, the SCRATCH directory can be used for work files for programs such as EDIT. You should create the SCRATCH directory on a disk volume with a large amount of free space. Use the CI PROT command to allow everyone to write access to the directory.

CI File Volume Mounting Considerations

Mounting a volume makes that volume and all the files on it available to the operating system. Dismounting a volume removes that volume and makes the files on it inaccessible to the system. These operations are not performed frequently except with removable media such as floppy disks, where disks must be mounted after they are installed and dismounted before they are removed.

To mount a volume, enter:

```
CI> mc <LU>
```

where lu is the LU number of the disk volume to be mounted. This must be a positive number.

For example, to mount a volume with disk LU number 12:

```
CI> mc 12
```

If the disk volume does not have a valid FMP or FMGR directory, you are prompted before the volume is initialized. This is to avoid the accidental corruption of volumes that are not of the FMP or FMGR types (for example, backup utility volumes). If the disk volume does have a valid FMP or FMGR directory, the volume is initialized.

The order in which disk volumes are mounted is unimportant. The only exception is when duplicate global directory names appear on two or more volumes. If a global directory on the newly mounted disk volume has the same name as a previously mounted global directory, the new directory is inaccessible.

The MC command does not place reserved blocks at the beginning of the volume. If reserved blocks are required, use the IN command.

Mounting a volume will initialize it if there is no valid data on the volume. Initializing a volume sets up information needed by the operating system, including the list of directories and the bit map for keeping track of space use. For more information on CI file volumes refer to the RTE-6/VM CI User's Manual (92084-90036).

FMGR File Access Considerations

Once a cartridge is mounted to a users session, files on that cartridge may be created, read from, modified, and purged. However, files on LU 2 and LU 3 are subject to the following access restrictions:

- All files on LUs 2 and 3 may be read by all system users.
- Users may create type 6 files on LUs 2 and 3 via file manager SP commands. A type file 6 may be purged only by the specific user who created it.
- When operating from the MANAGER.SYS account, the System Manager is given complete access to all files on LUs 2 and 3. The Manager may read, write, modify and purge all files, including type 6 files.
- When operating outside the session environment (i.e., from FMGR on the system console when not enabled as a session terminal) users are given complete access to all files on LUs 2 and 3.
- When users invoke transfer files residing on LUs 2 and 3, the commands within those transfer files are given complete access to all files on LUs 2 and 3.

The System Manager, when operating in the MANAGER.SYS account, is given complete access to all cartridges mounted on the system. This includes complete access to files on LUs 2 and 3, non-session cartridges, and cartridges mounted to group and private accounts. To access these cartridges, place the cartridge LU(s) in your SST. References to that cartridge should be through LU rather than CRN (the CRN might not be unique to your session). For example, to get a directory list of a cartridge mounted to some other user (assuming you are in the MANAGER.SYS account), enter the following commands:

```
:SL,30,30
:DL,-30
```

Note LU 30 is not, and cannot be, mounted to your session.

Non-Session File Access Considerations

Programs operating outside the session environment (i.e., without a SCB), may access system global cartridges and those cartridges mounted outside the session environment (non-session cartridges). They are given complete access to files on these cartridges, including LUs 2 and 3. Non-session programs, however, cannot access mounted session group or private cartridges. Likewise, programs operating under session control cannot access non-session cartridges. Since system global cartridges can be accessed in both non-session and session environments, they may be used for file sharing on a system wide basis. For example, programs operating in the non-session environment performing data communication or acquisition functions could update files on a system global cartridge. Session users would then have full access to these files.

CI File Access Considerations

Ownership of files is determined by the owner of the directory containing the files. All files in a directory are considered owned by the directory owner. The owner can change the protection status of files in that directory, which defines the read/write access allowed for the owner, for members of the owner's group, and for general users.

The owner of a CI volume is initially MANAGER.SYS, but ownership can be reassigned to another user, if desired. The owner of a volume can reassign ownership and define volume protection.

The owner of the directory (and no other users) can also reassign the directory ownership. The owner of a directory is the user who created it. The same is true for subdirectories. However, the owner of a subdirectory can be different from the owner of the directory that contains the subdirectory.

File protection is a security measure in the CI file system and is defined when a file is created or copied into a directory. It is specified in terms of read or write access allowed. It can be specified differently for owners, for group members, and for general users. The default protection is to allow the owner both read and write accesses and read access only for all other users.

Each file created assumes the protection status defined for the directory where the file resides. A copied file assumes the protection status defined for that file if one exists; if not, it assumes the protection of the directory where the file is copied into.

Disk Planning

The following sections help you to estimate disk cartridge and CI file volume requirements and plan your cartridge/volume configuration. To perform this planning function you should:

1. Estimate the size of your primary and auxiliary system disk subchannels (LUs 2 and 3). Determine the number and size of globally accessible cartridges in your system. The number of tracks on LUs 2 and 3 combined, cannot be greater than 1600, however, LU 2 must be made large enough to contain the operating system.

2. Determine the number and size of non-session cartridges.

3. Determine the number, size and allocation of cartridges that will be dedicated to users and groups operating in the session environment.

4. Determine the number and size of hierarchical file volumes that will be needed by users in the CI environment. A minimum disk volume of 25 megabytes is recommended for the standard hierarchical directories and for the RTE-6/VM software. If you have subsystems to install, you may have to increase the size of this volume or provide an additional hierarchical volume for the software.

5. Determine the number and size of cartridges in the spare cartridge pool.

Cartridge Requirements Worksheet

The cartridge requirements worksheet is provided to help you allocate your disk space and assign cartridges to various users. The information gathered here will be used in Chapter 3 when you determine your disk subchannel layout and in Chapter 6 when you initialize the session account system. An example worksheet is shown in Figure 3-3.

Before you start filling out this worksheet, find out the type and number of disks you have. If you will be using 9895, ICD, or MAC disks, the size will be specified in terms of number of tracks available on the disk. CS/80 disks are block oriented and their size is specified in terms of blocks. Using the total number of tracks or blocks as a base, start allocating cartridge space for your system global, non-session, dedicated group, dedicated private, hierarchical file volumes, and disk pool cartridges. Circle the intended use of the cartridge alongside the cartridge size allocation as shown in the example. As you fill out this worksheet, the right most column should reflect the number of unassigned tracks or blocks remaining on the disk at that point. When you finish filling out the worksheet this value should be zero.

As can be seen from the sample worksheet, the System Manager allocated the disk as follows:

- Three system cartridges are dedicated: for the system cartridge (LU 2), an auxiliary system cartridge (LU 3), and an additional system global cartridge.
- One non-session cartridge is dedicated that will be used for non-session programs.
- Three cartridges were dedicated to groups: one cartridge each is dedicated to the MKTNG, ACCTS and MANUF groups.
- Two cartridges dedicated to private users: JONES, and DUNN.
- The next two cartridges comprise the spare cartridge pool. These will be mounted to groups or private users on an as-needed basis.

- Two disk volumes are dedicated, which CI hierarchical files will use.

The following sections discuss some of the requirements and special considerations to be taken into account when planning each of the various cartridge types. Use the information provided in these sections along with your account planning matrix, and your general knowledge of the user base, to fill out your cartridge requirements worksheet.

Note Use the cartridge sizes specified in your cartridge requirements worksheet as a guideline for your disk subchannel layout (discussed in Chapter 4). The actual cartridge sizes will depend on additional considerations such as disk type, spare track allocations, and the physical layout of subchannels on your disk.

Cartridge and Volume Size Requirements

There are many criteria for selecting the size of a cartridge or volume. Among the factors you should take into consideration are:

- The amount of program development being done. Many users will require relatively little disk space in order to hold source files, documentation, relocatables, etc. On the other hand, if the project is large, or many versions must be kept on disk at the same time, you want to allocate more tracks to the users cartridge or the appropriate disk volume.
- The amount of word processing functions being done. If the user or group is storing large text files more tracks should be allocated.
- The number and size of IMAGE data bases/data sets, if used. Refer to the IMAGE Reference Manual for the information needed to calculate data base sizes.
- The number and size of data files.
- The number and size of virtual memory programs.
- The number of potential users of the cartridge or volume. You might multiply the intended number of users by some size constant to give a rough estimate of the optimal size.
- Approximately 3% of each subchannel on a MAC or ICD disk should be reserved for spare tracks. In the sample cartridge requirement worksheet for these disks, the number of tracks per cartridge takes into consideration the fact that several tracks will be used as spares. For CS/80 disks, there are areas available for sparing bad areas.

CARTRIDGE REQUIREMENT WORKSHEET FOR 9895, ICD, AND MAC DISCS

AVAILABLE TRACK SPACE ON DISC CONTROLLER 1644

CARTRIDGE TYPE	USER	#TRACKS	#TRACKS LEFT
S G P D N	Primary System Subchannel	200	1444
S G P D N	Auxiliary System Subchannel	100	1344
S G P D N	System-Wide File Sharing	100	1244
S G P D N	Non-Session Applications	50	1194
S G P D N	Mktg Group	50	1144
S G P D N	Accts Group	50	1094
S G P D N	Manfg Group	100	994
S G P D N	Private User Dunn	50	944
S G P D N	Private User Jones	50	894
S G P D N	Private User Packard	50	844
S G P D N	Pool Cartridge #1	25	819
S G P D N	Pool Cartridge #2	25	794
S G P D N	CI File Volume #1	600	194
S G P D N	CI File Volume #2	194	0
S G P D N			
S G P D N			
S G P D N			
S G P D N			
S G P D N			
S G P D N			
S G P D N			
S G P D N			
S G P D N			
S G P D N			

S = System Cartridge;
G = Group Cartridge;
P = Private Cartridge;
D = Disk Pool Cartridge;
N = Non-Session Cartridge or Hierarchical File Volume

Figure 3-3. Sample Cartridge Requirement Worksheet

CARTRIDGE REQUIREMENT WORKSHEET FOR CS/80 DISKS

AVAILABLE BLOCK SPACE 109824

CARTRIDGE TYPE	USER	#TRACKS	#BLOCKS/ TRACKS	#BLOCKS REMAINING
S	Primary System Subchannel	300	48	95424
S	Auxiliary System Subchannel	500	48	71424
S	System-Wide File Sharing	200	48	61824
N	Non-Session Applications	100	48	57024
G	Mktg Group	100	48	52224
G	Accts Group	75	48	48624
G	Manfg Group	125	48	42624
P	Private User Dunn	50	48	40224
P	Private User Jones	50	48	37824
P	Private User Packard	50	48	35424
D	Pool Cartridge #1	100	48	30624
D	Pool Cartridge #2	88	48	26400
N	CI File Volume #1	300	48	12000
N	CI File Volume #2	250	48	0
S G P D N				
S G P D N				
S G P D N				
S G P D N				
S G P D N				
S G P D N				
S G P D N				
S G P D N				
S G P D N				
S G P D N				

S = System Cartridge;
G = Group Cartridge;
P = Private Cartridge;
D = Disk Pool Cartridge;
N = Non-Session Cartridge or Hierarchical File Volume

Figure 3-3. Sample Cartridge Requirement Worksheets (cont.)

System Global FMGR Cartridge Requirements

System cartridges include the FMP cartridges on the primary and auxiliary system disk subchannels (LUs 2 and 3) and all other cartridges mounted to the MANAGER.SYS account.

The first area of the primary system subchannel will be used to contain the operating system, and its size will be determined by the number of modules included in the system at generation time. This Figure is displayed by the On-Line Generator when it has completed the system generation process. The remainder of the subchannel must then be divided at system initialization into the system scratch track and FMP areas. You may want to put LU 2 and LU 3 on fast disks because editing, program swapping, and other system related works occur on these disks.

The auxiliary system subchannel (LU 3) is used for additional FMP files and system scratch tracks. The auxiliary system cartridge is optional. Its use is recommended if your system will be heavily loaded (to provide additional swapping tracks) and/or you will be requiring additional FMP area. The combined size of LUs 2 and 3 may not exceed 1600 tracks.

You may wish to dedicate space for additional global cartridges in your system. These cartridges can be used for file sharing on a system wide basis. In particular, global cartridges are very useful in cases where files must be shared by different groups or by both session and non-session programs. Global cartridges should also be used for message and spool files. Even if you can foresee no requirement for global cartridges at this time, it is recommended that you dedicate one disk cartridge for this purpose. This will give users in different groups a means to share files with one another when the need arises.

Non-Session FMGR Cartridge Requirements

Certain applications require that programs be run outside of the session environment. Programs in the time list, or that operate continuously, should not be associated with a session since they will be terminated by the system whenever the session user logs off. Programs will operate in the non-session environment when they are dispatched from the system console in non-session mode or by detaching themselves from their session using library calls. These programs may then access system and non-session cartridges. You should determine the disk storage required by programs and subsystems operating in the non-session environment. If desired, this storage can be partitioned into more than one cartridge to isolate different non-session subsystem files from each other. If you decide not to allocate cartridges for this purpose, non-session programs will use the file space on LUs 2 and 3 and the other system global cartridges in your system.

Group and User FMGR Cartridge Requirements

Depending on user requirements, you should dedicate cartridges to various groups and/or private users. Cartridges are allocated to users by defining the cartridge LU(s) in their group or user account SST definition. These dedicated cartridges should not be specified in any other account SST (unless the cartridges will be traded between different group and private accounts). Chapters 6 and 7 describe the account definition process in detail.

It may be advantageous to allocate at least one cartridge to each group. This will be used by group members to share files with each other and to save information on a permanent basis. When required, users can allocate additional disk space for themselves by requesting cartridges from the spare cartridge pool. This space should be used on a temporary basis and will be returned to the pool when dismounted from the system. If possible, users should save files accessed infrequently on magnetic tape using WRITT or FC so that pool cartridges will be available for other uses.

Since disk space in the spare cartridge pool can be allocated to users as their needs and requirements dictate, it is strongly recommended that disk cartridges should be allocated from the disk pool. Cartridges should be dedicated to users only in the following situations:

- GROUP CARTRIDGES. It is recommended that all group cartridges be dedicated rather than from the spare cartridge pool. When allocated from the pool, the group runs the risk of losing all files on that cartridge if it is inadvertently dismounted from the system by a group member.
- CARTRIDGES ON USER REMOVEABLE DISK MEDIA. If you have users who will be inserting and removing private disk packs from the system, the cartridges on those packs should be dedicated to those users. They must NOT be included in the spare cartridge pool.
- USER TRADED CARTRIDGES. Cartridges that may be mounted to different private users or groups at different times, should be dedicated, when files on those cartridges need to be preserved.
- PRIVATE CARTRIDGES CONTAINING FREQUENTLY ACCESSED FILES. It is recommended that a cartridge be dedicated to a user when he will be accessing the same files on a long term basis and does not wish to place these files on a group or global cartridge. This will free the user from the risk of losing his files if the cartridge is inadvertently dismounted and returned to the spare cartridge pool. Examples of applications which might fall in this category are long term data base access, large word processing functions (e.g. documentation development), long term program development projects, etc.
- SPECIAL PURPOSE CARTRIDGES. Users requiring specific cartridge sizes or storage on specific disk areas should have those respective cartridges dedicated to them. For example, when virtual memory is being used, a cartridge large enough to hold the backing store file should be mounted to the user. The disk pool should not be used here since the location of a cartridge cannot be guaranteed and/or a cartridge of the required size may not be presently available.

Spare Cartridge Pool

After you have allocated space for your system global cartridges, non-session cartridges, and your dedicated group and private cartridges, the remaining space on your disk should be divided into cartridges for your spare cartridge pool. These cartridges will be allocated to private users and groups on an as-needed basis. The number and size of the cartridges in your pool will depend on several factors, including:

- Disk storage requirements of users who will be accessing cartridges from the cartridge pool.
- The estimated number of disk pool cartridges that will need to be mounted to private users/groups concurrently.
- The size of the remaining disk area. Since disk storage needs are sometimes difficult to anticipate, it is suggested that there be a broad spectrum of cartridge sizes in the pool. One way to plan the pool is to divide half your total remaining disk space into relatively small cartridges (say 50 to 100 tracks). These cartridges can be used by users to save a relatively small number of temporary files. Divide the other half of your disk pool area into successively larger cartridges. These cartridges can be used by individuals requiring relatively large amounts of disk storage. The allocation of cartridges from the spare pool will depend on the order of cartridge LUs in the pool (specified during accounts setup) and the parameters specified by the user in his allocate cartridge (:AC, CRN) command.

Meeting Changing Cartridge Requirements

System disk storage utilization is a dynamic variable and will vary as user applications and levels of sophistication change. When a user or group of users runs out of cartridge space, you can accommodate their needs via several means listed in order of preference.

- Dedicate a cartridge out of the spare cartridges pool just for their use. This will mean altering the definition of the spare cartridge pool and their account SST with the system accounts program ACCTS.
- If possible, allocate additional cartridges to that user.
- Trade cartridges with another user/group to obtain a larger cartridge. This will require exchanging the files on those cartridges and modifying affected user accounts.
- Increase the size of their cartridge(s) by regenerating the system. Certain applications will require all files to be on the same cartridge. If the cartridge can no longer accommodate all these files, you may be forced to regenerate the system specifying a new disk subchannel mapping. This has the disadvantage, however, that files on LU 3 and other cartridges affected by the subchannel redefinition will have to be saved before the new system is installed.
- Generate additional disk storage space (and corresponding disk subchannels) in to the system. Obviously, this will require adding more disk storage units to your system.

4

System Generation Response

This chapter will aid you in preparing specific responses to the Online Generator for the HP 92084A Operating System. It should be used in conjunction with the *RTE-6/VM Online Generator Reference Manual*, part number 92084-90010, and other appropriate configuration documentation such as subsystem configuration manuals. The system generation process can be broken down into the following steps:

1. DISK STRUCTURE PLANNING. Determine your disk subchannel configuration. If you have Session Monitor in your system, use the cartridge requirement worksheet (filled out in the previous chapter or this manual) as a basis for planning your disk.
2. I/O CONFIGURATION PLANNING. Determine the select code, LU, and EQT entry assignments for the devices in your system. Procedures are given in this chapter to aid you in planning your I/O configuration.
3. PREPARE GENERATOR RESPONSES. Prepare responses to the Online Generator by filling in associated worksheets. Generator responses are explained in this chapter in the context of these worksheets.
4. GENERATING YOUR SYSTEM. Running the Online Generator (RT6GN) to generate your system.
5. SYSTEM BACKUP. Backup your newly generated system.
6. INSTALLATION. After all these procedures have been followed and you are confident that your operating system has been properly generated, install the new system by following the procedures described in the Transferring the New Operating System chapter of this manual.

Note to the New User

Keep your first attempt at system generation as simple as possible. One of the major features of RTE is its flexibility and adaptability which is accomplished by allowing the user many options at generation time. This feature can be a mixed blessing to the new user as all the options lend complexity to the process. The best and quickest way for a new user to start is to build an answer file by modifying the sample answer file provided on the primary disk to suit your application. Follow the recommended generation guidelines unless you have specific requirements which cannot be met. Since most subsystems require additional steps, it is suggested that you exclude all non-standard subsystems (that is, subsystems not included in the RTE-6/VM product) from your first generation. An overview of the generation process is given below. Starting this way will guarantee a first success, help build your experience and allow more complex generations to be done easily.

After you have prepared your worksheets, run the Online Generator following the procedures described in the Online Generator Manual. You may wish to compare the information on your I/O Configuration Worksheet with your actual generator inputs.

When you are satisfied that the generation is correct,

Note **Back Up Your Disk!!**

This is very important! Do it before going any further! You must always be sure you can get back to a working system if a mistake was made and not caught. The primary disk shipped with your system contains the software you will need to generate all systems in the future, and must not be overwritten! Refer to the *RTE-6/VM Utility Programs Reference Manual*, part number 92084-90007, for disk backup information.

If you back up your disk by copying it to another disk, be sure that you do not overwrite any part of the factory-generated disk.

There is an alternative to disk backup, which is useful only if your disk drive has a fixed platter. Initialize the fixed platter using the FMGR and copy your newly-generated system file to that platter. Then, at the point in the switchover procedure where you are directed to place the proper disk cartridge in the disk drive, remove the factory-generated cartridge and place another one in the drive, making sure that the drive has come up to speed before continuing the switchover process. The SWTCH program will then copy to the new disk, without destroying data on the factory cartridge.

Another technique, which also requires a fixed disk surface, is to modify the generation answer file so that the system is generated to run on the fixed platter. Then SWTCH can be run to install the generated system to the fixed platter.

If the factory-generated disk is to remain in the disk drive, ensure that it is protected by means of the hardware protect switch. Then transfer the new system as described in the Transferring the New Operating System Chapter of this manual.

Set the factory-generated cartridge aside in a safe, clean place, to be used only when generating systems. Place another disk pack into the drive, which can be used for storing data. Under no circumstances should the factory generated cartridge be used as a day to day work disk.

When the system is booted up, test it according to the instructions in Chapter 6. If you notice anything peculiar, note the specific symptoms, and continue testing until you are satisfied that it has been well tested. If you noted any errors, consult the RTE-6/VM Online Generator Reference Manual, making use of examples and the factory-generation listing. Pay particular attention to those questions you answered differently from those shown in the examples. When you've identified the problems, replace the factory-generated cartridge in the disk drive, boot up that system, purge all copies of the previous generation and list file, pack the disk, edit the answer file, and rerun the generator.

When you have successfully generated and RTE system, and are familiar with the use of the Generator and SWTCH, make a copy of your generation answer file (so you can use it again if necessary) and then modify the original to include the other HP subsystems you want. Consult the appropriate subsystem manuals and configuration guides for the generation requirements of each subsystem. Generate the new system using this answer file and the procedures outlined in the first part of this chapter. Your new system may overlay your first system, but DO NOT

ALLOW YOUR FACTORY-GENERATED DISK TO BE OVERLAID! Boot the new system up, and test it, using the procedures described previously and the information provided in the system manuals. When satisfied that the subsystems work, make a copy of your generation answer file for subsequent backup if necessary.

Disk Planning

RTE-6/VM is a disk-based operating system in which the disk provides the primary storage area for the following items:

- Configured operating system.
- Relocated memory-resident library and programs.
- Relocated disk-resident programs.
- Relocatable library modules.
- Temporary storage for programs (source programs for editing, etc.).
- Temporary storage for swapped out programs.
- User files.

Disk storage is managed in terms of groups of contiguous tracks called subchannels (after generation, subchannels are referenced through logical unit numbers that are assigned in the I/O planning section). The primary purpose of the disk planning section is to configure available disk tracks into one or more subchannels. The operating system further defines the subchannels as system, auxiliary, and peripheral subchannels. The generator allows you to define a group of subchannels on a single disk controller. Multiple controllers are discussed here under the heading Multiple Disk Controllers.

System/Auxiliary Subchannels

The system and auxiliary subchannels contain tracks controlled by the system. A track usage table is maintained by the system for these subchannels. These tracks are considered system tracks and may be obtained from the system subchannel (LU 2) or the auxiliary subchannel (LU 3). The system tracks are used for:

- Program swapping.
- Online loading of programs.
- Scratch area for the line editor.
- Temporary storage (by user programs).

The difference between a system subchannel and an auxiliary subchannel is that the configured system (including the memory-resident system, the generator relocated disk-resident programs, and the relocatable library) is stored only on the system subchannel.

An auxiliary disk is not required but is sometimes useful for:

- Larger swap area.
- More system manager file space.
- Decreasing swapping time, since system swap tracks are allocated from the top of the available track list downward (that is, from the top tracks on LU 3 before LU 2). This feature permits the auxiliary disk to be used as a swapping disk. Because LU 3 can be on another disk or another controller, head movement is reduced, thus optimizing a system for speed (refer to Appendix A).

The combined size of a system and auxiliary subchannel is limited to 1600 tracks. This size may be reduced, depending on the type of disk used (for example, 400 tracks on a 7908 disk).

The user can also share tracks on LU 2 and LU 3 with the operating system. Four EXEC requests are provided to allow the user to request and release these tracks. Note that these tracks are not managed by the file system. They are managed in full track increments by the RTE-6/VM Operating System.

Note More than one system or type of system can be located on, and/or share a disk, and these systems may share tracks on one or more disks. In designating tracks, those that are shared should be included and declared during each system generation. The restriction is that any tracks of an RTE system that are assigned to LUs 2 or 3 (the system or auxiliary subchannel) must be unique to that RTE system. Remaining tracks on other disk subchannels can be assigned to more than one system.

Peripheral Subchannels

Disk subchannels other than system and auxiliary are classified as peripheral subchannels and must be assigned logical unit numbers greater than 6 but less than 64 (between 7 and 63). Note that if no LU is assigned for a peripheral disk, that disk cannot be accessed. Tracks on the peripheral subchannels are not subject to the operating system assignment and release mechanism. Management of these areas can be accomplished directly by user supplied programs or by the File Management Package. Peripheral subchannels to be used by the File Manager can be defined with up to 32767 tracks.

Multiple Disk Controllers

For the purposes of interactive subchannel definition, the generator assumes a single HP 13037B/C Multiple Access Controller (MAC), a HP 12821A Disk Interface for Integrated Controller Disks (ICD), or a HP 12821A Disk Interface for Command Set 80 (CS/80) disks. If a system has more than one controller or interface, a table must be constructed before beginning system generation. Refer to Appendix A for multiple disk controller information and assistance in constructing this table. You must include the appropriate disk driver and define an Equipment Table entry and the logical unit numbers for the subchannels defined (described in I/O STRUCTURE PLANNING).

The optional auxiliary subchannel may be placed on a different controller than the system subchannel. The preceding discussion applies in this case with the added requirement that the user specify the number of tracks in the subchannel when the generator inquires about the auxiliary option (refer to Chapter 2 of the Online Generator Manual).

Multiple CPU − HP 7905/7906/7920/7925 Systems

Multiple CPU operation (associated with only the HP 13037B/C disk controller) is supported by the SWTCH program, the bootstrap loader, and the DVR32 disk driver. More than one CPU can share one or more disk drives under the following conditions:

- System area (LU 2 and LU 3) for one CPU cannot occupy the same system disk tracks as that of another CPU.
- Systems may map tracks in the same peripheral disk area. However, they should share access to these areas only as described in Appendix B under MULTIPLE CPU − MAC SYSTEM OPERATION.
- The File Management System does NOT support multiple CPU operation.

Disk Configuration

Characteristics of the disk drives supported by RTE-6/VM are given in Table 4-1. Discussion of each disk type is given in the following sections. Refer to the appropriate manual for the disk drive used in your system for more details.

ICD and MAC Disk Surface Organization

Tracks on a subchannel must be contiguous. They may be allocated on a single surface or allocated by cylinders. A cylinder is a collection of tracks from several contiguous surfaces, a single track from each surface, each having the same track number. For example, cylinder 5 would be defined as a collection of track 5 from each surface of a disk, see Figure 4-2.

Allocating tracks in cylinder mode improves access time since contiguous tracks may be accessed by using different heads, thereby keeping physical head movement to a minimum. Using this mode on LUs 2 and 3 can improve swap and program load time considerably.

If more than one surface is to be used, tracks are cyclically allocated downward and back to the original surface when necessary. For example, a subchannel beginning with head 0 and using two surfaces will use head 0, head 1, and head 0 repeatedly, and in that order.

If a subchannel includes both fixed and removable platters (i.e., cylinder mode) flexibility is lost because the absence of either platter invalidates all data on the subchannel. Also, the relational alignment between two platters depends on drive orientation when the cartridges inserted. This makes track-to-track access time across platters unpredictable. In fact, it may be better or worse than on one platter, depending on alignment and the time required for software processing between tracks. Thus, it is suggested that cylinder mode not cross fixed and removable platter boundaries.

Note Be careful when defining disk subchannels; avoid including tracks in more than one subchannel. The generator assumes the disk subchannel organization is valid and performs no checks on the definition. Overlapping tracks between two subchannels is a common mistake and the user should be absolutely positive that this error is avoided, since it can have disastrous results on a running system. Remember that when a subchannel covers more than one surface, the starting head is incremented to determine that surfaces covered by that subchannel. In addition, remember that spares immediately follow each subchannel. To ensure correct subchannel definitions, the second part of the worksheet (see Figure 4-3) must be filled in correctly.

ICD and MAC Spare Tracks

Some tracks on a disk surface may be unusable. When such a track is encountered, another track may be assigned (provided spares are available) in its place by the system transfer program SWTCH or the disk initialization program FORMT. In this case the disk controller will automatically switch to that track on future references.

During generation, spare tracks on MAC and ICD disks can and should be assigned to each subchannel for this purpose. When a bad track is encountered during the system transfer or FORMT process, a subchannel may draw from its spares. Note that spare tracks are allocated on a subchannel basis and belong only to that subchannel. As a result, one subchannel cannot use spare tracks from another subchannel.

Spares immediately follow the main tracks for the associated subchannel and use the same surface organization. Spares are recommended even though they may not be used on a given disk. A subchannel or complete disk might later be copied to another disk where bad tracks are encountered, and all data would not fit if the receiving disk did not have sufficient spares.

With CS/80 Disks, this sparing process is transparent to the user.

Note Spare track assignment occurs only in SWTCH or FORMT and does not occur in the online disk driver.

Table 4-1. Compatible Disk Drive Characteristics

TRACK-ORIENTED DRIVES

Model	Recording Surfaces	Tracks/Surface	Driver	Words/Track
The following disk drives utilize the HP 13037B/C MAC controller. Any combination of eight of these drives may be connected to a single controller. The controller is interfaced to the computer through the HP 13175 interface card occupying one I/O slot. Up to 64 subchannels may be defined for this controller.				
7905	3	411	DVR32	6144
7906	4	411	DVR32	6144
7920	5	823	DVR32	6144
7925	9	823	DVR32	8192
Each of the following disk drives contains its own integrated controller which is connected to the computer through the HP 12821A ICD/CS/80 interface card occupying one I/O slot. On a single HP 12821A interface card, any combination of two of these drives may be connected and up to 64 subchannels may be defined (1).				
9895	2	77	DVA32	3840
7906H	4	411	DVA32	6144
7920H	5	823	DVA32	6144
7925H	9	823	DVA32	8192

BLOCK-ORIENTED DRIVES

Model	Number of Blocks	Block Size (in words)	Driver
Each of the following disk drives contains its own integrated controller which is connected to the computer through the HP 12821A ICD/CS/80 interface card occupying one I/O slot. On a single HP 12821A interface card, any combination of four of these drives can be connected and up to 64 subchannels can be defined (1).			
7907*	80256/platter	128	DVM33
7908	64750	128	DVM33
7911	109824	128	DVM33
7912	256256	128	DVM33
7914	516096	128	DVM33
7933	1579916	128	DVM33
7935	1579916	128	DVM33
CTD (2)	16319 or 65279	512	DVM33
7936	1201956	128	DVM33
7937	2232204	128	DVM33
7941	94752	128	DVM33
7942	94752	128	DVM33
7945	216831	128	DVM33
7946	216831	128	DVM33
7957	319095	128	DVM33
7958A	510552	128	DVM33
7958B	593872	128	DVM33
C2200 (335H)	1309840	128	DVM33
C2203 (670H)	2619792	128	DVM33

* 7907 contains two platters addressed as unit 0 (fixed) and unit 1 (removable).
1. A Cartridge Tape Drive (CTD) requires one subchannel.
2. Number of total blocks on a CTD depends on the cartridge tape size.

Command Set 80 Disk Configuration

Unlike disks previously described, Command Set 80 (CS/80) devices are block addressable. This allows RTE to treat these devices as a series of contiguous blocks. The physical arrangement of these blocks into surfaces and cylinders may be hidden from the user. Because of this, all CS/80 devices which have the same block size are logically identical except for the number of blocks they contain.

The completed worksheet describes each subchannel in terms of two values: the number of (logical) tracks and the number blocks per (logical) track. Unlike other disks, the position (starting surface and cylinder) and orientation (surface mode vs. cylinder mode) are not user-specifiable for Command Set 80 disks. All subchannels will begin at the next available block, and will be laid out in cylinder mode.

When filling out the worksheet in Figure 4-1, there are several important guidelines to remember.

- Spare Tracks: Command Set 80 devices include sparing at the hardware level. Therefore, the explicit allocation of spare tracks by the user is not required.

- Subchannel Size: The combined length of the system and auxiliary subchannels (LU 2 and LU 3) must not exceed 1600 tracks. If the auxiliary subchannel is not defined, the system subchannel must not exceed 1600 tracks. Similarly, a peripheral subchannel to be used by the file Management Package (FMP) must not exceed 32767 tracks.

- Blocks/Track: This must be a non-multiple of 7 between 1 and 64. If a multiple of 7 is used, such as 35, the system will not access this subchannel properly, and the generator will issue a warning (GEN ERR 84). If this parameter is not entered, the generator will use the default value of 48 blocks/track.

- Subchannel Numbering: Subchannels on a given disk interface (12821A card) are numbered sequentially from 0 to 63. Space allocated for holes or disk caches are not assigned subchannel numbers. Cartridge Tape Drives (CTD) will be assigned subchannel numbers by the generator.

- System Subchannel: The disk ROM Loader (12992J) will boot a system on a CS/80 disk only if the system's boot extension starts with block 0, volume 0, unit 0, HP-IB address 0. The boot extension must be located in block 0 and 1:

  - In the generator, define the system subchannel such that it starts at block 0. In this case, the program SWTCH will pace the new operating system at the start of this subchannel. The first two blocks of the system are the boot extensions.

If the boot extension is not located at block 0, the boot file produced by the generator must be used to boot the system.

Subchannels on a CS/80 disk are defined in a manner directly translatable into input for the generator. Refer to Figure 4-1 and fill in the blanks on the worksheet form according to the following instructions:

Step 1. for each disk volume or Cartridge Tape Drive (CTD) on your system, fill in the model number, HP-IB address, unit number, and volume number on a blank Command Set 80 (CS/80) worksheet. For all CS/80 disks, set the unit and volume number to 0. For the CTD, set unit number to 1 and volume to 0.

Step 2. Fill in the initial number of blocks for each disk.

Step 3. Sort the worksheets into the order in which the information they contain is to be given to the generator. If a CTD has the same HP-IB address as a disk (that is, an integrated CTD), a disk cache should be used. To do this, the information about the CTD must be entered before that of the disk (unit #1, volume #0).

Step 4. Allocate disk space and assign subchannel numbers. Subchannel numbers start with 0 and are assigned sequentially up to and including subchannel 63. For each worksheet, do the following:

If the model is CTD, then the device is a Cartridge Tape Drive (CTD). Assign the next sequential subchannel number to the CTD and go to the next worksheet. Remember the subchannel number so that you can assign a disk cache to it later.

If the model is a disk, then determine the number of blocks on the disk.

DISK	BLOCKS
7907*	80256/platter
7908	64750
7911	109824
7912	256256
7914	516096
7933	1579916
7935	1579916
7936	1201956
7937	2232204
7941	94752
7942	94752
7945	216831
7946	216831
7957	319095
7958A	510552
7958B	593872
C2200	1309840
C2203	2619792

* 7907 contains two platters addressed as unit 0 (fixed) and unit 2 (removable).

Once the size of the disk is filled in on the worksheet, you may allocate disk space as follows:

- For a subchannel, assign the next sequential subchannel number, and specify the number of tracks and the blocks per tracks. If you do not specify the blocks per track, the generator will use 48 as the default. Compute the number of blocks expended and subtract this number from the blocks remaining.

- For a disk cache, put the work CTD in the number of tracks column and the subchannel number of the CTD in the blocks per track column. The number of blocks expended will be 256. Subtract this number from the blocks remaining.

- For a hole on the disk, specify a negative number of tracks and the number of blocks per track. If you do not specify the blocks per track, the generator will use 48 as the default. Compute the number of blocks expended and subtract this number from the blocks remaining.

- If more space remains on the disk, allocate it for one of the above two uses by repeating the above instructions. To end space allocation on a disk volume, enter /E in the number of tracks column. The space remaining on the disk volume will not be accessible from the generated system.

Step 5. Determine which subchannel(s) will be the system and auxiliary subchannels and mark them accordingly.

At this point the worksheets provide all of the data required by the generator. This information is contained in all four fields of the device description and columns 2 and 3 of the allocation description.

MODEL 7908

HP-IB ADDRESS 1

UNIT 0

VOLUME 0

INITIAL NUMBER OF BLOCKS 64750

SUBCHANNEL # HOLE or DISK CACHE	# OF TRACKS (1)	BLOCKS/TRACK (2,3)	BLOCKS EXPENDED	BLOCKS REMAINING
System 0	300	48	14400	50350
1	200	48	9600	40750
2	200	48	9600	31150
3	200	48	9600	21550
Auxiliary 4	200	48	9600	11950
5	150	48	7200	4460
6	98	48	4704	
	/E			

(1) Enter CTD for Cartridge Tape Drive.

(2) If CTD entered in previous column, enter subchannel of CTD.

(3) If not specified, this parameter will default to 48.

Figure 4-1. Command Set 80 Worksheet

MODEL CTD

HP-IB ADDRESS 2

UNIT 1

VOLUME 0

INITIAL NUMBER OF BLOCKS N/A

SUBCHANNEL # HOLE or DISK CACHE	# OF TRACKS (1)	BLOCKS/TRACK (2,3)	BLOCKS EXPENDED	BLOCKS REMAINING
Assigned 7				

(1) Enter CTD for Cartridge Tape Drive.

(2) If CTD entered in previous column, enter subchannel of CTD.

(3) If not specified, this parameter will default to 48.

Figure 4-1. Command Set 80 Worksheet (Cont.)

MODEL 7911

HP-IB ADDRESS 2

UNIT 0

VOLUME 0

INITIAL NUMBER OF BLOCKS 109824

SUBCHANNEL # HOLE or DISK CACHE	# OF TRACKS (1)	BLOCKS/TRACK (2,3)	BLOCKS EXPENDED	BLOCKS REMAINING
8	500	48	24000	85824
9	300	48	14400	71424
10	250	48	12000	59424
11	250	48	12000	47424
12	250	48	12000	35424
13	200	48	9600	25824
14	200	48	9600	16224
15	200	48	9600	6624
DISK CACHE	CTD		256	6368
16	132	48	6336	32
	1E			

(1) Enter CTD for Cartridge Tape Drive.

(2) If CTD entered in previous column, enter subchannel of CTD.

(3) If not specified, this parameter will default to 48.

Figure 4-1. Command Set 80 Worksheet (Cont.)

MODEL 7912

HP-IB ADDRESS 0

UNIT 0

VOLUME 0

INITIAL NUMBER OF BLOCKS 256,256

SUBCHANNEL # HOLE or DISC CACHE	# OF TRACKS (1)	BLOCKS/TRACK (2,3)	BLOCKS EXPENDED	BLOCKS REMAINING
1 (SYSTEM)	400	64	25,600	230,656
2 (AUX.)	300	64	19,200	211,456
3	400	64	25,600	185,856
4	350	64	22,400	163,456
5	350	64	22,400	141,056
6	300	64	19,200	121,856
7	300	64	19,200	102,656
DISC CACHE	CTD	0	256	102,400
8	300	64	19,200	83,200
9	250	64	16,000	67,200
10	200	64	12,800	54,400
11	200	64	12,800	41,600
12	100	64	6,400	35,200
13	100	64	6,400	28,800
14	100	64	6,400	22,400
15	100	64	6,400	16,000
16	50	64	3,200	12,800
17	50	64	3,200	9,600
18	50	64	3,200	6,400
19	25	64	1,600	4800
20	25	64	1,600	3,200
21	25	64	1,600	1,600
22	25	64	1,600	0

(1) Enter CTD for Cartridge Tape Drive.

(2) If CTD entered in previous column, enter subchannel of CTD.

(3) If not specified, this parameter will default to 48.

Figure 4-1. Command Set 80 Worksheet (Cont.)

HP Disk Configuration

The HP 7905 Disk Drive is a single unit that contains two disk platters, one permanently mounted, and the other housed in a removable cartridge. Each disk platter has two surfaces; however, one surface of the 7905 fixed platter is used for timing purposes and is not available for data recording. Therefore, a single HP 7905 Disk Drive contains 3 surface (requiring three heads) and 1,233 tracks in 411 cylinders. Note that a cylinder consists of one track from each surface. For example, cylinder #3 is made up of the fourth track on surface 0, the fourth track on surface 1, and the fourth track on surface 2. See Figure 4-2 for a pictorial diagram of the HP 7905 disk platter organization. The following discussion provides the criteria for subchannel configuration. Each subchannel consists of a group of contiguous tracks on a single drive. One drive may contain several subchannels, and up to 64 subchannels may be defined for one controller. There is no fixed hardware relationship between a subchannel and a given disk area; it is your responsibility to define these relationships.

The completed disk worksheet describes each subchannel on a drive in the following terms; unit number of the drive, size of the subchannel in number of tracks, starting head and the cylinder numbers, surface organization, number of tracks, and number of spare tracks. In dividing up the HP 7905 disk tracks, bear in mind that the goal is to assign a logical unit number referencing a group of disk tracks.

When filling in the worksheet illustrated in Figure 4-3, there are several important rules and guidelines to remember:

- Surface Organization: Note that any HP 7905 subchannel using three surfaces must start on head 0.
- Spare Tracks: You should plan on about 1,200 usable tracks per drive, dividing the remaining 33 tracks as spares among the subchannels in proportion to their size.
- Subchannel Size: the combined size of the system and auxiliary subchannel (LU 2 and LU 3) must not exceed 1,600 tracks, excluding spares. A peripheral subchannel assigned to be used by the File Management Package may have up to 32,767 tracks.
- Subchannel Numbering: Subchannels on a given disk controller are numbered sequentially from 0 to 63. Do not skip or duplicate any numbers.
- System Subchannel: The disk ROM loader boots a system on a 7905 disk only if it starts at cylinder 0, head 0, 1, or 2 on drive 0. The RPL cylinder 0, head 0 or 2. Locating the system subchannel elsewhere requires that the bootstrap loader optionally produced during generation be used each time the system is booted up.

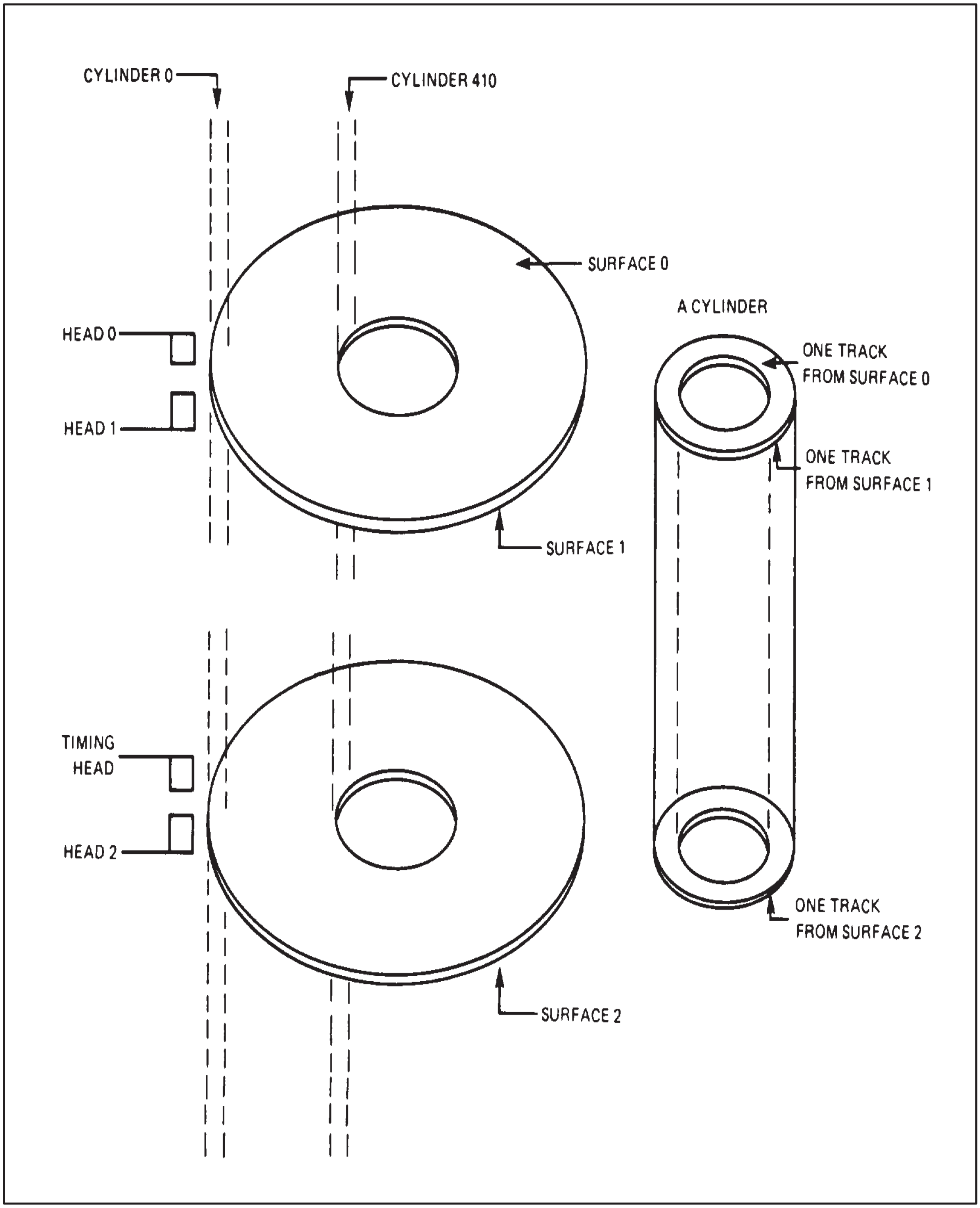


Figure 4-2. HP 7905 Disk

Subchannels on the HP 7905 are defined in a manner directly translatable for input to the generator. See Figure 4-3 and fill in the blanks on the worksheet form according to the following instructions.

Follow the instructions below for each HP 7905 drive.

Step 1. A hardware unit number is associated with each drive and is selected by a switch located behind the perforated front panel. Set the switch to the appropriate number and then write the number on the worksheet. No two disk drives should have the same number.

Note This hardware switch should not be repositioned while the drive is loaded (that is, active).

Step 2. The second part of the worksheet represents the three surfaces of the disk drive and is provided as an aid in dividing the surfaces into subchannels. For example, for subchannel 0, you could allocate 800 tracks for data and 8 tracks for spares, encompassing two surfaces. This makes a total of 808 tracks, which is 404 cylinders. The first cylinder contains the first and second addressable tracks:

First track = head #0, cylinder #0
Second track = head #1, cylinder #0

Divide up the surfaces, grouping the tracks into subchannels. Allow approximately 6 spare tracks for each 200 data tracks allocated. the number for the first cylinder of succeeding subchannel is found by adding the number of cylinders used by preceding subchannels. (To count cylinders, add tracks and spares, then divide by the number of surfaces.) In the example above, 404 cylinders were assigned to subchannel 0 (800 tracks plus 8 spares). Therefore the starting cylinder for subchannel 1 could be cylinder 404, head 0 or 1, or cylinder 0, head 2 (for a one-surface subchannel only), depending on how you assign the tracks.

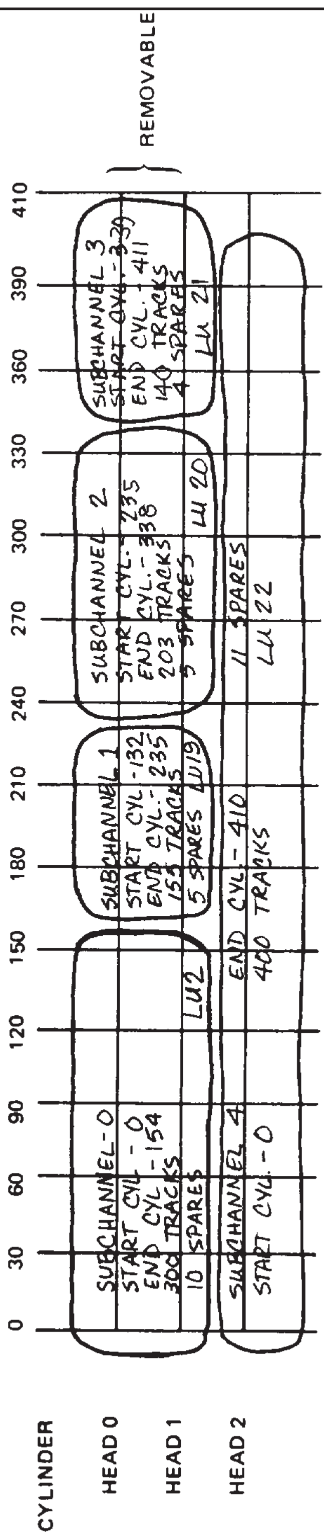
Step 3. When the third part of the worksheet is filled out, it provides the answers to all of the questions that the generator will ask about each subchannel. For the most part, the numbers are filled in from Step 2.

Fill in the blanks for all subchannels created in Step 2.

Determine which subchannel is the system and which subchannel the auxiliary (if any) and check the appropriate boxes.

STEP 1 FILL IN UNIT NUMBER: 0

STEP 2 TRACKS ARE SHOWN END-TO-END ON THREE SURFACES. USE PENCIL TO CIRCLE YOUR SUB-CHANNELS. WITHIN EACH CIRCLE WRITE THE FOLLOWING INFORMATION: THE SUBCHANNEL NUMBER; THE NUMBERS OF THE STARTING AND ENDING CYLINDERS; THE TOTAL NUMBER OF TRACKS, EXCLUDING SPARES; THE NUMBER OF SPARE TRACKS; AND THE LOGICAL UNIT NUMBER FOR EACH SUBCHANNEL.



STEP 3 TRANSLATE **STEP 2** TO NUMBERS:

SUBCHANNEL	# OF TRACKS, EXCLUDING SPARES	STARTING CYLINDER	STARTING HEAD	TOTAL # OF SURFACES INCLUDED IN SUBCHANNEL	NUMBER OF SPARES	SYSTEM? (√)	AUXILIARY? (√)
0	300	0	0	2	10	√	
1	155	155	0	2	5		
2	203	235	0	2	5		
3	140	339	0	2	4		
4	400	0	2	1	11		

Figure 4-3. HP 7905 Disk Worksheet Example

HP 7906(H) Disk Configuration

Except where otherwise noted the following sections apply to both HP 7906 and HP 7906H.

The HP 7906 Disk Drive is a single unit that contains two disk platters, one permanently mounted, and the other housed in a removable cartridge. Each HP 7906 disk platter has two surfaces available for data recording. One surface of the fixed disk is also used for timing purposes, but it is still available for data recording. Utilization of that surface by the system disk controller is transparent to the user. Therefore, a single HP 7906 Disk Drive contains 4 surface (4 heads), and 411 cylinders, giving 1,644 tracks. Note that a cylinder consists of one track from each surface. For example, cylinder #5 is made up of the sixth track on surface 0, the sixth track on surface 1, the sixth track on surface 2, and the sixth track on surface 3. See Figure 4-4 for a pictorial diagram of the HP 7906 disk platter organization.

The following discussion provides the criteria for configuring each disk into subchannels. Each subchannel will consist of a group of contiguous tracks on a single drive, and one drive may contain several subchannels. Up to 64 subchannels may be defined for one controller/disk interface. There is no fixed hardware relationship between a subchannel and a given disk area; it is your responsibility to define these relationships.

The completed disk worksheet describes each subchannel on a drive in the following terms: unit number/ICD address number of the drive, size of the subchannel in tracks, starting head and cylinder numbers, surface organization, number of tracks, and number of spare tracks. In dividing up the HP 7906 disk tracks, bear in mind that the goal is to define a logical unit number referencing a group of disk tracks.

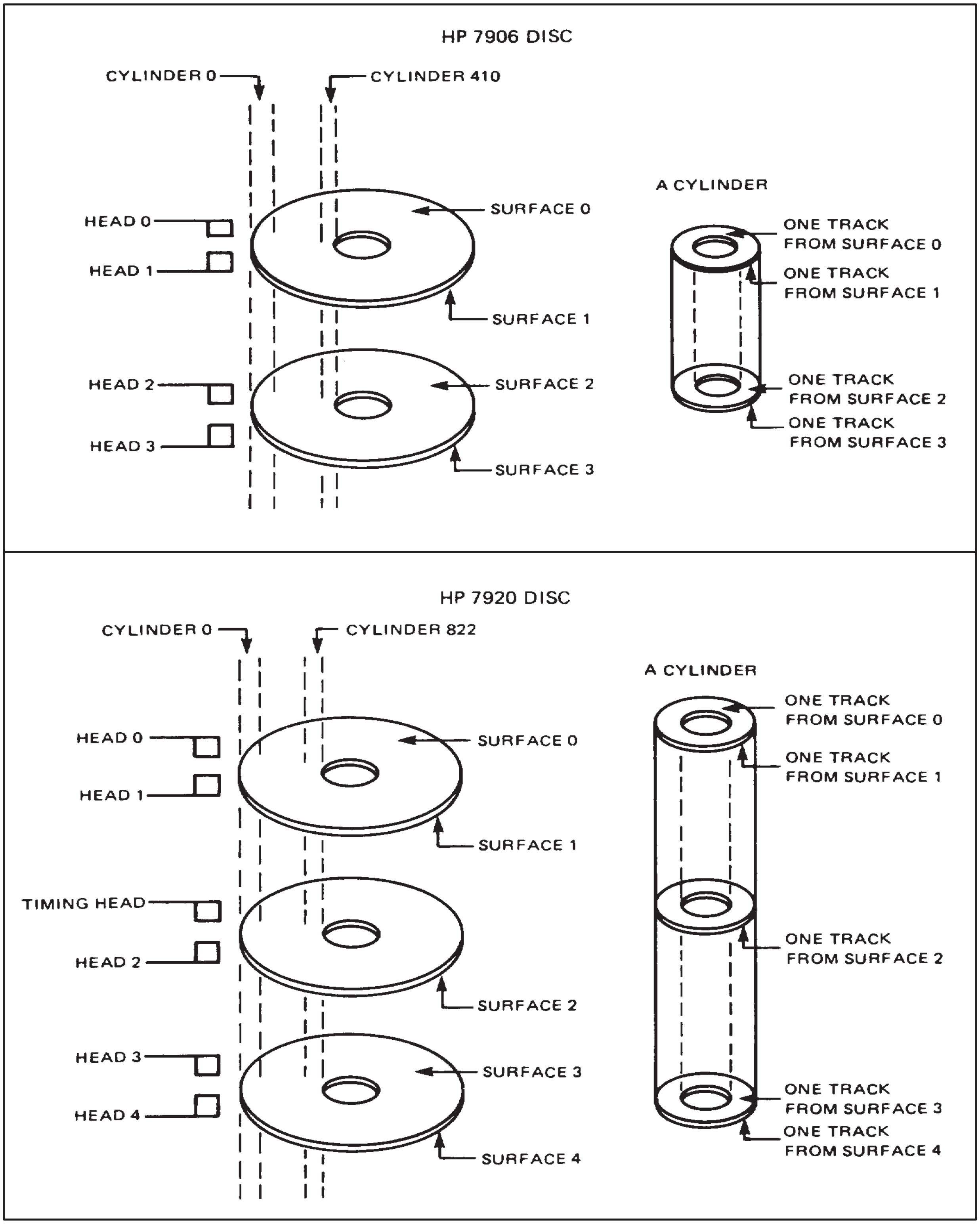


Figure 4-4. HP 7906 and 7920 Disks

When filling in the worksheet illustrated in Figure 4-5 there are several important rules and guidelines to remember:

- Surface Organization: Note that any subchannel using four surfaces must start on head 0; any subchannel using three surfaces must start on head 0 or 1, etc.
- Spare Tracks: You should plan on about 1,600 usable tracks per drive, dividing the remaining 44 tracks as spares among the subchannels in proportion to their size.
- Subchannel Size: The combined size of the system and auxiliary subchannel (LU 2 and LU 3) must not exceed 1,600 tracks, excluding spares.

  Similarly, a peripheral subchannel to be used by the File Management Package must not exceed 32,767 tracks, excluding spares.
- Subchannel Numbering: Subchannels on a given disk controller/interface are numbered sequentially from 0 to 63. Do not skip or duplicate numbers.
- System Subchannel: The disk ROM loader will boot a system on a HP 7906 disk only if it starts at cylinder 0, head 0, 1, 2, or 3 on drive 0. The RPL feature using the disk ROM loader boots a system on a 7906 disk only if it starts at cylinder 0, head 0 or 2. Locating the system subchannel elsewhere will require that the bootstrap loader optionally produced during generation be used each time the system is booted up.

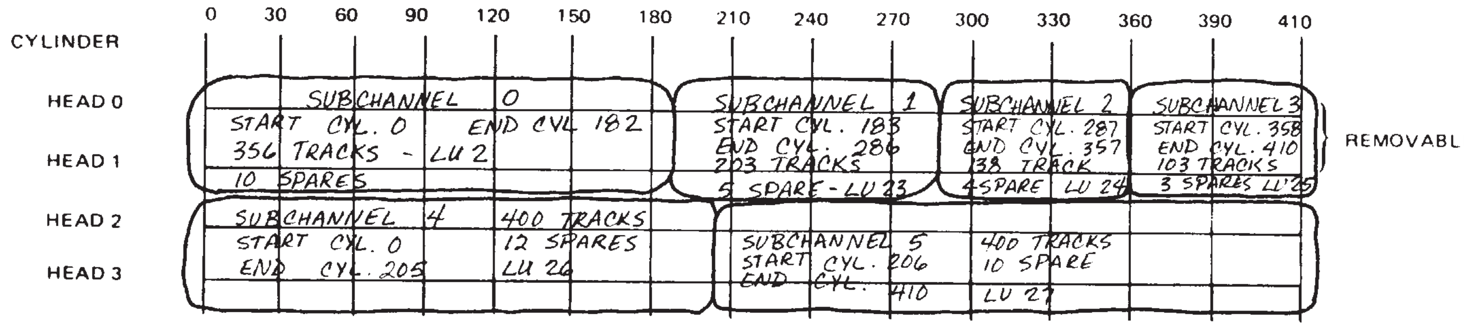
Subchannels on the HP 7906 are defined in a manner directly translatable for input to the generator. Complete the worksheet (see Figure 4-5) according to the following instructions for each HP 7906 drive.

Step 1. A hardware unit number is associated with each HP 7906 drive. An ICD address number is associated with each HP 7906H drive. Both numbers are selected by a switch located behind the perforated front panel. Set the switch to the appropriate number and then write the number on the worksheet. No two disk drives should have the same number.

Note This hardware switch should not be repositioned while the drive is loaded (that is, active).

STEP 1 FILL IN UNIT/ADDRESS NUMBER: 0

STEP 2 TRACKS ARE SHOWN END-TO-END ON FOUR SURFACES. USE PENCIL TO CIRCLE YOUR SUB-CHANNELS. WITHIN EACH CIRCLE WRITE THE FOLLOWING INFORMATION: THE SUBCHANNEL NUMBER; THE NUMBERS OF THE STARTING AND ENDING CYLINDERS, THE TOTAL NUMBER OF TRACKS, EXCLUDING SPARES; THE NUMBER OF SPARE TRACKS; AND THE LOGICAL UNIT NUMBER FOR EACH SUBCHANNEL.



STEP 3 TRANSLATE STEP 2 TO NUMBERS:

SUBCHANNEL	# OF TRACKS, EXCLUDING SPARES	STARTING CYLINDER	STARTING HEAD	TOTAL # OF SURFACES INCLUDED IN SUBCHANNEL	NUMBER OF SPARES	SYSTEM? (√)	AUXILIARY? (√)
0	356	0	0	2	10	√	
1	203	183	0	2	5		
2	138	287	0	2	4		
3	103	358	0	2	3		
4	400	0	2	2	12		
5	400	206	2	2	10		

Figure 4-5. HP 7906 Disk Worksheet Example

Step 2. The second part of the worksheet represents the four surfaces of the disk drive and is provided as an aid in dividing the surfaces into subchannels. For example, for subchannel 0, you could allocate 256 tracks for data and 8 tracks for spares, encompassing two surfaces. This makes a total of 264 tracks, which is 132 cylinders. The first cylinder contains the first and second addressable tracks;

first track = head #0, cylinder #0
second track = head #1, cylinder #0.

Divide up the surfaces, grouping the tracks into subchannels. Allow approximately 6 spare tracks for each 200 data tracks allocated. The number for the first cylinder of succeeding subchannels is found by adding the number of cylinders used by preceding subchannels. (To count cylinders, add tracks and spares, then divide by the number of surfaces.) In the example above, 132 cylinders were assigned to subchannel 0 (256 tracks plus 8 spares). Therefore the starting cylinder for subchannel 1 could be cylinder 132, head 0 or 1, or cylinder 0, head 2 or 3, depending on how you assign the tracks.

Step 3. When the third part of the worksheet is filled out, it will provide the answers to all of the questions the generator will ask about each subchannel. For the most part, the numbers are filled in from Step 2.

Fill in the blanks for all subchannels created in Step 2.

Determine which subchannel is the system and which subchannel the auxiliary (if any) and check the appropriate boxes.

HP 7920(H) Disk Configuration

Except where otherwise noted, the following sections apply to both HP 7920 and HP 7920H.

The HP 7920 Disk Drive is a single unit that contains three disk platters. Each data disk platter on the 7920 has two surfaces; however, one surface of the middle disk platter is used for timing purposes and is not available for data recording. Therefore, a single HP 7920 Disk Drive contains 5 surfaces (5 heads), and 823 cylinders, giving 4,115 tracks. Note that a cylinder consists of one track from each surface. For example, cylinder #7 is made up of the eighth track on surface 0, the eighth track on surface 1, the eighth track on surface 2, the eighth track on surface 3, and the eighth track on surface 4. See Figure 4-4 for a pictorial diagram of the 7920 platter organization.

The following discussion provides the criteria for configuring each disk into subchannels. Each subchannel consists of a group of contiguous tracks on a single drive, and one drive may contain several subchannels. Up to 64 subchannels may be defined on one controller/disk interface. There is no fixed hardware relationship between a subchannel and a given disk area; it is your responsibility to define these relationships.

The completed disk worksheet describes each subchannel on a drive in the following terms: unit number/ICD address number of the drive, size of the subchannel in tracks, starting head and cylinder numbers, surface organization, number of tracks, and number of spare tracks. In dividing up the HP 7920 disk tracks, bear in mind that the goal is to define a logical unit number referencing a group of disk tracks.

When filling in the worksheet illustrated in Figure 4-6, there are several important rules and guidelines to remember:

- Surface Organization: Note that any subchannel using five surfaces must start on head 0; any subchannel using four surfaces must start on head 0 or 1, etc.
- Spare Tracks: You should plan on at least 4,000 usable tracks per drive, dividing the remaining 115 tracks as spares among the subchannels in proportion to their size.
- Subchannel Size: the combined size of the system and auxiliary subchannel (LU 2 and LU 3) must not exceed 1,600 tracks, excluding spares. Similarly, a peripheral subchannel to be used by the File Management Package must not exceed 32,767 tracks, excluding spares.
- Subchannel Numbering: Subchannels on a given disk controller/interface are numbered sequentially from 0 to 63. Do not skip or duplicate any numbers.
- System Subchannel: The disk ROM loader will boot a system on a 7920 disk only if it starts at cylinder 0, head 0, 1, 2, or 3 on drive 0. The RPL feature using the disk ROM loader will boot a system on a 7920 disk only if it starts at cylinder 0, head 0 or 2. Locating the system subchannel elsewhere will require that the bootstrap loader optionally produced during generation be used each time the system is booted up.

Subchannels on the HP 7920 are defined in a manner directly usable as input to the generator. Fill in the blanks on the worksheet according to the following instructions. See Figure 4-6 for an example of a HP 7920 worksheet.

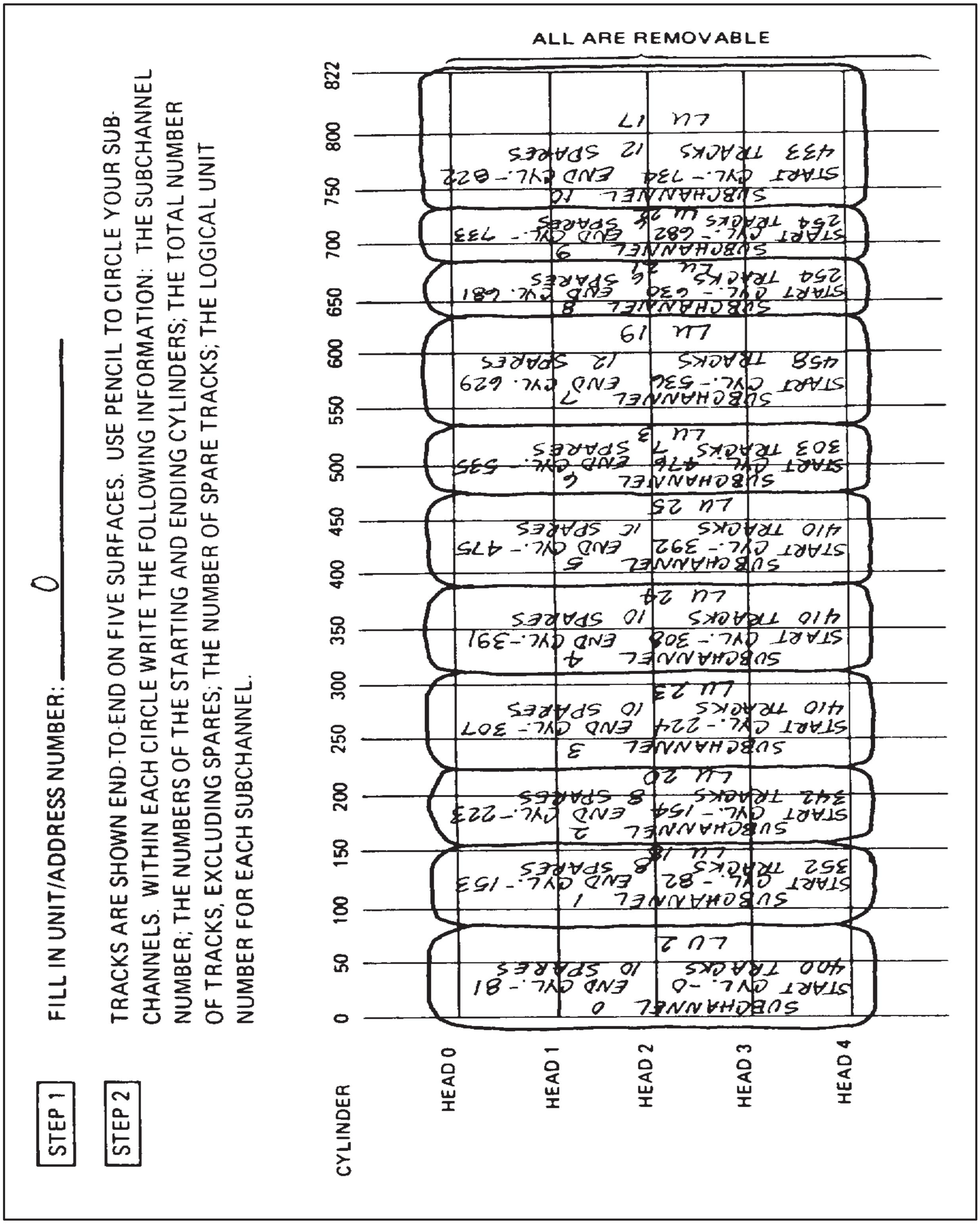


Figure 4-6. HP 7920 Disk Worksheet Example

STEP 3 TRANSLATE STEP 2 TO NUMBERS:

SUBCHANNEL	# OF TRACKS, EXCLUDING SPARES	STARTING CYLINDER	STARTING HEAD	TOTAL # OF SURFACES INCLUDED IN SUBCHANNEL	NUMBER OF SPARES	SYSTEM? (√)	AUXILIARY? (√)
0	400	0	0	5	10	√	
1	352	82	0	5	8		√
2	342	154	0	5	8		
3	410	224	0	5	10		
4	410	308	0	5	10		
5	410	392	0	5	10		
6	303	476	0	5	7		
7	458	536	0	5	12		
8	254	630	0	5	6		
9	254	682	0	5	6		
10	433	734	0	5	12		

Figure 4-6. HP 7920 Disk Worksheet Example (Cont.)

Follow the instructions given below for each HP 7920 drive.

Step 1. A hardware unit number is associated with each 7920 drive. An ICD address number is associated with each HP 7920H drive. Both numbers are selected by a switch located behind the perforated front panel. Set the switch to the appropriate number and then write the number on the worksheet. No two disk drives should have the same number.

Note This hardware switch should not be repositioned while the drive is loaded (that is, active).

Step 2. The second part of the worksheet represents the five surfaces of the disk drive and is provided as an aid in dividing the surfaces into subchannels. For example, for subchannel 0, you could allocate 256 tracks for data and 8 tracks for spares, encompassing two surfaces. This makes a total of 264 tracks, which is 132 cylinders. The first cylinder contains the first and second addressable tracks:

first track	=	head #0, cylinder #0
second track	=	head #1, cylinder #0.

Divide up the surfaces, grouping the tracks into subchannels. Allow approximately 6 spare tracks for each 200 data tracks allocated. The number for the first cylinder of succeeding subchannels is found by adding the number of cylinders used by preceding subchannels. (To count cylinders, add tracks and spares, then divide by the number of surfaces.) In the example above, 132 cylinders were assigned to subchannel 0 (256 tracks plus 8 spares). Therefore the starting cylinder for subchannel 1 could be cylinder 132, head 0 or 1, or cylinder 0, head 2, 3, or 4, depending on how you assign the tracks.

Step 3. When the third part of the worksheet is filled out, it will provide the answers to all of the questions the generator will ask about each subchannel. For the most part, the numbers are filled in from Step 2.

Fill in the blanks for all subchannels created in Step 2.

Determine which subchannel will be the system and which subchannel the auxiliary (if any) and check the appropriate boxes.

HP 7925(H) Disk Configuration

Except where otherwise noted, the following sections apply to both the HP 7925 and HP 7925H.

The HP 7925 Disk Drive is a single unit that contains five disk data platters and two platters for media protection only. Each data disk platter on the HP 7925 has two surfaces; however, one surface is used for timing purposes and is not available for data recording. Therefore, a single HP 7925 Disk Drive contains 9 surfaces (9 heads), and 823 cylinders, giving 7,407 tracks. Note that a cylinder consists of one track from each surface. For example, cylinder #7 would be made up of the eighth track on surface 0, the eighth track on surface 1, the eighth track on surface 2, the eighth track on surface 3, and the eighth track on surfaces 4, 5, 6, 7, and 8. See Figure 4-7 for a pictorial diagram of the 7925 disk platter organization.

The following discussion provides the criteria for configuring each disk into subchannels. Each subchannel consists of a group of contiguous tracks on a single drive. Each drive may contain several subchannels. Up to 64 subchannels may be defined for one controller/disk interface. There is no fixed hardware relationship between a subchannel and a given disk area; it is your responsibility to define these relationships.

The completed disk worksheet describes each subchannel on a drive in the following terms: unit number/ICD address number, size of the subchannel in tracks, starting head and cylinder numbers, surface organization, number of tracks, and number of spare tracks. In dividing up the HP 7925 disk tracks, bear in mind that the goal is to define a logical unit number referencing a group of disk tracks.

When filling in the worksheet illustrated in Figure 4-8, there are several important rules and guidelines to remember:

- Surface Organization: Note that any subchannel using nine surfaces must start on head 0; any subchannel using four surfaces must start on head 0 or 1, etc.

- Spare Tracks: You should plan on at least 7,200 usable tracks per drive, dividing the remaining 207 tracks as spares among the subchannels in proportion to their size.

- Subchannel Size: the combined size of the system and auxiliary subchannel (LU 2 and LU 3) must not exceed 1,600 tracks, excluding spares. Similarly, a peripheral subchannel to be used by the File Management Package must not exceed 32,767 tracks, excluding spares.

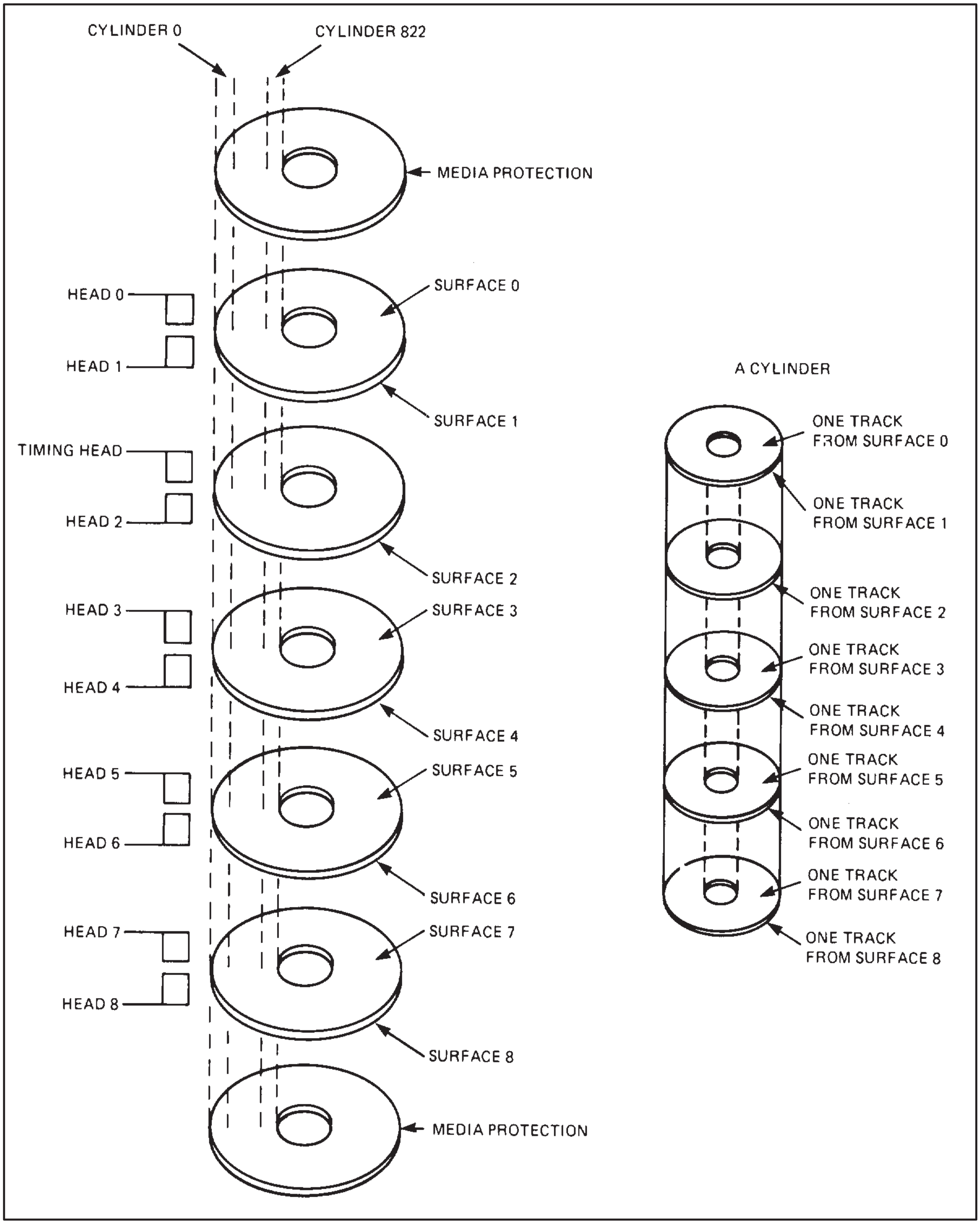


Figure 4-7. HP 7925 Disk

- Subchannel Numbering: Subchannels on a given disk controller/interface are numbered sequentially from 0 to 63. Do not skip or duplicate any numbers.

- System Subchannel: The disk ROM loader will boot a system on a HP 7925 disk only if it starts at cylinder 0, head 0, 1, 2, or 3 on drive 0. The RPL feature using the disk ROM loader will boot a system on a HP 7925 disk only if it starts at cylinder 0, head 0 or 2. Locating the system subchannel elsewhere will require that the bootstrap loader optionally produced during generation be used each time the system is booted up.

Subchannels on the HP 7925 are defined in a manner directly usable as input to the generator. See Figure 4-8 and fill in the blanks on the worksheet form according to the following instructions. Follow the instructions below for each HP 7925 drive.

Step 1. A hardware unit number is associated with each HP 7925 drive. An ICD address number is associated with each HP 7925H drive. Both numbers are selected by a switch located behind the perforated front panel. Set the switch to the appropriate number and then write the number on the worksheet. No two disk drives should have the same number.

Note This hardware switch should not be repositioned while the drive is loaded (that is, active).

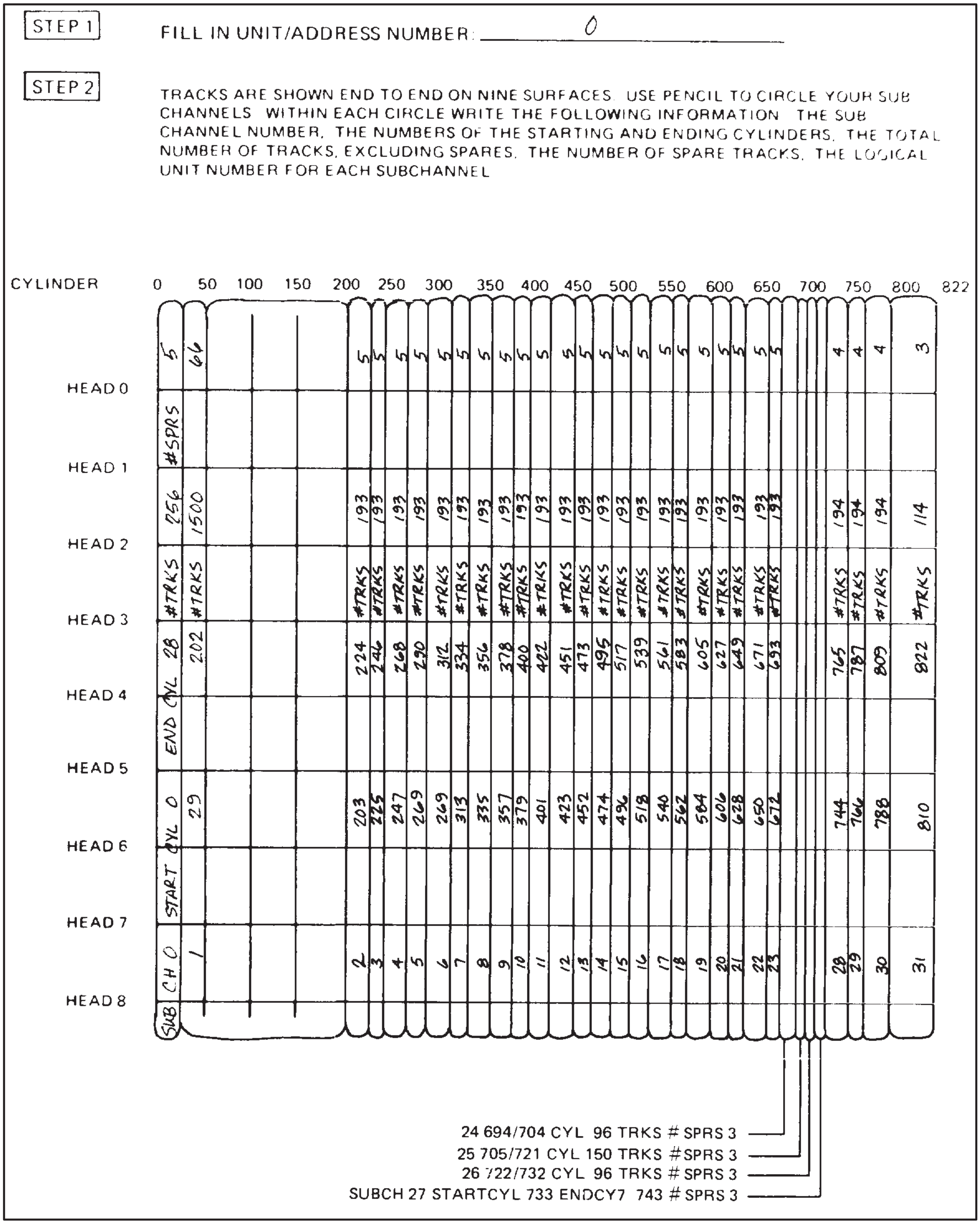


Figure 4-8. HP 7925 Disk Worksheet Example

STEP 3 TRANSLATE STEP 2 TO NUMBERS:

SUBCHANNEL	# OF TRACKS, EXCLUDING SPARES	STARTING CYLINDER	STARTING HEAD	TOTAL # OF SURFACES INCLUDED IN SUBCHANNEL	NUMBER OF SPARES	SYSTEM? ( )	AUXILIARY? ( )
0	256	0	0	9	5	✓	
1	1500	29	0	9	66		
2	193	203	0	9	5		
3	193	225	0	9	5		
4	193	247	0	9	5		
5	193	269	0	9	5		
6	193	291	0	9	5		
7	193	313	0	9	5		
8	193	335	0	9	5		
9	193	357	0	9	5		
10	193	379	0	9	5		
11	193	401	0	9	5		
12	256	423	0	9	5		✓
13	193	452	0	9	5		
14	193	474	0	9	5		
15	193	496	0	9	5		
16	193	518	0	9	5		
17	193	540	0	9	5		

Figure 4-8. HP 7925 Disk Worksheet Example (Cont.)

STEP 3 TRANSLATE STEP 2 TO NUMBERS:

SUBCHANNEL	# OF TRACKS, EXCLUDING SPARES	STARTING CYLINDER	STARTING HEAD	TOTAL # OF SURFACES INCLUDED IN SUBCHANNEL	NUMBER OF SPARES	SYSTEM? ( )	AUXILIARY? ( )
18	193	562	0	9	5		
19	193	584	0	9	5		
20	193	606	0	9	5		
21	193	628	0	9	5		
22	193	650	0	9	5		
23	193	672	0	9	5		
24	96	694	0	9	3		
25	150	705	0	9	3		
26	96	722	0	9	3		
27	96	733	0	9	3		
28	194	744	0	9	4		
29	194	766	0	9	4		
30	194	788	0	9	4		
31	114	810	0	9	3		

Figure 4-8. HP 7925 Disk Worksheet Example (Cont.)

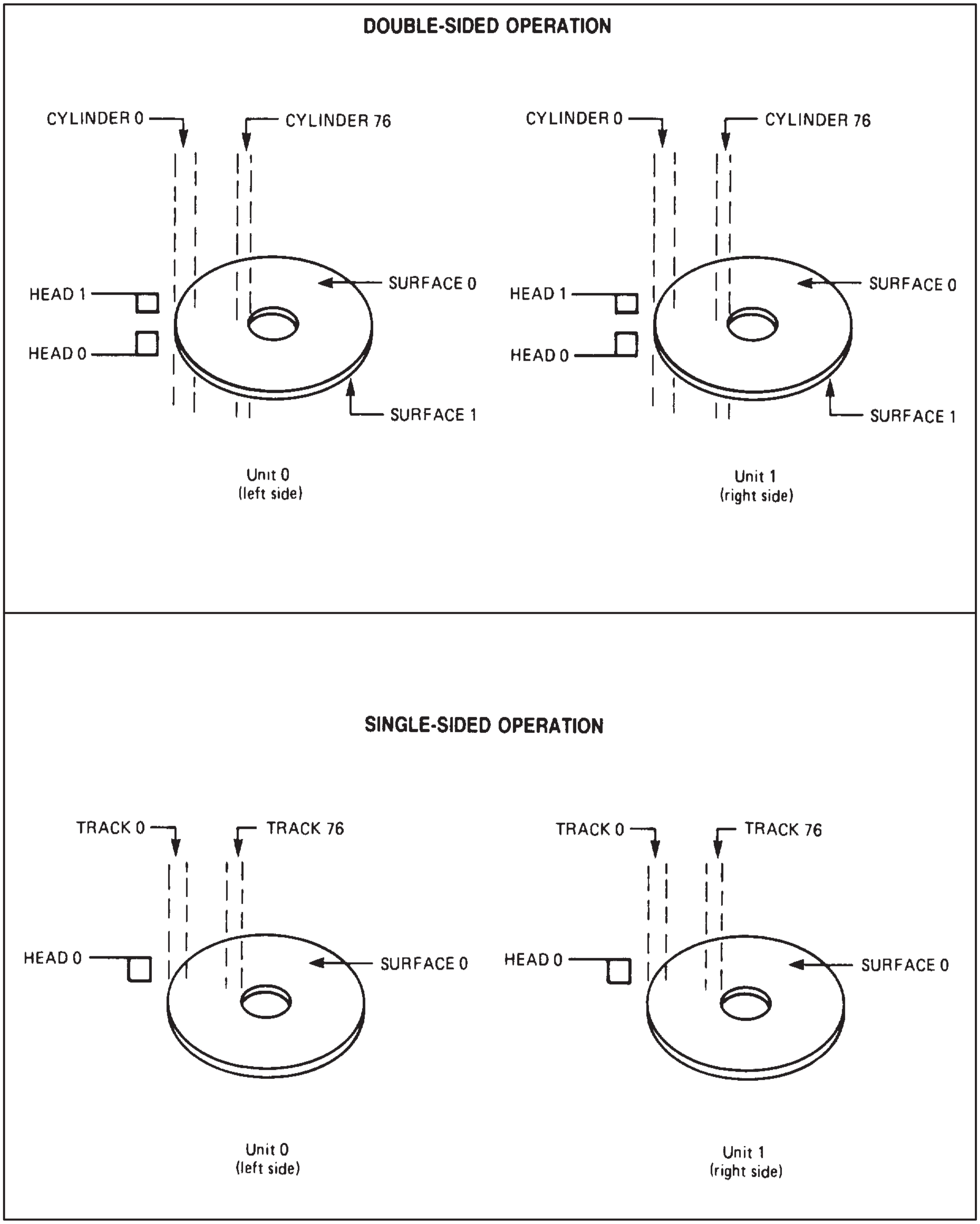


Figure 4-9. HP 9895 Disk

STEP 1 FILL IN ICD ADDRESS NUMBER: 1

STEP 2 ONLY ONE SUBCHANNEL PER DRIVE WILL BE DEFINED. THE FOLLOWING DEFINITION IS HP STANDARD DEFINITION FOR 9895 FLEXIBLE DISC

SUBCHANNEL	#OF TRACKS, EXCLUDING SPARES	STARTING CYLINDER	STARTING HEAD	TOTAL # OF SURFACES INCLUDED IN SUBCHANNEL	NUMBER OF SPARES	UNIT #
DOUBLE-SIDED OPERATION						
Ø	134	Ø	Ø	2	2Ø	Ø
1	134	Ø	Ø	2	2Ø	1
SINGLE-SIDED OPERATION						
Ø	67	Ø	Ø	1	1Ø	Ø
1	67	Ø	Ø	1	1Ø	1

Figure 4-10. HP 9895 Disk Worksheet

Step 2. The second part of the worksheet represents the nine surfaces of the disk drive and is provided as an aid in dividing the surfaces into subchannels. For example, for subchannel 0, you could allocate 244 tracks for data and 8 tracks for spares, encompassing nine surfaces. This makes a total of 252 tracks, which is 28 cylinders. The first cylinder contains the first and second addressable tracks:

− first track = head #0, cylinder #0
− second track = head #1, cylinder #0
− third track = head #2, cylinder #0
− fourth track = head #3, cylinder #0
:
:
:
− tenth track = head #0, cylinder #1
− eleventh track = head #1, cylinder #1
:
:
:

Divide up the surfaces, grouping the tracks into subchannels. Allow approximately 6 spare tracks for each 200 data tracks allocated. The number for the first cylinder of succeeding subchannels is found by adding the number of cylinders used by preceding subchannels. (To count cylinders, add tracks and spares, then divide by the number of surfaces.) In the example above, 28 cylinders were assigned to subchannel 0 (244 tracks plus 8 spares). Therefore, the starting cylinder for subchannel 1 would be cylinder 28.

Step 3. When the third part of the worksheet is filled out, it provides the answers to all the questions the generator will ask about each subchannel. For the most part, the numbers are filled in from Step 2.

Fill in the blanks for all subchannels created in Step 2.

Determine which subchannel is the system and which subchannel the auxiliary (if any) and check the appropriate boxes.

HP 9895 Disk Configuration

The HP 9895 Disk unit contains two disk drives and may use either single-sided or double-sided flexible disks. The drive mechanism determines which type of media is currently loaded in the drive by checking the position of the index hole on the flexible media. See Figure 4-9 for a pictorial diagram of the HP 9895 platter and unit organization.

Double-sided Operation. Each HP 9895 double-sided flexible disk has two surfaces available for data recording. Each drive contains 2 heads which may be positioned over 77 cylinders, giving a total of 154 tracks per drive. One subchannel per drive is defined; with an allocation of 134 tracks per subchannel, leaving 20 extras. this definition is an HP standard and should be used if FMGR compatibility across RTE Systems is desired. The two drives are distinguished by their unit numbers, either 0 or 1, indicating the left or right drive respectively. Figure 4-10 gives the worksheet entries for HP 9895 disks.

Single sided Operation. Each HP 9895 single-sided flexible disk has one surface available for data recording and thus only uses one of the two heads of the HP 9895 disk drive. The head may be positioned over 77 cylinders, giving 77 tracks per drive. One subchannel per drive is defined with an allocation of 67 tracks per subchannel leaving 10 extra. This definition is an HP standard and should be used if FMGR compatibility across RTE Systems is desired. The two drives are distinguished by their unit numbers, either 0 or 1, indicating the left or right drive, respectively. Figure 4-10 gives the worksheet entries for HP 9895 disks.

The HP 9895 disks do not have the sparing capability: instead, defective tracks are marked invisible, and the tracks following the defective track are renumbered by the FORMT program.

These disks cannot be used as a system disk.

I/O Structure Planning

The following sections will aid you in assigning select codes, logical unit (LU) numbers, and equipment table (EQT) entry numbers, for the devices to be included in your system. The I/O configuration worksheet should be filled out during the I/O planning process. The sample worksheet shown in Figure 4-11 should be referred to during each planning phase.

Devices and Interface Cards

Use the INTERFACE and DEVICE NAME columns in the I/O configuration worksheet to list the interface cards and devices in your system. List only interface cards and/or devices to be uniquely referred to by logical unit (LU) numbers in your system. The planning process is simplified if all devices attached to the same interface card are grouped together. List each interface card only once; leave the INTERFACE column blank for subsequent devices attached to the same card.

A sample list of devices normally assigned logical unit numbers is as follows:

- All terminals in the system, including terminals on multipoint or the multiplexer.
- All terminal cartridge tape units (CTU) and auxiliary printers connected to terminal drivers offering device support (for example, DVR05, DVA05).

- HP 7908/11/12 Cartridge Tape Drives.
- Every disk subchannel to be accessed by the File Management System or user application programs. Usually all disk subchannels are assigned logical unit numbers.
- All line printers, magnetic tape units, paper tape reader/punches, card readers, plotters, etc. Note that certain peripherals (card readers, HP 2608 printer) may require more than one LU to implement certain driver control and data conversion functions.
- Every communication line including DS/1000 links, DS/3000 link, RTE links and Multipoint control lines.
- Devices to be individually accessed by LU numbers or common interface buses (that is, HP-IB, HP 2250, Factory Data Link). You may also have to assign LU numbers to the interface buses themselves for control.
- Interface cards such as the WCS, TV interface.
- Custom user devices.

This list is by no means complete. You should refer to the appropriate subsystem manuals and configuration guides for more information.

If spooling is to be included in your system, a pseudo-device must exist for each concurrent spool operation. Each spool device should be listed in the worksheet since it will require one LU, EQT entry and Interrupt Table assignment. For an estimate of the number of spool devices to be configured into your system, refer to the Spooling System section in this chapter.

If automatic restart after power-fail is to be included in your system, the power-fail logic is treated as a device. This device should be entered on your worksheet.

Certain interface cards, such as the Time Base Generator (TBG) and privileged fence card, do not require LU and EQT entries. These cards have to be taken into account however, when planning your overall system select code assignments since each occupies an I/O slot (see below). If you make entries for these cards in your I/O configuration worksheet, it is recommended that a line be drawn through their LU and EQT columns.

Select Code Assignments

Every device controller connected to the computer must be plugged in to an I/O slot in the CPU. The operating system accesses device controllers by the address of their I/O slot, or Select Code. Device interrupt service priority is determined by select codes. When two or more device controllers request interrupt servicing concurrently, the controller with the lowest select code will be serviced first. Device controller select codes must be in the range 10 through 77 (octal).

Interface cards should be assigned to select codes according to the speed of interrupt response required by the I/O device. Interface cards for high-speed devices should be assigned higher priority addresses (that is, lower select codes) than low-speed devices. Devices requiring privileged interrupt are always assigned to the highest priority addresses (a privileged interrupt bypasses normal interrupt processing to achieve faster response for interrupts having the greatest urgency), while devices using DCPC transfers are assigned the lowest priority addresses. The one exception to this rule is in regard to the moving head system disk controller. For the fastest interrupt response, assign the moving head disk controller to the next available I/O slots after the Time Base Generator.

The following detailed steps show how to assign select codes to devices, starting at the highest priority address, octal select code 10. In addition to these steps, make certain that any peripheral devices or subsystems that use multiple I/O slots have their I/O cards together and in the relative order required by that device or subsystem.

A. Assign all devices that require privileged interrupt in order of decreasing response time requirements (that is, time from interrupt to service).

B. After the privileged devices, assign the privileged interrupt I/O card (note that this card is not necessary if no privileged devices exist).

C. Assign the Time Base Generator (TBG) I/O card.

D. Assign the moving head disk controller I/O card(s).

E. Assign all devices that do not use DCPC transfers in order of decreasing interrupt rate.

Note If a device uses DCPC for data transfers and still generates an interrupt for end-of-record (EOR) processing, the hardware priority of the device should be treated as a non-DCPC device, with the interrupt rate of the EOR condition determining its priority location. Some consideration should be given to the priority of a data transfer versus the priority of a record termination. Data transfers would normally be given priority over EOR interrupts of equivalent or even slightly slower interrupt rates.

F. Assign all devices that do use DCPC transfers in order of decreasing interrupt rate.

G. If an I/O extender is required and the extender does not have DCPC transfer capability, the order of steps e and f can be reversed so that all DCPC devices are in the computer mainframe. If this step is necessary, maintain the same relative order of interrupt rate assignment among the DCPC and non-DCPC devices.

H. If automatic restart after power-fail is to be included in your system, the power-fail logic is treated as a device. Assign it select code 4.

I. If spooling is to be included in your system, an unused select code must be assigned to each spool pseudo-device. Usually the spool devices are assigned high numbered select codes. If is recommended that you start your select code assignments at 77 and work downwards. For a discussion of the number of spool devices to configure in your system, refer to the Spooling System section in this chapter.

Refer to the SELECT CODE column in Figure 4-11 for sample select code assignments.

Logical Unit Assignments

A. Standard LU Assignments:

In the LU number column, make standard logical unit assignments (1-6) for appropriate devices. Standard logical unit assignments are as follows:

LU 1 System Console
LU 2 Primary System Disk Subchannel
LU 3 Auxiliary System Disk Subchannel (optional)
LU 4 Standard output Device
LU 5 Standard Input Device
LU 6 Standard List Device (line Printer)

The auxiliary system disk (LU 3) may be generated into your system when additional system scratch tracks or system files are required. (LU 3 can only be used as an auxiliary disk, not as another peripheral disk.) The standard output device may be a minicartridge or paper tape punch. The standard input device is usually a minicartridge or paper tape reader. If a magnetic tape unit is to be configured into the system, it is recommended that it be made logical unit 8.

B. Disk Subchannel Assignments.

Beginning with LU 10, consecutively assign LU numbers to your peripheral disk subchannels (other than LU 2 and LU 3). Note: Disk subchannels must be assigned LU numbers less than 64.

C. Non-Session Accessible Peripheral Assignments.

Assign logical unit numbers for those peripherals that must be accessed outside of the session environment. (This includes all peripherals in running without the Session Monitor.) The power-fail device is also in this category. Peripherals to be accessed outside the session environment must be assigned LU numbers less than 64.

Note If you are NOT using the Session Monitor, skip steps d and e.

D. Session Terminal Assignments.

Assign logical unit numbers to the keyboard/display subchannel of each session terminal (usually subchannel 0). Session terminal LU numbers must be in the range of 7 to 99.

E. Session Accessible Peripheral Assignments.

Assign logical unit numbers to the remaining devices in your system. Peripheral devices having LU numbers greater than 63 are only accessible from the session and batch environments. LU assignments for your spool devices should also be made at this point.

F. Spare LU Assignments.

You may wish to configure spare logical units into the system. Assign these units an EQT entry number of zero (the bit bucket). Spare logical unit numbers are used to point to devices not specified during generation (providing their EQT entries and drivers are configured). If the need arises and there are no spare LU numbers left in the system, you will have to switch another device LU to the new device or regenerate your system with additional LU numbers.

It is recommended that you include at least several spare LU numbers in your system. Systems using the Session monitor may use a total of 254 LU numbers; other systems are restricted to 63.

For sample LU assignments, refer to the LU column in Figure 4-11.

Summary of LU assignments:

7 <= Disk LU	<=63
7 <= non-Session LU	<=63
7 <= Terminal LU	<=99
7 <= Remaining Session LUs	<=254

Equipment Table Entry Assignments

There should be one Equipment Table (EQT) entry for every device controller (interface card). In cases where multiple devices are attached to the same controller, the same EQT entry number should be assigned to each device. For the MUX card, however, each port is assigned its own EQT entry number.

A. Assign your system disk subchannel (LU 2) and all other disk subchannels on that controller to EQT entry #1. If you do not have an auxiliary disk subchannel in your system, it is recommended that you assign LU 3 to EQT entry #0 (the bit bucket).

B. Beginning with EQT entry #2, other DCPC devices should be consecutively assigned EQT numbers in order of their DCPC priority.

C. Consecutively assign EQT numbers to the remaining devices in your system. (Remember, multiple devices on the same controller will share the same EQT number). You may want to match EQT number(s) and the LU number(s) assigned to the controller's device(s) to make the association easier to remember. These matching LU and EQT numbers will aid the user in operating the system after it is running; for example, when upping downed devices.

Note Certain HP subsystems (for example,DATACAP/1000, Multipoint, Multiplexer) require more than one EQT per controller. Consult the appropriate subsystem manuals and configuration guides for their EQT assignment procedures.

D. Assign the last (highest numbered) EQT to the power-fail device.

You now have enough information to form the basic structure of your system Device Reference Table, Equipment Table, and Interrupt Table. You should refer to the following sections of this manual and appropriate subsystem manuals and configuration guides for table parameter specifications.

Recall that generator inputs for the Device Reference Table (DRT) must be in order of increasing LU number. Inputs for the Equipment Table (EQT) must be in order of increasing EQT entry number. Inputs for the Interrupt table must be in order of increasing select code. The generator worksheets for the DRT and EQT tables are numbered by LU and EQT entry numbers. The Generator Interrupt Table worksheet is unnumbered (since select codes do not have to be contiguous). It is suggested that you fill in your Interrupt Table worksheet with the select codes to be configured into the system before filling out the rest of the worksheet.

RTE-6/VM I/O CONFIGURATION WORKSHEET PAGE 1 OF 4

Equipment Table Preparation

EQT Num.	Select Code (Octal)	Driver Name (DVyxx)	Buffered Output? (B)	DCPC? (D)	System Driver Area? (S)	Does Own Mapping (M)	EQT Ext. (Dec. Num. Of Words)	Time-Out (Dec. Num. Of 10 ms.)
1	12	DVM33		D				
2	13	DVR32		D				
3	14	DVA05	B				X=13	T=12000
4	15	DVB12	B				X=5	T=12000
5	16	DVA37	B				X=50	T=12000
6	17	DVR23	B	D				
7	21	DVA05	B				X=13	T=12000
8	22	DVA05	B				X=13	T=12000
9	23	DVA05	B				X=13	T=12000
10	24	DVA05	B				X=13	T=12000
11	25	DVA05	B				X=13	T=12000
12	26	DVA05	B				X=13	T=12000
13	27	DVA05	B				X=13	T=12000
14	67	DVZ12						
15	70	DVS43				M	X=18	
16	71	DVS43				M	X=18	
17	72	DVS43				M	X=18	
18	73	DVS43				M	X=18	
19	74	DVS43				M	X=18	
20	75	DVS43				M	X=18	
21	76	DVS43				M	X=18	
22	77	DVS43				M	X=18	
23	4	DVP43				M		
24								
25								

Figure 4-11. Sample I/O Configuration Worksheet

Device Reference Table Preparation

PAGE 2 OF 4

LU Number	Corresponding EQT	Subchannel	Description
1	3		System Console
2	1	1	System Disc
3	1	13	Aux Disc
4	3	1	Left CTU
5	3	2	Right CTU
6	4	0	2608 L.P.
7	4	3	2608 L.P.
8	14	0	2608 (Graphics)
9	5		HPIB
10	6		7970 Mag Tape
11	1	0	CTD
12	1	2	Disc Subchannel
13	1	3	"
14	1	4	"
15	1	5	"
16	1	6	"
17	1	7	"
18	1	8	"
19	1	9	"
20	1	10	"
21	1	11	"
22	1	12	"
23	1	13	"
24	1	14	"
25	1	15	"
26	1	16	"
27	1	17	"
28	1	18	"
29	1	19	"
30	1	20	"
31	1	21	"
32	1	22	"
33	1	23	Disc Subchannel
34	1	24	Disc Subchannel
35	1	25	"
36	1	26	"
37	1	27	"
38	1	28	"
39	1	29	"
40	1	30	"
41	1	31	"
42	1	32	"
43	1	33	"
44	1	34	"
45	1	35	"
46	1	36	"
47	1	37	"
48	1	38	"
49	1	39	Disc Subchannel
50	2	0	7906 Disc Subchannel
51	2	1	"
52	2	2	"
53	2	3	"
54	2	4	"
55	2	5	7906 Disc Subchannel
56	0		Spare LU
57	0		"
58	0		"
59	0		Spare LU
60	23		
61	7		2645 Terminal #1
62	8		2645 Terminal #2
63	9		2645 Terminal #3
64	10		2645 Terminal #4
65	11		2648 Terminal #5
66	12		2648 Terminal #6

Figure 4-11. Sample I/O Configuration Worksheet (Cont.)

Device Reference Table Preparation PAGE 3 OF 4

LU Number	Corresponding EQT	Subchannel	Description
67	13		2648 Terminal #7
68	7	1	Term.1 L.CTU
69	7	2	Term.1 R.CTU
70	7	4	Term.1 Aux.Pr
71	8	1	Term.2 L.CTU
72	8	2	Term.2 R.CTU
73	8	4	Term.2 Aux.Pr
74	9	1	Term.3 L.CTU
75	9	2	Term.3 R.CTU
76	9	4	Term.3 Aux.Pr
77	10	1	Term.4 L.CTU
78	10	2	Term.4 R.CTU
79	10	4	Term.4 Aux.Pr
80	11	1	Term.5 L.CTU
81	11	2	Term.5 R.CTU
82	11	3	Term.5 Graphics
83	11	4	Term.5 Aux.Pr
84	12	1	Term.6 L.CTU
85	12	2	Term.6 R.CTU
86	12	3	Term.6 Graphics
87	12	4	Term.6 Aux.Pr
88	13	1	Term.7 L.CTU
89	13	2	Term.7 R.CTU
90	13	3	Term.7 Graphics
91	13	4	Term.7 Aux.Pr
92	15		Spooling
93	16		"
94	17		"
95	18		"
96	19		"
97	20		"
98	21		"
99	22		Spooling

LU Number	Corresponding EQT	Subchannel	Description
100			
101			
102			
103			
104			
105			
106			
107			
108			
109			
110			
111			
112			
113			
114			
115			
116			
117			
118			
119			
120			
121			
122			
123			
124			
125			
126			
127			
128			
129			
130			
131			
132			

Figure 4-11. Sample I/O Configuration Worksheet (Cont.)

Interrupt Table Preparation

PAGE 4 OF 4

Select Code (Octal)	Option (EQT, PRG, ENT, or ABS)	Destination	Description
4	ENT	$POWR	Power Fail
12	EQT	1	7912 Disc
13	EQT	2	7906 Disc
14	PRG	PRMPT	System Console
15	EQT	4	2608 Line Prntr
16	EQT	5	HPIB
17	EQT	6	7970 MagTape
20	EQT	6	7970 MagTape
21	PRG	PRMPT	Terminal #1
22	PRG	PRMPT	Terminal #2
23	PRG	PRMPT	Terminal #3
24	PRG	PRMPT	Terminal #4
25	PRG	PRMPT	Terminal #5
26	PRG	PRMPT	Terminal #6
27	PRG	PRMPT	Terminal #7
67	EQT	4	2608 (Graphics)
70	EQT	15	
71	EQT	16	
72	EQT	17	
73	EQT	18	
74	EQT	19	
75	EQT	20	
76	EQT	21	
77	EQT	22	

Figure 4-11. Sample I/O Configuration Worksheet (Cont.)

Sample Worksheet Conventions

In this manual, the generator inputs are given in the context of the generator worksheets. It is recommended that these worksheets be filled in as you read this chapter. Worksheet inputs are keyed to the step numbers given in the Online Generator Manual. The step numbers may be used to cross reference the generator manual and associated worksheets. For example:

```
16d
    *
    *   RTE OPERATING SYSTEM
    *
    REL,%CR6S1              *OPERATING SYSTEM MODULES PART #1

    REL,%CR6S2              *OPERATING SYSTEM MODULES PART #2

    REL,%CR6S3              *OPERATING SYSTEM MODULES PART #3
```

The function and syntax of the above input are discussed in step 16d of the generator manual. These inputs would also be added to section 16d of the generator worksheets.

Note The file names (for example, %CR6S1) discussed in this chapter refer to relocatable files supplied on the HP 92084A primary disk.

Generator inputs for certain system resources are shown in the form (+n), where n is the number of resource units required in addition to those already allocated for other system components.

For example, a subsystem requiring three resource numbers would be indicated as follows:

```
28

    # OF RESOURCE NUMBERS?
    (+3)
```

You would therefore allocate an additional sixteen resource numbers to the current total. Note that the resource limits indicated in this manual and other configuration documentation will, in general, be a minimum value. The actual numbers should also include resources used by user application programs.

Generation variables are shown in lower case. These must be substituted with the desired values when your worksheets are filled in. For example, if a HP 7920 disk controller is assigned to EQT entry 1 with a select code of 12, and the worksheet example is:

```
22

    INTERRUPT TABLE

    sc,EQT,nn               *79XX Disk Controller
```

where:

sc is the select code of the disk.

nn is the assigned disk controller EQT entry.

Your worksheet should be filled as follows:

```
22

    INTERRUPT TABLE

    12,EQT,1                    *7920 Disk Controller
```

Device Configuration Inputs

The following sections describe the generation inputs required for many common peripherals. This information is organized by driver type. Most drivers supplied with the RTE-6/VM HP 92084A primary disk are discussed. For other drivers, consult the appropriate driver manuals, subsystem manuals or configuration guides.

Table 4-2 correlates peripherals and interface cards with their respective drivers. Note that certain devices may be supported by more than one driver and interface card. In this case, select the driver which supports the interface card in your configuration.

Note It is recommended that your I/O configuration worksheet be filled in before specific generator device configuration inputs are made. This worksheet will list the interface cards and LU accessible devices in your system, together with their select code, logical unit, and Equipment Table entry assignments.

Automatic Output Buffering

In the following sections, many of the EQT entry definitions specify the B (output buffering) parameter for devices. This parameter is optional but recommended. It causes the system to buffer output for the device into SAM. This will allow device output operations to proceed concurrently with program execution. It will also allow programs to be swapped out during output operations since buffers will not be in program partitions.

The buffering feature has no effect on input operations. Therefore, it makes no sense to specify it for input only devices (for example, card readers). This feature must NOT be enabled for disk devices.

The number and type of buffered devices in your system affects the amount of SAM required in your system. Refer to the Online Generator Manual for SAM generation Considerations.

The output buffering feature can be enabled/disabled on line via the system EQ command.

Table 4-2. Peripheral Device Interface Cards and Drivers

Device	Interface Card	Driver	Driver P/N
TERMINALS			
2382A Alpha Terminal			
2392A Display Terminal			
2393A Monochrome Graphics Terminal			
2397A Color Graphics Terminal			
2600A Terminal	12531/12880A	DVR00	
2600A Terminal	12792B	*	
2615A Terminal	12531/12880A	DVR00	
2615A Terminal	12792B	*	
2621A/P Terminal	12531D/12880A	DVR00	
2621A/P Terminal	12792B	*	
2621A/P Terminal	12966A	DVR05	
2621A/P Terminal (Modem)	12966A-002	DVA05	
2622A Data Entry Terminal			
2623A Graphics Terminal			
2624B Data Entry Terminal			
2625A IBM 3270 Terminal			
2626A Display Station			
2627A Color Graphics Terminal			
2628A HP WORD Terminal			
2635A Terminal	12531D/12880A	DVR00	
2635A Terminal	12792B	*	
2635A Terminal	12966A	DVR05	
2635A Terminal (Modem)	12966A-002	DVA05	
264x Terminal	12531D/12880A	DVR00	
264x Terminal	12792B	*	
264x Terminal	12966A	DVR05	
264x Terminal (Modem)	12966A-002	DVA05	
45610B HP 150 (as 2623A)			
45850/51A HP 150-II			
700/92 Terminal			
700/96 Terminal			
700/98 Terminal			
3081A Rugged Terminal			
91730A Multipoint Terminal	12970A	DVR07	
PRINTERS/PLOTTERS			
2225D Thinkjet			
2562C Line Printer			
2563A/B/C Line Printer			
2564A/B/C Line Printer			
2565A Line Printer			
2566A/B/C Line Printer			
2607A Line Printer	12845A/B	DVA12	
2608A Line Printer	26099A	DVB12	
2608S Graphics Printer	12821	DVC12	
2610A Line Printer	12845A/B	DVA12	
2613A Line Printer	12845B	DVA12	
2614A Line Printer	12845A/B	DVA12	
2617A Line Printer	12845B	DVA12	
2618A LIne Printer	12845B	DVA12	
2619A Line Printer	12845B	DVA12	
2631A Line Printer	12531D/12880A	DVR00	
2631A Line Printer	12845B	DVA12	

* Two possible configurations are available with the HP 12792B MUX: DDT05 and DVT00 for 2-page
DDV05 and DVM00 for 3-page

Table 4-2. Peripheral Device Interface Cards and Drivers (cont.)

Device	Interface Card	Driver	Driver P/N
PRINTERS (continued)			
2671G Thermal Printer			
2673G Thermal Printer			
2686A/+ Laser Printer			
2752A Teleprinter	12531C	DVR00	
2754A Teleprinter	12531C	DVR00	
2767A Line Printer	12653	DVR12	
2932A Printer			
2934A Printer			
82906A Impact Printer			
9866A Thermal Printer	12566	DVR00	
PLOTTERS			
7440A Plotter			
7470A Plotter			
7475A Plotter			
7550A Plotter			
7570A Plotter			
7580A/B Plotter			
7585A/B Plotter			
7586B Plotter			
DISK DRIVES			
7900/7901 Disk	13210A	DVR31	
7905/06/20/25 MAC Disk	13175	DVR32	
7905/06/20/25 MAC Disk #2	13175	DVP32	
7906H/20H/25H Disk	12821	DVA32	
7906H/20H/25H Disk #2	12821	DVC32	
CS/80 Disks	12821A	DVM33	
9121 Microfloppy Disk Drive	12821	DVA32	
9885A/M Floppy Disk	12732A/12733A	DVR33	
9895 Disk	12821	DVA32	
9895 Disk #2	12821	DVC32	
TAPE DRIVES			
7970E 9-Track Magnetic Tape Drive	13181A/13183A	DVR23	
7971A Magnetic Tape Drive			
7974A Magnetic Tape Drive	12821A	DVS23	
7978B Magnetic Tape Drive			
7979A Magnetic Tape Drive			
7980A Magnetic Tape Drive			
9144A CS/80 Cartridge Tape Drive			
1300H HP-IB DAT Tape Drive (C1511A)			

Table 4-2. Peripheral Device Interface Cards and Drivers (cont.)

Device	Interface Card	Driver	Driver P/N
MISCELLANEOUS			
12556A Universal Interface	12556A	DVM72	
12566B Universal Interface	12566B	DVM72	
12566B 40-Bit Register	12566B	DVR54	
12604A/B Universal Interface	12604A/B	DVM72	
12604B Data Source	12604B	DVR40	
12661A Universal Interface	12661A	DVM72	
12665 DS/1000 Hard Link	12665A	DVA65	
12770 Coupler	12665A	DVR66	
12771Serial Link Kit	12665A	DVR65	
12773 DS/1000 Modem Link	12773A	DVA65	
12773 Modem Link Kit	12773A	DVR65	
12889 DS/3000 Link	12889A	DVG67	
12978/13197 WCS	12978A/13197A	DVR36	
2313B A/D Converter	2314-60020	DVR62	
2320A Data Acquisition	2320A	DVR76	
2570A/2575A Coupler	12665A	DVR66	
2570A/2575A Coupler	12773A	DVR66	
2761A Card Reader	12602B	DVR15	
2895B Paper Tape Punch	12597	DVR00	
3840D/3845A DVM Scanner	28037	DVR45	
59310B HP-IB Interface	59310B	DVA37	
6129/30/31 Voltage Source	12661A	DVR70	
6940B/6941B Multi-Progr.	14550A/12665A	DVA72	
91200B TV Interface	91200B	DVA13	
91780 RJE/1000	12618A	DVR50	
92900A Serial Link	40280A	DVA47	
93012A Meter/Scanner	28037/2116-6123	DVR47	

Table 4-2. Peripheral Device Interface Cards and Drivers (cont.)

DEVICE	I/F CARD	DRIVER	DVR P/N
12556A UNIV. INTERFACE	12556A	DVM72	09580-16079
12566B UNIV. INTERFACE	12566B	DVM72	09580-16079
12566B 40 BIT REGISTER	12566B	DVR54	25117-93001
12604A/B UNIV. INTERFACE	12604A/B	DVM72	09580-16079
12604B DATA SOURCE	12604B	DVR40	29100-93001
12661A UNIV. INTERFACE	12661A	DVM72	09580-16079
12665 DS/1000 HARD LINK	12665A	DVA65	91750-16105
12770 COUPLER	12665A	DVR66	29004-93003
12771 SERIAL LINK KIT	12665A	DVR65	12665-93001
12773 DS/1000 MODEM LINK	12773A	DVA65	91750-16105
12773 MODEM LINK KIT	12773A	DVR65	12774-90001
12889 DS/3000 LINK	12889A	DVG67	91750-16108
12978/13197 WCS	12978A/13197A	DVR36	13197-16001
2313B A/D CONVERTER	2314-60020	DVR62	29009-93001
2320A DATA ACQUISITION	2320A	DVR76	02320-93002
2570A/2575A COUPLER	12665A	DVR66	29004-93003
2570A/2575A COUPLER	12773A	DVR66	29004-93001
+ 2600A TERMINAL	12531/12880A	DVR00	92084-16637
2607A LINE PRINTER	12845A/12845B	DVA12	92001-16020
2608A LINE PRINTER	26099A	DVB12	92062-16004
2608S GRAPHICS PRINTER	12821	DVC12	92068-16110
2610A LINE PRINTER	12845A/12845B	DVA12	92001-16020
2613A LINE PRINTER	12845B	DVA12	92001-16020
2614A LINE PRINTER	12845A/12845B	DVA12	92001-16020
+ 2615A TERMINAL	12531/12880A	DVR00	92084-16637
2617A LINE PRINTER	12845B	DVA12	92001-16020
2618A LINE PRINTER	12845B	DVA12	92001-16020
2619A LINE PRINTER	12845B	DVA12	92001-16020
+ 2621A/P TERMINAL	12531D/12880A	DVR00	92084-16637
2621A/P TERMINAL	12966A	DVR05	92001-16028
2621A/P TERMINAL (MODEM)	12966A-002	DVA05	92084-16607
2631A LINE PRINTER	12531D/12880A	DVR00	92084-16637
2631A LINE PRINTER	12845B	DVA12	92001-16020
+ 2635A TERMINAL	12531D/12880A	DVR00	92084-16637
2635A TERMINAL	12966A	DVR05	92001-16028
2635A TERMINAL (MODEM)	12966A-002	DVA05	92084-16607
2640A/B TERMINAL	12531D/12880A	DVR00	92084-16637
+ 2640B TERMINAL	12966A	DVR05	92001-16028
2640B TERMINAL (MODEM)	12966A/-002	DVA05	92084-16607

+ These terminals can use the MUX terminal drivers of the HP 12792B Multiplexer Subsystem. Two possible configurations are available: DDT05 and DVT00 for 2 page, and DDV05 and DVM00 for 3 page.

Table 4-2. Peripheral Device Interface Cards and Drivers (cont.)

DEVICE	I/F CARD	DRIVER	DVR P/N
+*2645A TERMINAL	12531D/12880A	DVR00	92084−16637
2645A TERMINAL	12966A	DVR05	92001−16027
2645A TERMINAL (MODEM)	12966A-002	DVA05	92084−16607
+*2647A TERMINAL	12531D/12880A	DVR00	92084−16637
2647A TERMINAL	12966A	DVR05	92001−16027
2647A TERMINAL (MODEM)	12966A-002	DVA05	92084−16607
+ 2648A TERMINAL	12531D/12880A	DVR00	92084−16637
* 2648A TERMINAL	12966A	DVR05	92001−16027
2648A TERMINAL (MODEM)	12966A-002	DVA05	92084−16607
+ 2752A TELEPRINTER	12531C	DVR00	92084−16637
2754A TELEPRINTER	12531C	DVR00	92084−16637
2761A CARD READER	12602B	DVR15	12602−90023
2767A LINE PRINTER	12653	DVR12	29028−90002
2895B PAPER TAPE PUNCH	12597	DVR00	92084−16637
3840D/3845A DVM SCANNER	28037	DVR45	91062−93003
59310B HPIB INTERFACE	59310B	DVA37	92084−16593
6129/30/31 VOLTAGE SOURCE	12661A	DVR70	25117−93005
6940B/6941B MULTIPROGR.	14550A/12665A	DVA72	29100−93003
7900/7901 DISC	13210A	DVR31	92067−16466
7905/06/20/25 MAC DISK	13175	DVR32	92067−16330
7905/06/20/25 MAC DISK #2	13175	DVP32	92067−16508
7906H/20H/25H/9895 DISK	12821	DVA32	92064−16553
7906H/20H/25H/9895 DISK #2	12821	DVC32	92067−16506
CS/80 DISKS	12821A	DVM33	92084−16650
7970 9 TRACK MAG TAPE	13181A/13183A	DVR23	92202−16001
7974A MAG TAPE	12821A	DVS23	92084−15050
91200B TV INTERFACE	91200B	DVA13	91200−16001
91730A MULTIPOINT TERM.	12970A	DVR07	91730−16001
91780 RJE/1000	12618A	DVR50	91780−90006
92900A SERIAL LINK	40280A	DVA47	92900−90005
93012A METER/SCANNER	28037/2116-6123	DVR47	93012−93001
9885A/M FLOPPY DISK	12732A/12733A	DVR33	92084−16713
9866A THERMAL PRINTER	12566	DVR00	92084−16637
9121 MICRO FLOPPY	12821	DVA32	92084−16708

* Use DVA05 hardwired for CTU equipped terminals 2645A , 2648A, and 2647A.

+These terminals can use the MUX terminal drivers of the HP 12792B Multiplexer Subsystem. Two possible configurations are available: DDT05 and DVT00 for two page, and DDV05 and DVM00 for three page.

Terminal Driver DVR00

This terminal driver is available in the following relocatable module:

```
16d

    REL,%DVR00              *Terminal/Reader/Punch Driver
```

The following information should be generated into the system for each terminal supported by driver DVR00:

```
20
    EQUIPMENT TABLE ENTRY

    EQT nn?

    sc,DVR00,T=30000,B      *Terminal EQT

21
    DEVICE REFERENCE TABLE

    LU = EQT # ?

    nn,type                 *Terminal # LU

22
    INTERRUPT TABLE

    sc,PRG,PRMPT            *Terminal #
```

where:

nn	is the EQT number assigned to the terminal.
LU	is the logical unit number assigned to the terminal.
type	is the device type. For 262X, 263S, and 264X terminals, type = 1. Otherwise, type = 0.
sc	is the select code of the terminal interface card.

For the system console (LU 1), and all dedicated terminals not to be handled by Session Monitor or MTM, the Interrupt Table entry above should be changed to:

```
22

    INTERRUPT TABLE

    sc,EQT,nn               *System Console (dedicated terminal)
```

The EQT timeout (T) determines the length of time programs can wait for terminal inputs before the driver issues a zero-length record. If the timeout is set too small, the operator may not have enough time to enter the required response. Long EQT timeouts may unnecessarily tie up system resources. In the Session Monitor, if no operator input is received during a period of five consecutive system timeouts, the system will automatically log the user off. The recommended T value of 30000 therefore, will allow 25 minutes before automatic log off. The timeout can be increased/decrease by adjusting the T parameter during generation or via the TO command. The value of T should not be less than 500. If the T parameter is omitted, timeouts will not occur on the terminal.

Refer to the *DVR05/DVA05 Driver Manual*, part number 92001-90015, for more detailed configuration information.

Note The most common generating error is matching the select code entry in the interrupt table to the EQT. Go back and check that the select code number is for the EQT that you specified.

Paper Tape Reader Driver DVR01

The paper tape photoreader driver is actually a part of driver DVR00. If relocatable %DVR00 is already specified in your worksheets, skip the following step.

16d

```
    REL,%DVR00              *DVR00/DVR01/DVR02 Driver
```

The recommended EQT, DRT, and Interrupt Table entries for each photoreader in your system are as follows:

20

```
    EQUIPMENT TABLE ENTRY

    EQT nn?

    sc,DVR01,T=xxx          *Photoreader LU
```

21

```
    DEVICE REFERENCE TABLE

    LU = EQT # ?

    nn,6                    *Photoreader LU
```

22

```
    INTERRUPT TABLE

    sc,EQT,nn               *Photoreader
```

where:

sc	is the select code of the photoreader interface card.
LU	is the assigned logical unit.
nn	is the assigned EQT entry number.
xxx	is the photoreader timeout value. A value of 300 to 500 (4-5 seconds) is recommended to allow reading of long leaders. You may wish to increase this value to allow the operator a few moments to realize there is no tape motion and ready the device before the timeout period expires (otherwise the system will set the device down).

For more information on photoreader configuration, refer to the *Driver DVR00 Manual*, part number 29029-95001.

Paper Tape Punch Driver DVR02

The paper tape punch driver is actually a part of driver DVR00. If relocatable %DVR00 is already specified in your worksheets, skip the following step.

16d

```
REL,%DVR00              *DVR00/DVR01/DVR02 Driver
```

The recommended EQT, DRT, and Interrupt Table entries for each paper tape punch in your system are as follows:

20

```
EQUIPMENT TABLE ENTRY

EQT nn?

sc,DVR02,B,T=xxx
```

21

```
DEVICE REFERENCE TABLE

LU = EQT # ?

nn,4
```

22

```
INTERRUPT TABLE

sc,EQT,nn
```

where:

sc	is the select code of the punch interface card.
LU	is the assigned output logical unit.
nn	is the assigned EQT entry number.
xxx	is the punch timeout value. A minimum value of 500 (5 seconds) is suggested to allow the operator a few moments to ready the device before the timeout period expires (and the system sets the device down).

For more information on punch configuration, refer to the Driver DVR00 Manual.

Terminal Driver DVR05/DVA05

This driver is available in one of two relocatable modules. One (and ONLY ONE) of these modules should be generated into the system.

For any HP 26xx terminals with cartridge tape units (CTUs), printers, or graphics, connected by modem or hardwired link:

```
16d

    REL,%DVA05              *DVA05 Complete Version (Modem/Periph. Support)
```

For a 26xx terminal without CTUs, printer, or graphics, connected by hardwired link:

```
16d

    REL,%0DV05             *DVR05 Minimum Version (hardwired/keyboard only)
```

The recommended EQT, DRT, and Interrupt Table definitions for each DVR05/DVA05 terminal in your system are as follows:

```
20
    EQUIPMENT TABLE ENTRY

    EQT nn?

    sc,DVR05,X=13,T=30000,B        *Terminal EQT

       (or sc,DVA05,X=13,T=30000,B for complete version)
21
    DEVICE REFERENCE TABLE

    LU = EQT # ?

    nn,                            *Keyboard Display LU

    LU = EQT # ?

    nn,1                           *Left CTU LU (optional)

    LU = EQT # ?

    nn,2                           *Right CTU LU (optional)

    LU = EQT # ?

    nn,3                           *Graphics LU (optional)

    LU = EQT # ?

    nn,4                           *Auxiliary Printer LU (optional)
22
    INTERRUPT TABLE

    sc,PRG,PRMPT                   *Terminal #
```

where:

nn — is the EQT number assigned to the terminal.

LU — are the logical unit numbers assigned to the terminal and associated peripheral devices. The assignments need not be to contiguous LUs. For LU assignment restrictions, refer to the Logical Unit Assignments section in this chapter.

sc — is the select code assigned to terminal interface card.

For the system console (LU 1), and all dedicated application terminals not being handled by Session Monitor or MTM, the Interrupt Table entry above should be changed to:

```
22
   INTERRUPT TABLE

   sc,EQT,nn              *System Console (dedicated terminal)
```

In the above inputs, DVA05 should be substituted for DVR05 in the EQT definition(s) if relocatable %DVA05 will be generated into the system.

The EQT timeout (T) determines the length of time programs can wait for terminal input before the driver issues a zero length record. If the timeout value is set too small, the operator may not have enough time to enter the required response. Long EQT timeouts may unnecessarily tie-up system resources. In the Session Monitor, if no operator input is received during a period of five consecutive system timeouts, the system will automatically log the user off. The recommended T value of 30000 therefore, will allow 25 minutes before automatic log off. The timeout can be increased/decreased by adjusting the T parameter during generation or via the TO command on line. The value of T should not be less than 500. If the T parameter is omitted, timeouts will not occur on the terminal.

Refer to the *DVR05/DVA05 Driver Manual*, part number 92001-90015, for more detailed configuration information.

Line Printer Driver DVR12

The HP 2767A line printer driver relocatable module is:

```
16d
    REL,%DVR12               *2767 Line Printer Driver
```

The recommended EQT, DRT, and Interrupt Table entries for each HP 2767 line printer in your system are as follows:

```
20
    EQUIPMENT TABLE ENTRY

    EQT nn?

    sc,DVR12,B,T=200         *2767 Line Printer EQT

21
    DEVICE REFERENCE TABLE

    LU = EQT # ?

    nn,                      *2767 Line Printer LU

22
    INTERRUPT TABLE

    sc,EQT,nn                *2767 Line Printer LU
```

where:

sc	is the select code of the line printer interface card.
LU	is the assigned logical unit.
nn	is the assigned EQT number.

a timeout value of two seconds (T=200) is recommended to accommodate printer top of form operations.

Line Printer Driver DVA12

This line printer driver is found in the following relocatable module:

```
16d

    REL,%DVA12              *26XX Line Printer Driver
```

The recommended EQT, DRT, and Interrupt Table entries for each line printer in your system using driver DVA12 are as follows:

```
20
    EQUIPMENT TABLE ENTRY

    EQT nn?

    sc,DVA12,B,T=xxx         *26xx Line Printer EQT

21
    DEVICE REFERENCE TABLE

    LU = EQT # ?

    nn,                      *26xx Line Printer LU

22
    INTERRUPT TABLE

    sc,EQT,nn                *26xx Line Printer
```

where:

sc — is the select code of the line printer interface card.

LU — is the assigned logical unit.

nn — is the assigned EQT entry number.

xxx — is the line printer timeout value. This should reflect the time it takes the printer to do a top of form operation. This value will depend on the type of line printer. Recommended timeout values are shown below.

2607A	200 lpm	600
2610A	200 lpm	200
2613A	300 lpm	120
2614A	600 lpm	100
2617A	600 lpm	100
2618A	1250 lpm	100
2619A	1000 lpm	100
2631A	180 cps	300

For more information on DVA12 printer configuration, refer to the *DVA12 Driver Manual*, part number 92001-90010.

Line Printer Driver DVB12

The HP 2608A line printer driver is found in the following relocatable module:

16d

```
REL,%DVB12               *2608 Line Printer Driver
```

The recommended EQT, DRT, and Interrupt Table entries for each 2608 line printer in your system are as follows:

20

```
EQUIPMENT TABLE ENTRY

EQT nn?

sc,DVB12,B,X=5           *2608 Line Printer EQT
```

21

```
DEVICE REFERENCE TABLE

LU = EQT # ?

nn,pwr                   *2608 Line Printer LU

LUP = EQT # ?

nn,3                     *Character Set Read Back LU (optional)
```

22

```
INTERRUPT TABLE

sc,EQT,nn                *2608 Line Printer
```

where:

sc	is the select code of the line printer interface card.
LU	is the assigned output logical unit.
pwr	indicates how the driver is to process power failures. If pwr=0, the driver will, after a power failure, attempt to restore the printer to its previous state (as much as possible) and resume output operations. If pwr=1, the driver will set the printer OFF LINE after a power failure.
nn	is the assigned EQT number.
LUP	is the assigned read back logical unit. This should be included if you are using the HP 9284A graphics package or you are spooling to the line printer.

Note that the driver automatically handles device timeout functions. Refer to the *DVB12 Driver Manual*, part number 92062-90004, for more details on HP 2608 configuration.

Line Printer Driver DVC12

The line printer driver is found in the following relocatable module:

```
16d
    REL,%DVC12                *Line Printer Driver
```

The recommended EQT, DRT, and Interrupt Table entries for each line printer in your system are as follows:

```
20
    EQUIPMENT TABLE ENTRY

    EQT nn?

    sc,DVC12,B                *Line Printer EQT for the 1st Printer
    sc,DVD12,B                *Line Printer EQT for the 2nd Printer

21
    DEVICE REFERENCE TABLE

    LU = EQT # ?

    nn

22
    INTERRUPT TABLE

    sc,EQT,nn                 *Line Printer
```

where:

sc	is the select code of the line printer interface card.
LU	is the assigned output logical unit.
nn	is the assigned EQT number.

Refer to the *DVC12 Line Printer Driver Reference Manual*, part number 92068-90022, for more details on line printer configuration.

TV Interface Driver DVA13

The TV interface driver is found in the following relocatable module.

```
16d

    REL,%DVA13              *91200B TV Interface Driver
```

If you are using the TV interface library described in the 91200B Programming and Operating Manual, the following relocatable module should be generated into your system.

```
16d

    REL,%TVLIB              *91200B TV Interface Driver
```

When your new system is operational, the TV interface may be tested with the TV verifier program (found in relocatable module %TVVER). It is recommended that this program be relocated online.

The recommended EQT, DRT, and Interrupt Table entries for each 91200B TV interface in your system are as follows:

```
20
    EQUIPMENT TABLE ENTRY

    EQT nn?

    sc,DVA13,D,T=4          *91200B TV Interface EQT

21
    DEVICE REFERENCE TABLE

    LU = EQT # ?

    nn                      *91200B TV Interface LU

    LUB = EQT # ?

    nn,2                    *B/W Mode LU (optional)

22
    INTERRUPT TABLE

    sc,EQT,nn               *91200B TV Interface
```

where:

sc — is the select code of the TV interface card. This is the single card in a Black and White (B/W) system or the master card (card A, red) of a color system.

LU	is the assigned logical units.
nn	is the assigned EQT entry number.
LUB	is the assigned B/W mode LU. It provides the user with a convenient means of checking the black and white appearance of a color program. This will only be useful in multiple card systems where users desire this capability.

European systems and certain output operations may require a timeout value larger than 40 milliseconds. For more information refer to the *DVA13 Driver Reference Manual*, part number 91200-90005.

Magnetic Tape Drive DVR23

The HP 7970 9-track magnetic tape driver is contained in the following relocatable module:

16d

```
    REL,%DVR23              *7970 Magnetic Tape Driver
```

The recommended EQT, DRT, and Interrupt Table entries for each magnetic tape controller in your system are as follows:

20

```
    EQUIPMENT TABLE ENTRY

    EQT nn?

    sc,DVR23,[B],T=700      *Magnetic Tape Controller EQT
```

21

```
    DEVICE REFERENCE TABLE

    LU = EQT # ?

    nn,unit                 *Magnetic Tape Controller Unit LU
      .
      .
      .
```

22

```
    INTERRUPT TABLE

    sc,EQT,nn               *Magnetic Tape Controller Lower sc
    sc+1,EQT,nn             *Magnetic Tape Controller Upper sc
```

where:

sc	is the select code of the lower magnetic tape interface card. (sc+1 is the select code of the upper card.)
LU	is the assigned logical unit. There should be a logical unit configured in your system for each magnetic tape unit.
unit	is the magnetic tape unit number (ranging from 0 to 3).
nn	is the assigned controller EQT entry number.

A device timeout of 7 seconds (T=700) will meet the ANSI standard of 25 feet of blank tape read before determining the tape to be blank. Timeouts are not used for tape positioning; timeouts are used only for reading and writing. The D option on the EQT entry (which specifies that a DCPC channel is required by the driver) is not required since DVR23 will allocate a DCPC channel automatically. The B (buffered) option is not required but is usually set to increase the throughput for short records. It can, however, have the adverse effect of causing some utilities to suspend due to lack of System Available Memory (SAM). HP supplied utilities use the "unbuffered bit" in their EXEC calls for large record transfers to eliminate this condition. For more information on magnetic tape configuration, refer to the *DVR23 Driver Reference Manual*, part number 92202-93001.

Magnetic Tape Drive DVS23

The HP 7974/7978 nine track magnetic tape driver is contained in the following relocatable module:

```
16d

    REL,%DVS23               *7974/7978 Magnetic Tape Driver
```

The recommended EQT, DRT, and Interrupt Table entries for each magnetic tape controller in your system are as follows:

```
20
    EQUIPMENT TABLE ENTRY

    EQT nn?

    sc,DVS23,D,X=yy          *Magnetic Tape Controller EQT

21
    DEVICE REFERENCE TABLE

    LU = EQT # ?

    nn,unit                  *Magnetic Tape Controller Unit LU
      .
      .
      .

22
    INTERRUPT TABLE

    sc,EQT,nn                *Magnetic Tape Controller
```

where:

sc — is the select code of the magnetic tape interface card. (sc+1 is the select code of the upper card.)

LU — is the assigned logical unit. There should be a logical unit configured in your system for each magnetic tape unit.

unit — is the subchannel (HP-IB address) for multiple magnetic tape units on the HP-IB but (ranging from 0 to 7).

nn — is the assigned controller EQT entry number.

yy — is the size of the EQT extension. Calculate the number of extension words required as follows:

Number of EQT extension words = 3* (the highest subchannel # which will be used +1) +2.

Since the maximum number of subchannels is eight, the maximum size of the EQT extension is 3 * (7+1)+2 = 26 words. The minimum size of the EQT extension is 3*(0+1)+2=5 words. For more information on magnetic tape configuration, refer to the *DVS23 for HP 7974/7978 Magnetic Tape Subsystem Installation and Programming Manual*, part number 92084-90040.

Disk Driver DVR32

The driver for the HP 13037B/C Multiple Access Controller (MAC) disks (HP 7905/06/20/25) is contained in the following relocatable module:

```
16d
    REL,%DVR32              *79XX Disk Driver
```

The recommended EQT, DRT, and Interrupt Table entries for the disk controller are:

```
20
    EQUIPMENT TABLE ENTRY

    EQT nn?

    sc,DVR32,D              *13037B/C Disk Controller EQT

21
    DEVICE REFERENCE TABLE

    LU = EQT # ?

    nn,sub                  *79xx Disk Subchannel
     .
     .
     .

22
    INTERRUPT TABLE

    sc,EQT,nn               *13037B/C Disk Controller
```

where:

sc — is the select code of the disk.

LU — is an assigned disk subchannel logical unit. A logical unit must be configured into your system for each subchannel, or the corresponding disk space will be inaccessible. The subchannel definitions and number should be determined from the disk worksheets filled out earlier in this chapter.

sub — is the disk subchannel number. It must be in the range 0 through 63.

nn — is the assigned disk controller EQT entry number. The system disk controller should be assigned EQT entry #1.

Note RT6GN will always build one track map table for all disks on the same controller as the system subchannel. When the HP 13037B/C is being used with peripheral disk subchannels (non-system) it is the user's responsibility to build the appropriate track map table ($TB32) and generate it into the system.

If a multiple HP 13037B/C configuration is desired, the user must include in the generation a track map table and a renamed version of DVR32 for each disk controller that does not contain the system subchannel. HP supplies one renamed version of the driver (DVP32) which may be generated into the system along with the correct Track Map Table ($TP32). Include entries in the Equipment Table, Device Reference Table, and Interrupt Table for the second controller. Refer to Appendix A for additional details.

Multiple Controller operation will allow an I/O operation to be active on each controller at the same time, thus providing greater throughput in systems that make intensive use of disk I/O.

If a disk is to be a system disk, it should be assigned EQT 1.

For more information on DVR32 configuration, refer to the *DVR32/DVA32 Driver Manual*, part number 92068-90012.

Disk Driver DVA32

The driver for the ICD disks (HP 7906H, HP 7920H, HP 7925H, and HP 9895) is contained in the following relocatable module:

```
16d

   REL,%DVA32              *79xxH Disk Driver
```

Note Refer to the *DVA32/DVR32 Driver Manual*, part number 92068-90012, for additional information concerning the correct timeout value for your system. If a timeout is not specified, the default value of 2 seconds will be assumed by DVA32.

The recommended EQT, DRT, and Interrupt Table entries for the disk controller are:

```
20
    EQUIPMENT TABLE ENTRY

    EQT nn?

    sc,DVA32,D,T=100          *12821A Disk Interface EQT

21
    DEVICE REFERENCE TABLE

    LU = EQT # ?

    nn,sub                    *79xxH Disk Subchannel
     .
     .
     .

22
    INTERRUPT TABLE

    sc,EQT,nn                 *12821A Disk Interface
```

The recommended EQT, DRT, and Interrupt Table entries for the disk controller are:

where:

sc	is the select code of the disk.
LU	is an assigned disk subchannel logical unit. A logical unit must be configured into your system for each subchannel, or the corresponding disk space will be inaccessible. The subchannel definitions and number should be determined from the disk worksheets filled out earlier in this chapter.
sub	is the disk subchannel number. It must be in the range 0 through 63.
nn	is the assigned disk controller EQT entry number. The system disk controller should be assigned EQT entry #1.

Note RT6GN will always build one track map table for all ICD disks on the same 12821A interface card as the system disk. When the 12821A is being used with peripheral (non-system) disk subchannels, it is the user's responsibility to build the appropriate track map table ($TA32) and generate it into the system.

If multiple HP 12821A cards are to be used, the user must supply a track map table and a renamed version of DVA32 for each additional card that does not contain the system subchannel. HP supplies one renamed version of the driver (DVC32) which may be generated into the system along with the correct Track Map Table ($TC32). Include entries in the Equipment Table, Device Reference Table, and Interrupt Table for additional interface cards. Refer to Appendix A for additional details.

Multiple ICD interface cards will allow an I/O operation to be active on each card at the same time, thus providing greater throughput in systems that make intensive use of disk I/O.

If a disk is to be the system disk, it should be assigned EQT 1.

ICD disks managed by DVA32 and CS/80 disks managed by DVM33 cannot be on the same HP 12821A card.

Disk Driver DVM33

The driver for CS/80 disks is contained in the following relocatable module:

```
16d
    REL,%DVM33               *CS/80  Disk Driver
```

Note The recommended timeout value for the CS/80 driver is at least 2 seconds. If using a CTD, this should be increased to 90 seconds (to handle rewind time). If a timeout is not specified, the default value of 2 seconds will be assumed by DVM33. Refer to the DVM33 Driver Manual for additional information concerning the correct timeout for your system.

The recommended EQT, DRT, and Interrupt Table entries for the disk controller are:

```
20
    EQUIPMENT TABLE ENTRY

    EQT nn?

    sc,DVM33,D,T=200        *12821A Disk Interface EQT

21
    DEVICE REFERENCE TABLE

    LU = EQT # ?

    nn,sub                  *CS/80 Disk Subchannel
      .
      .
      .

22
    INTERRUPT TABLE

    sc,EQT,nn               *12821A Disk Interface
```

where:

sc	is the select code of the disk.
LU	is an assigned disk subchannel logical unit. A logical unit must be configured into your system for each subchannel, or the corresponding disk space will be inaccessible. The subchannel definitions and number should be determined from the disk worksheets filled out earlier in this chapter.
sub	is the disk subchannel number. It must be in the range 0 through 63.
nn	is the assigned disk controller EQT entry number. The system disk controller should be assigned EQT entry #1.

Note RT6GN will always build one track map table for all CS/80 disks on the same 12821A interface card as the system disk. When the 12821A is being used with peripheral (non-system) disk subchannels, it is the user's responsibility to build the appropriate track map table ($TA32) and generate it into the system.

If multiple HP 12821A cards are to be used, the user must supply a track map table and a renamed version of DVM33 for each additional card that does not contain the system subchannels. HP supplies one renamed version of the driver (DVN33) which may be generated into the system along with the correct Track Map Table ($TN33). Include entries in the Equipment Table, Device Reference Table, and Interrupt Table for additional interface cards. Refer to Appendix A for additional details.

Multiple HP 12821A interface cards will allow an I/O operation to be active on each card at the same time, thus providing greater throughput in systems that make intensive use of disk I/O.

If a disk is to be the system disk, it should be assigned EQT 1.

Disks managed by DVM33 or DVA32 cannot be on the same 12821A card.

The program CSERR must be relocated as a memory-resident program in the generation when DVM33 or DVN33 is included in the generation. CSERR is an extremely useful program for identifying potential hardware and disk configuration problems. This program is scheduled by the driver when the CS/80 device returns a Fault error, a Reject error, or an Access error (as explained in the DVM33/DVN33 Driver Reference Manual). CSERR gets this status information from a 20-byte area in Table Area I at entry point $CS80. The drive status information is placed here on most error conditions, except as described below:

1. If the request results in a channel error, the driver does not return status in Table Area 1 since a channel error could make the status invalid and therefore misleading. (The status is the EQT should indicate a channel error has occurred in this case).
2. If the request was to a Cartridge Tape Device, the driver does not return status in Table Area I.
3. If the request to the driver was through a user call to a Direct Disk Control subroutine, the driver does not return the status in Table Area I since the user program is expected to interpret and handle errors for these types of calls. The user program usually receives all of the bad status normally placed in $CS80, or can obtain this information by using certain Direct Disk Control subroutines. For a description of the Direct Disk Control Subroutines refer to the *DVM33/DVN33 Driver Reference Manual*, part number 92084-90025.

The format of the CSERR printout is shown below. Refer to the DVM/DVN33 Driver Reference Manual for more information on interpreting the status information presented.

```
SCODE x EQT xx SUBCH xx
ADDRESS  x  QSTAT  x
xxxxxx xxxxxx xxxxxx xxxxxx
xxxxxx xxxxxx xxxxxx xxxxxx
xxxxxx xxxxxx
```

The information at entry point $CS80 contains the information printed out in the last three lines shown above. The first two lines (SCODE, EQT, SUBCH, ADDRESS, and QSTAT) are all passed into CSERR when it is scheduled by the driver.

Disk Driver DVR33

The HP 9885 M/S flexible disk driver is contained in the following relocatable module:

16d

```
REL,%DVR33                *9885 Flexible Disk Driver
```

The recommended EQT, DRT, and Interrupt Table entries for each flexible disk controller in your system are as follows:

20

```
EQUIPMENT TABLE ENTRY

EQT nn?

sc,DVR33,D                *Flexible Disk Controller EQT
```

21

```
DEVICE REFERENCE TABLE

LU = EQT # ?

nn,unit                   *Flexible Disk Unit
  .
  .
  .
```

22

```
INTERRUPT TABLE

sc,EQT,nn                 *Flexible Disk Controller Lower sc
sc+1,EQT,nn               *Flexible Disk Controller Upper sc
```

where:

sc — is the select code of the lower interface card. (sc+1 is the select code of the upper card.)

LU — is the assigned logical unit. There should be a logical unit configured into your system for each disk unit.

unit — is the disk unit number. This will be the unit number of the master/slave drive according to the drive number set on the rear of the device. Each drive is set to a different number from 0 to 3.

nn — is the assigned disk controller EQT entry number.

The device timeout is automatically set by the driver. For more information, refer to the *DVR33 Driver Manual*, part number 12732-90001.

HP-IB Interface Driver DVA37

The HP-IB driver is supplied in two versions. The two drivers are identical except that one provides service request (SRQ) capability and the other does not. SRQ service is desirable if you need to activate application programs or BASIC programs on HP-IB device interrupts. The HP-IB driver with SRQ capability is contained in the following relocatable module:

```
16d

    REL,%6DV37              *HP-IB Driver With SRQ
```

Otherwise, if SRQ capability is not desired, relocate the following driver (under no circumstances do you relocate both):

```
16d

    REL,%6DVA37             *HP-IB Driver Without SRQ
```

The HP-IB utility subroutine and message library are found in the following relocatable module:

```
16d

    REL,$IB6A               *HP-IB Utility Routine and Message Library
```

If you have BASIC in your system, and have included the HP-IB driver with SRQ capability, and wish to handle HP-IB interrupts in BASIC; then enter the following inputs:

```
16d

    REL,%SRQ.P              *SRQ/TRAP Program for BASIC
    REL,%BAMLB              *BASIC Memory-Resident Library

17
    PARAMETERS

    TTYEV,17                *Memory-Resident W/SSGA
    TRAP,30                 *Put in SSGA
```

The recommended EQT, DRT, and Interrupt Table entries for each HP-IB controller in your system are:

```
20
    EQUIPMENT TABLE ENTRY

    EQT nn?

    sc,DVA37,T=xxx,X=yy      *HP-IB Controller EQT

21
    DEVICE REFERENCE TABLE

    LU = EQT # ?

    nn,0                     *HP-IB Line Control LU

    LUD = EQT # ?

    nn,unit                  *HP-IB Device Unit
      .
      .
      .
22
    INTERRUPT TABLE

    sc,EQT,nn                *HP-IB Controller
```

where:

sc	is the select code of the lower interface card.
LU	is the assigned line control logical unit.
LUD	is the assigned auto addressing logical units for devices. Devices not assigned on auto addressing LU must be addressed through the line control LU. You should configure a logical unit in your system for each device to be auto-addressed. It is suggested that you configure spare LUs in your system to handle additional devices added at a later date.
unit	is the hardware address of the device. It must be in the range 1 through 31.
nn	is the assigned EQT entry number for the interface card.
xxx	is the maximum device timeout for the slowest device on the bus.
yy	is the size of the EQT extension. Calculate the number of extension words required as follows:

Number of EQT extension words = 7n + 27.

where n = the number of auto-addressable devices to be connected to the HP-IB.

Be sure to include enough extension words to allow for adding devices to the system at a later date. Since the maximum number of devices on a bus is 31, the largest EQT extension size is 244 = 7 * 31 + 27 words.

Refer to the DVA37 Driver Manual for a more detailed discussion of HP-IB configuration.

HP 12792D Eight Channel Multiplexer

Two versions of the D-Mux driver are available. The driver names are DV800_0.REL and DV800_1.REL. DV800_1 is a superset of DV800_0. DV800_0 is unable to drive slave devices (the CTUs or internal or external printers). You cannot generate both drivers into the same system because you will get duplicate entry point names.

The typical user will use DV800_1.REL as both drivers will fit into a two-page driver partition.

```
16d

        Links in Base
        REL,DV800_1.REL
        Links in Current
            or
        REL,DV800_0.REL              * The D-Mux driver (see above for
                                       choice about which one to use)
```

Note Use the “Links in Base” command for DV800_1.REL and the normal “Links in Current” command for DV800_0.REL. The driver has been optimized to conserve base page links if it starts on a page boundary. This is accomplished by the “Links in Base” command.

The D-Mux driver does not require DMA, so the mux can be installed in any slot (above the privilege fence, if you have one) including an I/O Extender.

Generation and initialization have been simplified for the D-Mux due to the use of GEN records in the driver. These records specify details such as Eqt Extension size, buffering, timeout, etc.

The GEN records have a suffix which indicates with which port the EQT is associated. They also cause the proper choice of protocol based upon the name.

The GEN records that are available are:

`Hp_term:0..7`	CRT that uses HP Enq/Ack protocol, does not speed sense, PRMPT enabled, device type 5
`Term:0..7`	CRT that uses Xon/Xoff protocol, does speed sense, PRMPT enabled, device type 0
`HP_Printer:0..7`	non−interactive printers that use Hp Protocol 9600 baud, program scheduling disabled, device type 12
`Printer:0..7:`	non−interactive printers that use Xon/Xoff protocol 9600 baud, program scheduling disabled, device type 12
`Hp2635:0..7`	hardcopy terminal that uses Hp protocol, does speed sense, PRMPT is enabled, device type 6

The example given on the next page uses HP terminals on all eight ports:

```
20
    EQUIPMENT TABLE ENTRY

    EQT nn?

    sc,DV800,Gr=Hp_Term:0          *EQT nn    D-MUX Port 0

    EQT nn+1?

    sc,DV800,Gr=Hp_Term:1          *EQT nn+1  D-MUX Port 1

    EQT nn+2?

    sc,DV800,Gr=Hp_Term:2          *EQT nn+2  D-MUX Port 2
      .
      .
      .
    EQT nn+7?

    sc,DV800,Gr=Hp_Term:7          *EQT nn+7  D-MUX Port 7
```

To decide which GEN record to use, determine which of the modes above is closest to what is needed by the device you will connect to a given port. Probably the most general class is that of "Term:x", so it is a good choice for "black boxes." You can use control calls to tailor the port after it is booted. If you wish, you can choose to not use the default GEN records and to manually provide all the parameters in the EQT entry. The RTE-6/VM Generator allows you to preset the Eqt Extension words, so you can specify the auto-configure parameters yourself. For more information refer to the *RTE-6/VM Serial Driver Reference Manual*, part number 92084-90050, and the *RTE-6/VM Online Generator Reference Manual*, part number 92084-90010.

One parameter you may want to change from the GEN records would be the timeouts. You can do this by specifying the "T=xxxxx" field before the "Gr=" field (for example, sc,DV800,T=177776,Gr=Hp_Term:0). This gives the longest possible timeout, almost 11 minutes.

```
21 DEVICE REFERENCE TABLE

    lu  =  EQT #?
     nn,0                        * D-MUX Port - 0        <LU lu>

    lu+1 = EQT #?
     nn+1,0                      * D-MUX Port - 1        <LU lu+1>

    lu+2 = EQT #?
     nn+2,0                      * D-MUX Port - 2        <LU lu+2>
    .
    .
    .
    lu+7 = EQT #?
     nn+7,0                      * D-MUX Port - 7        <LU lu+7>

    lu+8 = EQT #?
     nn,1                        * Left CTU on Port 0    <LU lu+8>

    lu+9 = EQT #?
     nn,2                        * Right CTU on Port 0   <LU lu+9>


22 INTERRUPT TABLE

    sc,ent,MPRDV                 * D-MUX
```

HP 12792B/C Multiplexer Drivers

The MUX drivers of the HP 12792B/C Multiplexer Subsystem are available in one of two possible configurations: a two-page driver-partition version and a three-page driver partition version.

The typical user will usually generate the three-page version, which provides all the capabilities available on the MUX.

```
16d
    REL,%DVM00
    REL,%PVM00
    REL,%DDV05
    REL,%DDV12
    REL,%$DVTB
      .
      .
      .

17
    PARAMETERS

    PVM00,13
      .
      .
      .

20
    EQUIPMENT TABLE ENTRY

    EQT nn?

    sc,DVM00,B,X=31,T=30000
```

Eight (8) EQTs, one for each port, must be provided *unless* the MUX is being used with an HP 37214A Systems Modem. In that case, only seven EQTs are required because the eighth port is used by the driver to communicate with the modem controller.

```
21
    DEVICE REFERENCE TABLE
    lu = EQT # ?
    nn                        *Keyboard Display LU
    lu+1 = EQT # ?
    nn+1                      *Keyboard Display LU
      .
      .
      .
    lu+7 = EQT # ?
    nn+7                      *Keyboard Display LU
    LU = EQT # ?
    nn,1                      *Left CTU LU (optional)
    LU = EQT # ?
    nn,2                      *Right CTU LU (optional)
    LU = EQT # ?
    nn,3                      *Graphics LU (optional)
    LU = EQT # ?
    nn,4                      *Auxiliary LU (optional)
22
    INTERRUPT TABLE
    sc,PRG,PRMPT
```

CTUs, graphics LU, and auxiliary printers are allocated through additional LUs with the indicated subchannels on the EQT for that type.

If your system must remain at two-page driver partitions, then make the following changes only to step 16d:

```
16d
    REL,%DVT00          Note:  DDV12, Line Printer Driver, is omitted.
    REL,%PVM00                 This configuration also does not support
    REL,%DDT05                 CTUs or program MODEM.
    REL,%$DVTB
      .
      .
      .
    The remaining steps are the same.
```

Refer to the *HP 12792B 8-Channel Asynchronous Multiplexer Configuration Guide*, part number 5955-8868, for more detailed information.

Software Components and Resource Requirements

The following sections discuss the software modules, resources, and generation parameters required for the following components:

- Firmware Configuration
- RTE Operating System
- File Management System
- Spooling System
- Libraries
- Utilities
- Session Monitor
- Multi-Terminal Monitor
- Command Interpreter

Firmware Configuration

In HP 1000 M/E/F-Series computers, there are many subroutines implemented in firmware. Those subroutines implemented in firmware and their instruction opcode equivalents must be identified to the system by entering the RP commands in the CHANGE ENTS? section of the generation.

RT6GN command files are supplied with RTE-6/VM which contain RP commands for all the standard firmware available on M/E/F-Series computers for RTE-6/VM. Depending on the firmware installed on your system, these command files may be entered in your generation answer file using the TR command. For example, if your F-Series computer contains firmware for FFP, SIS, and VIS, you would put the following commands in your generation answer file.

```
18

   CHANGE ENTS?

    TR,]RT6OS        * RTE-6 Operating System firmware RP's

    TR,]F^FFP        * FFP Firmware RP's

    TR,]RT6VM        * RTE-6 VMA firmware RP's

    TR,]F^SIS        * SIS firmware RP's

    TR,]F^VIS        * VIS firmware RP's

    TR,]F^FPB        * Floating Point Box
```

Following is a list of standard M/E/F-Series firmware and the associated RT6GN command files.

Description **Command File**

M-Series
FFP - Fast FORTRAN processor ......]M^FPP

E-Series
RTE-6 Operating System firmware .....]RT60S
FFP - Fast FORTRAN Processor ......]E^FFP
VMA - Virtual Memory Access ....]RT6VM

F-Series
RTE-6 Operating System firmware .....]RT60S
FFP - Fast FORTRAN Processor ......]F^FFP
VMA – Virtual Memory Access ....]RT6VM
SIS – Scientific Instruction Set ....]F^SIS
VIS – Vector Instruction Set ....]F^VIS
Floating Point Box ....]F^FPB

On the following pages are listings of the supplied RT6GN command files with comments, where appropriate, on their use.

```
* ]M^FFP 92084-17273 REV.2440 <841005.1603>
*
* RTE-6/VM M-Series Fast Fortran Processor Firmware Entry Points
*
*
*     ***** FFP ENTRY POINTS *****
*
DBLE ,RP,105201          * CONVERT REAL TO EXTENDED REAL
SNGL ,RP,105202          * CONVERT EXTENDED REAL TO REAL
.DFER,RP,105205          * 3 WORD MOVE (EXTENDED REAL TRANSFER)
.XPAK,RP,105206          * NORMALIZE, ROUND AND PACK WITH EXPONENT
*                        *   AN EXTENDED REAL MANTISSA
.XCOM,RP,105215          * COMPLEMENT AN EXTENDED REAL UNPACKED
*                        *   MANTISSA IN PLACE
..DCM,RP,105216          * COMPLEMENT AN EXTENDED REAL
DDINT,RP,105217          * TRUNCATE AN EXTENDED REAL
.XFER,RP,105220          * 3 WORD MOVE (EXTENDED REAL TRANSFER)
.GOTO,RP,105221          * TRANSFER CONTROL TO LOCATION
..MAP,RP,105222          * CAL THE ADR OF A 2 OR 3D ARRAY ELEMENT
.ENTR,RP,105223          * TRANSFER THE TRUE ADDRESS OF PARAMETERS
*                        *   USED IN A SUBROUTINE CALL
.ENTP,RP,105224          * SAME AS .ENTR, EXCEPT MUST BE THIRD
*                        *   INSTRUCTION AFTER THE ENTRY POINT
.PWR2,RP,105225          * CALCULATE REAL X AND INTEGER N, Y=X*2**N
.FLUN,RP,105226          * UNPACK REAL (EXPONENT IN A, LOWER PART OF
*                        *   MANTISSA IN B)
*
$SETP,RP,105227          * SET UP A LIST OF POINTERS
*                        * NOTE: $SETP REPLACES .SETP AS OF 1913
*
.PACK,RP,105230          * CONVERT SIGNED MANTISSA OF REAL INTO
*                        *   NORMALIZE REAL FORMAT
.XADD,RP,105213          * EXTENTED REAL ADDITION
.XSUB,RP,105214          * EXTENDED REAL SUBTRACTION
.XMPY,RP,105203          * EXTENDED REAL MULTIPLY
.XDIV,RP,105204          * EXTENDED REAL DIVIDE
*
****************************************************************
*
*     XADD, XSUB, XMPY AND XDIV ARE USED FOR FTN INTERFACES
*
XADD ,RP,105207          * EXTENDED REAL ADDITION
XSUB ,RP,105210          * EXTENDED REAL SUBTRACTION
XMPY ,RP,105211          * EXTENDED REAL MULTIPLCATION
XDIV ,RP,105212          * EXTENDED REAL DIVISION
*
*
```

```
* ]RT6OS 92084-17271 REV.5010 <880617.1559>
*
* RTE-6/VM E/F-Series Operatng System Firmware Entry Points
*
*       *** USER CALLABLE OP SYS ENTRY POINTS
*           (F AND E SERIES IN RTE-6/VM ONLY)  *****
*
$LIBR,RP,105340          * EMULATE SYSTEM ENTRY $LIBR
$LIBX,RP,105341          * EMULATE SYSTEM ENTRY $LIBX
$SIP ,RP,0               * USE $SIP ONLY IF THE SYSTEM IS
*                        *   PRIVILEGED OR A MICROINSTRUCTION
*                        *   IS STORED IN A TRAP CELL
.FNW ,RP,105345          * FIND WORD WITH USER INCREMENT
.LLS ,RP,105347          * LINKED LIST SEARCH
.CPM ,RP,105352          * COMPARE WORDS IN MEMORY
.ENTN,RP,105354          * ENTRY POINT RESOLVER
.ENTC,RP,105356          * ENTRY POINT RESOLVER
.STIO,RP,105344          * Configure I/O instructions for drivers
.dspi,rp,105357          * show low 6 bits of A register in
*                          the display indicator register on front panel
*
```

Notes:

1) The command "$SIP,RP,0" should be left in the command file only if a trap cell contains a midrocode instruction. This is true if you are using DS/1000-IV driver DVA65, or any other driver or routine that places a microcode instruction in the trap cells.

   If this is not your situation, you can comment out the line which RP's $SIP.

2) The OS microcode is located at address 10000B on the F-Series and 26000B on the E-Series computer. It must be on the FEM board on the F-Series computer. FEM switch settings:

   1001101000 F-Series
   1001110110 E-Series

```
* ]E^FFP 92084-17274 REV.2440 <841005.1605>
*
* RTE-6/VM E-Series Fast Fortran Processor Firmware Entry Points
*
*
*     ***** FFP ENTRY POINTS *****
*
DBLE ,RP,105201              * CONVERT REAL TO EXTENDED REAL
SNGL ,RP,105202              * CONVERT EXTENDED REAL TO REAL
.DFER,RP,105205              * 3 WORD MOVE (EXTENDED REAL TRANSFER)
.XPAK,RP,105206              * NORMALIZE, ROUND AND PACK WITH EXPONENT
*                            *   AN EXTENDED REAL MANTISSA
.XCOM,RP,105215              * COMPLEMENT AN EXTENDED REAL UNPACKED
*                            *   MANTISSA IN PLACE
..DCM,RP,105216              * COMPLEMENT AN EXTENDED REAL
DDINT,RP,105217              * TRUNCATE AN EXTENDED REAL
.XFER,RP,105220              * 3 WORD MOVE (EXTENDED REAL TRANSFER)
.GOTO,RP,105221              * TRANSFER CONTROL TO LOCATION
..MAP,RP,105222              * CAL THE ADR OF A 2 OR 3D ARRAY ELEMENT
.ENTR,RP,105223              * TRANSFER THE TRUE ADDRESS OF PARAMETERS
*                            *   USED IN A SUBROUTINE CALL
.ENTP,RP,105224              * SAME AS .ENTR, EXCEPT MUST BE THIRD
*                            *   INSTRUCTION AFTER THE ENTRY POINT
.PWR2,RP,105225              * CALCULATE REAL X AND INTEGER N, Y=X*2**N
.FLUN,RP,105226              * UNPACK REAL (EXPONENT IN A, LOWER PART OF
*                            *   MANTISSA IN B)
*
$SETP,RP,105227              * SET UP A LIST OF POINTERS
*                            * NOTE: $SETP REPLACES .SETP AS OF 1913
*
.PACK,RP,105230              * CONVERT SIGNED MANTISSA OF REAL INTO
*                            *   NORMALIZE REAL FORMAT
.CFER,RP,105231              * MOVE 4 WORDS (COMPLEX TRANSFER)
*                            *   (Note: Not in E-series before 1978)
.XADD,RP,105213              * EXTENDED REAL ADDITION
.XSUB,RP,105214              * EXTENDED REAL SUBTRACTION
.XMPY,RP,105203              * EXTENDED REAL MULTIPLY
.XDIV,RP,105204              * EXTENDED REAL DIVIDE
*
****************************************************************
*
*     XADD, XSUB, XMPY AND XDIV ARE USED FOR FTN INTERFACES
*
XADD ,RP,105207               * EXTENDED REAL ADDITION
XSUB ,RP,105210               * EXTENDED REAL SUBTRACTION
XMPY ,RP,105211               * EXTENDED REAL MULTIPLCATION
XDIV ,RP,105212               * EXTENDED REAL DIVISION
*
*
```

```
* ]RT6VM 92084-17272 REV.2440 <840925.1058>
*
* RTE-6/VM E/F-Series Extended/Virtual Memory Area Firmware Entry Points
*
*
*   ***  EMA/VMA ENTRY POINTS (E AND F SERIES IN RTE-6/VM ONLY)  ****
*
*
.PMAP,RP,105240          * MAP EMA/VMA PAGE IN MAP REGISTER
$LOC ,RP,105241          * MEMORY RESIDENT NODES LOAD ON CALL
.IMAP,RP,105250          * SINGLE INT FTN4X ARRAY CALC. + MAP
.IMAR,RP,105251          * SINGLE INT SUBSCRIPT ARRAY CALC.
.JMAP,RP,105252          * DOUBLE INT FTN4X ARRAY CALC. + MAP
.JMAR,RP,105253          * DOUBLE INT SUBSCRIPT ARRAY CALC.
.LPXR,RP,105254          * TWO DEF POINTER ADD & MAP
.LPX ,RP,105255          * A&BREG POINTER + DEF OFFSET & MAP
.LBPR,RP,105256          * ONE DEF POINTER & MAP
.LBP ,RP,105257          * MAP POINTER IN A&BREG
*
*
```

```
* ]F^FFP 92084-17275 REV.5000 <870205.0913>
*
* RTE-6/VM F-Series Fast Fortran Processor Firmware Entry Points
*
*
*     ***** FFP ENTRY POINTS *****
*
DBLE ,RP,105201          * CONVERT REAL TO EXTENDED REAL
SNGL ,RP,105202          * CONVERT EXTENDED REAL TO REAL
.DFER,RP,105205          * 3 WORD MOVE (EXTENDED REAL TRANSFER)
.XPAK,RP,105206          * NORMALIZE, ROUND AND PACK WITH EXPONENT
*                        *   AN EXTENDED REAL MANTISSA
.XCOM,RP,105215          * COMPLEMENT AN EXTENDED REAL UNPACKED
*                        *   MANTISSA IN PLACE
..DCM,RP,105216          * COMPLEMENT AN EXTENDED REAL
DDINT,RP,105217          * TRUNCATE AN EXTENDED REAL
.XFER,RP,105220          * 3 WORD MOVE (EXTENDED REAL TRANSFER)
.GOTO,RP,105221          * TRANSFER CONTROL TO LOCATION
..MAP,RP,105222          * CAL THE ADR OF A 2 OR 3D ARRAY ELEMENT
.ENTR,RP,105223          * TRANSFER THE TRUE ADDRESS OF PARAMETERS
*                        *   USED IN A SUBROUTINE CALL
.ENTP,RP,105224          * SAME AS .ENTR, EXCEPT MUST BE THIRD
*                        *   INSTRUCTION AFTER THE ENTRY POINT
.PWR2,RP,105225          * CALCULATE REAL X AND INTEGER N, Y=X*2**N
.FLUN,RP,105226          * UNPACK REAL (EXPONENT IN A, LOWER PART OF
*                        *   MANTISSA IN B)
*
$SETP,RP,105227          * SET UP A LIST OF POINTERS
*                        * NOTE: $SETP REPLACES .SETP AS OF 1913
*
.PACK,RP,105230          * CONVERT SIGNED MANTISSA OF REAL INTO
*                        *   NORMALIZE REAL FORMAT
.CFER,RP,105231          * MOVE 4 WORDS (COMPLEX TRANSFER)
*
*
..FCM,RP,105232*         * COMPLEMENT A REAL
..TCM,RP,105233*         * NEGATE A DOUBLE REAL
.BLE ,RP,105207*         * CONVERT REAL TO DOUBLE REAL
.NGL ,RP,105214*         * CONVERT DOUBLE REAL TO REAL
*
* (*) Available in F-series date code 1920 or greater (rev of firmware >=4)
```

```
* ]F^SIS 92084-17277 REV.5000 <870205.1120>
*
* RTE-6/VM F-Series Scientific Instruction Set Firmware Entry Points
*
*
*     ***** SIS ENTRY POINTS (IN F SERIES ONLY) *****
*
TAN  ,RP,105320          * TANGENT
SQRT ,RP,105321          * SQUARE ROOT
ALOG ,RP,105322          * NATURAL LOGARITHM LN(X)
ATAN ,RP,105323          * ARCTANGENT
COS  ,RP,105324          * COSINE
SIN  ,RP,105325          * SINE
EXP  ,RP,105326          * EXPONENTIAL E**X
ALOGT,RP,105327          * LOGARITHM LOG10(X)
TANH ,RP,105330          * HYPERBOLIC TANGENT
*
TRNL ,RP,105331          * EVALUATE THE QUOTIENT OF 2 POLYNOMIALS IN
*                        * DOUBLE PRECISION
DPOLY,RP,105331*         * EVALUATE THE QUOTIENT OF 2 POLYNOMIALS IN
*                        * DOUBLE PRECISION
*                        * NOTE: DPOLY REPLACES TRNL AS OF 1926 (SAME
*                        *       ROUTINE DPOLY IS USED IN OTHER SUB-
*                        *       ROUTINES SUCH AS DCOS AND DSIN)
*
/CMRT,RP,105332*         * RANGE REDUCTION FUNCTION
/ATLG,RP,105333*         * COMPUTE (1-X)/(1+X) IN DOUBLE PRECISION
.FPWR,RP,105334*         * COMPUTE X**I FOR REAL X AND UNSIGNED INTEGER I
.TPWR,RP,105335*         * COMPUTE X**I FOR DOUBLE REAL X AND UNSIGNED
*                        *   INTEGER I
*
* (*) Available in F-series date code 1920 or greater (rev of firmware >=3)
```

```
* ]F^VIS 92084-17278 REV.2440 <840925.1058>
*
* RTE-6/VM F-Series Vector Instruction Set Firmware Entry Points
*
*
*   ***** VIS ENTRY POINTS (F SERIES IN RTE-6/VM ONLY) *****
*
.VECT,RP,101460          * FIRST OF TWO WORDS (USED BY SOFTWARE IN %VLIB)
*                        *   TO GET TO TWO WORD OPCODES
VPIV ,RP,101461          * PIVOT ROUTINE
VABS ,RP,101462          * ABSOLUTE VALUE ROUTINE
VSUM ,RP,101463          * SUM THE ARRAY ELEMENTS
VNRM ,RP,101464          * SUM THE ABSOLUTE VALUE OF THE ELEMENTS
VDOT ,RP,101465          * DOT PRODUCT ROUTINE
VMAX ,RP,101466          * FIND THE LARGEST  ARRAY ELEMENT
VMAB ,RP,101467          * FIND THE LARGEST  ARRAY ELEMENT(ABSOLUTE VALUE)
VMIN ,RP,101470          * FIND THE SMALLEST ARRAY ELEMENT
VMIB ,RP,101471          * FIND THE SMALLEST ARRAY ELEMENT(ABSOLUTE VALUE)
VMOV ,RP,101472          * COPY AN ARRAY INTO AN OTHER ARRAY
VSWP ,RP,101473          * EXCHANGE ELEMENTS OF TWO ARRAYS
.DVCT,RP,105460          * FIRST OF TWO WORDS (USED BY SOFTWARE IN %VLIB
*                        *   TO GET TO TWO WORD OPCODES)
DVPIV,RP,105461          * PIVOT ROUTINE FOR DOUBLE REAL ARRAYS
DVABS,RP,105462          * ABSOLUTE VALUE ROUTINE FOR DOUBLE REAL ARRAYS
DVSUM,RP,105463          * SUM THE ARRAY ELEMENTS FOR DOUBLE REAL ARRAYS
DVNRM,RP,105464          * SUM THE ABSOLUTE VALUE OF THE ELEMENTS IN A
*                        *   DOUBLE REAL ARRAY
DVDOT,RP,105465          * DOT PRODUCT ROUTINE FOR DOUBLE REAL ARRAYS
DVMAX,RP,105466          * FIND THE LARGEST  ARRAY ELEMENT IN A DOUBLE
*                        *   REAL ARRAY
DVMAB,RP,105467          * FIND THE LARGEST  ARRAY ELEMENT IN A DOUBLE
*                        *   REAL ARRAY (ABSOLUTE VALUE)
DVMIN,RP,105470          * FIND THE SMALLEST ARRAY ELEMENT IN A DOUBLE
*                        *   REAL ARRAY
DVMIB,RP,105471          * FIND THE SMALLEST ARRAY ELEMENT IN A DOUBLE
*                        *   REAL ARRAY (ABSOLUTE VALUE)
DVMOV,RP,105472          * COPY A DOUBLE REAL ARRAY INTO ANOTHER DOUBLE
*                        *   REAL ARRAY
DVSWP,RP,105473          * EXCHANGE ELEMENTS OF TWO DOUBLE REAL ARRAYS
*
*
```

```
* ]F^FPB 92084-17276 REV.5000 <870205.0918>
*
* RTE-6/VM F-Series Floating Point Box Firmware Entry Points
*
*
*     ***** 3-WORD ENTRY POINTS (IN F SERIES ONLY) *****
*
.XADD,RP,105001          * EXTENDED REAL ADDITION
.XSUB,RP,105021          * EXTENDED REAL SUBTRACTION
.XMPY,RP,105041          * EXTENDED REAL MULTIPLCATION
.XDIV,RP,105061          * EXTENDED REAL DIVISION
.XFXS,RP,105101          * EXTENDED REAL TO INTEGER FIX
.DINT,RP,105101          * EXTENDED REAL TO INTEGER FIX (NOTE .DINT FOR
*                        *   FTN INTERFACE, SAME ENTRY POINT AS .XFXS
.XFXD,RP,105105          * EXTENDED REAL TO DOUBLE INTEGER FIX
.XFTS,RP,105121          * INTEGER TO EXTENDED REAL FLOAT
.IDBL,RP,105121          * INTEGER TO EXTENDED REAL FLOAT  (NOTE: FTN
*                        *   INTERFACE SAME ENTRY POINT AS .XFTS)
.XFTD,RP,105125          * DOUBLE INTEGER TO EXTENDED REAL FLOAT
*
*     ***** 4-WORD ENTRY POINTS (IN F SERIES ONLY) *****
*
.TADD,RP,105002          * DOUBLE REAL ADDITION
.TSUB,RP,105022          * DOUBLE REAL SUBTRACTION
.TMPY,RP,105042          * DOUBLE REAL MULTIPLY
.TDIV,RP,105062          * DOUBLE REAL DIVIDE
.TFXS,RP,105102          * DOUBLE REAL TO INTEGER FIX
.TINT,RP,105102          * DOUBLE REAL TO INTEGER FIX  (NOTE: FTN
*                        *   INTERFACE SAME ENTRY POINT AS .TFXS)
.TFXD,RP,105106          * DOUBLE REAL TO DOUBLE INTEGER FIX
.TFTS,RP,105122          * INTEGER TO DOUBLE REAL FLOAT
.ITBL,RP,105122          * INTEGER TO DOUBLE REAL FLOAT (NOTE: FTN
*                        *   INTERFACE SAME ENTRY POINT AS .TFTS)
.TFTD,RP,105126          * DOUBLE INTEGER TO DOUBLE REAL FLOAT
*
* **** DOUBLE INTEGER ENTRY POINTS (FFP) (IN F SERIES ONLY) *****
*
.DAD ,RP,105014*         * DOUBLE INTEGER ADDITION
.DSB ,RP,105034*         * DOUBLE INTEGER SUBTRACTION
.DMP ,RP,105054*         * DOUBLE INTEGER MULTIPLICATION
.DDI ,RP,105074*         * DOUBLE INTEGER DIVISION
.DSBR,RP,105114*         * DOUBLE INTEGER SUBTRACTION (REVERSED)
.DDIR,RP,105134*         * DOUBLE INTEGER DIVISION (REVERSED)
.DNG ,RP,105203*         * DOUBLE INTEGER NEGATE
.DIN ,RP,105210*         * DOUBLE INTEGER INCREMENT
.DDE ,RP,105211*         * DOUBLE INTEGER DECREMENT
.DIS ,RP,105212*         * DOUBLE INTEGER INCREMENT AND SKIP IF 0
.DDS ,RP,105213*         * DOUBLE INTEGER DECREMENT AND SKIP IF 0
.DCO ,RP,105204*         * DOUBLE INTEGER COMPARE
*
.FIXD,RP,105104          * REAL TO DOUBLE INTEGER FIX
.FLTD,RP,105124          * REAL TO DOUBLE INTEGER FLOAT
*
* (*) Available in F-series date code 1920 or greater (rev of firmware >=2)
```

Operating System

The RTE-6/VM Operating System is contained in the following modules:

```
16d
    *
    *   RTE OPERATING SYSTEM
    *
    REL,%CR6S1              *Operating System Modules Part 1
    REL,%CR6S2              *Operating System Modules Part 2
    REL,%CR6S3              *Operating System Modules Part 3
    REL,%$CNFG              *Configurator Extension
```

Powerfail/Auto-Restart

If you desire powerfail/auto-restart capability for your system, enter the following inputs:

```
16d
    REL,%6DP43              *Powerfail Driver
    REL,%4AUTR              *Restart Utility

20
    EQUIPMENT TABLE ENTRY

    EQT nn?

    4,DVP43,M               *Powerfail EQT Entry

21
    DEVICE REFERENCE TABLE

    LU = EQT # ?

    nn                      *Powerfail LU

22
    INTERRUPT TABLE

    4,ENT,$POWR             *Powerfail
```

where:

LU	is the assigned logical unit.
nn	is the assigned EQT entry. The powerfail EQT should be the last EQT entry assigned.

The powerfail restart utility, AUTOR, is responsible for re-enabling terminals and outputting messages indicating the time of the failure. Additional user specified functions can be performed after restart by modifying the AUTOR source file: %4AUTR and reloading the utility.

File Management System

The File Management system is contained in the following relocatable modules:

```
16d
    *
    *   FILE MANAGEMENT SYSTEM PROGRAMS
    *
    REL,%BMPG1
    REL,%BMPG2
    REL,%BMPG3
    REL,%FMP6,NOLIB
    NOLIB
    REL,%CI                 *CI Utility
    REL,%CISU6              *CI RTE-6 Library
    REL,%CI000              *CI Messages
    REL,%CRLIB              *CI Library
    REL,%CR000              *CI Library Messages
    REL,%CIX                *CIX Utility
    REL,%CX000              *CIX Messages
    LIB
```

The File Management system consists of the following components:

FMGR. FMGR provides the interactive interface between the user and the file system (%BMPG1). In multi terminal environments a copy of FMGR will normally be provided for each user. The System Manager should allocate a minimum of two long blank ID segments for every terminal on the system. In addition a minimum of one resource number should be allocated to FMGR and each copy to permit LU locks. It is recommended that sufficient resource numbers be allocated (32 is an average figure for resource numbers).

```
28
   # OF RESOURCE NUMBERS?
    (+nn+1)
   --------
30
   # OF BLANK ID SEGMENTS?
    (+nn+1)                    ("long" ID segments)
   --------
```

where:

nn is the number of Session or MTM terminals configured into the system.

CI. The CI Utility (and its companion program CIX) provide the interactive interface to the hierarchical file system. CI may be used as an alternative to FMGR as the program scheduled at log-on time for users. The System Manager should allocate a minimum of three blank ID segments for every terminal on the system that will be running CI at log-on time.

```
36
   MODIFY PROGRAM PAGE REQUIREMENTS

   CI,32

   CIX,32
```

FMP LIBRARY. The FMP libraries (%BMPG3 and $FMP6) consist of a set of subroutines that are appended to user programs which access File Management System disk files. These subroutines are optionally stored (in relocatable form) on disk in the system library area. The library %BMPG3 contains subroutines which access the FMGR file system, and $FMP6 contains subroutines which access the CI file system.

D.RTR. D.RTR is the system file directory manager. D.RTR is called upon by all copies of File Manager, CI, and programs accessing the file system. If is responsible for mounting cartridges, manipulating file directories, and allocating additional file space when files overflow their extents. D.RTR uses free memory to maintain the list of global directories and open flags for CI files. For this reason, D.RTR must be sized up in the generation. 28 pages are recommended. D.RTR also uses EMA in 8-page chunks for caching I/O to disk. D.RTR requires, and defaults to, one 8-page chunk, but more EMA can be assigned in 8-page increments, up to 128 pages. Each 8-page chunk is used as a separate caching buffer. Depending on the application, more buffers will improve performance by reducing the number of disk accesses for D.RTR. Because it is used by so many modules, D.RTR should be assigned a high priority relative to other programs in the system. D.RTR is defaulted as a disk-resident program with a priority of 1. In systems with a great deal of file activity, you may want to ensure that D.RTR is always memory-resident. This will eliminate any potential time required to swap D.RTR into main memory from disk. To ensure that D.RTR is memory-resident, assign it to a partition and reserve that partition for D.RTR's exclusive use:

```
35
DEFINE PARTITIONS:
PART nn?
37,BG,R                          *D.RTR's 37-page Partition

SUBPARTITIONS?                   *No subpartition on D.RTR partition
NO
36
MODIFY PROGRAM PAGE REQUIREMENTS?
D.RTR,28,24
  .
  .
  .

39
ASSIGN PROGRAM PARTITIONS?
D.RTR,nn
```

where:

nn is the 37-page partition assigned to D.RTR.

The System Manager may optionally protect the peripheral disk subchannels from alteration by user EXEC calls. Use of this feature forces usage of the file management system when modifying peripheral disks. It also prevents use of the online COPY and RSTOR utilities on peripheral cartridges. Due to this constraint, this feature is NOT recommended for most systems:

```
18
CHANGE ENTS?
$PDSK,AB,1                       *Protect Peripheral Disks
```

Spooling System

The spooling system operates in conjunction with the File Management system to automatically provide spool capability within batch jobs or sessions. In addition, the spooling system allows programmatic control of spooling operations via SMP calls.

It is recommended that spooling be included in your system if:

1. Users have access to common system peripherals like line printers. The spool system synchronizes access to selected peripherals when accessed from different sessions and batch jobs.
2. Peripheral device EXEC I/O calls (to selected LUs) must be diverted to or from disk. The spool system will divert output (input) operations destined for peripheral devices to FMP disk files.
3. Users tie up peripherals for long periods of time, though with minimal use.
4. Allow tasks (that is, compiling, listing, etc.) to complete rapidly and return to user rather than waiting for I/O completion.
5. Perform tasks first and then have output occur at low-activity period (lunch, evenings, etc.).
6. Allow batch jobs to be processed.
7. Restart and perhaps re-direct output of a task if output was lost or partly destroyed due to device failure without restarting the test.
8. Heavy program development use. Note that utility COMPL is used to provide automatic outspooling and other features for all supported languages. Refer to the *RTE-6/VM Utility Programs Reference Manual*, part number 92084-90007, for a description of COMPL.

If you decide to include spooling in your system, the following modules (in addition to the File Management System modules described in the previous section) must be included in your generation.

```
16d
    *
    *   SPOOLING SYSTEM
    *
    REL,%SPOL1              *Spooling Modules Part #1
    REL,%SPOL2              *Spooling Modules Part #2
```

A brief description of the major components in the spooling system is given below. For a detailed description of the operation of the spooling system and these modules refer to the *RTE-6/VM Batch and Spooling Reference Manual*, part number 92084-90006.

JOB — Spooling of Batch Jobs is initiated by running program JOB. This program controls the phase known as inspooling.

SMP — SMP monitors the spooling process including maintaining the spool directory, assigning outspool files, and monitoring the output spooling program, SPOUT.

SPOUT SPOUT takes the output from the outspool files and directs the output to the actual devices.

GASP GASP is an interactive utility which is used to initialize the spool system and control the inspool and outspool processes with operator commands.

DVS43 DVS43 is a system driver which reroutes standard EXEC I/O calls into spool files.

SP.CL SP.CL is a spool communication area which resides in Table Area II.

The spool monitor programs have the default priority and program types shown in Table 4-3.

Table 4-3. Spool Monitor Programs

Program	Priority	Size (pages)	Program Type
JOB	30	6	130 (Real-time Disk-resident)
GASP	30	11	3 (Background Disk-resident)
SMP	30	6	130 (Real-time Disk-resident)
EXTND	10	2	129 (Memory-resident)
SPOUT	11	2	1 (Memory-resident)
DVS43	– –	–	0 (System Module)
SP.CL	– –	–	13 (Table Area II Module)

–

* Size includes base page.

Generally, optimal performance is provided by using these default values. For some programs, the program type code may be changed during generation if the rules stated below are observed.

JOB may be any disk-resident type as long as it does not compete for the same partition as FMGR. There should be enough partitions to avoid competition.

For best performance, SMP should be left real-time disk-resident. SMP must not be made memory-resident.

SPOUT is normally memory-resident. If SPOUT cannot be memory-resident (due to space limitations), you can ensure that SPOUT will reside in memory at all times by assigning it to a partition and reserving that partition exclusively for SPOUT:

```
35
   DEFINE PARTITIONS:
   PART nn?
   2,BG,R                    *Spout's Partition

37
   ASSIGN PROGRAM PARTITIONS?

   SPOUT,nn
```

where:

nn		is the partition to be reserved.

Note If both SPOUT and D.RTR are real-time (or background) disk-resident programs, you should have at least 2 real-time (or background) partitions defined. If both programs must contend for the same partition, a deadlock situation may occur when SPOUT needs to create an extent for a spool file.

I/O CONFIGURATION. In RTE-6/VM you must have a spool pseudo-device generated into your system for every concurrent spool operation. Each spool operation requires one LU and one EQT. The number of spool LU numbers and EQT entries generated in your system should depend on the amount of anticipated spool system usage.

Estimate the number of spool LU numbers and EQT entries needed for your system with the following in mind:

- In the session environment each user may set up several spools. Each SL spool command uses ONE spool LU. If the NOW attribute is specified, TWO spool LU numbers are used.
- A Batch job normally uses TWO spool LU numbers plus that number used for any SL commands. One spool LU is used for LU 5 and one is used for LU 6. If NO is specified as an option on the job statement, two spool LUs are used for LU 6.

If your system does not have Session Monitor, the number of concurrent spool operations depends on the mix and type of batch jobs in the system and the programs using spooling through FMP calls. If is suggested that you configure the system to allow at least four concurrent spool operations. If you are using the Session Monitor, spooling operations can be initiated from all sessions able to execute the spool SL command, batch jobs initiated from these sessions, and batch jobs initiated outside the session environment. If possible, allow for at least one concurrent spool operation per terminal and several for batch jobs.

Spool LUs can be greater than or less than 64, but the following restrictions should be noted:

1. Spool LUs set up in a session environment will have their LUs redirected through the user's Session Block (SCB) and thus will appear to be less than 64. Standard EXEC and REIO calls can be used to access these spool LUs.
2. Spool LUs set up outside of session cannot be redirected through a user's SCB. Therefore, the user may see a spool LU that is greater than 63 and must use XLUEX and XREIO calls to access it.

SMP allocates spool LUs from the highest number down in a session environment and from the lowest numbers up in a non-session environment in an attempt to minimize this problem.

If spool LUs are to be accessed from outside of session it is recommended to allocate some less than 64.

Use the spooling portion of the I/O configuration worksheet to make your device LU and EQT entry assignments. The generator inputs described below give the spool DRT, EQT, and interrupt table specifications after the assignments have been made.

In general, if the system is being used for program development and batch is not used, eight spool LUs would allow up to eight concurrent compilations via the utility program COMPL (one per compilation). Thus eight spool LUs is a fairly common number for program development.

Note Each spool EQT will require an additional 33 words in System Table Area I. If you have many spool EQT entries in your system, Table Area I may overflow to an additional page, thereby reducing logical address space available for program use. (The logical address space of extended background programs will not be affected.)

The following table definitions must be made for each spool device to be configured in the system:

```
20
    EQUIPMENT TABLE ENTRY

    EQT nn?

    sc,DVS43,M                *Spool DEV EQT

21
    DEVICE REFERENCE TABLE

    LU = EQT # ?

    nn                        *Spool DEV # LU

22
    INTERRUPT TABLE

    sc,EQT,nn                 *Spool DEV #
```

where:

sc	is the select code of the spool device.
nn	is its assigned EQT entry.
lu	is its assigned logical unit.

RESOURCE REQUIREMENTS. Two class numbers should be allocated for the spool monitor; one for outspooling and one for SMP. Note that many other HP subsystems use class numbers. If DS/1000 is not included in your system, a minimum of 16 class numbers are recommended.

```
26
   # OF I/O CLASSES?
   (+16)
```

The Batch LU switch table should be configured as follows:

```
27
   # OF LU MAPPINGS [,# OF SCHEDULE PROGRAMS]?
   nn+2
```

where:

nn — is the maximum number of SL Commands expected by batch jobs initiated outside the Session environment. A total of 8 is recommended.

Four resource numbers should be allocated for the spooling system:

```
28
   # OF RESOURCE NUMBERS [,DEBUG TABLE SIZE]?
   (+16)
```

SAM REQUIREMENTS. The SPOUT program attempts to keep four requests in System Available Memory (SAM) for each device to which it is outspooling. For optimum performance, spool system SAM requirements should be estimated as follows:

(# Outspool devices) X (Max record size) X (Queue depth)

where:

# Outspool devices	is the number of peripheral devices in the system to which spooled output will be sent.
Max record size	is the largest expected outspool record in SAM for each device.
Queue depth	is the number of requests SPOUT attempts to keep in SAM for each outspool LU (default is 4). Follow the guidelines in the RTE-6/VM Batch and Spooling Reference Manual for setting queue depth.

For the outspool record, you may assume a maximum record size of 68 words plus a 10-word SAM header. For example, in a system with spooling to two line printers and a paper tape punch, the optimum amount of additional SAM for SPOUT would be 3 x 78 x 4 = 936 words.

If you cannot generate this much additional SAM you may experience a degradation in system performance. As a bare minimum, generate enough SAM for one outspool device. Thus, 78 words x 4 records = the minimum SAM required by SPOUT. As a general rule, the more SAM, the better.

Libraries

Several relocatable library files are provided with the RTE-6/VM operating system. Table 4-4 shows libraries used by system utilities. The table and the following guidelines will help determine which libraries should be included in the system generation.

There are two reasons for including a library file in the system generation:

1. To be used by the generator for loading programs.

2. To include the library in the system disk library.

Libraries that are used by the generator for loading programs during the generation are automatically included in the system disk library unless the NOLIB option is used when relocating the library.

The following guidelines will help in determining which library files should be included in the system disk library:

- Libraries in the system disk library are automatically search by LOADR, MLLDR, and LINK when loading programs online.

- Libraries not included in the system disk library have to be searched explicitly.

- Libraries to be automatically searched by LINK during online loading do not have to be in the system disk library. They may be put into the snap file for LINK when the system is indexed by LINDX.

- Library entry points having names that are 6 or more characters in length cannot be included in the system disk library. Libraries that have some entry points with 6 or more characters would be relocated with the NOLIB command to avoid having only a subset of the library file included in the system disk library. These libraries should only be included in the generation if they are needed by programs being loaded by the generator.

- Reducing the number of library files in the system disk library will reduce the amount of space taken on the system disk.

The following libraries contain routines used by operating system utilities, subsystems, compilers, and user programs. They must be relocated at generation time for a minimum system.

A. %BMPG1, %BMPG2, and %BMPG3
File Management Package Libraries
Required by the File Management Package and programs using the FMGR file system.

B. $SYLB6 System Subroutine Library
Required by most system utilities, compilers, and user programs needing specific operating system services.

C. $MATH, $FLIB, and $FOLDF
Math/Formatter Libraries
Used by most system utilities, compilers, and user programs needing the mathematical routines and user-callable routines in these libraries. Use $FOLDF for programs using FORTRAN I/O statements accessing only FMGR files.

D. $LDRLN   Loader Library
Used by LOADR, MLLDR.

```
16d
    *
    *
    *
    REL,%BMPG1              *File Management Library

    REL,%BMPG2              *File Management Library

    REL,%BMPG3              *File Management Library

    REL,$SYLB6              *System Library

    REL,$MATH               *Math/Formatter Library

    REL,$FLIB               *Math/Formatter Library

    REL,$FOLDF              *File I/O Library

    REL,$LDRLN              *Loader Library
```

At generation time, the following libraries can be relocated for the loading of programs that need them. The NOLIB option, however, should be used to prevent having portions of these libraries in the system library.

A. PASCAL.LIB and PASCAL_FMGR.LIB (access to FMGR file system only)
Pascal Libraries
Used to load Pascal compiler and programs with access to the CI file system or the FMGR file system.

B. SHSLB.LIB   Pascal Short Heap/Stack Library
Used to load Pascal programs, but is shorter and has less functionality than the heap/stack management routines of PASCAL.LIB (refer to Pascal/1000 Reference Manual). SHSLB.LIB can be used instead of PASCAL.LIB to reduce program size.

C. $ONLIN   Online Physical Backup Library
Used to load physical backup utilities.

D. $BEGGT and $BCKUP
Online Physical Backup Library
Used to load physical backup utilities.

E. $FMP6   File System Library (CI file system).

F. $FCLBA   FORTRAN Compiler Library (CI file system).

G. $FNEWF   FORTRAN File I/O Library (CI file system).
$FNEWF supercedes $FOLDF. Use $FNEWF for programs using FORTRAN I/O statements accessing CI and FMGR files.

The system manager decides whether or not to generate the remaining libraries. Doing so will cost system size, but will be convenient for the user loading programs that use these libraries.

The following libraries contain user-callable routines. If heavily used, they should be generated into the system.

A. %DECAR — Decimal String Library
Used by HP subsystems and user programs using decimal string arithmetic.

B. %DBUGR — Debugger Library
Used by programs loaded with the LOADR command OP,DB.

C. $IB6A — HP-IB Library
Used by user programs calling HP-IB interface routines.

D. $VLB6A or $VLB6B
VIS Libraries
Used by programs using VIS routines.

E. $MLSLB — MLS Program Library
Required to load multilevel segmented programs with MLLDR.

Although the following libraries can be generated into the system, it is probably not worthwhile, as they are used only to load specific programs or system utilities:

A. $DSCLB — SWTCH and FORMT Library
Used by system utilities interfacing with MAC and ICD disks.

B. $DTCLB — CS/80 Disk Utility Library
Used by system utilities interfacing with CS/80 disks.

C. $FDSLB — FORTRAN File I/O Library for use with DS.
Required for programs that use the FORTRAN I/O OPEN statement with the "NODE =" file control specifier. These programs must use the $FILES compiler directive with the DS option in the main. $FDSLB includes calls to the DS/1000 Remote File Access (RFA) routines to access remote FMGR files. Programs that use DS File Transparency to access remote CI or FMGR files do not need $FDSLB. HP-supplied RTE-6 and DS/1000-IV programs do not need $FDSLB.

D. $RBLIB — MLS Support Library
Required to load MLLDR, SGMTR, and SXREF.

E. $RSLIB — READR/SAVER Library
Required to load the READR and SAVER utilities.

F. $UTLIB — Utility Library
Used to load system utilities.

G. $VCLIB — VMA Firmware Diagnostic Library
Required to load VMACK.

H. $6FCLB — FORTRAN Compiler Library
Required to load the FTN4X compiler.

I. PASCAL_FMGR_ALT.LIB — Altered version of PASCAL_FMGR.LIB
Required only by altered Pascal programs. Refer to Pascal/1000 Configuration Guide for further information.

J. PASCAL_ERR_ALT.REL Altered version of PASCAL_ERR.REL (shorter Pascal Run-Time Error Reporter provided with Pascal/1000)
Required only by altered Pascal programs when program sizes need to be reduced. PASCAL_FMGR_ALT.LIB must be generated first if at all. Run-time error message number replace the messages reported by PASCAL_FMGR_ALT.LIB, reducing program size.

K. SHSLB_ALT.LIB Altered version of SHSLB.LIB
Required only if altered Pascal programs are generated and program sizes need to be reduced. PASCAL_FMGR_ALT.LIB must be generated first if at all. The shorter Pascal heap/stack management routines in SHSLB_ALT.LIB replace those in PASCAL_FMGR_ALT.LIB, reducing program size.

L. $ACCLB ACCTS library

M. $EMCLB RTE-IVB EMA Compatibility Library
Used to make programs that used EMA references for RTE-IVB systems compatible with RTE-6/VM firmware instructions. For example, converts .EMAP calls to .LBP. Not used by any RTE-6/VM software.

$FNEWF versus $FOLDF

If FORTRAN file I/O will be performed by a user program, the library $FNEWF *or* $FOLDF must be searched. Use the following guide to determine which library to use:

Use $FOLDF if: your program uses FORTRAN I/O calls such as OPEN, CLOSE, READ, WRITE, INQUIRE, etc. that access FMGR files *only*.

Use $FNEWF if: your program uses FORTRAN I/O calls such as OPEN, CLOSE, READ, WRITE, INQUIRE, etc. that access CI and/or FMGR files.

If $FOLDF is used, relocate it at generation time.

$FNEWF, however, can only be relocated at generation time if the NOLIB option is used. The NOLIB option is required so that $FNEWF is searched for programs being loaded during the generation, but it will not be placed into the system library. Note that if $FNEWF is used, the CI file system library $FMP6 must also be searched to supply the general file access routines for the system.

For loading programs online that require $FNEWF, LINK can be made to search $FNEWF automatically if it is placed into the system snap file (along with $FMP6). For example, include $FNEWF in the LINDX command as follows when building your system snap file:

```
LINDX,SNAP.6::2,$FNEWF::LIBRARIES,PASCAL.LIB::LIBRARIES,$FMP6.LIB::LIBRARIES,+NL
```

Refer to the *RTE-6/VM Online Generator Manual*, part number 92084-90010, for information on generating libraries with the NOLIB command and the *RTE-6/VM Link User's Manual*, part number 92084-90038, for information on including libraries in the system snap file.

Table 4-4. RTE-6/VM Libraries and Utilities

UTILITY	$FMP6	$LDRLN	%DECAR	$DSCLB	$DTCLB	$UTLIB	$PLIB	$SHSLB	$RBLIB	$RSLIB	$ED1K6	$FCLIB	$BCKUP	$BEGGT	$ONLIN	$VCLIB	$MLSLB	$6FCLB	$FNDLB	$FDSLB	$IB6A	$VLB6A	$VLB6B	$EMCLB	$ACCLB
ACCTS																									X
CLOAD	X					X																			
CLOSE	X																								
CMD	X						X																		
COMPL	X					X																			
DL	X																								
DRRPL		X																							
DRREL																									
DSRTR	X																								
EDIT	X										X														
EDITR																									
FC			X		X							X													
FOWN	X																								
FORMC					X																				
FORMT				X																					
FPACK	X																								
FREES	X																								
FSCON	X																								
FVERI	X																								
GENIX	X						X																		
HELP	X																								
INDXR	X																								
IS	X																								
KEYS	X																								
KYDMP	X																								
LGTAT																									
LI	X																								
LIF	X																								
LINDX	X																								
LINK	X	X																							
LOADR		X																							
MACRO	X																								
MERGE	X																								
MLLDR	X	X							X																
MSAFD																									
OLDRE	X																								
PATH	X																								
PCOPY			X		X		X						X	X	X										
PRIN0	X																								
PRINT	X																								

Table 4-4. RTE-6/VM Libraries and Utilities (Cont.)

UTILITY	LIBRARY: $FMP6	$LDRLN	%DECAR	$DSCLB	$DTCLB	$UTLIB	$PLIB	$SHSLB	$RBLIB	$RSLIB	$ED1K6	$FCLIB	$BCKUP	$BEGGT	$ONLIN	$VCLIB	$MLSLB	$6FCLB	$FNDLB	$FDSLB	$IB6A	$VLB6A	$VLB6B	$EMCLB	$ACCLB
PRSTR			X		X		X						X	X	X										
PSAVE			X		X		X						X	X	X										
PSPAR			X				X						X	X											
READR										X															
READT						X																			
RT6GN	X																								
SAFD																									
SAVER										X															
SCOM	X																								
SGMTR	X								X																
SWTCH	X			X	X	X																			
SXREF									X																
T5IDM																									
TRFAS	X																								
TVVER																									
VMACK																X									
WHOSD	X																								
WHZAT																									
WRITT						X																			

NOTES: 1. X indicates a library required to load the utility.
2. The required system libraries $6SYLB, %BMPG3, $MATH, $FLIB, and $FOLDF are not shown.
3. The short Pascal library $SHSLB can be used with $PLIB.

System Utilities

UTILITY LOADING CONSIDERATIONS. Utilities may be permanently included in the system using one of the following procedures:

1. Utilities can be generated into the system. When a program is generated into the system, the generator permanently allocates ID segments and disk storage for it. When the system is booted up, the utility is automatically defined to the operating system. No blank temporary ID segments are required to run the utility, except if a copy of the program is made.

2. Utilities can be added as Type 6 files. The utility disk image is stored in an FMP file on LU 2 or LU 3. When the utility is run (or RPd), a blank ID segment is allocated for it. Type 6 files are created by loading the utility with LINK or using the online LOADR, saving the utility (and any segments) via FMGR SP commands, and releasing the temporary ID segments of the utility and segments with the OF command. Type 6 files are system specific; that is, they are not generally transportable from one system to another. Type 6 programs cannot be run from breakmode unless they have been previously RPd.

3. Utilities can be loaded on line as permanent programs. When utilities are added to the system in this manner, the LOADR permanently allocates blank ID segments and disk tracks for them. The LOADR allocates disk space in track multiples for each program loaded on line. If less than a full track is required, the remaining space on the track is unavailable for other uses. Since this method of adding programs to the system uses disk space and ID segments least efficiently, the methods described above are preferable.

MULTI-TERMINAL USE. When run in the Session or MTM environment, programs permanently added to the system (with one of the above methods) are automatically copied for each user. This feature allows multiple copies of a utility to be active at one time. The first three characters of the program name are concatenated with the terminal LU or session number.

For example, assume a user on terminal LU 13 types:

```
:RU,EDIT
```

The system will create a copy of EDIT and actually run EDI13.

Certain utilities, for example, SWTCH, should not be automatically copied by the system. If they will be generated into the system, 128 should be added to their program type in the parameters phase of your generation. For example, to inhibit copying of SWTCH, (which is normally type 4) the program type should be set to 128 + 4 = 132:

```
17
  PARAMETERS

  SWTCH, 132
 -----,--------,----,----
```

UTILITY RELOCATABLES. Refer to Table 4-5 for the various relocatable file names for the utilities supplied in the standard 92084A disk. It is recommended that at least the following subset of these utilities be generated into the system:

```
16d *
    * *RTE UTILITIES
    *
     REL %LGTAT      *Track Assign. Table Status
     ---,------ ------ ------ ------
     --
     REL %$LDR       *LOADR
     ---,------ ------ ------ ------
     --
     REL %WHZAT      *WHZAT Utility
     ---,------ ------ ------ ------
     --
     REL %T5IDM      *Short ID Segment Manager
     ---,------,---,---,---
     --
```

It is recommended that the remaining utilities be loaded into the system by using the INCI.CMD and LOAD6.CMD files. These CI command files contain all necessary sizing commands and allow easy updating of the system.

Table 4-5. Utility Relocatable File Names

Program	Segments	Relocatable File	Description	Documentation
LOADR	LODR1 LODR2 LODR3 LODR4	%$LDR	Online Loader	A
MLLDR	MLLD1 MLLD2 MLLD3 MLLD4 MLLD5 MLLD6	%MLLDR %MLLDA %MLLDB	Online MLS Loader	A
SGMTR	SGMT1 SGMT2 SGMT3 SGMT4 SGMT5 SGMT6	%SGMTR	Segmenter Utility	A
SXREF	SXRE1 SXRE2 SXRE3 SXRE4 SXRE5 SXRE6	%SXREF	Cross Reference Utility	A
LGTAT		%LGTAT	Track Assignment Table Status Utility	B
WHZAT		%WHZAT	Program/Partition Status Utility	B
CMD		CMD.REL	Help Function Utility	B
HELP		%HELP	Help Utility	B
GENIX		GENIX.REL	Indexing Utility	B
SCOM		SCOM.REL	File Compare Utility	B
MERGE		MERGE.REL MERGE.R000	File Merge Utility	B
DRRPL		%DRRPL	Online Driver Relocation Utility	B
READT		%READT	File Cartridge Restore Utility	B

Table 4-5. Utility Relocatable File Names (Cont.)

Program	Segments	Relocatable File	Description	Documentation
WRITT		%WRITT	File Cartridge Save Utility	B
READR		%READR	File Restore Utility	G
SAVER		%SAVER	File Backup Utility	G
PSAVE		%PSAVE	Disk Save Utility	B
PRSTR		%PRSTR	Disk Restore Utility	B
PCOPY		%PCOPY	Disk Copy Utility	B
FORMT		%FORMT	ICD/MAC Disk Format Utillity	B
FORMC		%FORMC	CS/80 Disk Format Utility	B
COMPL		%COMPL	Program Compilation Utility	B
CLOAD		%CLOAD	Compile and Load Utility	B
KEYS		KEYS.REL	Soft Key Utility	B
KYDMP		KYDMP.REL	Soft Key Dump Utility	B
RT6GN	RT6G1 RT6G2 RT6G3 RT6G4 RT6G5 RT6G6 RT6G7 RT6G8 RT6G9	%RT6GN	Online Generator	C
SWTCH	SWSG1 SWSG2 SWSG3	%SSTCH	System Installation Utillity	D
T5IDM		%T5IDM	Short ID Segment Handler	D
VMACK		%VMACK	VMA Firmware Verifier	E
TVVER		%TVVER	91200 TV Interface Verifier	F

Documentation Key for Table 4-5.

A	92084-90008	RTE-6/VM Loader Reference Manual
B	92084-90007	RTE-6/VM Utility Programs Reference Manual
C	92084-90010	RTE-6/VM Online Generator Manual
D	92084-90009	RTE-6/VM System Managers Manual
E	91711-90006	HP 91711B Diagnostic Manual
F	91200-90006	HP 91200B TV Interface Kit, Programming and Operating Manual
G	92068-90016	READR/SAVER Utility Reference Manual

BUFFER SPACE CONSIDERATIONS. Some utilities require additional space to dynamically construct buffer space areas or symbol tables. Standard RTE utilities needing additional space are shown with their size requirements in Table 4-6. If any of these utilities is to be generated into your system, it is recommended that the minimum partition size be overridden.

For example, assuming LOADR is being generated into the system, the minimum page requirements might be overridden as follows:

```
36
    *
    * MODIFY PROGRAM PAGE REQUIREMENTS?
    * INCREASE UTILITY BUFFER AREAS

    LOADR,28
```

The partition sizes given in Table 4-6 should be used as guidelines only. If you are developing very large programs, the partition sizes may need to be increased. The appropriate Generator input values should be sized to the largest partition generated into your system (excluding EMA partitions). If possible, optimum performances can be obtained by making the partition the same size as the maximum large background partition allowed in your system. This number is given by the generator before partition definitions are made (it is usually 27-29 pages).

Note that the above input does NOT require that the FORTRAN Compiler be generated into your system.

Table 4-6. Programs Requiring Buffer Space in Partitions

Program Name	Minimum* Recommended Override (Pages)	Suggested Override (Pages)
LOADR	--	28
MLLDR	22	32 *
SXREF	--	32 *
GENIX	20	32 *
SCOM	--	32
DRREL	17	18
DRRPL	16	18
PSAVE	28	28
PRSTR	28	28
PCOPY	28	28
FORMT	18	18

* If these programs are loaded online as Extended Background, they can be sized to 31 or 32 pages. The remaining utilities can be loaded online after the system has been brought up.

Generation vs. Online Loading

Generating fewer programs into your system will increase the speed of your generation and possibly make better use of system resources.

In general, the following guidelines can be observed when deciding whether to generate a program into the system:

- Programs scheduled at system startup. Programs scheduled by the operating system at system startup should be generated into the system since they must be permanently allocated an ID segment and system disk tracks. (It is also possible to permanently add them online with LOADR, but disk space may not be used as efficiently because the disk storage is allocated in # of tracks.) Generated program disks storage is allocated in # blocks (128 words per block). System programs in this category are FMGR and the session ACCTS program.

  You can cause a user application program to be scheduled at system startup by adding 80 to its program type during the generator parameter definition phase (refer to the Online Generator Manual).

- Programs scheduled from break mode. ID Segments of programs run in break mode must have been previously defined to the system. This can be accomplished either by: generating the program in the system: issuing an RP command from file manager (assuming a type 6 file exists for the program); permanently adding the program to the system with LOADR; temporarily loading the program with LOADR (usually done with programs under development). If a program is to be run from break mode frequently, it is suggested that it be generated into the system or RPd into the system from the WELCOM File at system startup. Otherwise, users should define programs to the system only when actually needed. This should reduce unnecessary use of ID segments (a valuable system resource). Among the HP utilities that may be regularly scheduled from break mode are WHZAT, LGTAT, and HELP.

- Programs required during system installation. Certain utilities are used to facilitate installation. Obviously, the LOADR, or LINK and LINDX are required to add new programs to the system. The editor can be used to create and modify file manager command files, system message files, documentation files, and utility command files. File restore utilities may be used to retrieve HP and user relocatable binaries from tape. These programs can then be loaded on line and type 6 files created for them.

- Memory-resident requirements. If you want to eliminate disk swap time as a factor in program operation, you can follow one of three procedures (listed in the order of most efficient memory utilization):

  1. Generate the program into the system as a memory-resident program (type 1 module).

  2. Generate the program into the system as disk-resident, but assign it to a partition and reserve that partition.

  3. After generation, load the program on line via LOADR and assign it a partition; reserve that partition by reconfiguring memory.

If you decide to load certain programs online rather than during generation, their relocatable files will obviously have to be accessed after the new system has been brought up. Users may also require that their files be accessible on the new system. One way of making files immediately accessible to the new system is to have a common disk subchannel definition between the new and old systems. Files to be used on the new system can be stored on a cartridge identified by a common subchannel before system switchover. This cartridge can then be mounted on the new system.

Magnetic tape is another convenient medium for file transportation, if both new and old systems support have compatible magnetic tape units. Files can be grouped individually with file manager commands or entire cartridges can be copied via the READT/WRITT and FC utilities. Using the CTD tape cartridges and the FC utility is another method for file transportation.

When none of the above methods can be employed, use some other compatible media (for example, cartridge tape units, paper tape, DS links, etc.).

Transportable Type 6 Files

Certain type 6 files created on one RTE-6/VM Operating System can be copied to another RTE-6/VM Operating System and executed. These files are called transportable type 6 files. This is useful when switching to a new RTE-6/VM operating system in that not all programs must be reloaded.

Transportability of a type 6 file is determined by the File Manager when the type 6 file is RPd. The type 6 file contains a copy of the original ID segment and additional information about the system on which it was loaded.

Below are the requirements for type 6 file transportability and an explanation of how the conditions are checked.

a. The firmware replacements for the microcode that the program uses must be the same on both systems. There is no check for this, but the program will not execute properly if they do not match.

b. The initialization code word located in the system cartridge list is compared to the initialization code word stored in the type 6 file when created. The initialization code word is the sum of certain base page locations that define locations and sizes of system tables. If the initialization code word of the current system matches the initialization word stored in the type 6 file, then the systems are identical or similar enough for the type 6 file to be transportable.

   If not, the transportability bit and the load point are checked. If the transportability bit is set and the program's load point on the current system is the same as the load point when the program was originally loaded, then the type 6 file is also transportable.

   The transportability bit is set by LOADR, MLLDR, or LINK if the program does not access any entry points that can be at different locations. The LOADR or MLLDR LE option will report references to nontransportable entry points.

The guidelines for program transportability are given below:

a. Load the program as an Extended Background program. Extended Background programs have the same load point on all RTE-6/VM operating systems.

b. Do not directly access system entry points. System entry points placement is dependent on the generation and size of system tables.

c. Do not access SSGA entry points. Again, these entry points may vary from system to system.

Session Monitor

If you are not using the Session Monitor in your system, skip this section. If you are using the Session Monitor, include %SMON1 and %SMON2 during generation. The accounts program (%ACCTS) can be included in the generation or loaded online, as described below:

```
16d
    * Session Monitor

    REL,%SMON1            *Session Monitor Modules #1

    REL,%SMON2            *Session Monitor Modules #2

    REL,%ACCTS            *ACCTS Program

    REL,$ACCLB            *ACCTS Library
```

The Software Components contained in these relocatable modules are briefly described below.

PRMPT Session Break Mode Interrupt Processor. This program is responsible for issuing the break mode prompt and queuing command inputs to the command processor.

RPN Session Monitor Break Mode Command Processor. Handles all break mode commands queued onto it by PRMPT. RPN will either route commands to the operating system or process them itself (depending on the command).

LOGON Session Monitor Log On Processor. This processor is scheduled by PRMPT when there is no active session on the terminal. It accepts the user log-on ID, checks the ID against the account file, and sets up the session.

LGOFF Session Monitor Log Off Processor. This processor is scheduled by the session copy of FMGR at log off. LGOFF is responsible for updating the accounts file and releasing system resources allocated to the session.

ACCTS Session Accounts Management Program requires library $ACCLB. ACCTS is used by the system manager to initialize, maintain, and backup the account system. Normally it is run only by the system manager. Although it is not required to generate ACCTS into the system, it is recommended since it is scheduled during system startup. If you are loading ACCTS online, it is important to remember to load it as a large background program. It cannot be loaded extended background as it accesses the Subsystem Global Area (SSGA).

!BITM A table used by session modules to indicate whether terminals are enabled for break mode interrupts.

$YCOM Used by the system to activate LOGON and RPN to process break mode interrupts from the system console when enabled as a session terminal.

The default module respective priority levels, sizes, and program types are shown in Table 4-7.

Table 4-7. Session Monitor Programs

Program	Size (K)	Priority	Program Type
PRMPT	3	5	1 – (Memory-resident)
RPN	4	5	3 – (BG Disk-resident)
LOGON	11	50	3 – (BG Disk-resident)
LGOFF	9	90	3 – (BG Disk-resident)
ACCTS	17	90	20 – (large BG Disk-resident and access to SSGA)
!BITM	(7 words)	– –	15 – (Table Area I)
$YCOM	2	10	1 – (Memory-resident)

The Session Monitor software requires four class numbers:

```
26
   # OF I/O CLASSES?
  (+4)
```

Session Monitor requires SAM for storage of session control blocks and the spare cartridge pool. If the Session memory allocation algorithm is used during account system initialization (described in Chapter 6), SAM requirements will depend on the Session Limit:

Session Limit < 20: (70-Session Limit) * Session Limit

Session Limit > 20: Session Limit * 50

At the very minimum, you should allocate an additional 50 words of SAM per session terminal. This should be increased if the terminals have automatic output buffering enabled. Refer to the Online Generator Manual for a more detailed discussion of SAM requirements and usage.

Every Session terminal requires entries in the Device Reference Table, Equipment Table, and Interrupt Table. Refer to your I/O configuration worksheet for the LU, EQT, and select code assignments. Table definitions for session terminals will depend on the type of terminal and interface. For recommended generator inputs, refer to the Device Configuration section in this Chapter.

```
27
   # OF LU MAPPINGS [# OF SCHEDULED PROGRAMS]?
  (+0),(+1)
```

Scheduled programs are those programs that serial (terminal) drivers schedule on unexpected interrupts. The Session Monitor scheduled program is PRMPT. The Schedule Programs Table requires 5 bytes for each program. The Generator enters PRMPT as the first program at Generation time.

Multi-Terminal Monitor

NOTE: If you are not using the Multi-Terminal Monitor in your system, skip this section.

The Multi-Terminal Monitor is contained in the following relocatable modules:

16d
```
* Multi-Terminal Monitor

REL,%6MTM                *Multi-Terminal Monitor

REL,%NSESN               *Dummy Non-Session Module
```

These modules contain the following programs:

PRMPT MTM break mode interrupt processor. This program is responsible for issuing the break mode prompt and queuing inputs to the command processor.

RPN MTM break mode command processor. Handles all break mode commands queued onto it by PRMPT. RPN will either route commands to the operating system or process them itself (depending on the command).

The size, priority and program types for these programs is shown below:

Program	Size (K)	Priority	Program Type
PRMPT	3	10	1 (Memory-resident)
RPN	3	10	3 (BG Disk-resident)

MTM requires one class number for communication between PRMPT and RPN.

26
```
# OF I/O CLASSES?

(+1)
```

At the very minimum you should allocate an additional 50 words of SAM per MTM terminal. This should be increased if the terminals have automatic output buffering enabled. Refer to the Online Generator Manual for a more detailed discussion of SAM requirements and usage.

Every MTM terminal requires entries in the Device Reference Table, Equipment Table, and Interrupt Table. Refer to your I/O configuration worksheet for the LU, EQT, and select code assignments. Table definitions for MTM terminals depend on the type of terminal and interface. For recommended generator inputs, refer to the section titled DEVICE CONFIGURATION in this chapter.

Non-Session and Non-MTM Systems

If you will not be generating either Session Monitor or MTM in your system, relocate the following module:

16d

```
    REL,%NSESN                  *Dummy Non-Session Module
```

5

Transferring the New Operating System

SWTCH Program

After you have completed the online generation of your RTE-6/VM Operating System, the new system will reside on the disk in a Type 1 FMP file. Use the SWTCH program to activate and transfer your new system to the system disk of the new configuration.

When you are finished with the generation, you should always back up your disk. A minimum backup would consist of a physical backup (PSAVE) of LUs 2 and 3. Save the other disk LUs if the disk configuration will be modified by the new generation. It is important that you can always get back to a working operating system in case a planning mistake was made during generation. The disk shipped with your system contains the software you will need to generate all systems in the future and must not be overwritten by any RTE generation, even one that is error-free. Hewlett-Packard provides utility routines for disk backup, verification, and restoration. Consult the RTE-6/VM Utility Programs Reference Manual for disk backup instructions.

After assuring that your factory-generated disk has been suitably protected and cannot be destroyed by the switch, follow the procedures that are given in this section for executing the SWTCH program.

Glossary

The following terms will be used in the description of SWTCH:

CS/80 disks	The Command Set 80 (CS/80) disks have their own command set. They use an HP 12821A interface card and the DVM33 disk driver. These disks cannot be on the same card as non-CS/80 disks. CS/80 disks include such disks as HP 7908, 7911, 7912, 7914, 7933, 7935, 7937, and 7958.
MAC disks	Multiple Access Controller disk drives use the HP 13037 disk controller and the online disk driver, DVR32. The HP 7905, 7906, 7920 and 7925 models are MAC disks.
ICD disks	Integrated Controller Disks have their own controller in each disk drive. They use the HP 12821 interface card and the DVA32 disk driver. The HP 7906H, 7920H, 7925H and 9895 models are ICD disks.

host system	The current RTE operation system under which SWTCH is executing.
host configuration	The hardware system on which the host system is executing.
destination system	The RTE operating system that was defined during system generation.
destination configuration	The hardware system where the destination system will execute.
target disk	The portion of a disk in the host configuration where SWTCH will store the destination system. Target disk does not refer to a complete platter, but to the specified subchannel only.
target select code	The select code of the I/O slot where the target disk is plugged in. Applies to 7900 switches only.
target disk LU	A logical unit number in the host system which references any disk subchannel on the target disk. This LU is not affected by SWTCH. It is a point of reference for SWTCH to find the select code of the target disk driver. Applies to MAC, ICD, and CS/80 switches.
batch mode	SWTCH executes without user intervention. Batch mode is disallowed when a YES response is given for the subchannel initialization option parameter.

Types of Transfers

The SWTCH program offers flexibility in transferring your new operating system. For example, you may transfer your new RTE system to an I/O configuration that differs from the current I/O configuration. In this case, the destination can still be booted up using the RTE-6/VM I/O reconfiguration procedure (refer to Chapter 10 of this Manual for more information).

Below is a summary of the basic types of transfers offered by SWTCH.

1. SWTCH can transfer the new system to the current host system thereby replacing the host system while saving its file structure. Be sure to back up your host system.

2. SWTCH can transfer the new system to the target disk in the host configuration. You have the option of preserving the file structure contained on any previous system disk subchannel that exists on the target. The destination system can then be booted up with a different I/O configuration than the host.

   For example, the select code of the 7906 system disk controller may be number 12 in the host system. An HP 7906 system that has been generated may have the system disk controller in select code 13. Using a target select code of 12, SWTCH will allow you to store the destination system on the target disk. When SWTCH completes the transfer, you may physically change the I/O cards of the host configuration to the proper slots for the destination configuration. After the necessary cards are moved you can boot up the destination system.

3. SWTCH can transfer the new system to the host disk drive, where the system cartridge has been replaced by a temporary target cartridge for the duration of the SWTCH process. This temporary target can be transported to a system having the destination configuration and that system booted up. The original system cartridge can be placed back in the host and all activity there is resumed where it was suspended by SWTCH.

   SWTCH provides maximum protection for MAC, ICD, and CS/80 switches by suspending all I/O to the target disk while SWTCH is executing. For 7900 switches, the system must be quiescent to keep from corrupting the target disk.

4. SWTCH can transfer a newly generated ICD based system to a MAC target disk drive containing a temporary target cartridge. This target disk is later transported to an ICD destination configuration where it can then be booted up. Only the MAC driver, DVR32 needs to have been generated into the host. The opposite case of transferring a new MAC based system to an ICD target disk drive for later installation in a MAC destination system is also possible.

   For example, suppose that the host system is configured with a 7906 MAC disk. DVR32 (Rev. 2001 or later) is the only driver generated into this system. RT6GN is run on the host system to create an FMP type 1 file containing an ICD based system. SWTCH is run on the host system. At the proper time, the host's system cartridge is removed, and is replaced with the target cartridge. The destination system is installed on the cartridge by SWTCH, using the online DVR32 MAC disk driver. SWTCH tells the operator to remove the target cartridge and replace the host system LU 2 cartridge. Control is passed back to the host when SWTCH terminates. All disk I/O resumes where it was suspended by SWTCH. The target cartridge can then be transported to an ICD configuration and booted up.

   Unlike the ICD to MAC transfers described above, SWTCH can only transfer a newly generated CS/80 based system to a target CS/80 disk type from a host ICD, MAC, or CS/80 based system.

   The driver for the CS/80 disk (DVM33) must be generated into the host system before the transfer can take place.

In order to understand the flexibility available with SWTCH, it is helpful to understand how SWTCH communicates with the disk drivers.

For HP 7900 disks, SWTCH has its own internal driver. SWTCH asks for the select code of the target HP 7900 drive and then the platter where the new system will be stored. The host may be an HP 7900 based system or it may be MAC, ICD, or CS/80 based without the HP 7900 disk driver configured into the system.

For MAC and ICD disks, the appropriate driver DVR32 or DVA32 must have been generated into the host system. SWTCH asks for the target disk LU (see the glossary at the front of the chapter for the definition of target disk LU). It then asks for the MAC hardware unit number or the ICD address number where the new system will be stored.

If you are transferring a newly generated ICD based system to a MAC target disk drive, or vice versa (see the above paragraphs about SWTCH transfer types), only the appropriate driver for the host disk drive needs to be present in the host system. For CS/80 disks, the driver DVM33 must have been generated into the host system. SWTCH asks for the targeted disk LU (just as for ICD or MAC disks) and then asks for the HP-IB address, unit number, and volume number, where the new system will be stored. If specified, the unit and volume numbers must be zero (0). The unit number and volume number will default to zero (0) when not entered.

SWTCH Options

In addition to the various types of transfers possible with SWTCH, the following options are available.

Autoboot:	The autoboot option can be specified so SWTCH automatically boots up the new system on the completion of transfer. The destination configuration must be the same as the host configuration. Note that if the bootstrap loader was sent to a file during the generation process, the file should be punched or written out before SWTCH is executed.
Filesave:	The filesave option gives you the opportunity to save all the files on the target disk. The target system subchannel definition must be the same as the destination system subchannel. If this match does not occur, SWTCH warns you that information on the target disk will be destroyed and give you the option of proceeding.
Purge Type 6 Files:	SWTCH provides the option of saving or purging the type 6 files (memory image program files) existing in the file structure of the target disk.
Subchannel Initialization:	The destination disk system subchannel is initialized automatically. For non-CS/80 disks, SWTCH gives you the option of initializing all of the destination disk subchannels, none of them, or interactively allows you to specify which subchannels are to be initialized. Do not confuse disk subchannel initialization with FMGR disk cartridge initialization. For CS/80 disks, SWTCH initializes only the system subchannels.

SWTCH Loading Instructions

SWTCH must be loaded only as a regular large background (type 3 or 4) program, requiring 23 pages. SWTCH references the disk utilities libraries $DSCLB, $UTLIB, and $DTCLB, and makes use of special tables in the system. If the disk utilities libraries are not generated into the host system, these libraries may be searched when loading online by using the command file #SWTCH:

```
RU,LINK,#SWTCH
```

SWTCH can use the CI file system, thereby requiring that you provide the CI file system Library during loading. You need not, however, have the CI file system installed on the system. (That is, you do not need to load the CI file system version of D.RTR.) In summary, you must load SWTCH with the CI file system library, but you need not load any other CI file system code to successfully run SWTCH.

SWTCH does a core-lock, and if the host system does not allow a BG program to do a core-lock, then it will abort with the SC07 message.

Caution The revision of SWTCH should be the same as the revision of the RT6GN program used to produce the system file or errors can occur when running SWTCH.

SWTCH Operating Instructions

Use the RU command to schedule SWTCH for execution. You may specify any or all of the seven parameters with the RU command or enter them interactively as responses to SWTCH prompts.

The command is issued in the following form:

```
:RU,SWTCH,namr,{ scB/    },[ addr/unit/pltr ],autoboot,filesave,Type-6,init
               { disc LU }  [ addr:unit:vol  ]
                            [      -1        ]
```

where:

namr
: is the FMP file descriptor of the file that contains your generated system. This file must exist on a standard host system subchannel. If a target cartridge is to be inserted for the SWTCH process, the file must not exist on the cartridge that is to be swapped out for the target.

scB/disk LU
: sc: for the 7900 disk, sc is the select code of the target disk controller (octal value with a B as the terminating character). This target select code does not need to be configured into either the host or the destination RTE system. It is used as a means of specifying the correct controller I/O card for the transfer. SWTCH configures its own driver to this select code.

  disk LU: for switching MAC, ICD, or CS/80 based systems, the target disk LU is the logical unit number of any disk subchannel on the target disk. The LU is not affected by SWTCH. It is a reference for SWTCH to find the select code of the target disk driver. The target disk driver, DVR32 for MAC disks, DVA32 for ICD disks, or DVM33 for CS/80 disks, must be present in the host system. Neither LU 2 nor LU 3 should be specified as the target disk LU because the system does special checks to protect these LUs. If LU 2 or LU 3 is specified for the target disk and that disk, while being initialized, is found to contain more sectors per track than the host systems LU 2 or LU 3, SWTCH will be aborted with an IO07 error.

addr/unit/platter
: Note that the addr/unit/platter is for ICD or MAC disks and the addr:unit:vol is for a CS/80 disk. A prompt for the right disk information will be issued, based on the disk LU, if any other syntax is used.

  Enter −1 to default to the value defined at generation.

  address − for ICD disks, enter the target ICD address number (0-7) where the new system will be stored.

  unit − for MAC disks, enter the hardware unit number (0-7) where the new system will be stored.

  platter − for HP 7900 disks, enter the logical surface number where the new system will be stored (0, 2, 4, or 6 for the fixed platter; 1, 3, 5, or 7 for the removable platter).

addr:unit:volume
: address − enter the HP-IB address (0-7) for the target CS/80 disk, where the new system will be stored. Enter −1 to use the values defined during generation for the disk LU specified as the second parameter.

unit – the unit number (0-14) associated with the addr for the target CS/80 disk where the new system will be stored. Enter the unit number. If not entered, the default of zero (0) will be used.

volume – the volume number (0-7) associated with the unit for the target CS/80 disk where the new system will be stored. Enter the volume number. If not entered, the default of zero (0) will be used.

The disk system will be transferred to the subchannel that was defined as LU 2 during system generation.

autoboot — is the automatic boot-up option.

Specify Y (yes) to attempt an automatic boot-up following the transfer of the new system. The host configuration must match the destination configuration. See the paragraph titled Autoboot Specification for more detail on this match. Specify N (no) to deny automatic boot-up.

filesave — is the filesave option.

Specify Y (yes) to attempt saving the target disk's current file structure during the transfer.

Specify N (no) to deny saving the target disk's current file structure.

type 6 — is the option to purge type 6 files.

Specify Y (yes) to purge the target disk's type 6 files during the transfer.

Specify N (no) to deny purging the target disk's type 6 files.

Note Some type 6 files can be executed only on the operating system on which they were created.

init — is the subchannel initialization option.

Specify Y (yes) to request initialization of destination disk subchannels other than the system subchannel. SWTCH will prompt you for each subchannel that was defined to be on the same disk controller (MAC disks) or interface card (ICD disks) as the system subchannel.

Note that SWTCH will not initialize subchannels defined on the 9895 floppy disk. This must be done with the FORMT utility. For CS/80 disks, SWTCH will not initialize subchannels other than LU 2 or LU 3. Bad areas on these subchannels may be spared with the FORMC utility.

Specify N (no) to deny additional subchannel initializations. Batch mode is implied.

You can omit any of the above parameters from the command entry string. If any parameters were omitted, a comma must be specified as a place holder for each of the omitted leading parameters. Omitted trailing parameters do not require a place holder. During execution, SWTCH displays a prompt message for any omitted or illegally specified parameters. If the response entered interactively is invalid, SWTCH will reissue the prompt.

Examples:

`:RU,SWTCH,NEWGEN::17`	Only the file name (with a cartridge label)is specified. SWTCH will request the other six parameters.
`:RU,SWTCH`	No parameters are specified. SWTCH will request all of the parameter information.
`:RU,SWTCH,,,,Y`	Only the autoboot option is specified. SWTCH will request the omitted information.

If you specify all of the parameters and a NO response was entered for the subchannel initialization option, batch mode is implied and SWTCH will execute without your intervention. However, if FMP files within the new system will be destroyed at the target subchannel, you will be warned and asked for permission to continue.

SWTCH displays the following message at the beginning of its execution:

Warning **All activity must be terminated before system transfer process.**

Remember that the transferred system may be corrupted if other processes continue while SWTCH is executing.

HP 7900 Switches: For switching HP 7900 based systems, remember that SWTCH has its own internal driver. Therefore, the interrupt system is turned OFF during the transfer process, and you must be careful to terminate ALL system activity before initiating this process.

If this precaution is not observed strictly, the new system may be corrupted as it is written on the target disk. The host system may also be damaged, because output normally going to the host LU 2 will be on the target disk.

MAC/ICD/CS/80 Switches: For switching MAC, ICD, and CS/80 systems, SWTCH uses the online drivers and locks all disks on the same EQT of the target disk for the duration of the switch. All loads, swaps, and all other I/O to these disks will be held off by SWTCH to protect both the target cartridge and the integrity of the host system.

Although it is not strictly necessary to have a quiescent system during ICD/MAC/CS/80 switches, the performance will be severely degraded for the entire duration of SWTCH. For example, if PRMPT and RPN are disk-resident, all session terminals will appear to be dead until SWTCH terminates and unlocks the disks.

Filename Specification

SWTCH performs a validity check on the FMP file name specified by the filename parameter. The file named must exist as an FMP file in the host system and must be an RTE-6/VM system generated by RT6GN. This file must also be a Type 1 file beginning with the header records followed by the track 0, sector 0 boot extension and it cannot be an extended file. If the type 1 file has extents then store it to a file with no extents as follows:

```
:ST,OLDSYS::XX,NEWSYS::XX:1:-1
```

If this validity check fails, SWTCH displays the short version of the message below. If an error occurs during the FMP OPEN call, SWTCH displays the entire message indicating which FMP error occurred.

```
ILLEGAL FILENAME [ -FMP ERR XXXX]
FILE NAME OF NEW RTE SYSTEM?
```

At this point a valid file name of an RTE-6/VM system must be entered.

If the filename parameter was omitted from the RU command entry string, SWTCH requests:

```
FILE NAME OF NEW RTE SYSTEM?
```

You enter the FMP file descriptor of the file that contains your new system.

Note SWTCH can be aborted at any time it is requesting input with the !! command. If a file name begins with the exclamation characters (!!), precede the file name with a blank character.

After the file name has been entered, SWTCH reads the file and displays the time and date of generation. The time is the same as that appearing at the beginning of the generator answer file. For example:

```
FILE NAME OF THE NEW RTE SYSTEM?           RTE6::1234

RTE-6/VM SYSTEM GENERATED 11:30 AM MON., FEB 16, 1981
```

If a SWTCH segment is missing, SWTCH issues the following error message:

```
SWTCH SEGMENT MISSING
```

and terminates. To correct the error, reload SWTCH, making sure that all segments are correctly loaded.

Destination I/O Configuration

Then SWTCH displays the I/O configuration of the new system:

```
NEW SYSTEM I/O CONFIGURATION:
SELECT CODE cc PRIVILEGED INTERRUPT (if present)
SELECT CODE cc TBG
SELECT CODE cc TYPE= ee
    .     .                (in order of select code number)
    .     .
SELECT CODE cc TYPE= ee
```

where:

cc	is the I/O select code.
ee	is the equipment type code (the last two digits of the driver name; e.g., 05 for DVR05).

System Subchannel Definition

SWTCH derives the destination system's select code and subchannel from the file and displays the following message:

```
NEW SYSTEM (LU 2) SELECT CODE = cc SUBCHANNEL = ss
```

where:

cc and ss	are the actual select code and subchannel numbers.

Depending on the disk model of the new system, SWTCH reports the system subchannel definition:

For ICD and MAC disks:

```
#TRACKS        nnnn        FIRST CYL        cccc
HEAD #         hhhh        #SURFACES        ssss
ADDR/UNIT      uuuu        #SPARES          pppp
#SECTORS/TRACK                              kkkk
```

where:

nnnn	is the number of tracks
cccc	is the first cylinder number
hhhh	is the starting head number
ssss	is the number of surfaces
uuuu	is the MAC hardware unit number or ICD address number
pppp	is the number of spares
kkkk	is the number of (64-word) sectors/track.

For CS/80 disks:

```
 #OF TRACKS             nnnnn ADDRESS          a
 UNIT #                 ii VOLUME#             v
 STARTING BLOCK ADDRESS          bbbbbbbb
#OF 128-WORD BLOCKS/TRACK              tt
```

where:

nnnnn	is the number of tracks (1-32767)
a	is the HP-IB address (0-7)
ii	is the unit number (0-14)
v	is the volume number (0-7)
bbbbbbbbb	is the relative starting block address (0-1579916)
tt	is the number of physical (128-word) blocks per track (0-64)

Disk LU/Select Code Specification

If the select code or target disk LU parameter is omitted from the RU command entry string, SWTCH prompts:

For HP 7900 disks: `TARGET SELECT CODE FOR NEW SYSTEM (XX OR " "CR)`

For a HP 7900 disk, you respond with the octal select code (XX) of the correct controller I/O card, or a space followed by a carriage return. The select code number specified may refer to a select code in the host system, the destination system, or it may be a select code that is not configured into either system (SWTCH will configure its own driver to the select code specified). Entry of `" "CR` results in a default to the destination select code defined during the generation of the new system. If the select code is invalid, SWTCH will issue the warning:

```
ILLEGAL TARGET
```

For ICD, MAC, and CS/80 disks: `TARGET DISK LU FOR NEW SYSTEM?(XX)`

For ICD, MAC, and CS/80 disks, you respond with a decimal disk LU number that refers to DVA32, DVR32, or DVM33, respectively, in the host system as it is presently configured. This LU is not affected by SWTCH; it is used as a reference for SWTCH to find the select code of the target disk drive. If you enter LU 2 or 3, SWTCH will scan your system for some other LU that points at the same EQT. If no other LU can be found, SWTCH prints the message:

```
--AN LU OTHER THAN 2 OR 3 MUST EXIST FOR THIS DEVICE.

--ILLEGAL TARGET
```

At this point, you should terminate SWTCH by entering !! in response to the repeated question and assign some other LU to the same EQT using the system LU command. An SL command may also be needed because if you are in session you must also have this LU in your Session Switch Table. SWTCH can then be re-run. If the target disk LU does not point to a disk subchannel on the target disk, SWTCH will issue the warning:

```
ILLEGAL TARGET
```

There is no default allowed for ICD, MAC, and CS/80 target disks, i.e., you must enter a decimal disk LU rather than `"  "CR`. This prevents you from accidentally overlaying your system disk.

Address/Unit/Platter Specification

If the address/unit/platter is omitted from the RU command entry string or is of illegal form (see the runstring), SWTCH asks:

```
TARGET ADDRESS/UNIT/PLATTER FOR NEW SYSTEM? (X OR "  "CR)
```

You respond with one of the following:

For 7900 disks	enter the logical surface number 0, 2, 4, or 6 for the fixed disk; 1, 3, 5, or 7 for the removable platter where the new system will be stored.
For MAC disks	enter the hardware unit number (0-7) where the new system will be stored.
For ICD disks	enter the ICD address number (0-7) where the new system will be stored.

Entry of `"  "CR` results in a default to the destination value defined during generation.

Note that this hardware unit number or address number does not have to exist in the host system. This allows the user to plug a temporary target disk drive into the 13037 controller or ICD bus for the duration of SWTCH without regenerating to include the drive in the host system. The only requirement is that the temporary target disk drive must be connected to the same controller or bus as the target disk LU specified above.

The flexibility provided by the select code and address/unit/platter specifications permits temporary storage of your destination system on a target disk cartridge. Notice that you can boot up your destination system only on the destination select code and address/unit/platter that was specified during the generation process (unless the disk select code is changed during I/O reconfiguration).

Address:unit:volume Specification

If the address:unit:volume parameter is omitted from the RU command entry string or is of illegal form (see the runstring), SWTCH asks:

```
TARGET ADDRESS:UNIT:VOLUME OF CS/80 DISK FOR NEW SYSTEM?
(X:0:0 or " "CR
```

You respond with the CS/80 disk HP-IB address (0-7) where the new system will be stored. The unit number (0-7) specifies which unit of the addressed drive is to be used. At this time the only disk unit that is not 0 is the 7907 removable cartridge which is unit 1. The volume number must be zero (0) if specified. If the address:unit:volume is different then the one for the host system, you must have this target disk definition generated into the host system.

If just the address is entered, then the unit and volume numbers will default to zero (0).

An entry of `" "CR` results in a default to the destination address defined during generation for the disk LU specified.

VOLUME must always be zero!

Disk Cartridge Exchange

Except when batch mode is implied, SWTCH reminds you that the correct disk cartridge must be in place at the target address/unit/platter number or address:unit:volume for CS/80 disks. The following message is displayed:

```
NOW IS THE TIME TO INSERT CARTRIDGE
IN TARGET ADDRESS/UNIT/PLATTER.(" "CR TO CONTINUE)
```

or

```
IN TARGET ADDRESS:UNIT:VOLUME FOR CS/80
```

When this occurs even the operating system platter (LU 2) may be removed and another cartridge inserted. The absolute output file, however, must not reside on the removed cartridge, nor should it lie in the area of the target subchannel.

Perform the appropriate action and signal SWTCH to continue by entering `" "CR`.

From this point on, a !! response will cause SWTCH to issue a modified abort sequence of:

```
TRANSFER CANCELLED

IF RETURNING TO HOST SYSTEM, TARGET CARTRIDGE
MUST NOW BE REPLACED BY HOST CARTRIDGE
(" "CR TO CONTINUE)
```

After the host system is replaced, entering `" "CR` will issue the message:

```
SWTCH TERMINATED
```

Filesave Specification

If the filesave parameter is omitted from the RU command entry string, SWTCH requests:

```
SAVE FILES AT TARGET?  (Y OR N)
```

You respond with a Y (yes) to save files on the system subchannel (subject to the match conditions described in the following paragraphs), or with an N (no) to indicate that no files are to be saved.

A match must exist between the subchannel definition already on the target disk and the destination system subchannel definition. In other words, for ICD or MAC disks, the first track, the number of tracks, the number of surfaces, and the starting head of both subchannel definitions must be the same. For CS/80 disks, the track maps defined during generation, the starting block number, the number of blocks/track, and the number of tracks must all agree. SWTCH reads from the target disk area in order to determine a match. An FMP file directory must exist on the last track of the target disk subchannel in order to save the existing file structure.

If the match conditions fail, a warning is displayed, followed by a request for your permission to continue.

For ICD or MAC disks, the following message is displayed:

```
INFORMATION STORED ON ADDRESS/UNIT/PLATTER x OF TARGET SELECT
CODE yy WILL BE DESTROYED.

OK TO PROCEED? (Y OR N)
```

For CS/80 disks, the following message is displayed:

```
INFORMATION STORED ON ADDRESS:UNIT:VOLUME a:u:v OF TARGET
SELECT CODE yy WILL BE DESTROYED.

OK TO PROCEED? (Y OR N)
```

If you respond with a Y (yes), the information on address/unit/platter x or address:unit:volume a:u:v of target select code yy may be destroyed. A no (N) will prevent the destruction of this information.

Caution After verifying that target disk files can be saved, SWTCH attempts to find the cartridge directory on the target disk. If successful, it displays the directory and queries as follows:

```
THIS CL LOOKS REASONABLE, IF YOU AGREE AND YOU WANT TO
SAVE IT ANSWER YES, ELSE NO.  SAVE CL?  (Y OR N?)
```

You should verify that the CL looks reasonable and that it contains only entries that will not vary because of the new configuration. That is, each LU will still point to the same physical area of the disk. If the CL is reasonable and you wish to save it, enter Y. If you wish to proceed without saving the CL, enter N.

If the CL is not found, is rejected by answering N above, or is corrupt (which may be discovered after part of the CL is printed) this warning and prompt appear:

```
CL WAS NOT FOUND OR WAS CORRUPT OR REJECTED.
FILES ON LU 2 WILL BE SAVED.

OK TO PROCEED? (Y OR N)
```

If you proceed at this point, the following caution applies:

Since RT6GN places a new cartridge directory with a null master security code at the end of the new RTE-6/VM system generated, no cartridges will be mounted after the system transfer. SWTCH makes no attempt to preserve any files on the auxiliary subchannel (LU 3) or any other peripheral disk subchannel. Unless you request additional subchannel initializations, only the area occupied by the system subchannel on the disk is accessed by SWTCH. Therefore, it is your responsibility to save any of these files before the transfer.

If you save the CL the following caution applies:

SWTCH does not verify that the target disk description and organization match the generated description except for LU 2. Before running SWTCH you must dismount any cartridges that will no longer match the description.

Overlaid FMP Files

If the new system will overlay any of the existing FMP files on the system subchannel (LU 2 only) of the target disk, a warning message is displayed, followed by a request for your permission to continue:

```
NEW SYSTEM WILL DESTROY SOME FMP FILES.
OK TO PROCEED? (Y OR N)
```

Type 6 Specification

SWTCH provides the option of saving or purging (for the destination system) the target disk's type 6 files during the transfer. The match conditions described under the heading FILESAVE SPECIFICATIONS must be met in order to save type 6 files on the system subchannel.

If the target file structure is to be saved and the type 6 parameter is omitted from the RU command entry string, SWTCH displays:

```
PURGE TYPE 6 FILES? (Y OR N)
```

You respond with a Y (yes) to purge the type 6 files, or with an N (no) to save them.

Type 6 files contain a program in memory-image format that the system assumes is ready to execute. Type 6 files are created by LINK and the FMGR Save Program (SP) command. The first two sectors of the file contain ID segment information.

When a type 6 file is restored with the Restore Program (RP) command, an ID segment is set up for that program in memory. Note that such a program can usually execute only on the system within which it was created. The base page linkages and the setup word will be specific to the system in which the program was created and determine whether or not a program can execute on more than one system. Refer to the Transportable Type 6 Files section in Chapter 4 for more information on program transportability.

You may want to save type 6 files in those situations where you switch back and forth (using the SWTCH program) between RTE systems and do not wish to reload your programs after each switch.

Subchannel Initialization

SWTCH reformats the ICD/MAC disk track areas defined for RTE subchannels by writing their physical track and sector addresses in the preamble of each sector. For the system code area, the preambles are set to indicate write-protected tracks. When a defective track is encountered during the initialization of an ICD or MAC disk subchannel, a spare track is assigned to it. The preamble of the defective track indicates that it is defective and gives the address of the spare track that is replacing it so the disk controller will automatically switch to that track in future references. The preamble of the spare track indicates that it is acting as a spare, and the address of the defective track it is replacing. An ICD or MAC disk must be formatted before SWTCH initialization because SWTCH must check and acknowledge a previously detected defective track.

CS/80 disks have the feature of hardware block sparing. The user does not have to specify to spare. SWTCH will report any bad blocks (refer to the section on Bad Track Information) as it initializes all tracks to a common value.

If the init parameter is omitted from the RU command entry string, SWTCH requests:

```
INITIALIZE SUBCHANNELS? (Y OR N)
```

Respond with a Y (yes) to continue with the initialization requests for the disk subchannels, excluding the system subchannel. A no (N) response terminates SWTCH after only the system subchannel has been transferred and initialized.

SWTCH will prompt you (interactively) for each additional subchannel defined for the system disk controller or HP 12821A interface during generation. The actual subchannel initializations will follow the system transfer process. If the disk is new, if it has any write protect flags written on it, or if you are changing the subchannel definition for this pack, the disk must be initialized with the SWTCH or FORMT program. If you do not wish to disturb the information that is contained on the disk in the subchannel's designated area, you should respond with an N (no) to the initialization request.

HP 7900 Subchannel Initialization

For each HP 7900-based disk subchannel defined during the generation, SWTCH prompts:

```
INITIALIZE SUBCHANNEL s? (Y OR N)
```

You respond with a Y (yes) to initialize subchannel s, an N (no) to indicate that the subchannel is not to be initialized, or a /E to terminate the initialization prompting.

If the subchannel is to be initialized, SWTCH asks:

```
TARGET PLATTER?  (X OR " "CR)
```

You respond with the logical surface number (0, 2, 4, or 6 for the fixed platter; 1, 3, 5, or 7 for the removable platter) where subchannel s is to be initialized. Entry of `" "CR` results in a default to the destination platter that was defined for subchannel s during generation. Specifying a target platter that is identical to the target platter for the system subchannel is not allowed. If you specify matching target platters, SWTCH will reissue the prompt.

ICD and MAC Subchannel Initialization

For ICD and MAC disk subchannels, the subchannels are grouped according to their generation-defined MAC hardware unit number or ICD address select number. For those subchannels having their destination address/unit the same as the system subchannel's address/unit, the target unit will automatically be that target unit specified for the system subchannel. SWTCH displays:

```
TARGET ADDRESS/UNIT u FOR SUBCHANNELS XX,YY,...,ZZ
```

and then asks:

```
INITIALIZE SUBCHANNEL XX? (Y or N)
  :
INITIALIZE SUBCHANNEL ZZ? (Y or N)
```

After each prompt, you respond with a Y (yes) to initialize subchannel nn, an N (no) to indicate that the subchannel is not to be initialized, or a /E to terminate the initialization prompting.

You respond with a MAC unit number or ICD address number (0-7) for this group of subchannels or with a /E to decline initialization for the group. Entry of " "CR results in a default to the destination address/unit LU. Specifying a target address/unit that is identical to the target address/unit for the system subchannel is not allowed. If you specify matching target address/units, SWTCH will reissue the prompts.

Note SWTCH reports the DESTINATION ADDRESS for 9895 disks, but does not prompt for TARGET ADDRESS/UNIT, since the FORMT utility must be used for floppy disk initialization.

If you did not respond with a /E to the TARGET ADDRESS/UNIT question, SWTCH prompts:

```
INITIALIZE SUBCHANNEL XX? (Y or N)
  :
INITIALIZE SUBCHANNEL ZZ? (Y or N)
```

After each prompt, you respond with a Y (yes) to initialize subchannel nn, an N (no) to indicate that the subchannel is not to be initialized, or a /E to terminate initialization prompting for this group of subchannels.

SWTCH stops prompting when all generation-defined subchannels have been prompted for initialization, or when a /E has been entered. Actual initialization will be done (by SWTCH) following the system transfer.

Autoboot Specification

Automatic boot-up of the new system may occur following the transfer and initialization operations if the first five of the following six conditions are true. The sixth condition must also be true if both systems have a privileged interrupt card.

1.	Target disk select code	=	Destination disk select code
2.	Target disk address/unit/platter or address:unit:volume	=	Destination disk address/unit/platter or address:unit:volume
3.	Host TBG select code	=	Destination TBG select code
4.	Host system console select code	=	Destination system console select code
5.	Target disk type (ICD vs. MAC vs. CS/80)	=	Destination system disk type (ICD vs. MAC vs. CS/80)
6.	Host privileged interrupt select code	=	Destination privileged interrupt select code

If the automatic boot-up conditions are true and the autoboot parameter is not specified in the RU command string, SWTCH prompts:

```
AUTO BOOTUP? (Y OR N)
```

If any one of the automatic boot-up conditions is false, SWTCH displays the following message:

```
PRESENT CONFIGURATION DOESN'T PERMIT AUTO BOOT-UP.
```

SWTCH will check if the host system disk has the same addr/unit/subch values as the target. If these values are equal, SWTCH displays the following message:

```
DISK IN HOST SYSTEM DRIVE WILL BE OVERLAYED
```

This is a warning message indicating that the new system is being installed on the host system LU 2. No attempt is made to determine if the new system will overlay the system file. This file should be saved before switching the systems.

If it is not possible to return to the host system following the transfer operation, or if a transfer or initialization was done to the same address/unit/platter as the host (LU 2) and automatic boot-up is not to be done, SWTCH displays this message:

```
SYSTEM WILL HALT AFTER TRANSFER COMPLETION
```

If everything proceeds normally and if batch mode is not implied, SWTCH requests final permission for system transfer. The following message is displayed:

```
READY TO TRANSFER.  OK TO PROCEED? (Y OR N)
```

Respond with N (no) to deny the transfer at which time SWTCH will abort with the abort sequence messages (refer to the Abnormal Termination Messages section).

Respond with Y (yes) to proceed. At this point the host system is shut down and the transfer begins. Track sparing is done for the ICD, MAC, or CS/80 subchannels. If appropriate, SWTCH reports, under the following headings, the names of any files that are overlaid or purged during the system transfer:

```
OVERLAID FMP FILES:
   file list
```

or

```
TYPE 6 FILES PURGED:
   file list
```

Next, the new system subchannel is installed on the target disk (doing sparing as needed). The message,

```
INSTALLING SYSTEM SUBCHANNEL XX
```

is displayed as the transfer begins. At this point, SWTCH does another validity check on the system file to make sure the operator has not accidentally removed the cartridge containing the new system. (This may have happened when the operator was given a chance to remove the host and insert a target disk cartridge.) If the FMP file containing the new system has been removed, SWTCH displays the message,

```
DISK CARTRIDGE CONTAINING NEW SYSTEM FILE
XXXXXX HAS BEEN REMOVED FROM DISK DRIVE
```

(where XXXXXX is the filename), and aborts.

After successfully installing the system subchannel, SWTCH initializes any other subchannels requested by the operator, and prints the message:

```
INITIALIZING SUBCHANNEL XX
```

for each subchannel specified.

Normal Termination Messages

After system installation and subchannel initialization are complete, SWTCH checks again to see if the host system may have been overlaid. (It checks for a match between the target address/unit/platter or address:unit:volume and the host address/unit/platter number or address:unit:volume.) If there is a possibility that the host system was overlaid, and the automatic boot-up is not to be done, SWTCH displays the warning:

```
IF TRANSFERRING CONTROL TO NEW SYSTEM,
IT MUST BE BOOTED AFTER SWTCH TERMINATES.
```

Next, before I/O to the target disk is allowed to resume, SWTCH gives the user an opportunity to remove a temporary target cartridge, by displaying the message:

```
IF RETURNING TO HOST SYSTEM, TARGET CARTRIDGE MUST NOW BE
REPLACED BY HOST CARTRIDGE      (" "CR TO CONTINUE)
```

If in batch mode, or auto-boot mode, the above message is skipped, since it requires an interactive `" "CR` response from the user. Finally SWTCH prints the normal termination message:

```
SWTCH FINISHED
```

If the host system LU 2 has been overlaid with the new system, SWTCH halts with a 102077B in the T-Register.

Abnormal Termination Messages

There are three places during the interactive phase when you can stop SWTCH before the transfer starts:

1. When issuing the filename of the new system (abort with !!).
2. When asked to insert the target cartridge (abort with !!).
3. When asked if it is OK TO PROCEED? (answer with NO).

In the last two cases the abort sequence listed below will be issued.

If an error condition makes it necessary to terminate SWTCH, the user is given a chance to reinstall the host system disk cartridge before SWTCH unlocks the disk controller interface card of the target disk drive. This feature is useful when the operator wants to return to the host system and resume all I/O that was held off by the disk lock.

The abort sequence is:

```
TRANSFER CANCELLED.

IF RETURNING TO HOST SYSTEM,TARGET CARTRIDGE MUST NOW BE
REPLACED BY HOST CARTRIDGE (” ”CR TO CONTINUE)

SWTCH TERMINATED.
```

If the error condition which caused the abort occurred before the user was given an opportunity to insert a temporary disk cartridge, the abort sequence is simply:

```
TRANSFER CANCELLED.
SWTCH TERMINATED.
```

Bad Track Information

Defective tracks are reported as follows:

```
BAD TRACK PLATTER x
000yyy
```

where:

x	is the platter number.
000yyy	is the logical track number needed when initializing the File Manager on the subchannel.

Bad tracks on the ICD and MAC disks are automatically spared to tracks set aside by the generator for that purpose. Bad tracks reported and spared will not prevent operation of the system and should not be specified during File Manager initialization on the subchannel.

Defective tracks are reported as follows:

```
BAD TRACKS SUBCHANNEL xx

                  LOGICAL   CYL   HEAD  ADDR/UNIT
 BAD TRACK          tttt    cccc    h       u
 SPARED TO          tttt    cccc    h        u
```

where:

tttt is the logical track number (relative to the beginning of the subchannel).

cccc is the physical cylinder number on the disk pack.

h is the physical head number on the disk pack.

u is the ICD address or MAC hardware unit number of the target disk drive.

Tracks on a CS/80 disk are logically assigned during generation. The CS/80 disk controllers do hardware sparing by blocks and are therefore invisible to SWTCH. SWTCH may display the following messages.

```
WARNING: ONE SPARE PHY. TRACK LEFT. CONTACT SYSTEM MANAGER
```

Each physical track has one extra block for sparing. If a second block is required, one of several extra tracks is used to spare out the whole physical track. If the next to the last spare track is used, then the above warning is issued. Several more sparing operations may occur if the disk needs only one spare block per track. However, the disk will only be able to use one more extra physical track for sparing if the need arises. This warning indicates that there is something seriously wrong with the drive that is using up all the spares. Back up all your files on the disk and perform a complete diagnostic check on the disk drive subsystem.

```
PHYSICAL BLOCK nnnnnnnn SPARED FOR mmmmmm BLOCKS
```

where:

nnnnnnnn is the starting block address.

mmmmmm is the number of blocks spared.

There may be several sets of blocks that are spared. The above message will be issued for each set. This information should be available to all the users of any logical tracks and associated LUs that have been spared.

```
WARNING!!!  SPARE OUTSIDE SUBCHANNEL !!!
```

This message is displayed with the physical block message if the block that was spared was outside the system subchannel. Since a CS/80 physical track can contain more than one logical track, it is possible that a bad block in another subchannel can be spared.

SWTCH Example

For this example, the user is in this situation:

The user has a cartridge where he or she wants to put a newly generated system.

- The user wants to save the cartridge's file structure. Remember that Type 6 files can only execute on the system on which they were created.
- The new system (destination) will use an I/O configuration that is different from the host's and, therefore, will not be eligible for the autoboot option (specifically, the select codes for the devices will be changed).
- The target cartridge with the new system installed on it may be the current host system. The host system disk will be overlaid.
- Or, it may be a temporary target replacing the host only for the duration of SWTCH.
- Comments within the body of the example are in lowercase letters.

```
   :RU,SWTCH
                               * No parameters are specified so
                               * SWTCH is not in batch mode.  SWTCH
                               * will prompt for all parameters.
```

Warning **All activity must be terminated before system transfer process. Enter !! in response to any question to abort**

```
FILE NAME OF NEW RTE SYSTEM?
RTE06H::SS

RTE-6/VM SYSTEM GENERATED 11:30 AM MON., FEB 16, 1981

NEW SYSTEM I/O CONFIGURATION:

SELECT CODE 14 TBG
SELECT CODE 04 TYPE=43
SELECT CODE 10 TYPE=32
SELECT CODE 11 TYPE=32
SELECT CODE 12 TYPE=05
SELECT CODE 13 TYPE=23
SELECT CODE 15 TYPE=23
SELECT CODE 16 TYPE=12
SELECT CODE 17 TYPE=05
SELECT CODE 20 TYPE=05
SELECT CODE 21 TYPE=05
SELECT CODE 23 TYPE=05
SELECT CODE 25 TYPE=05
```

```
NEW SYSTEM (LU 2) SELECT CODE= 13  SUBCHANNEL= 00

#TRACKS   0256    FIRST CYL   0000
HEAD #    0000    #SURFACES   0002 * destination system subchannel
ADDR/UNIT 0000    #SPARES     0006 * definition.
#SECTORS/TRACK                0096

TARGET DISK LU FOR NEW SYSTEM?  (XX)
2
                                * defining the target disk driver

TARGET ADDRESS/UNIT/PLATTER FOR NEW SYSTEM? (X OR " "CR)
0
                                * defining the specific target disk

NOW IS THE TIME TO INSERT CORRECT CARTRIDGE IN
TARGET ADDRESS/UNIT/PLATTER.   (" "CR TO CONTINUE)

                                * host system LU 2 may be replaced
                                * by a temporary target cartridge
                                * now.

SAVE FILES AT TARGET?   (Y OR N)
Y

 LU L-TRK  CR     LU L-TRK  CR    LU  L-TRk  CR       LU L-TRK   CR
  2  399     2      3  395    3    41   202  <CIFS>   21  399    50
 20  401    51     11  202   R4    54  1029   17      52 1029    22
 17  502    RE     30  395   GU    51  1069  <CIFS>   56 1029  <CIFS>
 33  202    24     15  502   ED    16   502   XX      43  202    HH
 22  401    X4     14  202 1799    42   202  179      35  202    GG

THIS CL LOOKS REASONABLE.  IF YOU AGREE AND YOU WANT TO SAVE IT
ANSWER YES, ELSE NO.  SAVE CL?  (Y OR N)
Y

PURGE TYPE 6 FILES?   (Y OR N)
N
INITIALIZE SUBCHANNELS  ?  (Y OR N)
Y
                                * prompting for subchannel
                                * initialization begins here.

TARGET ADDRESS/UNIT  0 FOR SUBCHANNELS 01, 02, 03, 04, 05, 06, 07,

                                * these subchannels were configured
                                * to address/unit 0.

INITIALIZE SUBCHANNEL 01?  (Y OR N)
 /E
                                * terminates subchannel prompts for
                                * this address/unit.

DESTN. ADDRESS/UNIT  1 FOR SUBCHANNELS  08, 09, 10,

                                * unit 1 is a 9895 floppy disk.
```

```
DESTN. ADDRESS/UNIT  2 FOR SUBCHANNELS  11, 12, 13, 14, 15, 16,
TARGET ADDRESS/UNIT?  (XX OR ” ”CR)
                                * default is the destination
                                * address/unit specified during
                                * generation.

INITIALIZE SUBCHANNEL 11?  (Y OR N)
N

INITIALIZE SUBCHANNEL 12?  (Y OR N)
Y

INITIALIZE SUBCHANNEL 13?  (Y OR N)
/E
                              * terminates subchannel initialization
                              * prompts for this unit.

DESTN. ADDRESS/UNIT  3 FOR SUBCHANNELS 17, 18, 19, 20, 21, 22, 23,
24, 25, 26, 27, 28, 29, 30, 31,
TARGET ADDRESS/UNIT?  (XX OR ” ”CR)
/E
                                * terminates destination
                                * address/unit prompts.

PRESENT CONFIGURATION DOESN’T PERMIT AUTO BOOT-UP.

DISK IN HOST SYSTEM WILL BE OVERLAID.
READY TO TRANSFER. OK TO PROCEED?
YES

INSTALLING SYSTEM SUBCHANNEL  00
BAD TRACKS SUBCHANNEL 00
           LOGICAL CYL HEAD ADDR/UNIT
BAD TRACK   0004  0002  00  00
SPARED TO   0256  0128  00  00
BAD TRACK   0015  0007  01  00
SPARED TO   0257  0128  01  00

INITIALIZING SUBCHANNEL 12

IF TRANSFERRING CONTROL TO NEW SYSTEM, IT
MUST BE BOOTED AFTER SWTCH TERMINATES.

IF RETURNING TO HOST SYSTEM,TARGET CARTRIDGE MUST NOW
BE REPLACED BY HOST CARTRIDGE   (” ”CR TO CONTINUE)
                            * if LU 2 of host system was removed, it
                            * may be reinstalled now.

SWTCH FINISHED
```

SWTCH Error Conditions

An appropriate message will be displayed for any errors encountered during the execution of SWTCH. If SWTCH is aborted because of a disk error on the system subchannel, the system on the disk may not be a workable system.

Error conditions that result in an error message may be encountered at the following points in the execution of SWTCH:

1. While SWTCH is testing for the file structure on the target disk; i.e., it is reading from the target disk in a non-initialization mode.
2. While SWTCH is transferring the destination system to the target disk.
3. While SWTCH is initializing the remainder of the destination system subchannel.
4. While SWTCH is initializing one of the remaining disk subchannels.

Below are listed the SWTCH error messages, their meanings, and the suggested actions to be taken if any of the errors occur:

INVALID DISK SPECIFICATIONs ss

Disk Types:	7900/ICD/MAC/CS/80.
Meaning:	Disk specifications do not conform to system disk type, or track areas are too large. This can occur in two places. The first is attempting to read the disk after the save file option was specified. The second is attempting to initiate a write to the target disk. SWTCH is aborted (ss is the destination subchannel causing the error) if this error occurs on the system subchannel. HP 7900 switches terminate immediately. If initializing a non-system subchannel, SWTCH aborts this subchannel initialization and proceeds to the next.
Action:	Redefine track areas of generated destination system and regenerate.

READY DISK AND PRESS RUN ss	**7900 Disk**
−−ENTER " "CR ss	**ICD/MAC/CS/80 Disks**

Disk Types:	HP 7900/ICD/MAC/CS/80.
Meaning:	The disk device is not ready. For HP 7900 disks the system executes a HALT 33B (ss is the destination subchannel causing the error).
Action:	Ensure that the disk drive is ready and press RUN on the CPU control panel, or enter `"  "CR`.

TURN OFF DISK PROTECT−−PRESS RUN ss **7900 Disk**
−−ENTER " "CR ss **ICD/MAC/CS/80 Disks**

Disk Types:	HP 7900/ICD/MAC/CS/80.
Meaning:	The disk protect switch is in the PROTECT position. For HP 7900 disks, the system executes a HALT 32B (ss is the destination subchannel causing the error).
Action:	Turn off the switch and press RUN on the CPU control panel, or enter " "CR. (This switch is called the READ-ONLY switch for the HP 7920 model disk.)

PARITY OR DATA ERROR TRACK yyy ss

Disk Types:	HP 7900/ICD/MAC.
Meaning:	Read parity/decode error. Ten attempts have been made to read or write to disk track yyy. SWTCH is aborted (ss is the destination subchannel causing the error) if the error occurs on the system subchannel.
Action:	For an HP 7900, disk recovery is not possible. ICD or MAC disks will proceed to the next subchannel rather than aborting SWTCH.

TURN ON FORMAT SWITCH−−PRESS RUN ss **7900 Disk**
−−ENTER " "CR ss **/CS/80ICD/MAC Disks**

Disk Types:	HP 7900/ICD/MAC.
Meaning:	The Format switch is not in the ON position. For 7900 disks the system executes a HALT 32B (ss is the destination subchannel causing the error).
Action:	Set the Format switch to ON and press RUN on the CPU control panel or enter " "CR.

DEFECTIVE CYLINDER − TRACK XXXX ss

Disk Types:	HP 7900/ICD/MAC.
Meaning:	Disk error. SWTCH is aborted (ss is the destination subchannel causing the error).
Action:	Recovery is not possible.

LIMIT OF 10 BAD TRACKS EXCEEDED ss (7900 Disk only)

Disk Types:	HP 7900
Meaning:	More than ten bad tracks exist on a subchannel. SWTCH is aborted (ss is the destination subchannel causing the error).
Action:	Redefine the track area and regenerate, or get a new disk.

OUT OF SPARES XX (ICD and MAC DISKS only)

Disk Types:	ICD/MAC.
Meaning:	All available spare tracks have been used up. If this error occurs while installing the system subchannel, SWTCH will abort. When initializing a peripheral subchannel, SWTCH will abort this subchannel and proceed to the next.
Action:	Define more spare tracks for the problem subchannels, and regenerate.

UNABLE TO INITIALIZE SUBCHANNEL XX (ICD/MAC DISKS only)

Disk Types:	ICD/MAC.
Meaning:	Because of one of the previous error conditions, SWTCH soft aborted subchannel xx and will proceed to the next.
Action:	None.

OUT OF SPARES

Disk Types:	CS/80.
Meaning:	All available spare tracks have been used up.
Action:	If this error occurs at any time, data contained on the disk may be corrupt. Contact the HP Customer Representative.

QSTAT ERRORS

Disk Types:	CS/80.
Meaning:	SWTCH has received bad status back from the driver.
Action:	The disk is returning hardware errors. Check to see that all cables are connected and secure. If the error reoccurs, contact your HP representative.

FULL STATUS xxxxx xxxxx xxxxx xxxxx

Disk Types: CS/80.

Meaning: SWTCH prints the full status returned by the driver whenever it detects an internal error.

Action: The disk is returning hardware errors. Check to see that all cables are connected and secure. If the error reoccurs, contact your HP representative.

RELEASE ERROR

Disk Types: CS/80.

Meaning: SWTCH was unable to successfully issue a release to the disk.

Action: The disk is returning hardware errors. Check to see that all cables are connected and secure. If the error reoccurs, contact your HP representative.

DISK ERROR

Disk Types: CS/80.

Meaning: A hardware error has been detected by the sparing routine.

Action: The disk is returning hardware errors. Check to see that all cables are connected and secure. If the error reoccurs, contact your HP representative.

ILLEGAL ERROR

Disk Types: CS/80.

Meaning: An illegal error code was returned by the disk. May indicate a hardware malfunction.

Action: The disk is returning hardware errors. Check to see that all cables are connected and secure. If the error reoccurs, contact your HP representative.

WARNING !!! SPARE OUTSIDE SUBCHANNEL !!!

Disk Types:	CS/80.
Meaning:	While performing a spare operation on a physical track, the sparing routine detected and spared a bad block belonging to a logical track outside the system subchannel. This is a warning message only.
Action:	None

The following errors are all described in the DVM33 Driver Manual. Additional information relevant to SWTCH is noted below the message.

CHANNEL PARITY	UNIT FAULT
ILLEGAL OPCODE	DIAGNOSTIC RESULT
MODULE ADDRESSING	OPERATOR REQUEST
ADDRESS BOUNDS	DIAGNOSTIC REQUEST
PARAMETER BOUNDS	INTERNAL MAINTENANCE
ILLEGAL PARAMETER	POWER FAIL
MESSAGE SEQUENCE	RELEASE COMPLETED
MESSAGE LENGTH	ILLEGAL PARAMETER OPERATION
CROSS-UNIT	UNINITIALIZED MEDIA
CONTROLLER FAULT	NO SPARES

NOT READY. MAKE READY AND ENTER “ “CR

Disk Types:	CS/80.
Meaning:	If SWTCH detects that the disk is offline, it issues this warning message then pauses to allow the user to make the disk ready.
Action:	Put disk online.

WRITE PROTECT, TURN WRITE PROTECT OFF AND ENTER “ “CR

Disk Types:	CS/80.
Meaning:	When SWTCH cannot write on the disk due to write protect being on, it issues the above message then allows the user to turn off write protect.
Action:	Turn off write-protect and enter “ “CR.

NO DATA FOUND

Disk Types:	CS/80.
Meaning:	Disk area has never been formatted or written to.
Action:	Format the disk.

UNRECOVERABLE DATA

Disk Types:	CS/80.
Meaning:	A warning message printed by SWTCH when bad data is found on the disk. An attempt will be made to spare any bad blocks on the track.
Action:	None.

END OF FILE

Disk Types:	CS/80.
Meaning:	Switch program error.
Action:	Reload SWTCH and try to switch again. If the error reoccurs, you may have a corrupt copy of SWTCH or the software is incorrect; call your HP representative.

END OF VOLUME

Disk Types:	CS/80.
Meaning:	Switch program error.
Action:	Reload SWTCH and try to switch again. If the error reoccurs, you may have a corrupt copy of SWTCH or the software is incorrect; call your HP representative.

MARGINAL DATA

Disk Types:	CS/80.
Meaning:	A warning message printed when marginal data is suspected on the disk. The suspect area is spared if needed.
Action:	None.

MAINTENANCE TRACK OVERFLOW

Disk Types:	CS/80.
Meaning:	The disk error logging tracks overflowed.
Action:	Reload SWTCH and try to switch again. If the error reoccurs, you may have a corrupt copy of SWTCH or the software is incorrect; call your HP representative.

NON-DISK SUBCHANNEL

Disk Types:	CS/80.
Meaning:	Trying to initialize a non-disk subchannel (ss) (i.e., a cartridge tape drive).
Action:	SWTCH will prompt for next subchannel.

6

System Initialization

After you have generated and installed your new system, perform the following steps to make your system operational:

1. Boot up your new system.

2. Initialize your primary and auxiliary system cartridges (LU 2 and LU 3).

3. Check your system. Perform simple checks that check the operation of commands and devices for proper functioning. Check for generation errors. If appropriate, run the RTE reconfigurator to correct these errors.

4. Install utilities. Utilities not generated into the system should be configured into the system with LINK or the Online Loader. This chapter gives the online installation procedures for utilities included on the HP 92084A primary disk.

5. Create the system WELCOM file. The WELCOM file is a FMGR command file that is automatically executed at system startup. It can be used to enable terminals, initialize subsystems, set up ID segments, correct generation errors, and pack cartridges.

6. Install various user transfer files, documentation files, and support files. If desired, the HELP utility message file !HELP can be installed on the system and modified to suit the needs of your particular installation. On MTM systems, you may install a transfer file called the HI file to be executed when users invoke their copy of FMGR. For session systems, the System Manager may install specific transfer files to give low capability users access to special high capability level commands or to perform various privileged system functions.

7. Initialize the spooling system. If spooling has been configured into the system, you must initialize the spooling system by running GASP. This process is described in the *RTE-6/VM Batch and Spooling Reference Manual*, part number 92084-90006.

8. Create a backup copy of your system on magnetic tape or disk. The procedures required to save/restore your system on disk or magnetic tape are given in the *RTE-6/VM Utility Programs Reference Manual*, part number 92084-90007.

This chapter discusses these steps in greater detail. For your convenience, it is recommended that they be followed in the order presented.

Standard Boot-Up Procedures

System boot-up is the process of loading the operating system software into memory so that it is ready for execution. For CS/80 disks, the Bootstrap Loader is used to load the system directly. For ICD/MAC disks, boot-up begins by using either the Disk Loader ROM or Bootstrap Loader to load the Boot Extension into memory from track 0, sector 0 of the system disk subchannel. The Boot Extension, in turn, loads the operating system into memory.

At this point, you have the option of either completing a standard system boot-up procedure as described in this section or reconfiguring the current I/O and memory assignments as described in Chapter 10, Memory and I/O Reconfiguration. In a standard boot-up, the operating system immediately completes the rest of the initialization process as follows:

1. Displays a SET TIME message.
2. Executes a startup program (optional).
3. Passes control to the File Manager (FMGR), which schedules the following programs to initialize the various subsystems:

   a. LPAIR,−32768 (to initialize the DATASAFE subsystem, if present);
   b. GASP,−1 (to initialize the spooling system);
   c. ACCTS,−1 (to initialize the session environment);
   d. LOADR,−1,−1 (to create the system entry point file $SYENT).

   FMGR then tries to execute a procedure file named WELCOM. If the WELCOM file does not exist on LU 2, the FMGR displays a FMGR −006 error message.

If memory and/or I/O reconfiguration are to be performed during system boot-up, completion is delayed and an interactive Configurator program is scheduled via S-Register settings to make the new memory and I/O assignments. At the end of the reconfiguration process, control is returned to the system to complete the boot-up procedure as described above.

Use the procedures described below to perform a standard system boot-up. Use the procedures described in Chapter 10 to perform a boot-up with I/O and memory reconfiguration.

Boot Loaders and Boot Extension

The Disk Boot Extension (for non-CS/80 disks) can be loaded into memory from the disk using either the Disk Loader ROM or Bootstrap Loader.

Disk Loader ROM

The Disk Loader ROM can be used to load the Boot Extension if the Boot Extension resides on physical track 0, sector 0 of the system disk with the following address: HP-IB 0, unit 0, volume 0, block 0. Refer to the *HP 12992 ROM's Installation Manual*, part number 12992-90001, for a description of the S-Register setting to load the Boot Extension into memory. An example of a standard system boot-up using the 12992J 7908/7911/7912/7914/7933/7935/7937/7958 Disk Loader ROM is as follows:

1. Select the S-Register for display on the computer front panel.
2. Press CLEAR DISPLAY.
3. Set the S-Register bits as follows:

Bits:	**Enter:**
0-2	Surface number for MAC/ICD disk, unit number for CS/80 disk.
3-4	0 (reserved).
5	0 for standard boot-up of disk.
6-11	Octal select code of the disk.
12	1 for MAC/ICD disk, 0 for CS/80 disk.
13	0 (reserved).
14-15	Loader ROM selection (number of the ROM cell containing the Disk Boot Loader).

4. Press STORE.
5. Press PRESET, IBL and PRESET (again) to load contents of Disk Loader ROM.
   A successful load is indicated if the OVERFLOW indicator does not light up.
6. Press RUN.

Example:

1. Assume a standard boot-up from ROM #2, with an HP 7908 in select code 21.
2. Set the S-Register = 102100. Press STORE.
3. Press PRESET, IBL, PRESET (again) and RUN.

Bootstrap Loader

The Bootstrap Loader is used to load the Boot Extension into memory if the Boot Extension does not reside on physical track 0, sector 0 of the system disk or if the Disk Loader ROM is not available. The procedure is as follows:

1. Select the S-Register for display on the computer front panel.
2. Press CLEAR DISPLAY.
3. Set the S-Register bits as follows:

Bits:	**Enter:**
0-5	0
6-11	Octal select code of input device (for example, 264x Display Terminal).
12-13	0
14-15	Loader ROM select code.

4. Press STORE.
5. Press PRESET, IBL and PRESET (again) to load the bootstrap Loader. A successful load is indicated if the OVERFLOW indicator does not light up.
6. Press RUN.

When the HLT 77B occurs, clear the S-Register, set the P-Register to octal 100 and press RUN to continue.

Boot Extension Execution

The disk Boot Extension uses the S-Register to communicate with the configurator program (see Chapter 9). Do NOT change the S-Register contents until the system boot-up procedure is completed and the SET TIME message is displayed.

System Track Allocation

The system maintains complete control over the allocation and ownership of disk tracks on the system (LU 2) and auxiliary system (LU 3) subchannels. Track control is maintained via the Track Assignment Table (TAT). Peripheral disks (NOT LU 2 or LU 3) are not managed through the TAT.

Figure 6-1a shows the structure of the system disk subchannel (LU 2). This subchannel has three distinct areas. The first area, starting at track 0, is the system area. A memory image of the operating system, drivers, and all programs loaded at generation time are stored here. In addition, this location contains the system library relocatable modules and an entry point directory.

The second area forms the System Scratch track pool. System Scratch tracks are used in a variety of ways:

- Scratch tracks can be allocated to programs requesting scratch disk space with EXEC calls. If you need to run application programs in the system making extensive use of system scratch tracks, their requirements should be considered when estimating the number of scratch tracks to be configured in your system.
- Scratch tracks can be allocated for swap space. A contiguous block of available scratch tracks must exist for each program swapped out from memory on to disk. Generally, the amount of scratch tracks used for swapping is determined by the number of active programs contending for the same memory partitions.
- Scratch tracks are allocated to contain programs added to the system with the Online Loader. At least one track is allocated for each program added to the system in this manner.
- Scratch tracks are used for the logical source (LS) and load and go (LG) areas. These areas were used by pre-RTE-IVB Compilers, Assemblers, and Loaders when accessing source programs (LS) or relocatable binaries (LG) on disk. In the session environment, these areas are not generally accessible.

There must be a minimum of 8 tracks in the scratch track pool on LU 2, however, a minimum of 50 is recommended. If the EMA or MLS features of RTE-6/VM are being used, a larger system scratch track area may be necessary to allow swapping of large arrays. The additional space needed can be determined by dividing the EMA program size by the number of words per track (that is, 6144 words for 7900/05/06(H)/20(H) type disks, 8192 words for 7925(H) type disks, 3840 words for 9895 type disks or the number of blocks/logical track multiplied by the number of words per block for all CS/80 type disks.

The scratch track pool begins at the next available track following the system area. The upper boundary of this area is determined the first time a generated system is booted up. This boundary is set up with the FMGR LU 2 cartridge initialize command (refer to FMGR Initialization section of this chapter).

The LU 2 FMP area comprises the third area on the system subchannel. This area is used for type 6 files, transfer files, and other files to be made accessible to all system users.

An auxiliary system subchannel (LU 3) can be used to extend your system file space (for example, for additional type 6 files) and/or provide additional scratch tracks for swapping, etc. A sample LU 3 configuration is shown in Figure 6-1b. The boundary between the scratch track area and FMP area on LU 3 is determined by the FMGR LU 3 cartridge initialize command.

When initializing LU 2 and LU 3 at system startup, you have to make a trade-off between the number of tracks allocated for the scratch track pool and the number allocated for the FMP areas on these cartridges.

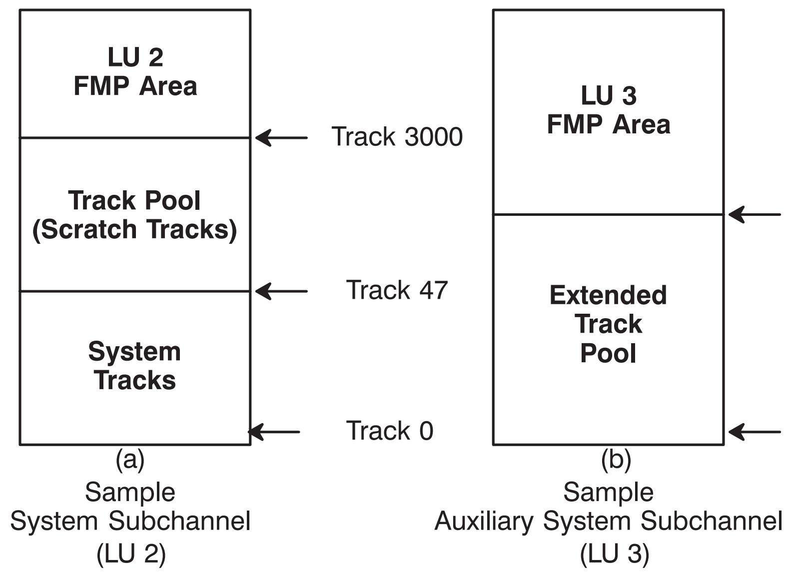


Figure 6-1. Sample System and Auxiliary Subchannels

FMGR Initialization

In order to use FMGR, an FMP area must be initialized on LU 2. The first time the system is started up after generation, an FMP area exists on LU 2 only if a request to save files was made during system switchover (refer to Chapter 5). If not, FMGR asks you to initialize the FMP area on LU 2. It displays the message FMGR 002 and then issues the standard prompt:

```
SET TIME
FMGR 002
  :
```

You should enter an IN command to initialize the FMP area on LU 2. This command specifies the number of directory tracks on the cartridge, the FMGR master security code, an ASCII identifier for the FMP area, and any bad tracks on that cartridge. Refer to the Terminal User's Reference Manual for IN command syntax.

Caution If you assign a master security code here, remember it because it cannot be recovered. The master security code is always reset to the null character (control@control@) during SWTCH.

The FMP starting track must be at least 8 tracks greater than the last track used by the system. The system size in tracks is reported in the generation dialog. The system scratch track area begins on the next track. It extends up to the first track specified in your IN command. For example, if you have the following: LU 2 has 500 tracks, you wish to allocate 50 scratch tracks, and your system needs 47 tracks; then the first track of your LU 2 FMP area is the 98th track (47+1+50).

If an auxiliary system disk subchannel (LU 3) has also been configured into your system, a portion (or all) of the tracks on this subchannel may be used for system scratch tracks. The remainder is used for the LU 3 FMP area. The number of tracks used for the FMP area is determined by the way LU 3 is initialized. The first time the system is booted up after generation, FMGR prompts:

```
FMGR 003
  :
```

You should now enter an IN command to initialize this cartridge. The system scratch track area precedes the FMP area on this LU. Therefore, the number of scratch tracks allocated on LU 3 is determined by the start of the FMP area specified in the IN command.

Before FMGR is initialized at system start up, it obtains all available tracks on the system and auxiliary system disk. After successful initialization with the IN commands, or upon subsequent system restarts, only those tracks residing in the FMP area of these LUs are assigned to the file management system. The cartridge directory tracks on LU 2 (and LU 3 if a cartridge exists on it) are assigned to the file directory manager, D.RTR. After the FMP tracks have been reserved, FMGR transfers control to the WELCOM file. If the LU 2 FMP area has just been initialized, this does not exist; so error FMGR −006 is generated and control is passed to the system console.

Example: To initialize the system subchannel with a FMP area starting at track 100 and the auxiliary system subchannel.

`SET TIME`

`FMGR 002`	(Request system subchannel initialization).
`:IN,XX,-2,2,SYSTEM,100`	(Start FMP area at track 100. Set master SC to XX).
`FMGR 003`	(Request auxiliary system subchannel initialization).
`:IN,XX,-3,3,AUX,50`	(Start FMP area at track 50 on the auxiliary system subchannel).
`FMGR-006`	(FMGR fails to find WELCOM and transfers to system console.)

`:`

If there is no auxiliary system subchannel configured into your system, FMGR does not request LU 3 initialization.

If you respond with a command other than IN to the prompts FMGR 002 or FMGR 00, the error message FMGR 004 is issued. If you correctly enter IN but request a starting track that is not available, FMGR 005 is issued. The first available track and sector can be obtained at this point by entering `??`.

System Tests

Simple and easy to use test procedures are provided below. They cannot test for all possible generation errors, but they do exercise the software and equipment sufficiently well to test for major hardware failures or generation errors. You may wish to supplement them with tests of your own.

Test your system using operator commands. For example, enter TI several times; if the time-of-day message that is printed does not change, the Time Base Generator is installed in the wrong slot, or is not working. Run the File Manager, list the directory and list some of the source files. Create some simple source files using EDIT, preferably a simple FORTRAN program, which you should compile, load and run. Dump source files to any output device, and re-submit the output tape, listing it. It should be identical to the disk file. If you notice anything peculiar, note the specific symptoms, and continue testing until you are satisfied that it has been well tested. If you noted any errors, consult the RTE generation manual examples, your own and the factory-generation listing. Pay particular attention to those questions you answered differently from those shown in the factory examples. When you've identified the problems, restore your previous system, boot-up that system, purge all copies of the previous generation boot, output, and list files, pack the disk, edit the answer file, and re-run the RTE generator.

Note that certain errors pertaining to I/O table definitions and memory partition definitions can be corrected by running the RTE-6/VM Reconfigurator (described in Chapter 10).

If generation errors can be corrected with operator commands (for example, LU reassignments with system LU commands), these commands may be inserted in the WELCOM file for execution at system startup.

File System Conventions

There should be a standard system of file conventions in order to simplify the installation and maintenance of your file system. You may follow the Hewlett-Packard conventions given below or you may design your own.

- The primary and auxiliary system FMP cartridges (LUs 2 and 3) are initialized with Cartridge Reference Numbers (CRNs) of 2 and 3, respectively.
- Files for general user access (such as utility type 6 files, standard transfer files, system documentation files, etc.) are protected from inadvertent destruction by a general security code such as RT.
- In an FMGR environment if users agree to use a convention for naming files, it is much easier to examine (and remember) the contents of a disk. It is a standard FMGR convention to attach significance to the first character of a file name as follows:

&	Program source code file
%	Binary relocatable code file
!	Binary absolute object code file
(	Program setup procedure file
)	Program cleanup procedure file
#	Loader command file
”	Documentation, information, or list file
*	General purpose file manager command transfer file
$	Relocatable library file

Type 6 files begin with alphabetic characters.

- In the CI hierarchical file system environment the file name is supplemented by a secondary name called a file type extension. The file type extension is used to indicate the type of information in the file: text, binary data, etc. It consists of a period and up to four characters appended to the file name. For example, in the filename parameter EDIT.RUN, the file type extension is .RUN. Blank type extensions are allowed; the period can be omitted if the type extension is blank. When specifying a file, the file type extension must be included if one exists.

  Standard type extensions should be used when files contain standard information. For example, all executable program files should have type extension .RUN, and all CI command files should have file type extension .CMD. The standard file type extensions are listed below.

.abs	absolute file
.cmd	CI command file
.dat	data file
.dbg	debug file
.dir	directory or subdirectory entry
.doc	documentation file
.err	error message file
.help	help file
.hlp	help file
.ftn	FORTRAN source file
.ftni	FORTRAN include file
.lib	library or merged relocatable
.lod	LINK/LOADR command file
.lst	list file
.mac	Macro source file
.maci	Macro include file
.map	LINK/LOADR load map listing
.merg	merge command file
.mnf	manual numbering file
.mrg	merge command file
.pas	Pascal source file
.pasi	Pascal include file
.rel	relocatable (binary) file
.run	executable program file
.snf	software numbering file
.snp	system snapshot file
.stk	command stack file
.sys	system file
.txt	text file

- In systems generated without the Session Monitor, dummy fill files can be used to control the space on LUs 2 and 3. This space should be reserved for type 6 files and transfer files to be made available to all users. A fill file can be created on LUs 2 and 3 to take up unused space remaining at the end of the FMP cartridge. This prevents user files from being stored here since file space is unavailable. When files are to be added to FMP LUs 2 or 3, the fill files can be temporarily purged and then recreated once the files have been added. Fill files can be created on these cartridges with the commands:

  `:CR,FILL02:RT:2:3:-1` (create fill file on LU 2)
  `:CR,FILL03:RT:3:3:-1` (create fill file on LU 3)

  Note that, if the Session Monitor is used on your system, fill files should not be created because user access to LUs 2 and 3 is automatically restricted.

- The system generation listing is stored in file “SYSTM on a globally accessible cartridge. Generation parameters can then be easily accessed by users, such as systems programmers, when troubleshooting or modifying the system. For example, to copy the generation map from LU 5 onto LU 2, specify:

  : ST,5,“SYSTM:RT:2::−1

  Normally the list file is rather large, so it may be desirable to put it on a cartridge other than LU 2 or LU 3.

- If the Multi-Terminal Monitor (MTM) is used in your system, and you have enough disk space, it is a good idea to logically associate each terminal with a specific disk LU. You can put an identifying tag on each terminal to tell the terminal user which disk LU he should use. There is actually nothing to prevent users from using LUs not assigned to them in this fashion. However, if they agree to the rules, interference among users can be minimized.

Utility Installation and CI Initialization

Three transfer files are provided for the ease of installing utilities and supporting software for CI. The first file (*DOSNP) is used to create the SNAP file for LINK. The second file INCI.CMD is used to link the CI file system utilities and optionally initialize the CI file system. The third file LOAD6.CMD links the rest of the utilities supplied with the operating system. This file can be initiated from INCI.CMD if desired, allowing a simple, one-step utility installation procedure.

The instructions to use these files are in the files themselves and should be read thoroughly before starting.

If your system is part of the DS (Distributed Systems) network, you need to install the CI DS file transparency software. Refer to the DS Transparency Software Installation section of this manual for details.

The LOAD6.CMD file uses LINK primarily to load all the programs. For those few programs loaded by LOADR, or for programs you load yourself with LOADR, the following discussion describes procedures and recommendations for loading in this manner.

After the system is operational, programs can be added in one of two ways: they may be permanently added with the On-Line Loader (i.e., OP,PE loader directive), or type 6 files can be created for them. The latter method is preferable because better use is made of available disk space and ID segments are not permanently allocated to the utilities (refer to the System Utilities and Libraries section in Chapter 4).

Follow the steps below for each utility that you want to add to the system after generation:

1. Load the utility temporarily using the On-Line Loader. The loader can be run to take its inputs either from the terminal or from a command file. Refer to the Relocating Loader Reference Manual for a description of the loader operating parameters.

2. Type 6 files should be created for each utility and its segments. The utility (main and segments) should then be aborted (i.e., :OF,name) to deallocate the ID segments occupied by the temporary load. To assist you in this process, you can create a general purpose transfer file (usually called *SP) to make a type 6 file of a program and then remove the program from the system:

```
(Session *SP) (MTM*SP)
:SP,1G:RT              :PU,FILL02:RT:2
:OF,1G                 :SP,1G:RT:2
::                     :OF,1G
                       :CR,FILL02:RT:2:3:-1
                       ::
```

3. If the utility has segments, create file manager command transfer files to allocate and deallocate short ID segments for the utility segments. Usually, transfer files used to set up short ID segments for a utility are named "(prog" where "prog" is the utility's name. Similarly, transfer files which perform clean up functions (that is, release ID segments) are called ")prog". Utility T5IDM may be used for this purpose. T5IDM is described at the end of this section.

It is recommended that all type 6 files and associated transfer files be given security codes. Normally there is no need to set up ID segments for utility main type 6 files since this is automatically done when the utility is run with the file manager :RU command. Normally, this command creates a copy of the utility for that user. This allows several users to operate the utility at the same time. Note that the same set of utility segments (identified in the system with short ID segments) can be shared by different copies of the utility main. They should not be copied.

WELCOM File

Every time the system is booted up, the system schedules FMGR to execute commands from the file WELCOM. This file must be on LU 2 and should contain the file manager and operating system commands necessary to initialize your system. A sample WELCOM file listing is shown in Figure 6-2. This WELCOM file performs the following functions which should be included, if applicable, in your WELCOM file:

- COMMAND ECHO (line 1). A severity level of one can be specified to inhibit the echoing of WELCOM file commands to the system console. The IH parameter in this case inhibits echoing of the SV command itself.

- OPERATOR MESSAGE (lines 2-5). It is suggested that you print a start up message on the system console indicating that the initialization process has begun. The file manager :TE command provides a convenient way of doing this on a line by line basis.

- ALTER SYSTEM PARAMETERS (lines 6-9). The WELCOM file can alter system parameters in order to correct generation errors and/or change system default values. The commands are of the form:

  ```
  :SY<operating system command>.
  ```

  Types of commands you may wish to place here are:

  1. Commands which alter I/O table definitions. The LU command can be used to redefine logical unit/equipment table (EQT)/subchannel assignments. The EQ command can be used to enable/disable automatic output buffering on selected devices. The BL command can alter the automatic buffering limits. The TO command can be used to change device timeout values.
  2. Commands which control program execution. The PR command can change program priority levels. The IT command places programs on the time list and/or change execution intervals. The ON and RU commands schedule programs for execution. The QU command alters the system time slicing parameters.

3. Commands which alter partition definitions. The AS command assigns programs to execute in specific partitions. The UR command releases a previously reserved partition.

   The *Terminal User's Reference Manual* describes the operating system commands in detail.

- PACK SYSTEM CARTRIDGES (line 10). Reference is made to a user created transfer file called *PACK. Sample listings of *PACK for both session and non-session users are shown below:

```
(MTM/non-session version)     (Session version)
:PU,FILL02:R7:-2              :PK,-2
:PK,-2                        :PK,-3
:CR,FILL02:RT:-2:3:-1 ::
:PU,FILL03:RT:-3
:PK,-3
:CR,FILL03:RT:-3:3:-1
::
```

```
0001      :SV,1,IH
0002      :TE,***************************************
0003      :TE,*  RTE-6/VM 92084 SYSTEM REV XXXX    *
0004      :TE,*          GENERATED     XXXXXX      *
0005      :TE,***************************************
0006      :SYEQ,1UN
0007      :SYLU,50,15,1                          *  PACK must reside on a cartridge
0008      :SYBL,100,400                             not being packed.
0009      :SYQU,90,2000
0010      ::*PACK
0011      ::*STIME
0012      :RP,FTN7X
0013      :RP,MACRO
0014      :RP,EDIT
0015      :RU,CI,/SYSTEM/CI_WELCOME.CMD
0016      ::*COPY,FMGR,FMG07  *                  COPY, FMGR is necessary only in an
0017      ::*COPY,FMGR,FMG09                     MTM/non-session environment.
0018      ::*COPY,FMGR,FMG15
0019      ::*COPY,FMGR,FMG22
0020      ::*COPY,FMGR,FMG23
0021      ::*COPY,FMGR,FMG24
0022      ::*COPY,FMGR,FMG25
0023      :RU,DINIT,*DINIT
0024      :CT,7,20B,RTE IS UP....TERMINAL #7....STRIKE ANY KEY
0025      :CT,9,20B,RTE IS UP....TERMINAL #9....STRIKE ANY KEY
0026      :CT,15,20B,RTE IS UP....TERMINAL #15....STRIKE ANY KEY
0027      :CT,21,,,RTE is up ... Hit any key
0028      :CT,22,,,RTE is up ... Hit any key
0029      :CT,23,,,RTE is up ... Hit any key
0030      :CT,24,,,RTE is up ... Hit any key
0031      :CT,25,,,RTE is up ... Hit any key
0032      :CT,26,,,RTE is up ... Hit any key
0033      :CT,27,,,RTE is up ... Hit any key
0034      :CT,28,,,RTE is up ... Hit any key
0035      :TE,
0036      :TE,      <<<<INITIALIZATION COMPLETE>>>>
0037      :EX
```

Figure 6-2. Sample WELCOM File

In the above listings, references to LU 3 should be omitted if an auxiliary system cartridge is not configured into your system.

Once ID segments are allocated to programs on LU 2 or 3 identified as type 6 files (via file manager :RP or :RU commands), these cartridges cannot be packed. Packing recovers disk space returned to the system when files on these cartridges are purged. It is strongly suggested that these cartridges be packed using the WELCOM file at system start up. It may not be possible to do so (and reclaim unused disk space) at a later time.

Caution If you include a pack (PK) command in the WELCOM file and pack the cartridge containing the WELCOM file, FMGR aborts if the PK command moves WELCOM file. In the sample WELCOM file (see Figure 6-2) *PACK must reside on a cartridge not being packed.

- SET SYSTEM TIME (line 11). It is recommended that the system time be set correctly during initialization. Certain HP subsystems (and user application programs may) make use of the system time for scheduling and accounting purposes. Therefore, the time should be set correctly before these subsystems are initialized. In the sample WELCOM file, reference is made to a transfer file *STIME which queries the operator for the time. When run, *STIME queries the operator with:

  ```
  ENTER DATE/TIME AS FOLLOWS::,MONTH,DAY,HOUR,MIN,SEC[,PM][,YEAR]
  :PA,,WHERE PM IS ENTERED AFTER NOON.
  ```

- SETUP ID SEGMENTS (lines 12-14). At system start up, ID segments can be set up for programs that are used frequently, making access to these programs more efficient. Programs in this category might be compilers, assemblers, utilities, etc. Multiple copies of the program can share the same short ID segments. When the main program is run, for example, FTN7X, file manager makes a copy of it (FTN07, FTN09, etc.) for that particular user.

  Note that programs loaded by LINK do not use short ID segments. Only programs loaded by LOADR and MLLDR use short ID segments.

- EXECUTE CI WELCOME FILE (line 15). some functions can only be performed from CI. This might include RP'ing files from the /PROGRAMS directory or making a copy of CM from CI (RP CI CM).

  CM is a copy of CI which is RPed with the name CM. If PRMPT (the program scheduled when an unsolicited terminal interrupt is detected) finds CM RPed on the system, it allocates a class number to communicate with CM and schedules CM passing it this class number. If PRMPT fields an unsolicited interrupt on a terminal that is logged on and CM is waiting on its class, PRMPT will put out the 'CM>' prompt and post a read on CMs class. In its CM mode, CI will handle most CI requests, one exception being TR or implied TR commands. In addition, RU commands are changed to XQ. The rule is that CM does not want to wait but instead wants to be available to the nexy user. To do this, it must get back to the class wait state (or PRMPT will not do its CM> prompt, but instead will use the 's=xx command ?' prompt). To enable 'CM>' prompting you must RP CM in the welcome file as follows:

  ```
  CI> RP,CI,CM
  ```

  This command must be done from CI as FMGR uses a different meaning for the second RP parameter.

  If you generate CI into your system you will need to create a file from which to RP CM. The easy way to do this is:

  `:SP,CI:RP:2` from FMGR

You may then RP it from this file as:

```
RP,CI::2,CM                from CI
```

or you may remame the file to CM and then RP it directly from FMGR. The SP,CI need be done only once right after the system is switch and is first booted.

- FILE MANAGER COPIES (lines 6-22). Copies of the File Manager need only be made for systems operating with the Multi-Terminal Monitor (MTM). If you are not using this system, SKIP this step.

  To make copies of the File Manager (FMGR), you first need to make a type 6 file of FMGR:

  ```
  :SP,FMGR:RT:2
  ```

  This must be done before the WELCOM file is executed. The transfer file *COPY referenced in the sample WELCOM file can be used to make copies of FMGR for each terminal:

  ```
  :RN,1G:RT,2G
  :RP,2G
  :RN,2G:RT,1G
  ::
  ```

  For example, executing ::*COPY,FMGR,FMG07 causes the following commands to be executed.

  ```
  :RN,FMGR:RT,FMG07       (rename FMGR)
  :RP,FMG07               (RP in copy)
  :RN,FMG07:RT,FMGR       (rename it back)
  ```

  When an MTM terminal key is struck in break mode and the terminal's copy of FMGR is dormant, that copy of FMGR is scheduled by MTM (that is, PRMPT). If a copy has not been created for the terminal, or if it is not dormant, the terminal user gets the standard MTM break mode prompt: LU>.

- SUBSYSTEM INITIALIZATION (line 23). The WELCOM file can be used to initialize subsystems during system start up. In this example, DS/1000 is initialized from an answer file. Note that certain subsystems, such as DS/1000, should be initialized at start up automatically. At bootup, you must generate ACCTS into the system or load it permanently (via LOADR PE or RP commands). If ACCTS is not permanent, it does not execute and a SESSION NOT INITIALIZED error message is displayed when you try to enable a terminal for Session. If you on-line load (but not permanently) and save ACCTS as a type 6 file, you should schedule it in the WELCOM file by RPing it and its segments and :RU,ACCTS,−1 or :RU,ACCTS,namr. This is recommended early in the WELCOM file before enabling terminals, to ensure that ACCTS is allocated enough SAM.

- ENABLE TERMINALS (lines 24-31). Terminals must be enabled before they respond to break mode interrupts (and log-on interrupts for Session Monitor). The file manager :CT command provides a convenient means of enabling terminals and sending out terminal initialization messages. In the sample WELCOM file, three point-to-point terminals (e.g., 2645's using 12966 I/F cards) are enabled (lines 24-26). Eight LUs on the multiplexer are enabled also (lines 27−34).

  Terminal Initialization commands are generally of the following form:

  ```
  :CT,LU,20B,sub,string
  ```

where:

LU	is the terminals keyboard/display logical unit. In session systems, LU must be in the range 7-99. In MTM systems, LU must be in the range 7-63.
sub	is the initialization subfunction code. It may be omitted for all terminals operating with DVR00 or DVR05. For other terminals (for example, multipoint, multiplexer, DVA05 modem links, etc.) consult the appropriate driver manuals, subsystem manuals, and configuration guides.
string	is an ASCII message to be sent to the terminal upon initialization. This message might indicate the terminals LU and/or give instructions.

- INITIALIZATION COMPLETED MESSAGE (lines 32 through 33). It is suggested that the operator be informed when a successful initialization has been completed.
- WELCOM FILE TERMINATION (line 34). The WELCOM file should be terminated with an EX command (which terminates FMGR) instead of a TR (or ::) which merely transfers back to the system console.

Note Whenever FMGR encounters an error when processing WELCOM file commands, transfer is made to the system console. Additional commands may then be entered at this time. Transfer can be made back to the WELCOM file by entering TR.

HI File (MTM Only)

If the Multi-Terminal Monitor (MTM) is included in your system, you can optionally specify a set of commands to be executed whenever a terminal's copy of File Manager is started up by MTM. The commands must reside on transfer file HI.

Typically, HI file commands dump out softkey definitions to terminals (these can be created by the KEYS utilities), display system welcome messages, and send system status messages to the terminal. You may also wish to use them to set up File Manager global parameters and the File Manager severity level. Note that the HI file should be general purpose in nature as it is executed by all MTM File Manager users.

Example:

```
:SV,1,,IH              (do not echo commands)
:DU,"SOFTK,0G          (dump soft key files to terminal)
:DP,***
:DP,***                (WELCOME TO RTE-6/VM)
:DP,***                REV 2001 2/11/80
:DP,***
:DU,"SYSMS,0G          (dump system message file to terminal)
:SV,0                  (set severity level)
::                     (transfer to terminal)
```

If a HI file cannot be found, File Manager takes its initial input from the terminal.

Help Utility

The HELP utility provides assistance to system users in a variety of ways. It can provide detailed explanations of errors, provide information on system or subsystem related features, and serve as an index to documentation.

All HELP messages are obtained from the file descriptor specified when the utility is run. If no file is specified, the file HELP.HLP::SYSTEM is used. If the user wishes a set of messages other than those contained in HELP.HLP::SYSTEM to be displayed, a new file may be created through the use of the GENIX utility and then included as a parameter in the runstring.

The key to be searched for is also included in the runstring. This key may be from 1 to 24 characters in length. All leading and trailing blanks are deleted, and all lowercase characters are mapped to uppercase.

The file to be searched by HELP must be a type 1 sequential file. It must be disk-resident and created by the utility GENIX. This file should reside on a system (global) disk so that it may be accessed by any session or non-session user.

Help schedules the program CMD to print the text associated with the key given in the runstring or SCB. CMD then searches the file specified in the runstring for the required key. The keys are stored at the start of the type 1 file in a balanced binary tree. The keys are forward and reverse link-listed to allow fast printing. Currently, the number of keys is limited by the heap size of the GENIX utility. A 28K partition provides sufficient space for GENIX to generate a keyed indexed sequential file with 1050 keys.

HELP Operation in the Session Environment

The Session Monitor provides special error handling for users under session control. The SCB associated with each session contains space for an error code. Whenever a subsystem detects an error, it calls library subroutine PTERR to put the error in the user's SCB. As a result, the SCB contains the error code for the most recent error that occurred in a session.

When Help is invoked by a session user and a particular keyword is not specified, Help uses the error code in the SCB as the keyword. Help does not clear the error code in the SCB. If desired, you may replace the HP supplied HELP Utility Program with your own. The same scheme may be used by the system manager if he chooses to have the SCB updated with errors occurring in his system's application software.

More information about the HELP and GENIX utilities are contained in the Utilities Reference Manual.

CMD Utility

The CMD utility is used to print expanded error messages and command syntax messages to the user's terminal. When called programmatically, it provides a means of generating help functions for any interactive program on the RTE operating systems. CMD also supports an interactive mode that allows a user to examine all the keys that are in a file and specify when additional information about a key should be printed.

All help messages are obtained from a file descriptor specified in the runstring. If CMD cannot find the file specified in the runstring in the current working directory, CMD searches ::SYSTEM directory for the file. If a file is not specified in the runstring, the default file CMD.HLP::SYSTEM is used. If CMD.HLP::SYSTEM cannot be found, then !CMD::0 is used. If CMD is scheduled by the HELP program, the file HELP.HLP::SYSTEM is used. If HELP.HLP::SYSTEM cannot be found, then !HELP::0 is used.

If you want to display a set of messages other than those contained in the default file, you can create a new file using the GENIX utility and then use this file as a parameter in the runstring.

The key to be searched for is also included in the runstring. This key may be from 1 to 24 characters in length. All leading and trailing blanks are deleted, and all lowercase characters are mapped to uppercase.

The file to be searched by CMD must be a type 1 sequential file. It must be disk-resident and created by the utility GENIX. This file should reside on a system (global) disk so that it may be accessed by any session or non-session user.

The keys are stored at the start of the type 1 file in a balanced binary tree. The keys are forward and reverse link-listed to allow fast printing. Currently, the number of keys is limited by the heap size of the GENIX utility. A 28K partition provides sufficient space for GENIX to generate a keyed indexed sequential file with 1050 keys.

More information about the CMD and GENIX utilities are contained in the Utilities Reference Manual.

Spool System Initialization

The spooling system is initialized by running GASP. Refer to the Batch and Spooling Reference Manual for a complete description of the spooling system and initialization procedures.

System Protection

It is strongly recommended that once your new system has been installed and initialized, it should be backed up onto magnetic tape or another disk. This allows you to recover your system in the event of a disk hardware failure or a system crash.

When your system is installed with the SWTCH utility, the disk track preambles for the system area on LU 2 are set to indicate protected tracks. The system area can be protected from being written over by switching your format switch to OFF mode (HP 7905/06/20/25) or your override switch to protect mode (HP 7900). Note that the system scratch track and FMP areas on LU 2 are not affected by these switches (that is, they are on unprotected tracks). If your system tracks are protected, the Loader cannot perform permanent load, replace or purge operations. Also, you cannot permanently change I/O or memory definitions with the Reconfigurator.

Knowledge of the master security code gives users complete access to all File Manager files. It is strongly suggested that you do not publicize it.

Changing Auxiliary Cartridges

If your auxiliary cartridge (LU 3) is on a removable disk subchannel, physically separated from the primary system subchannel (LU 2), certain procedures have to be followed when changing auxiliary cartridges.

Where possible, auxiliary cartridges should be changed when the system is down (that is, Halted). When the system is restarted (bootstrapped), the system checks to see if an FMP area has already been initialized on LU 3. The FMP file area tracks are assigned to the FMP and the directory tracks are assigned to D.RTR. If the cartridge has not been initialized, a FMGR 003 is issued and the user must initialize the cartridge with an IN command (refer to the FMGR Initialization section in this chapter).

If your auxiliary cartridge must be changed when the system is active, the following steps should be performed:

1. Run the LGTAT utility to determine if there are tracks on LU 3 used by programs other than the file management package (that is, LU 3 tracks should either be unassigned, assigned to FMP, or assigned to D.RTR). Program swap tracks and tracks assigned to programs for temporary storage can be unassigned by entering :OF,prog. LU 3 tracks containing programs permanently added to the system, can be unassigned by purging them with LOADR. If any LU 3 tracks are still assigned to programs when the auxiliary cartridge is changed, unpredictable (potentially disastrous) results occur.

2. The FMP area on the new auxiliary cartridge must be initialized to the same first track as the old cartridge (preferably track 0 since this prevents the loader or system from placing a program in this area.)

   To change auxiliary cartridges, use the DC command as follows:

   `:DC,-3` ⟵ this ensures that all files are closed

   Remove the cartridge from the drive and insert the replacement.

   `:DC,-3` ⟵ places new cartridge in disk directory

   Note that MC is not used to mount the cartridge. This is because DC remounts the cartridge as part of its procedure when the logical unit is 2 or 3.

3. If the new auxiliary cartridge has not been initialized, FMGR locks it and a subsequent attempt to initialize the cartridge results in FMGR 059 error message. The error occurs because the directory tracks are already assigned to D.RTR. You must, therefore, release the D.RTR tracks and then re-assign them by scheduling FMGR. After assigning the D.RTR tracks, FMGR terminates and you must schedule it again in order to enter FMGR commands. This special case, where FMGR terminates immediately, occurs only when the D.RTR tracks are unassigned.

   Example:

`:EX`		
`$END FMGR`		
`*RT,D.RTR`	⟵	release the D.RTR tracks
`*RU,FMGR`	⟵	scheduling FMGR assigns D.RTR tracks on LU 2; FMGR terminates
`*RU,FMGR`	⟵	re-schedule FMGR
`:IN,SC,-3,AUX`	⟵	initialize new auxiliary cartridge on LU 3

   Be sure that the new auxiliary FMP area cartridge has been initialized to the same starting track as the previous one.

Short ID Handler

T5IDM is only used by segmented programs that have been loaded by LOADR. LINK loaded programs keep the segment information in the main and do not use short ID segments.

The short ID handler T5IDM and its interface routine SEGLD can be used to dynamically install and release short ID segments for segmented programs (overlays). The advantage of SEGLD is that many segments can be called with only one free short ID segment in the system.

```
CALLING SEQUENCE: CALL SEGLD (INAM,IERR[,IP1 THRU IP5])
```

where:

`INAM`	is the segment name.
`IERR`	is the error return.
IP1 thru IP5	are optional parameters passed to the segment in INAM.
	Refer to the RTE-6/VM Programmer’s Reference Manual for details.
Error return:	If SEGLD returns, an error occurs. In this case either the name passed in INAM is not a program segment or the segment cannot be found.

To be accessed by T5IDM, a main program and all of its segments must be saved (SP’d) as type 6 files. The main and all of its segments must be on the same LU.

The names of SP’d program segments must not be changed (by RN, for example) because the relationship between the main program and its segments would be lost.

T5IDM produces only short ID segments. If a short ID segment is not available, T5IDM does not use a long ID segment.

When T5IDM builds an ID segment it copies the necessary information from the type 6 file into internal tables and then into the short ID segment. When the program segment completes execution, the short ID segment is released so that the system can reuse it. If the same segment is called again and T5IDM still has the segment information in its internal tables, it builds the short ID segment without referring to the type 6 file.

Included in T5IDM's internal tables are the starting locations of the program segments' type 6 files. If these addresses are incorrect, the program may abort with a DM or MP error.

How could the address be incorrect? This example illustrates one possibility. Suppose that a segmented program is loaded, SP'd, OF'd, and then run. SEGLD schedules T5IDM to build short ID segments. T5IDM has no information on the program segments, so it looks at the type 6 files. Now suppose that the program is OF'd and the type 6 files are purged. The program is reloaded, SP'd, OF'd and run again. It is possible that T5IDM's internal tables contain program segment information from the program's previous run. If so, T5IDM uses it, ignoring the type 6 files. Since the type 6 files were purged and resaved, they may not be in the same locations. If not, the program segment starting locations in T5IDM's Tables are wrong, and the program may abort with a DM or MP error.

If this happens, run T5IDM with a parameter of −1 (RU,T5IDM:IH,−1). This flushes the tables and forces T5IDM to get program segment information from the type 6 files. Currently executing segmented programs are not affected.

DS Transparency Software Installation

The DS Transparency software allows access to files on other RTE-6/VM or RTE-A systems connected via the DS/1000-IV network. Installation of the DS network is described in the DS manuals.

Two monitors, DSRTR and TRFAS, are used by the DS transparency software. DSRTR is called when the system is the host system originating the access request, and TRFAS is called when the system is the remote system receiving the access request. Both monitors are required at all systems using the DS transparency software for remote file access.

Install the DS transparency as follows:

1. Load DSRTR and TRFAS for your system, using the supplied load files and relocatables. Both can be generated into the system, or they can be loaded online. It is recommended that DSRTR and TRFAS (if not generated into the system) be loaded permanently with LOADR; the files #DSRTR and #TRFAS can be used with LOADR. The DS/1000-IV software also must be generated into your system.

2. DSRTR must be sized up in the generation if it is generated into the system. This may be done in the Modify program page requirements? section. Twenty-three (23) pages is recommended.

3. Make sure that TRFAS is scheduled by DINIT. This can be done by specifying the monitor scheduling as /D, using the default set of monitors. If you list the monitors explicitly, you must include TRFAS. Do not schedule DSRTR from DINIT.

4. Initialize DS by running DINIT.

You can test the DS file access setup by entering a CI command such as:

```
LI welcom::2>27
```

where the local system is node 27. This should list the file, although the following message is also displayed.

```
DSRTR:  No such file NODENAMES::SYSTEM
```

The NODENAMES file is used to associate node names with node numbers. DSRTR looks for this file the first time DSRTR is scheduled. If DSRTR finds such a file, DSRTR reads the file to build a table of names for node numbers.

The NODENAMES file should contain entries of the form:

```
 * <comment>
```
or
```
node# nodename [comment]
```

As an example:

```
*Test System 1
1  SYS1
*Test System 2
2  SYS2
*Central Systems
3  Central1  Central System #1
4  Central2  Central System #2
```

Numbers must be separated from names by one or more spaces. Comments are optional and names (up to 16 characters are allowed) must conform to the file naming conventions.

When DSRTR is first scheduled after boot-up, the NODENAMES file is read into memory. If the file is modified, DSRTR needs to re-read the file. Scheduling DSRTR with the “don’t clone” option (:IH) and no parameters causes DSRTR to re-read the NODENAMES file:

```
RU,DSRTR:IH
```

When a DS transparency request comes in from another node, and the request does not include an explicitly specified account name, the request is handled under the default logon account defined when DS was initialized. (If no default account was defined, the request is handled non-session.) The first time such a request comes in, a session is established under this default name, This session remains active for any subsequent requests. (If the session is shut down for any reason, it is set up again when needed.)

DS transparency software uses the same amount of SAM as other DS functions. You have problems if you only have 1K words of SAM, so you should allocate about 4K words of SAM. Remote file access through the DS transparency software can be accomplished through nodes that do not have TRFAS or DSRTR as long as the monitors are set up at the source and destination systems.

Remote file access performance is probably a factor of four slower than local access, depending on link speed and network topology.

7

Session Monitor Initialization

This chapter describes three aspects of accounts system operation: system initialization, setting up new group accounts, and setting up new user accounts. For other aspects of account system operation, such as altering accounts or backing up the accounts system, refer to Chapter 8. It is suggested that you complete the accounts planning matrix and cartridge requirements worksheets (refer to Chapter 3) before following the procedures outlined in this chapter. You should also have your generation listing and user application notes (for example, questionnaires) handy as they will also prove useful during this process.

Session LU Definition

Session LU assignments may be predefined by the System Manager in one of two ways:

- Account SST Definition. When group and user accounts are defined, Session LU to System LU mappings may be included in the definitions. When the user logs on to the system, those mappings will be included in the Session Switch Table (SST) of his Session Control Block (SCB). In this chapter, the term Account SST refers to the Session LU to System LU mapping contained in the individual group and user account file definitions.

- Configuration Table Definition. The Configuration Table allows you to define Session LU to System LU mappings for various terminals in the system. When a user logs on to a terminal with entries in the Configuration Table, those LU mappings associated with his terminal are automatically included in his SST. Typically, entries in the Configuration Table are made for auxiliary printers, cartridge tape units (CTUs) and other devices (for example, instrumentation) associated with terminals. The Configuration Table is contained in the account file.

Note that a session user can only access devices defined in his SST. When defining user accounts and the Configuration Table, you should insure that each user has access to the resources required by his application.

An overall scheme for assigning session LU numbers in your system should be developed before defining group accounts, user Account, and Configuration Table SST entries. This prevents conflicting definitions where the same session LU is mapped to different system LUs. The following paragraphs discuss where Account SST entries are defined and considerations to be taken when planning the session LU assignment scheme for your system.

Session LU Allocation Worksheet

This worksheet provides a framework for the allocation of session LU numbers in the system. It should be referred to when setting up your Account SST and Configuration Table definitions. To adequately fill out this worksheet, you should rely upon the following items: the account planning matrix and cartridge requirements worksheet (filled out in Chapter 3), the system generation listing, and your general knowledge of the user community (for example, user questionnaires, applications knowledge, etc.).

A sample Session LU Allocation Worksheet is shown in Figure 7-1. Use this example in conjunction with the instructions given below to fill out your own worksheet.

1. Standard LU Allocations. In the session environment LU 1 is always the keyboard display LU. LU 2 and LU 3 are the primary and auxiliary system cartridges. LUs 4 and 5 are the users standard input and output devices. Typically, they are assigned to terminal CTUs or paper tape reader/punches. LU 6 should be reserved for the standard list device. It is suggested that auxiliary list devices (for example, terminal auxiliary printers) be assigned to some other LU (LU 7 in the example). This prevents conflicts when users require access to both printers. LU 8 should be reserved for the system magnetic tape unit.
2. Disk Cartridge LU Allocations. In the session environment all disk cartridge session LU numbers must be identical to the cartridge system LU numbers. Use the worksheet to indicate what session LU numbers are to be used for disk pool cartridges and dedicated private, group, and system global cartridges. Disk Cartridge LUs dedicated for non-session use should not be included here. For your convenience later on, it is suggested that you indicate who owns which cartridges and, in the case of disk pool cartridges, cartridge sizes. Refer to the disk requirements worksheet and your system generation listing for this information.
3. Subsystems LU Allocation. Certain subsystems may require access to peripherals by their system LU numbers. For example, if DS/1000 is configured into your system, session communication LU numbers should be the same as the system communication LU numbers. It is therefore suggested that these session LU numbers be reserved for this purpose in the worksheet. This insures that when you are adding DS capability to an Account SST, it does not conflict with previous Account SST or Configuration Table definitions.
4. Station LU Allocations. A set of session LU numbers should be reserved for devices specifically associated with stations. This assures that user account and Configuration Table SST definitions does not conflict with each other regardless of the station a user logs on at. You might wish to reserve session LUs 4, 5, and 7 for station left CTUs, right CTUs, and auxiliary printers, respectively. These default LUs can be used to access similar types of devices from any system station. You need only reserve as many session LU numbers as is required to accommodate your largest station (that is, with the greatest number of associated peripheral devices).
5. Group and User Peripheral LU Allocation. A set of session LU numbers should be reserved for devices specifically associated with group and/or user accounts. It is suggested that one set of LUs be reserved for groups and one set for users. Note that although many groups (users) share the same session LU numbers in their account SST, they are not necessarily mapped to the same system LUs. The session LU allocation worksheet should be used as a guideline only. In some cases you may find a need for more session LUs of a certain type than what you have provided for in your worksheet. When such conditions arise you have to use session LU numbers allocated for other things (for example, cartridges that the user is unlikely to access).

LU#	Purpose	LU#	Purpose
1	Terminal Keyboard	34	Disk Cartridge, Pool, 101 tracks
2	Primary System Cartridge	35	Disk Cartridge, Pool, 102 tracks
3	Auxiliary System Cartridge	36	Disk Cartridge, Pool, 102 tracks
4	Standard Input	37	Disk Cartridge, User, Dunn
5	Standard Output	38	Disk Cartridge, Pool, 102 tracks
6	Standard List	39	Disk Cartridge, User Diskkey
7	Auxiliary List	40	Disk Cartridge, group, LC
8	System Mag Tape Unit	41	Disk Cartridge, Pool, 75 tracks
9	<available>	42	<available
10		43	<available>
11		44	
12		45	
13		46	
14	Stations	47	
15	Peripherals	48	
16		49	
17		50	
18		51	
19	Disk Cartridge, group, General	52	
20	Disk Cartridge, group, FP	53	
21	Disk Cartridge, group, HP	54	
22	Disk Cartridge, Pool, 203 tracks	55	
23		56	
24		57	
25		58	Disk Cartridge, Pool, 75 tracks
26	DS Links	59	Disk Cartridge, Pool, 203 tracks
27		60	Disk Cartridge, Pool, 203 tracks
28	Disk Cartridge, group, General	61	Disk Cartridge, Pool, 50 tracks
29	Disk Cartridge, Pool, 203 tracks	62	<available>
30	Hierarchical File Volume, 800 tracks	63	<available>
31	Hierarchical File Volume, 600 tracks		
32	Disk Cartridge, Pool, 102 tracks		
33	Disk Cartridge, Pool, 102 tracks		

Figure 7-1. Session LU Allocation Worksheet

Running ACCTS

Program ACCTS is responsible for the initialization, maintenance and overall control of your session monitor system. Inputs to ACCTS are made either interactively from a terminal or directly from an answer file. Unless, you are familiar with ACCTS operation, it is suggested that ACCTS be run interactively:

```
:RU,ACCTS,,namr
```

All ACCTS messages and operator inputs are recorded on the list file namr. For example:

```
:RU,ACCTS,,"LIST
```

directs all ACCTS messages and inputs to file "LIST. This file should be saved as it may prove useful when diagnosing initialization errors or when creating an answer file.

Caution If a list file is specified with previous list output in it, this previous information is lost.

Initialization Dialogue

When program ACCTS is run after the operating system is installed and initialized, it begins with the following message and prompt:

```
SESSION NOT INITIALIZED
ENTER IN,LO,/TR OR /HE
```

The /HE command can be entered at any time to get a list of valid commands or (if entered immediately after an error) to schedule HELP for a detailed error explanation. The /TR command can be used at any time to transfer to an answer file. The LO command can be used to rebuild the accounts system from a previously backed up account system file. (Refer to Chapter 8 for detailed descriptions of the LO, /TR, and /HE commands.)

Enter IN to start the initialization sequence.

ACCTS will first request the DISK LU on which the accounts file is to be located.

```
ENTER DISK LU FOR ACCTS FILE:
```

Caution For all session subroutines to operate properly, the subsystem disk cartridge must be mounted as a system disk.

Program ACCTS will prompt:

```
SESSION LIMIT?
```

Enter the maximum number of active sessions to be allowed in your system at any one time. This should be the number of session terminals in your system (including the system console if it may be operating in session mode). If batch jobs are submitted from sessions in your system, you should add one to this number. For example, for a system with five session terminals, a system console to be operated in session mode and batch jobs to be submitted from sessions, a session limit of 7 would be entered.

```
SESSION MEMORY ALLOCATION? (Y OR N)
```

The Session Monitor requires a block of system memory to contain active Session Control Blocks (SCBs) and the list of cartridges LUs in the spare cartridge pool. Enter Y if you want ACCTS to use the session memory allocation algorithm to calculate the amount of system memory for SCBs. The algorithm is as follows:

Session Limit <=20: (70 − (Session Limit))*(Session Limit)
Session Limit = 20: 50 * (Session Limit)

It is recommended that the session memory allocation algorithm be used unless you have a very large user account or Configuration Table, or have a large number of UDSPs, SST definitions, or limited system memory space.

Enter N to override this algorithm and manually set the memory allocation size. In this case, ACCTS asks:

```
NO. WORDS TO ALLOCATE?
```

Enter the decimal number of words to be allocated from system memory for session use. Refer to Appendix C for a description of internal SCB formats. Be sure you have sufficient SAM to contain active SCBs. Allocate SAM or reduce the SCBs in number/size, as required.

```
NUMBER OF USER ACCOUNTS?
```

Enter the maximum number of user accounts to be defined in your system. This quantity can be derived from the total number of check marks made in your account planning matrix plus an additional amount (for example, 5-10) for future users.

```
NUMBER OF GROUP ACCOUNTS?
```

Enter the maximum number of group accounts to be defined in your system. This quantity can be derived from the number of groups listed in your account planning matrix plus an additional amount (for example, 3) for future groups.

The number of accounts specified in the above two questions are used by ACCTS to determine the size of the account file. The account file is organized into records of 64 words. Each user and group account definition requires at least one record. This accommodates approximately 30 user Account SST entries (mappings) and 55 group Account SST entries. If one record is not large enough to accommodate an account entry, ACCTS allocates an additional record for that definition. When creating the accounts file during initialization, ACCTS allocates a record for each group and user account and several additional records. The number of additional records is approximately 20% of the number of user accounts. If more than 10% of your user and/or group account definitions are large (that is, requiring more than one record), you should increase the number of accounts specified.

```
SYSTEM MESSAGE FILE?
```

Enter a file name (filename:sc:crn) of the file to be output to each users session terminal at log on. Enter “ ” (blank) for no file. The system message file provides a convenient means for you to share informational messages on a system wide basis. Some of the items you might want to place in the system message file are:

- Scheduled preventive maintenance down time.
- New software or hardware additions to the system.
- Procedures to follow when using the system.
- Greetings.

Note that the message file does not have to exist when specified at this time.

The system message should be short and to the point. Otherwise, parts of it are apt to get overlooked by users at log on. A sample message file is shown in Figure 7-2.

```
PROMPT STRING?
```

Enter " " if you want users to be prompted with the default PLEASE LOG ON: when attempting to log on to the system. Otherwise, enter a string of up to 20 characters for the log on prompt. ACCTS always appends a backarrow (underscore) to the prompt to suppress a carriage return/line feed at the end of the prompt.

```
LOCATION OF MESSAGE FILES?
```

Enter the CRN (+ number) or −LU of the cartridge to which user message files (accessed with file manager SM or ME commands) are directed. This cartridge must be mounted as a global system disk in order for the message file mechanism to work properly for all users. If " " (space) is entered, message files are directed to LUs 2 and 3.

```
                    ****WELCOME****
                    RTE-6/VM REV xxx (date)
 1.   The system will be down for PM Saturday from 0800 to 1000

 2.   We have installed an additional lineprinter.
      For most users it is LU 7

 3.   Any problems contact Dave x2629
```

Figure 7-2. Sample Message File

```
STATION CONFIGURATION (Y OR N)?
```

Enter Y if you wish to define a Configuration Table for your system. The Configuration Table allows you to associate default session logical unit/system logical unit mappings for various stations in your system. When a user logs on from a station with entries in the Configuration Table, those mappings are automatically included in his SST. For example, say a terminal has an auxiliary printer assigned to system LU 90. You can make this printer the standard output device (LU 6) for every user logging on from that station by specifying a Configuration Table entry mapping session LU 6 to system LU 90.

The keyboard/display session LU is always LU 1 and therefore does not require a Configuration Table entry.

If N is entered in response to the station configuration question, no Configuration Table is defined at this time. In this case, the next question asks for a disk pool LU, see below.

If Y is entered, the next prompt will be:

```
STATION LU?
```

Enter the first (next) station terminal keyboard/display LU to have entries in the Configuration Table. Enter /E if no additional stations are to be included in the Configuration Table. Note that station LU numbers may not exceed 99. To redefine your entire station Configuration Table, enter /A.

```
SESSION LU, SYSTEM LU?
```

Enter the session LU/system LU mapping for this station. ACCTS continues to prompt for station LU mappings until a /E is entered. At that time, it asks for the next station (STATION LU?, see above). To redefine all Configuration Table entries for this station, enter /A.

Note that session LU numbers must be in the range of 4 to 63. System LU numbers must be in the range of 0 to 254.

If a session LU has been defined for a station more than once, ACCTS responds with:

```
DUPLICATE SESSION LU XXX
OVERRIDE PRIOR DEFINITION (Y OR N)?
```

Enter Y if you want the last definition mode to be included in the Configuration Table, thus removing the prior definition. Enter N if the last definition is to be ignored.

At log on, if a users account definition and the station Configuration Table entries contain conflicting mappings for the same session LU, the user is informed with a LGON 06 error. You can prevent this from happening by reserving a set of session LU numbers exclusively for the Configuration Table (refer to your Session LU Allocation Worksheet).

```
DISK POOL LU?
```

Enter the first (next) disk logical unit to be included in the spare cartridge pool. ACCTS continues to prompt with this question until you terminate the spare cartridge pool definition by entering /E.

ACCTS does not verify that the LU number is a disk LU, and accepts any number entered. However, when the disk pool is listed, the non-disk LU driver name is given in the size column of the list. Non-disk LUs can be removed from the disk pool using the ALTER,ACCT command, described in Chapter 8 of this manual.

The order in which you input disk pool LUs determines the order of the spare cartridge list. This gives you a degree of control over their allocation. Whenever a cartridge is allocated from the spare cartridge pool with an AC command, the system allocates the first unused cartridge in the list greater than or equal to the size specified in the command. If no cartridge size is specified, the first unused cartridge in the list is allocated.

It is suggested that two criteria be used when determining the order of LUs in the spare cartridge pool. If you want to order your cartridges so that users get the smallest possible cartridge that meets their needs, enter disk pool LUs in order of increasing subchannel size. If you want the system to give allocation priority to one disk unit over another (for example, cartridges on fixed disk platters before cartridges on removable platters), enter the disk pool LUs for the primary device first. In practice, you should use a combination of the above criteria to determine the order of disk cartridge pool LUs for your system.

To change the order of your spare cartridge LUs during this phase, enter /A. ACCTS responds with:

```
REDEFINE DISK POOL (Y OR N)?
```

Enter Y to start over. Any other input will abort the accounts initialization process and terminate ACCTS.

```
PASSWORD FOR MANAGER.SYS?
```

Enter a character string for the MANAGER.SYS account password. REMEMBER THIS PASSWORD. Without it, you will not be able to run ACCTS. The password can consist of up to ten of the following characters: A through Z, a through z, 0 through 9, !, ”, #, $, %, &, ’, (,), ;, <, =, >, ?, \,], ^, −. The characters , (comma), . (period), and / (slash) are not allowed.

Caution It is important that this password be kept secret from most system users as it will allow users access to the entire account structure and all files.

At this point ACCTS completes the Session Monitor initialization process. It performs the following functions:

1. Creates and initializes the session account file.
2. Creates the following accounts:
    a. Group SYS
    b. Group SUPPORT
    c. Group GENERAL
    d. User MANAGER.SYS
    e. User ENGINEER.SUPPORT
3. Allocates and initializes system memory for SCBs and the spare cartridge pool.

Refer to Figure 7-3 for a sample initialization dialogue up to this point.

If the amount of System Available Memory (SAM) required for initialization is unavailable, ACCTS responds with:

```
XXXX WORDS REQUESTED
XXXX WORDS AVAILABLE
ENTER NO. OF WORDS OR /E
```

It is suggested that you enter /E and reboot. This message may be caused if a large enough contiguous block of System Available Memory (SAM) is unavailable due to fragmentation. After you reboot, rerun ACCTS. If this message is not repeated, the problem has been solved.

If the above message persists after rebooting the system, there is not enough System Available Memory (SAM) generated in your system. You have several options:

1. Reconfigure your system to add more System Available Memory. It is recommended that you have at least 2K more SAM than is required by ACCTS.
2. Reduce your session limit. As a rule of thumb, reduce your session limit by: ((Amount Requested)-(Amount Available)+2000)/50. Enter a number smaller than that indicated by the XXXX WORDS AVAILABLE message. Reduce your session limit using the ALTER,ACCTS command (see Chapter 7), and reboot.
3. Regenerate your system with more System Available Memory. If you cannot add more SAM by reconfiguration, you will have to regenerate your system. (You have to reduce system table areas or system common to achieve this increase).
4. Enter an amount at least 2000 less than that indicated by the XXXX WORDS AVAILABLE message. This is likely to degrade both session monitor and overall system performance (depending on your systems particular SAM).

(User inputs are underlined)

```
:RU,ACCT,,"ACCTI                                  List to file "ACCTI
SESSION NOT INITIALIZED
ENTER IN,LI,/TR OR /HE   IN                       Start initialization
SESSION LIMIT?  10
SESSION MEMORY ALLOCATION?  (Y OR N)  Y           Let ACCTS determine size
NUMBER OF USER ACCOUNTS?   40
NUMBER OF GROUP ACCOUNTS?  8
SYSTEM MESSAGE FILE?   "SYSTEMS:RT:2              Message File On LU 2
PROMPT STRING?
                                                  Default To PLEASE LOG ON:
LOCATION OF MESSAGE FILES?                        Default To LU 2 And LU 3
STATION CONFIGURATION (Y OR N)?
Y
STATION LU     15
SESSION LU,SYSTEM LU?
4,98                                              Left CTU
SESSION LU,SYSTEM LU?
5,99                                              Right CTU
SESSION LU,SYSTEM LU?
7,100                                             Auxiliary Printer
SESSION LU,SYSTEM LU?
/E
STATION LU?     16
SESSION LU,SYSTEM LU
      :                                           Additional definitions
SESSION LU,SYSTEM LU?
/E
STATION LU?      21
SESSION LU,SYSTEM LU?
13,142                                            Instrumentation
SESSION LU,SYSTEM LU?
/E
STATION LU?     /E                                End Configuration Table
DISK POOL LU?    30                               Define Spare Cartridge
DISK POOL LU?    32                               Pool LUs
DISK POOL LU?    33
     :                                            More definitions
DISK POOL LU?    60
DISK POOL LU?    /E                               End Disk Pool definition
PASSWORD FOR MANAGER.SYS?   RDS>WED               Input ACCTS password
NEXT?
```

Figure 7-3. Sample Account System Initialization Dialogue

Group Account Definitions

After the account system has been initialized and ACCTS has verified the password, it prompts for the next command with:

```
NEXT?
```

To start your group definitions, enter:

```
NEW,GROUP
```

ACCTS will respond with:

```
GROUP NAME?
```

Enter the name of the group for which an account is to be created. The name must consist of one to ten of the following characters: A through Z, a through z, 0 through 9, !, ", #, $, %, &, ', (,), ;, <, =, >, ?, [, ,], ^, _. The characters , (comma), . (period), and / are invalid. If you enter /A, the NEW,GROUP command is aborted.

Note that all group names must be unique. Enter the first (next) group name listed in your Account Planning Matrix worksheet.

After the group name is entered, ACCTS will ask:

```
SST DEFINITION?  (ENTER SESSION LU, SYSTEM LU OR ENTER /E)
```

Enter a session LU/system LU mapping to be associated with this group. ACCTS continues prompting for entries until a /E is entered. Session LU numbers must be in the range 4 to 63. System LU numbers must be in the range 0 to 254. For each disk cartridge to be dedicated to this group, enter:

```
cartridge LU,cartridge LU
```

Session LUs assigned to disk cartridges must be identical to their respective system LUs.

You should also enter definitions here for those devices that are to be associated with this group. These devices will optionally be accessible to members of the group, depending on how each group member's user account is defined.

It is suggested that session LU numbers assigned to group devices be in the range indicated by your session LU allocation worksheet (if possible). This prevents conflicts with user account SST and Configuration Table definitions.

After the current group account SST definitions are terminated with /E, ACCTS responds with:

```
NEXT?
```

If you have more groups to define, enter additional NEW,GROUP commands. Otherwise, you are ready to define your user accounts. Note that group accounts must be defined prior to their member's user accounts.

Refer to Figure 7-4 for a sample group account definition.

(User inputs are underlined)

```
NEXT?
NEW,GROUP
GROUP NAME?
INVEN                                  Assign group name INVEN
SST DEFINITION? (ENTER SESSION LU, SYSTEM LU, OR ENTER /E)
40,40                                  Dedicated group cartridge
SST DEFINITION?
41,101                                 Device to be associated with group
SST DEFINITION?
6,6                                    Allow group access to line printer
SST DEFINITION?
/E                                     Terminate Account SST definition
NEXT?
```

Figure 7-4. Sample NEW,GROUP Command Dialogue

User Account Definitions

New user accounts can be added with the command:

```
NEW,USER
```

ACCTS will then ask:

```
USER NAME?
```

Enter the name of the user for which an account is to be created. The name must consist of one to ten of the following characters A through Z, a through z, 0 through 9, !, ”, #, $, %, &, ’, (,), ;, <, =, >, ?, [, \,], ^, _. The characters , (comma), . (period), / and “ ” are invalid. If you enter /A the NEW,USER command is aborted.

Refer to he Account Planning Matrix Worksheet for user names. Note that user names must be unique within groups. To simplify accounts system maintenance, it is suggested that each system user be assigned a unique user name.

After the user name is input, ACCTS will ask:

```
GROUP NAME?
```

Enter the name (in the format described above) of an existing group account to which this user belongs. Enter “ ” (blank) to use the default group GENERAL.

```
USE GROUP SST (Y OR N)?
```

Enter Y if group devices (defined in the group Account SST) are to be accessible in the users session. This will cause all devices in the group Account SST to be automatically included in the user Account SST. (This applies to devices currently defined in the group Account SST as well as those devices added later on with the ALTER,GROUP command.) Enter N if group Account SST entries are not to be included in the user Account SST.

```
USER PASSWORD?
```

Enter the user account password. The password may consist of up to ten of the following characters: A through Z, a through z, 0 through 9, !, ”, #, $, %, &, ’, (,), ;, <, =, >, ?, [, \,], ^, _. The characters , (comma), . (period), and / are invalid. Enter a “ ” (space) if no password is required for this account.

It is suggested that the user account password be obtained from the user himself. This will make it easier to remember and reduce the likelihood of other users breaking accounts security.

```
USER’S PRIMARY PROGRAM?  (FMGR or CI)
```

Enter the name of the primary program for this session (must be FMGR or CI). This is the program which executes the HELLO file (see below).

Note If CI is the primary program, the HELLO file name must still have a maximum of only six characters and the file must reside on a directory or cartridge that is compatible as a ‘namr’ subparameter (that is, six-character file name and two-character file directory).

```
NO. OF UDSPS, DEPTH?
```

Enter two values for the User-Definable Directory Search Path (UDSP), the first value for the number of UDSPs (to be allocated when the user logs on) and the second for the depth (number of entries) per UDSP. Both values must fall between 0 and 8, and if one value is 0 both must be 0. The amount of SCB memory space required by the UDSP tables is calculated as follows: (UDSPs * DEPTH) * 2+3.

```
USER HELLO FILE?
```

Enter the file name (filename:sc:crn) of the command file to be transferred to by the user’s copy of FMGR when the user logs on. Enter " " (space) if no such file is to be transferred to at log on.

User HELLO files must reside on cartridges accessible from the users session. You may wish to place user HELLO files on cartridge LU 2 or LU 3 to prevent users from modifying these files. Recall also that command transfer files on these cartridge LUs are granted special capabilities by the system. Commands in these transfer files may modify files on LU 2 and LU 3 and may execute any file manager or break mode command.

For users with CI as their primary program their hello file can be on a CI directory as long as the file descriptor conforms to FMGR namr format (e.g., GARY::HI, where HI is a global directory).

In many applications it may be advantageous to give a user complete control over his HELLO file. In this case, the HELLO file should reside on a cartridge completely accessible to the user (e.g., private or group cartridge). You may wish to create system transfer files on LUs 2 and 3 to perform various functions associated with session initialization. User HELLO files can then invoke these transfer files when appropriate.

Example: (HELLO file line numbers are for reference purposes only):

```
00001 :SV,4,,IH
00002 :RU,BASIC
00003 :EX,SP
```

This example illustrates how the HELLO file can be used to create a particular application environment. Setting the file manager severity level to 4 (in line 00001) inhibits file manager from echoing HELLO file commands on the terminal. More importantly, if an error occurs, file manager does not transfer control to the user's terminal.

When a user logs on with this HELLO file, he is brought immediately into BASIC. After exiting BASIC, he is automatically logged off the system (in line 00003). If the user is given a low enough capability level (see below) he is unable to interfere with this process. Note that instead of running BASIC, the HELLO file can run any set of programs (for example, automatic test programs for instrumentation, data base access programs/utilities, text editors etc.). Basically, HELLO files used in this manner allow you to present a customized system to the user. If desired, user accounts can be structured to keep interaction with the operating and file management systems to a minimum after logging on.

Example:

```
00001 :SV,2,,IH
00002 :RU,KYDMP,0G,DEVKYS
00003 :SL,6,,,6
00004 :CA,9,"BANNR
00005 :DU,9G,6
00006 :DP, YOUR GROUP CARTRIDGE LU IS 35
00007 :DP, YOUR PRIVATE CARTRIDGE LU IS 43.
00008 :DP, OUTPUT TO LU 6 WILL AUTOMATICALLY BE SENT TO
00009 :DP, THE PRINTER WHEN YOU LOG OFF.
00010 :DP, TO PRINT THIS OUTPUT SOONER, TYPE TR, DISPOSE
00011 :SV,0,,IH
00012 ::
```

This HELLO file might be used for individuals doing program development. The severity code (line 00001) is set to inhibit command echoing but allow transfers to the terminal in case of errors. In line 00002, the KYDMP utility is run to dump a set of softkey definitions to the terminal. You might set up softkeys to run utilities (e.g., EDIT,COMPL,LOADR), list files, log off, etc. Softkey definitions are easily set up with the KEYS utility. Refer to the Terminal User's Reference Manual for a description of KYDMP and KEYS.

The SL command in line 00003 will cause the system to setup a spool file for LU 6. Output to this LU will then be diverted to this spool file. The spool file is automatically placed in the queue for output to the printer when (either) the user logs off, a :CS,6 is executed, or another :SL,6,... command is executed. The spool system automatically controls access to the printer so output from different users will not be interspersed.

The HELLO file then proceeds to send a banner to the spool file to identify the users output (lines 00004 and 00005). This is useful in environments where many individuals are making use of the printer at one time. Note that a file manager global parameter is setup to indicate the particular banner file to be associated with the user. Globals provide a convenient method for system transfer files to communicate with each other. In this example, global 9 may be used by other system transfer files requiring a user banner file. A transfer file referred to as DISPOSE in the example, can be created to release spool files for output and to create a new spool file and banner on LU 6:

```
:SL,6,,,6
:DU,9G,6
::
```

Note Spool files are a potentially scarce system resource, use them wisely. It is suggested that you examine user requirements carefully before automatically setting up spool files in HELLO files. You should not be automatically creating spool files for users with little likelihood of using them. The number of spool files that can be active at any one time will be determined by the number of spool EQT entries generated into your system. Note that the COMPL and CLOAD utilities create spool files for list output, eliminating the need for HELLO file spool creation in many cases.

At the end of the sample HELLO file, the severity level is set to 0 (line 00011) and a transfer is made to the users terminal (line 00012).

```
USER CAPABILITY?
```

Enter the user capability level. The capability level must be an integer in the range 1 to 63. Enter " " (space) for the default capability level of 30.

The user capability level determines the subset of file manager and break mode commands the user will be allowed to execute. A user with capability level 20, for example, will be allowed to execute those commands assigned capability levels of twenty or less. The file manager and break mode command tables are used to associate capability levels with commands. If desired, these tables can be substituted with your own tables during generation to alter capability level assignments. Refer to Appendix C for command table formats.

Table 7-1 lists the file manager and break mode capability assignments defined in the command tables as supplied by HP. The various capability levels are summarized as follows:

1 Users may only transfer to command files or log off. Transfers will only be meaningful if command files reside in LU 2 or LU 3 since higher capability commands may be invoked from these files. No break mode commands are acted upon at this capability level. (Users will, however, still receive the break mode prompt).

10 Users can list files, obtain system status, obtain system Table definitions, send and receive messages, mount and dismount cartridges, and up downed devices.

20 Users may create and manipulate files and pack cartridges.

30 This level is intended for the general application programmer. Users may run programs, abort programs, and create type 6 files.

40 This level allows for manipulation of file manager globals.

50 This level enables users to add entries to their SST (potentially giving them access to any system device). Users can place programs in the time list, schedule programs, assign programs to partitions, and adjust priority levels. At this level, programs do NOT have to be necessarily associated with the users session. Level 50 should be reserved for users who are very knowledgeable about the system (e.g., systems programmers, support personnel, etc.,).

60 This level should be reserved exclusively for the support personnel. Access to all system commands is permitted.

63 This level is reserved for the system manager and/or group manager. In addition to all the capabilities of level 60, accounts of this level will be able to create, purge, and alter users within his group and alter the group wide parameters. This level is required by a group manager to perform group FMGR functions such as initializing a group FMGR LU.

After the user capability level is entered, ACCTS asks:

```
MAXIMUM DISK CARTRIDGES?
```

Enter the maximum number of group and/or private cartridges that the user can have mounted to his session at any one time. You should enter an integer from 0 to 60. Enter " " (blank) to use the default limit of 2.

The disk cartridge limit should reflect the number of dedicated private and group cartridges to be accessible to the user plus an additional amount (usually one) for scratch cartridges to be mounted from the spare cartridge pool. For example, say the user's group has two dedicated group cartridges, and the user himself has one dedicated private cartridge. Allowing for one additional scratch cartridge, you would enter 4 in response to this question.

Note that system global cartridges and CI volumes are automatically mounted to the user session at log on and should NOT be included in the cartridge limit.

```
SST DEFINITION?  (ENTER SESSION LU, SYSTEM LU OR ENTER /E)
```

Enter the first (next) session LU/system LU mapping to be associated with this user account. ACCTS will continue prompting for entries until a /E is entered. Session LUs must be in the range 4 to 63. System LU numbers must be in the range 0 to 254.

For each disk cartridge to be dedicated as a private cartridge for this user, enter:

```
cartridge LU,cartridge LU
```

Session LUs assigned to disk cartridges must be identical to their respective system LUs.

You should also enter definitions here for those devices that are to be associated with this user account. Session LU numbers assigned to these devices should be in the range indicated by your session LU allocation worksheet for dedicated user peripherals (if possible). This will prevent conflicts with group Account SST and Configuration Table definitions.

After the user's Account SST definitions are terminated with /E, ACCTS will respond with:

```
NUMBER OF SST SPARES?
```

Enter the number of spare SST entries to be included in the user's SCB at log on. This must be in the range 0 to 60. Enter " " (space) for the default value of 0. Spare SST entries are used when users mount cartridges to their session with the AC command, and create new session LU definitions with the SL command. Certain utilities (e.g., COMPL, CLOAD) also use spare SST entries for spooled list output.

Note The total number of spare SST entries configured into the SCB at log on will be the sum of the NUMBER OF SST SPARES plus the MAXIMUM DISK CARTRIDGES as defined in the users account.

The SST has a limit of 70 entries. The sum of the following cannot be greater than 70:

the number of system spares +
the number of mounted system disk cartridges +
the number of hierarchical disk volumes +
LU 1 +
the number of user SST definitions +
the number of group SST definitions +
the number of entries in the Station Configuration Table +
the maximum number of user disk cartridges

If the user will be initiating spooling operations with the SL command or utilities, you should allocate one SST spare for each concurrent operation. For many applications, one will be sufficient.

ACCTS will now ask:

```
LINK TO AN EXISTING ACCOUNT?  (ENTER " " OR USER.GROUP/PASSWORD)
```

If this account is to be linked to an existing user account, enter the account name in USER.GROUP format (or USER.GROUP/PASSWORD if a password exists for the account). Otherwise, enter " "(space). This feature allows several users to share the same set of private disk cartridges.

At this point ACCTS will create the user account in the account file. If it finds conflicting User and Group account SST definitions, ACCTS will report:

```
CONFLICTING SST DEFINITION - ASSUMING USER DEFINITION
USER: SES LU XX, SYS LU XX GROUP: SES LU XX, SYS LU YY
```

When Group and User account SST definitions specify different system LU mappings for the same session LU, the group definition is ignored.

Now ACCTS asks if this user is to have a user account defined in another group:

```
NEXT GROUP OR /E?
```

Enter the group indicated by the next column checked ( ) for that user in the account planning matrix worksheet. This group must have been defined prior to this point with the NEW,GROUP command. If the user does not belong to any more groups enter /E. A " " (space) entered here will default the group name to GENERAL. Note that the new account is linked to the previously defined user account and shares the same account definition except for group Account SST entries.

If the name of an existing group is entered, ACCTS asks whether to include the group Account SST in the user Account SST:

```
USE GROUP SST (Y OR N)?
```

ACCTS proceeds to define an account for the user in that group. It reprompts for additional groups with NEXT GROUP OR /E? until /E is entered.

Refer to Figure 7-5 for a sample NEW,USER command dialogue.

After the NEW,USER command has completed, ACCTS prompts for the next command with:

```
NEXT?
```

At this time, you may define additional group and user accounts. Refer to Chapter 8 for the procedures required to alter and back up accounts.

Table 7-1. Command Capability Level Assignments

File Manager	Break Mode	Capability Level
EX HE SY TR	HE OP	
AC LI TE CL MC WH DC ME ?? DL *SL SM	$BL *SL UP +BR ST RS *EQ TE $QU FL TI WH *TO	10
AN DP RN CN DU ST CO LL SV CR PK CT PU		20
+OF JO RP EO RT CS RU AB SP TL	+GO RU +OF SZ +SS RT +WS RS +VS	30
CA IF PA SE		40
LO SL	UR IT AS ON PR	50
IN VL OF CU	BR LU TO BL OF *AG DN QU UL EQ SS GO TM	60

* Single Parameter Only
\+ Program must be under sessions control
$ No Parameters permitted

(User inputs are underlined)

NEXT?	
NEW,USER	
USER NAME?	
JOHNSON	Definition for user JOHNSON
GROUP NAME?	
INVEN	Create account JOHNSON.INVEN
USE GROUP SST (Y OR N)?	
Y	Include group Account SST definitions
USER PASSWORD?	
CLARK	
USER'S PRIMARY PROGRAM? (FMGR OR CI)	
FMGR	
NO. OF UDSPS, DEPTH?	
2,4	Two UDSPs with 4 entries each
USER HELLO FILE?	
*CJHEL:RT:2	User Account HELLO file on LU 2
USER CAPABILITY?	
30	Use standard capability level
MAXIMUM DISK CARTRIDGES?	
3	
SST DEFINITION?	
8,8	Allow user access to mag tape
SST DEFINITION?	
50,50	Device to be associated with account
SST DEFINITION?	
51,200	
SST DEFINITION?	
/E	Terminate user Account SST definitions
NUMBER OF SST SPARES?	
1	One SST spare (for spool operations)
LINK TO AN EXISTING ACCOUNT? (ENTER " " OR USER.GROUP/PASSWORD)	
	Do not link to existing account
NEXT GROUP OR /E?	
HP	
USE GROUP SST (Y OR N)?	
Y	
NEXT GROUP OR /E?	
FP	
USE GROUP SST (Y OR N)?	
Y	
NEXT GROUP OR /E?	
LC	
USE GROUP SST (Y OR N)?	
Y	
NEXT GROUP OR /E?	
/E	
NEXT?	

Figure 7-5. Sample NEW,USER Command Dialogue

8

Maintaining the Account System

The Session Monitor Accounts System is maintained by means of the Accounts Setup program ACCTS. This program is run by the System Manager to build, maintain, and backup the system account file. It is also used to startup and shutdown the Session Monitor and to perform other account maintenance functions.

Session Monitor Account File

The account file must be set up before any user can log on to the system. The Account Setup Program (ACCTS) provides the System Manager the capabilities to build and maintain the account file. ACCTS allows new accounts to be added to the account file, existing account definitions to be modified, selected accounts to be deleted from the account file, and account parameters to be changed. The account file may be saved in a disk file or a backup medium (for example, magnetic tape). If necessary, it may be restored from the backup file (or medium). The Account File is comprised of the following components:

- Account File Header
- Active Session Table
- Configuration Table
- Spare Cartridge Pool
- User-Group ID Map
- Account Directory
- User and Group Account Entries

The overall account structure is shown in Figure 8-1. The following sections describe the various Account File components in detail.

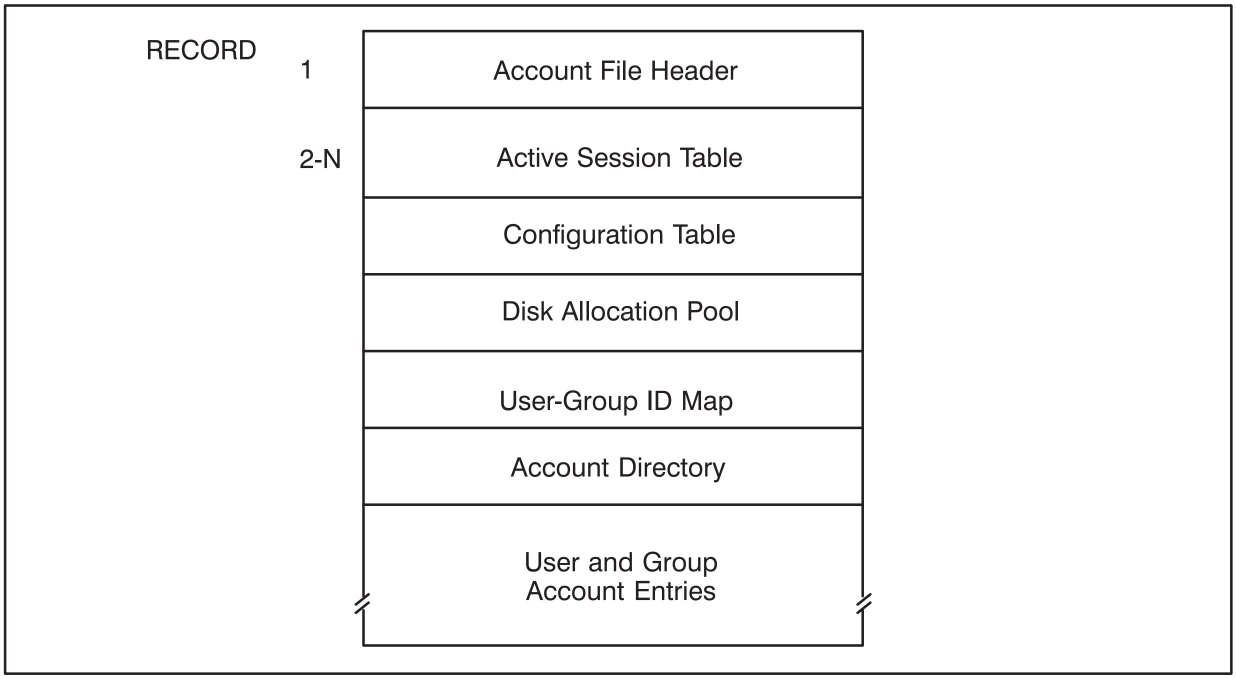


Figure 8-1. Account File Structure

Account File Header

The account file header contains the following information.

- File record pointers of various account tables and directories.
- Resource parameters used during Session Monitor initialization and to control access to the system.
- Session operating parameters such as the System message file NAMR, log on prompt string, etc.

Active Session Table

The active session table contains a list of all users currently logged on, their station, and the time of log on.

Configuration Table

The Configuration Table contains default logical unit definitions for specific stations (terminals) in the system. Each station defined in the Configuration Table has a set of default logical units, which are included in the user Session Switch Table (SST), when logging on from that station. The default logical unit associated with the station itself is always logical unit 1 (LU 1). The following example illustrates the use of the Configuration Table.

3		length of entry
30	1	station (terminal) LU
34	4	default left CTR (LU 34) to LU 4
35	5	default right CTR (LU 35) to LU 5
4		length of next entry
40	1	station (terminal) LU
44	4	default left CTR (LU 44) to LU 4
45	5	default right CTR (LU 45) to LU 5
57	6	default printer (LU 57) to LU 6
0		end of Configuration Table

The left and right cartridge tape units (CTU's) at station LU 30 can be accessed by a session user at this station as LU 4 and LU 5, respectively. Similarly station LU 40 has left and right CTU's that are to be accessed by session users at this station as LU 4 and LU 5. Also associated with station LU 40 is a dedicated line printer (actually LU 57), to be accessed by session users at this station as LU 6.

Only those stations with default LUs in addition to the station LU (Session LU 1) will require entries in the Configuration Table.

Spare Cartridge Pool

The Spare Cartridge Pool is a table of disk logical units assigned to individual users or groups when they require scratch disk space. When a session user requests a disk cartridge via the Allocate Cartridge (AC) command, a spare (unused) cartridge is allocated from this pool. The cartridge is then marked as taken and identified with the user who allocates the cartridge. The cartridge is not returned to the pool until it is dismounted from the system cartridge list.

User/Group ID Map

Every group account and private user accounts are identified in the system internally by a 12-bit account ID number. When a user logs on, both the 12-bit group ID and 12-bit private ID will be placed in the user's SCB. Private user accounts linked to each other are given the same private 12-bit ID.

The account ID number is used by the system to control access to cartridges. When a user mounts a cartridge on the system, his group or private ID is placed in the system cartridge list along with the cartridge LU. Users sharing the same group or private account ID are then permitted to mount the cartridge to their session. Since linked accounts share the same private ID, the same set of private cartridges can be shared by these accounts.

The system uses a 4096 bit map to keep track of allocated account numbers. When an account is defined, and a new account ID must be assigned to it, the system will allocate an unused number (indicated in the map with a 0 bit) and mark it as assigned (that is, 1-bit). Group accounts are allocated lower numbered account IDs and private accounts are allocated higher numbered IDs.

Account Directory

The account directory contains a list of all the user and group accounts defined in the system. The USER.GROUP character string identifier is saved here together with the corresponding account ID numbers and pointers to the actual account definitions.

Group and User Account Entries

These entries define the various operating parameters for all the accounts defined in the system. User account file entries contain the following primary components:

- Account Password. A password may optionally specified with each account. It may be up to ten characters in length.
- User Hello File NAMR. Each user account file entry may be used to define the name of a Hello file. The Hello file is either a file manager procedure file or a CI command file. It is immediately transferred to when a user logs onto the system. For this reason, it must reside on a disk which is already mounted to the user's session when he logs on. Refer to Chapter 7 for a more detailed discussion of Hello files and examples.
- Command Capability Level. The command capability level (integer in the range 1 to 63) defines the subset of file manager and break mode commands that the user may execute. A user assigned capability level 20, for example, is allowed to execute only those commands which have been defined as requiring capability level 20 or less.
- Account SST Entries. Each account can have SST entries defined specifically for it. This may be done to include dedicated disk cartridges and/or peripherals in the user operating environment. In addition to predefined SST entries, each account can be allocated a specified number of spare SST entries to be used during the session as needed (for example, for disk pool cartridges or to reference additional peripherals).

- User Message file NAMR. The message file will contain messages sent to the user by other users of the system. Message files are manipulated by file manager :ME and :SM commands.
- Connect Time and CPU Usage. The connect time indicates the total time (in minutes) that the user has been logged on since his account was initialized or last reset by the system manager (via the ACCTS RESET command). The CPU usage is similar except that it indicates actual CPU time and it is stored in seconds. The last time the user logged off with this account is also recorded.
- Disk Limit. The disk limit specifies the maximum number of group and private cartridges the user can have mounted at any one time.
- Private and Group account ID numbers. Each group account file definition contains the following information:
- Group account ID number.
- Cumulative connect and CPU usage time. These times are similar to those described above. They are the sum of all group member connect and CPU usage times.
- Group SST entries. These entries are optionally included in each group members SCB at log-on. They may define peripheral or dedicated cartridges belonging to the group. Two special accounts are predefined in the Session Monitor. The MANAGER.SYS account (user=MANAGER, group=SYS) is intended for the System Manager. The System Manager has the most extensive capabilities of any user on the system. Among these capabilities are privileged access to all disk cartridges on the system and access to the Account Setup Program. Initially a password is specified for the MANAGER.SYS account during account system initialization. The system manager can modify this password afterwards with the ALTER, USER command.

Warning **It is important that the MANAGER.SYS account be protected with an unpublicized password since any user able to successfully log on as the system manager will have access to protected file domains and will possess the ability to modify any account.**

The ENGINEER.SUPPORT account, like the MANAGER.SYS account, is predefined and should not be purged. This support account is for the use of Hewlett Packard support engineers. Its account capabilities do not include access to all file domains, but will allow the support engineer to execute all system commands.

In addition to the SYS and SUPPORT group accounts, a GENERAL group account is also predefined. When a new user is added to the account system but no group is specified for this user (that is, the default is used), the user is made a member of the GENERAL group. Note that the general group account initially has no group SST entries defined, but SST entries can be added with the ALTER,GROUP command.

Accounts Program Operation

Responses to the ACCTS program are provided using two modes of operation, interactive or direct. In the interactive mode, commands are input from the terminal keyboard. ACCTS prompts the user for each input.

In the direct mode, commands are supplied to the ACCTS program from a disk file or a logical input unit (that is, from a command file).

The system manager can alternate between these two modes at any point at which the ACCTS program is waiting for input by using the TRANSFER command.

To run the account setup program, enter:

```
:RU,ACCTS[,control[,list[,echo]]]
```

where:

control — If specified, control is the name of a file or logical unit number of a device from which a command file will be retrieved.

If control is omitted, or is a logical unit of an interactive device (terminal), ACCTS will operate in interactive mode. It will take its inputs from, and output prompt messages to, the user's terminal (if operating on session) or the system console (if operating non-session). For batch jobs commands is taken from LU 5.

If control is −1, ACCTS is scheduled to initialize Session. This can be done in the WELCOM file if ACCTS is not a permanent program but was loaded on-line, has an account file, and was RPed. Refer also to WELCOM FILE in Chapter 6, System Initialization.

list — If specified, list is the name of a file or logical unit number of a device on which all prompts and responses will be recorded.

If the list parameter is omitted, ACCTS prompts and responses are not sent to a list file/device (this can be altered later with the TRANSFER command).

ECHO — If specified, ACCTS prompts and responses are sent to the user's log device. The log device is the users terminal (if operating in session) or the system console (if operating in non-session mode or within a batch job). If the control parameter is omitted, the ECHO parameter should also be omitted since prompts will automatically be sent to the terminal. Note that when ACCTS is echoing prompts and responses to a list file/device all prompts will be preceded by an asterisk-blank (*).

Example:

```
:RU,ACCTS,5,LSTFIL:YL,ECHO
```

This command schedules ACCTS to take its input from logical unit 5, record prompts and responses on list file LSTFIL (with file security code YL), and echo the prompts and responses to the user's terminal for monitoring purposes.

Example:

```
:RU,ACCTS,ANSFIL:-1:1000,,ECHO
```

This command schedules ACCTS to take its inputs from answer file ANSFIL (with file security code −1 on cartridge 1000), generate no list file, and echo prompts and responses to the user's terminal.

Example:

```
:RU,ACCTS
```

This command schedules accounts to take its inputs in interactive mode from the user's terminal and generate no list file.

Example:

```
:RU,ACCTS,,LSTFIL
```

This command schedules accounts to take its inputs in interactive mode from the user's terminal and echo prompts and responses to file LSTFIL.

The procedures to follow when initializing the account system are described in Chapter 6. They are summarized in this chapter under ACCOUNT SYSTEM INITIALIZATION.

After the account system has been initialized and ACCTS was scheduled from the system console, ACCTS requests a password before it accepts any commands:

```
PASSWORD?
```

The user must supply the password defined for the MANAGER.SYS account. Upon verification of the password, ACCTS prompts:

```
NEXT?
```

Any legal account system command (except IN) can be entered here. A summary of legal accounts commands is shown in Table 8-1.

Table 8-1. ACCTS Command Summary

Command	Description	Page No.
ALTER,ACCT ALTER,GROUP ALTER,USER	Alters global Session Monitor parameters. Alters attribute(s) defined for groups. Alters attribute(s) defined for users.	8-27 8-17 8-18
EXIT	Terminates the account setup program.	8-12
HELP	Lists valid commands and schedules HELP utility	8-9
IN	Initializes the account file; can be entered only when no account file exists.	7-4
LIST,ACCT LIST,GROUP LIST,USER	Lists session-wide information Lists one or more group account entries. Lists one or more user account entries.	8-22 8-21 8-22
LOAD	Rebuilds the account system from an UNLOADED account file.	8-30
NEW,GROUP NEW,USER	Creates an account file entry for a new group. Creates an account file entry for a new user.	8-13 8-14
PASSWORD	Alters the password in the account of the session in which ACCTS is running.	8-20
PURGE,ACCT PURGE,GROUP PURGE,USER	Purges the entire account structure. Removes a group from the account file. Removes a user from the account file.	8-29 8-23 8-23
RESET,GROUP RESET,USER	Clears group time clocks. Clears user time clocks.	8-19 8-20
SD,SESSION SD,0 SD	Shuts down specified session Disable system console as a session terminal. Shutdown entire Session Monitor System.	8-24 8-24 8-25
SU	Restarts the session system after a shut down.	8-26
TELL	Sends a message to a single active user or group or to all active sessions.	8-10
TRANSFER	Transfers control from one LU or file to another.	8-11
UNLOAD	Creates a backup copy of the account file.	8-29
/ABORT /HELP /TRANSFER	Aborts current command. Schedules HELP from within a command. Invokes TRANSFER from within a command.	8-12 8-9 8-11

Command Syntax

Each ACCTS command consists of one of the commands shown in Table 8-2 followed by, in many cases, a parameter list. The parameter list contains one or more parameters that specify operands for the command. The parameter list is required in some commands, but is optional or prohibited in others. Optional parameters are shown enclosed in brackets in the command formats. Within the list, any delimiter can be surrounded by any number of blanks.

Whenever ACCTS is run after it has been initialized, it prompts immediately for a command with NEXT?, if not run interactively, to process commands from the control namr.

General Commands

There are a number of general commands for use by the system manager when running ACCTS. These are: HELP, TELL, TRANSFER, EXIT, and /ABORT.

HELP

The HELP command lists the various ACCTS command and schedules the HELP utility.

```
HE[LP],<error number>[,<list>]
HE[LP]  [,<keyword>[,<list>]]
```
or
```
/HE[LP],<error number>[,<LU>]
```

where:

keyword	is the name of the command about which further explanation is desired. The default is a list of all commands with brief descriptions. If keyword is numeric, the HELP utility is scheduled to expand the error.
list	is the list device to which the explanation is written. The default is LU 1.
error number	is the number of the error which is to expand. If omitted, the most recent error posted to SCB is expanded.

The HELP and /HELP commands are interchangeable with the exception that /HELP may be entered from within general commands, (for example, NEW,USER) and HELP may not.

If HELP is entered immediately after an ACCT nnn error, the HELP utility will be scheduled to supply information on the particular error. Otherwise, HELP supplies a list of all commands (keyword omitted) or a brief description of a specific command (keyword supplied).

Example:

```
NEXT?
NE,USER?
ACCT-nnn
HELP                (schedules HELP immediately after error message)
```

Example:

```
NEXT?
HELP                    (list all commands)
```

Example:

```
NEXT?
HELP,NEW                (list NEW command description)
```

Example:

```
NEW,GROUP
GROUP NAME? LC
SESSION LU,SYSTEM LU? 1,10
ACCT-XXX
SESSION LU,SYSTEM LU? /HE       (schedule HELP from within command)
     :
SESSION LU,SYSTEM LU? 9,10
```

TELL

The TELL command allows any user to send a message to a specific user or group of users who are logged on.

```
TE[LL],user.group,[namr[,message]]
```

where:

user.group — is the currently logged-on user to whom the message is to be sent. A . (period) may be specified to indicate that all users currently logged-on are to receive the message at their terminal. @.group may be used to send a message to all currently logged-on members of the same group.

namr — is a file name or device logical unit number containing the message to be sent to the user(s). If both namr and message parameters are specified, the message contained in the namr is transmitted first.

message — is an ASCII string to be sent to the user(s). The entire TELL command line, including this message string, is limited to a maximum of 80 characters.

Example:

```
TELL,.,SHUTD

The SYSTEM will be shut down for PM in 5 minutes.
Please Log off by then.
```

Example:

```
TELL, JIM.HP,,Please release your spool files.
```

TRANSFER

The TRANSFER command allows the system manager to alternate between interactive and direct command input modes. Command format is:

```
                                       [,NO[ ECHO]]]]
TR[ANSFER][,control[,list<
                                       [,EC[HO]]]]
or
                                       [,NO[ ECHO]]]]
/TR[ANSFER][,control[,list<
                                       [,EC[HO]]]]
```

where:

control — is the logical unit of filename (name:sc:crn) from which all further commands are read. If this parameter specifies an interactive device, prompt messages will be displayed on this depth of ten levels (see below). If control is omitted, ACCTS accepts command input from the previous control file or LU. A negative integer −N specified for this parameter causes the control input to go back N files or the LU specified N levels previously.

list — is the logical unit or filename (name:sc:crn) where all prompts and responses are listed. A O specified for this parameter stops listing to the current list file or LU. If list is omitted, the current list file remains unchanged. Note that all ACCTS prompts are preceded by a * in the list file.

ECHO — enables echoing of all prompts and responses to the log device. When ACCTS is in interactive mode (that is, log device same as control device), prompts and responses are not echoed to this device. The ECHO mode remains in effect until changed with the NOECHO parameter in another TRANSFER command.

NOECHO — disables echoing of all prompts and responses to the log device. The NOECHO mode remains in effect until changed with an ECHO parameter in another TRANSFER command. If both ECHO and NOECHO are omitted, the current echo mode remains in effect.

The TRANSFER and /TRANSFER commands are interchangeable except that /TRANSFER may be entered from within general commands (for example, NEW,USER) and TRANSFER may not.

You may enter the TRANSFER command from the terminal to transfer control to an answer file (or input device). The answer file may contain a TRANSFER command to transfer control to yet another answer file or device. At this point, transfers are said to be nested two deep. This nesting process can continue to a depth of ten levels. To transfer control back to the control file (or device) in effect at the previous level, a TRANSFER command with a null control parameter should be specified (for example, TR). To transfer control back to the control file/device in effect N levels from the current level, a TR,−N should be entered.

Note that an end-of-file condition (or control D input from the terminal) is interpreted as a /TR command and therefore transfers control back to the previous level.

When an error occurs, ACCTS automatically transfers to the terminal (if it is not already the control device). The operator is then re-prompted for a response. A subsequent /TR or TR entered from the terminal will transfer back to the control file/device in effect when the error occurred.

The operator can force a transfer to the terminal at any time by breaking the ACCTS program (that is, BR breakmode command). ACCTS prints ACCT 000 and the prompt for the next command. A transfer can be made back to the control file or device in effect at the time of the break by entering a TR or /TR.

Caution List output always starts at the beginning of the list file. If the list file specified in the TRANSFER command has already been specified in another TRANSFER command or the ACCTS runstring, the original list data will be lost.

Example: Assume file LUFILE contains the following entries:

```
7,10
9,53
10,105
/E
TR
```

```
NEXT?
NEW,GROUP                                (Operator inputs are underlined)
GROUP NAME? WED
SESSION LU,SYSTEM LU? /TR,LUFILE,,ECHO
*SESSION LU,SYSTEM LU?
7,10
*SESSION LU,SYSTEM LU?
9,53
*SESSION LU,SYSTEM LU?
10,105
*SESSION LU,SYSTEM LU?
/E
*NEXT?
TR
NEXT?
```

EXIT

The EXIT command enables the System Manager to terminate the ACCTS program. The command format is:

```
EX[IT]
```

ABORT

The /ABORT command allows the System Manager to abort the current command or subfunction within a command. The command format is:

```
/A[BORT]
```

If /ABORT is entered within a command, the command will not be acted upon and therefore, will have no effect on system operation. If /ABORT is entered as a general command (that is, in response to NEXT?) it terminates the ACCTS program.

Example:

```
NEXT?
NEW,GROUP
GROUP NAME? FP
SESSION LU, SYSTEM LU? 8,9
SESSION LU, SYSTEM LU? /A
NEXT?
```

Group FP is not defined because the command was aborted.

Adding New Accounts

New accounts can be added to the account file by using the NEW,USER and NEW,GROUP commands. New accounts can be added during and after system initialization and the accounts may be used as soon as they are defined.

NEW,GROUP

The NEW,GROUP command is used to enter a new group into the Account File. The command format is:

```
NE[W],G[ROUP]
```

ACCTS prompts for the group name with:

```
GROUP NAME?  <group name>
```

The group name must consist of 1 to 10 of the following ASCII characters: A through Z, a through z, 0 through 9, !, ", #, $, %, &, ', (,), ;, <, =, >, ?, [, \,], ^.

ACCTS then prompts for group account SST definition:

```
SST DEFINITION? <Session LU, sys LU> or </E>
```

Enter session LU and system LU separated by a comma. ACCTS prompts for additional SST entries until /E is entered.

Example:

```
NEXT?
NEW,GROUP
GROUP NAME? FP
SST DEFINITION? 8,8
SST DEFINITION? 6,10
SST DEFINITION? 6,12
SST DEFINITION? /E
NEXT?
```

Refer to the Group Account Definitions section in Chapter 7 for a more detailed discussion of the NEW,GROUP command and associated considerations.

NEW,USER

The NEW,USER command is used to enter a new user account definition into the Account File. The command format is:

```
NE[W][,U[SER]]
```

To define an account for a user, the following information is required:

- User name
- Group name(s)
- Password
- User's primary program
- User hello file
- Capability
- Disk limit
- SST entries
- SST spares

When the NEW,USER command is entered interactively, ACCTS prompts with:

```
USER NAME?  <user name> or </E>
```

Enter the name of the user for whom an account is to be created. The name must consist of 1 to 10 of the following ASCII characters: A through Z, a through z, 0 through 9, !, ", #, $, %, &, ', (,), ;, <, =, >, ?, [, \,], ^, and _. Note that within groups, the user name must be unique.

```
GROUP NAME?  <group name>
```

Enter the name of an existing group to which this user is to be included. (The NEW,GROUP command is used to create and define new group accounts.)

ACCTS then asks whether the group SST (if one exists) is to be used with:

```
USE GROUP SST (Y OR N)?  <Y> or <N>
```

If the group account does not currently have a group SST defined, a Y response may still be entered. If Y is entered, then when a group SST is defined for this group, it is mapped into this user's addressing space.

The next prompt from ACCTS is:

```
USER PASSWORD?  <password>
```

The password may consist of up to 10 of the following ASCII characters: A through Z, 0 through 9, !, ", #, $, %, &, ', (,), ;, <, =, >, [, \,], and <−, ^. An ASCII space entered for the password signifies no password required. Following the password, ACCTS prompts for the user's primary program:

```
USER'S PRIMARY PROGRAM? (FMGR or CI) <program name>
```

Enter the name of the primary program for this session (must be FMGR or CI). This is the program which executes the HELLO file (see below).

Note If CI is the primary program, the HELLO file name must still have a maximum of only six characters and the file must reside on a directory or cartridge that is compatible as a 'Namr' subparameter (a maximum of two ASCII characters). For example, a global directory named HI could be created to hold the HELLO files of users with CI as their primary program.

The next prompt from ACCTS is:

```
-----NO. OF UDSPS, DEPTH?
```

Enter two values, one for the number of User−Definable Directory Search Paths (UDSPs) to be allocated when the user logs on and the other for the depth (number of entries) per UDSP. Both numbers must fall between 0 and 8 and if one number is 0 then both must be 0. The amount of SCB memory space required by the UDSP Table can be calculated as follows: (#UDSPs * DEPTH) * 2 + 3.

ACCTS next prompts for the user's HELLO file:

```
USER HELLO FILE?  <name>
```

Enter the name of the user HELLO file. The HELLO file must reside on a disk which is already mounted when a user logs on.

An ASCII space entered for the user HELLO file indicates no HELLO File. ACCTS next prompts for the user's capability:

```
USER CAPABILITY?  <capability level>
```

Enter the user capability level, an integer from l to 63. The first user defined in the account file is the System Manager, who must have the highest capability defined. Following the capability, ACCTS prompts for the maximum number of disk cartridges which the user is allowed to have mounted at any given time.

```
MAXIMUM DISK CARTRIDGES? <total number of disk cartridges>
```

Enter the maximum number of private and/or group cartridges which the user can have mounted to a session at any given time. ACCTS next prompts (and continues to prompt) for the SST entries:

```
SST DEFINITION?  <session LU,system LU> or </E>
```

Enter system LU, session LU or enter /E to terminate the list. ACCTS next prompts for the number of spare SST entries:

```
SST SPARES?  <number of SST spares>
```

Enter the number of SST spares, an integer. Finally, ACCTS will ask whether the user is to be linked to an existing account. This allows the user access to files that the user may own as a member of a different account.

```
LINK TO AN EXISTING ACCOUNT?  <blank> or <user.group/password>
```

Enter a blank or user.group/password. The user.group name specified must be the name of one of the user's existing accounts. Note that if this existing account is protected with a password, the password must be specified.

At this point, the user's account is set up. If currently being run in the MANAGER.SYS session, ACCTS asks for the next group that this user is to be included in:

```
NEXT GROUP OR /E?  <group name> or </E>
or
USE GROUP SST (Y OR N)?  <Y> or <N>
```

The new account is linked to the previously defined user account and shares the same account definition except for group account SST entries. The group account must have been previously defined with the NEW,GROUP command. ACCTS prompts for additional groups until /E is entered.

Default values may be used for defining many of the user attributes. Table 8-2 describes these default values. The default value is used if " " (ASCII space) is entered when ACCTS prompts for a user attribute (such as capability or password).

Table 8-2. NEW,USER Command Defaults

Attribute	Default Value
GROUP NAME	GENERAL
USE GROUP SST	yes
USER PASSWORD	no password
USER'S PRIMARY PROGRAM	FMGR
USER HELLO FILE	no hello file
USER CAPABILITY	30
MAXIMUM DISK CARTRIDGES	2
SST SPARES	5
LINK TO AN EXISTING ACCOUNT	no (blank)

Refer to the User Account Definitions section in Chapter 7 for a more detailed discussion of the NEW,USER command and associated considerations.

```
MODIFYING OLD ACCOUNTS
```

The ALTER and RESET commands allow modification of specific user or group account attributes in the account file. For example, a user may wish to change the log-on password or a group leader may request that the capability levels defined for the members of the group be raised.

ALTER,GROUP

The ALTER,GROUP command allows the modification of attributes defined in group accounts. When an attribute is modified, the change is made to the account file, but it does not apply to users currently logged on; it will take effect for the next log-on by users belonging to this group. The command format is:

```
AL[TER],G[ROUP],group
```

where:

group is the name of the group account to be modified. Specify GENERAL to modify the general group account.

ACCTS prompts with:

```
NEW GROUP NAME or /?  <groupname> or </>
```

If the group name is to be changed, enter the new group name. This changes the name of the group in the group account definition and all user accounts in this group. This prompt is not given if the GENERAL group was specified in the ALTER,GROUP command because the group name GENERAL cannot be changed. If the group name does not need to be changed, enter a /.

ACCTS then prompts for group SST modifications:

```
SST DEFINITION? (Enter Session LU, System LU, or enter /E)
SESSION LU,SYSTEM LU?  <session LU,system LU> or </E>
```

Enter a new or modified group SST entry (session LU, system LU). ACCTS continues to prompt for SST entries until /E is entered. Note that if the same session LU is specified more than once, the last one entered will be the value in effect after modification. Specifying – for the system LU deletes the entry for the specified session LU from the group Session Switch Table (SST).

Example:

1. To add or modify a SST entry for system LU 12, session LU 11.

   ```
   ALTER,GROUP,<groupname>
   NEW GROUP NAME or /? /
   SST? 11,12
   SST? /E
   ```

   If an entry already exists for session LU 11, the associated system LU in the entry is changed to 12. If no entry exists for session LU 11, a new entry is added with session LU 11 associated with system LU 12.

2. To delete an existing SST entry for session LU 11.

   ```
   AL,G,group
   GROUPNAME? NEWNAME
   SST? 11,-
   SST? /E
   ```

3. To change the name of an existing group and all user accounts of the group.

   ```
       AL,G,group
   GROUPNAME? NEWNAME
   SST? /E
   ```

ALTER,USER

The ALTER,USER command allows the modification of any of the following attributes defined for a user: password, hello file, capability level, disk cartridge limit, SST entries, number of SST spares, and whether or not to use the group SST. When an attribute is modified, the change is made to the account file, but it does not apply to users currently logged on; it takes effect for users who subsequently log-on to a session. The command format is:

```
AL[TER],[U[SER]], user.group
```

where:

user.group — is the user and group name assigned to the user in the NEW,USER command. user.@, @.group and @.@ are also valid, where @ means all.

ACCTS prompts for the attributes as shown below. If no change to the value of the attribute is desired, enter / (slash). If the user specified is one unique account (i.e., a @ was not specified) the user name can be changed and the account can be assigned to a different group. The account may be linked to another account. For single account alters ACCTS issues the next prompt.

```
NEW USER NAME?
```

If the user NAME is to be changed enter new user name. This will change the name of the user. If the user name does not need to be changed enter a /.

Then ACCTS prompts with:

```
NEW GROUP?
```

If the account is to be assigned to a different group, enter the new groupname. If the account is to remain unchanged, enter /.

```
GROUP SST (Y OR N)?
```

Enter either Y or N. This attribute indicates whether the group SST for this user is to be mapped into the user account SST.

ACCTS prompts:

```
PASSWORD?
```

Enter the new account password to be assigned to the user. To delete the password assigned to the account, enter “ ” (blank).

```
USER'S PRIMARY PROGRAM?  (FMGR or CI)
```

Enter the name of the primary program for this session.

```
NO. OF UDSPS, DEPTH
```

Enter two values, each between 0 and 8, the first representing the number of UDSPs and the second the depth (number of entries) per UDSP.

```
HELLO FILE?
```

Enter the name of the new file to which control is transferred when the user logs on.

```
CAPABILITY?
```

Enter the new capability level, an integer from l to 63, to be assigned to the user.

```
MAXIMUM DISK CARTRIDGES?
```

Enter the new maximum number of private and/or group cartridges which the user may have mounted to his session at any given time.

```
SESSION LU,SYSTEM LU?
```

Enter the new or modified SST entry. ACCTS repeats this prompt until /E is entered. If the same session LU is specified more than once, the last value specified is used. Specifying – for the system LU deletes the entry for the specified session LU.

```
SST SPARES?
```

Enter the number of spare entries in the SST to be allowed.

```
LINK TO AN EXISTING ACCOUNT?
```

Enter a blank or / to effect no change in the accounts association. Enter a User.Group/password if you want the account linked to a different user.

RESET,GROUP

The RESET,GROUP command clears the processor-usage and/or connect-time clocks for a specific group or all groups. The command format is:

```
                          [,CP[U]]
RE[SET  ],G[ROUP],group<
                          [,CO[NNECT]]
```

where:

group	is the name of the group for which time clocks are to be reset. @ specifies that all group time clocks be reset.
CPU	is a specification that only the processor-usage counter is to be reset for the specified group (optional parameter).
CONNECT	is a specification that only the connect-time usage counter is to be reset for the specified group (optional parameter).

Note that resetting the group clocks does not affect the individual user clocks for members of that group. User clocks can be reset with the “RESET,USER, .” command.

Example:

To reset the group connect-time and CPU usage clocks for group HP:

```
RESET,GROUP,HP
```

RESET,USER

The RESET,GROUP command clears the processor-usage and/or connect-time clocks for a specific group or all groups. The command format is:

```
                              [,CP[U]]
RE[SET],U[SER],user.group  <
                              [,CO[NNECT]]
```

PASSWORD

The PASSWORD command allows any user to change his own password. ACCTS asks for his current password first.

```
ENTER CURRENT PASSWORD
```

As the password is input, it is not echoed. When this is verified with the password in the account, ACCTS asks for the new password.

```
ENTER NEW PASSWORD
```

Again, the input is not echoed. ACCTS then displays the new password just entered and ask the user to verify its correctness. Once the password is verified, ACCTS changes the password in the Accounts File. All future references to the account require the new password. ACCTS then over-prints the password several times to obscure it.

Displaying Account Information

The LIST,USER and LIST,GROUP commands will list user or group account file entries. Unless specified, user passwords and account ID numbers will not be listed. One of the attributes to be listed, which might be of some accounting use, is the total user or group connect-time. The LIST,ACCT command will list system information, including the users currently logged on, the status of the spare cartridge pool, etc.

Note that any of the LIST commands can be terminated by breaking the ACCTS program. This can be done by entering the BR (break mode) command. ACCTS responds with:

```
ACCT 000
```

and then prompts for the next command.

LIST,GROUP

The LIST,GROUP command lists the specified group account entries.

```
LI[ST],G[ROUP],group[,list[,ID]]
```

where:

group	is the name of the group whose account file entry is to be listed. @ may be specified to indicate all group accounts (default).
list	is the list device (logical unit or file name) to which the listing is to be directed. The default value is the list file device or the log device terminal if no list device is specified.
ID	is an optional parameter that includes the group account ID number in the listing.

Example:

To list the group account for the group HP.

```
LI,G,HP,,ID
```

LIST,USER

The LIST,USER command lists user account definitions. The command format is:

```
LI[ST],[U[SER]],user.group[,list[,PA[SS][,ID]]]
```

where:

user.group — is the name of the user whose account file entry is to be listed. @.group may be specified to list all users in a group. user.@ may be specified to list account definitions for a user belonging to several groups. @ may be specified to list account file entries for all users (default). A . (period) may be specified to list all users and all group accounts.

list — is the list device (logical unit or namr) to which the listing is to be directed. The default is the current list file/device or the log device (terminal) if no list device has been specified.

PASS — is an optional parameter that includes the user password in the account listing.

ID — is an optional parameter that includes the user account ID number in the account listing.

LIST,ACCT

The LIST,ACCT command is used to list session information. This command lists:

1. The name of the system message file, the cartridge CRN or LU where user message files are stored and the session limit.
2. The currently active sessions.
3. The current status of cartridges in the spare cartridge pool.
4. The Configuration Table.

```
                     [,AC[TIVE SESSIONS]]]
                     [,PO[OL]]]
LI[ST],A[CCT][,list<
                     [,CO[NFIGURATION TABLE]]]
L1,ACCTS,,ALL
```

Purging Accounts

The PURGE,USER and PURGE,GROUP commands delete accounts (user or group) from the account file.

PURGE,GROUP

The PURGE,GROUP command removes a group from the account file. All users belonging to the group are also removed from the account file.

```
PU[RGE],G[ROUP],group
```

where:

group — is the name of the group whose account entry is to be purged. @ can be specified to purge all user and group accounts with the exception of the MANAGER.SYS, ENGINEER.SUPPORT and GENERAL accounts.

The program prompts for verification:

```
GROUP group TO BE PURGED (Y OR N)?
```

Note that this command does not affect users currently logged on under this group account.

PURGE,USER

The PURGE,USER command deletes a user from the account file. The user is not able to log-on again until a new account file entry for this user is created with the NEW,USER command. All disk cartridges for the user account to be purged are reassigned to the SYS group account.

```
PU[RGE],[U]SER]],user.group
```

where:

user.group — is the name of the user whose account is to be purged. @.group may be specified to purge all users in a group, but leave the group account intact. The MANAGER.SYS account cannot be purged. PU,U,@ will purge all users in the GENERAL group.

The program prompts for verification:

```
USER user.group TO BE PURGED (Y OR N)?
```

Note that this command does not affect users currently logged on to this account.

Session Monitor System Control

The system control commands are used to perform the following: terminate individual sessions, terminate all sessions and shut the session monitor system down, disable the system console as a session terminal, and restart the session monitor after it has been shutdown. These commands are described in the following paragraphs.

SD,SESSION

The SD command is used to terminate a session. This command performs the following: logs the specified session off the system, terminates associated processes, and releases session related resources. The command format is:

```
                [,SP or SG]
SD,session<
                [,RP or RG]
```

where:

Session	is the session identifier of the user to be logged off. Normally this is the station LU.
SP or RP	is an optional parameter entered to save (SP) or to remove (RP) the session private cartridges. The default is SP.
SG or RG	is an optional parameter entered to save (SG) or to remove (RG) the group cartridges. The default is SG.

The SD,session command is entered to log a particular user off, remove all programs associated with that user's session, close and release the associated spool files, and release the session control block (and extensions). After this command is entered, the following message is displayed on the specified station terminal:

```
SESSION ABORTED BY SYSTEM MANAGER
```

SD,0

The SD,0 command is used to disable the system console as a session terminal. This command is used only after the EN command has been entered at the system console to enable it as a session terminal. The disable system console command format is:

```
SD,0
```

If there is an active session at the system console, the system console will revert to its standard operation when the session user logs off.

Example:

Assuming the System Manager is on the system console, and it is not enabled as a session terminal:

```
:RU,ACCTS
PASSWORD?
NEXT?
SD,0
NEXT?
EXIT
ENDACCTS
:
```

SD

The SD command is used to shut down the Session Monitor. The following functions are performed when this command is entered:

1. Prohibit future users from logging on, leaving current sessions unaffected.
2. Terminate all current sessions and session related batch, jobs and spools.
3. Completely deallocate session monitor system resources.

The command format is:

```
SD [,RE[LEASE MEMORY]]
```

where:

RELEASE MEMORY is a specification to release all system memory resources allocated to Session Monitor.

When this command is entered, ACCTS responds with:

```
DO YOU REALLY WANT TO SHUT DOWN THE SESSION SYSTEM (YES OR NO)?
```

Enter YES to shut down the Session Monitor. Enter NO to terminate this command. System operation will not be affected.

If YES is entered, ACCTS asks for a shut down message:

```
SHUT DOWN MESSAGE (20 CHARS)
```

This message will be displayed whenever users try to log on the system. Enter " " (blank) for the default message: SESSION SHUT DOWN.

At this point, new users will be unable to log on, but currently active sessions will remain unaffected. If there are active sessions, ACCTS prints the active sessions, jobs and spools. ACCTS then asks if these sessions are to be shut down now:

```
TO SHUT DOWN "NOW" WE MUST ABORT THE ABOVE PROCESSES!!
ABORT THE ABOVE PROCESSES (YES OR NO)?
```

Enter NO if these sessions, jobs, and spools are to remain active until their normal completion. Enter YES if all session related activity is to be terminated in the system immediately.

If YES is entered, ACCTS proceeds to log all users off, terminate any session related batch jobs in progress (or waiting to be run), and close and release all open session related spool files. If the RELEASE MEMORY parameter was specified in the SD command, the memory allocated at bootup (or at least startup) is deallocated and returned to the system.

Caution It is strongly recommended that the RELEASE MEMORY parameter NOT be specified in the SD command if session monitor is to be restarted at a later time before rebooting the system. If restarted under these circumstances, the session memory area may permanently fragment System Available Memory, possibly severely degrading system performance.

After the session monitor has been shut down, ACCTS displays the following messages when prompting for each new command:

```
SHUT DOWN
NEXT?
```

SU

The SU (Start Up) command restarts the session monitor after a shut down. The command format is:

```
SU
```

After the SU command is entered, users will again be able to log on the system. If the system was shut down with an SD,RE command, session monitor will re-allocate memory for itself at this time. The prompt password is only requested when running accounts outside of session control.

Example:

```
:RU,ACCTS
PASSWORD?  <password>  (Not echoed)
SHUT DOWN
NEXT?
SU
NEXT?
```

Accounts System Maintenance

The ALTER,ACCT and PURGE,ACCT commands allow either altering or purging of the entire account structure. The UNLOAD and LOAD commands are provided for maintenance of the account file. The UNLOAD command is used to transfer the contents of the account file to a logical unit or another file. This provides a backup of the account file. The LOAD command is used to rebuild the account file, if necessary.

ALTER,ACCT

The ALTER,ACCT command is used to change the following:

1. The maximum number of active sessions allowed by the Session Monitor.
2. The system message file.
3. The disk allocated to the Session Monitor disk pool.
4. The Configuration Table.

The command format is:

```
AL[TER],A[CCT]
```

ACCTS prompts for the attributes shown below. If no change to the value of the attribute is desired, enter / (slash). To change a value to the default or delete it enter “ ” (space).

```
SESSION LIMIT?
```

Enter a non-negative integer signifying the maximum number of active sessions. If set to zero, all users are turned away with a SESSION LIMIT EXCEEDED message. This does not affect users who are currently logged-on.

```
CHANGE MEMORY ALLOCATION (Y OR N)?
```

Enter Y or N, *only*; do not enter / (slash). Enter Y to change the amount of memory allocated for session control blocks using the memory allocation algorithm. The memory is allocated at system startup. ACCTS will display the MESSAGE FILE? prompt if N was entered. If Y was entered, the following prompt is displayed:

```
NO. of WORDS?
```

This question is asked only if the memory allocation algorithm is not to be used (Y entered above). Enter the decimal number of words to be allocated for Session Monitor at start up.

```
MESSAGE FILE?
```

Enter the namr (file name:sc:crn) of the system message file. The default is no message file.

```
PROMPT STRING?
```

Enter the log-on prompt string of up to 20 characters. The default is PLEASE LOG ON:.

```
LOCATION OF MESSAGE FILES?
```

Enter the cartridge reference number (CRN) or negative disk LU of the cartridge where message files are located. Default is LU 2 and LU 3. At this time the above changes are posted to the Account File.

```
ADD DISK LU (Y OR N)?
```

If Y is entered, ACCTS prompts with DISK LU?

Enter the logical unit of the disk to be added to the spare cartridge pool. ACCTS repeats this prompt until /E is entered.

```
PURGE DISK POOL LU?
```

If Y is entered, ACCTS prompts with DISK LU?

Enter the logical unit of the disk to be purged from the disk pool. ACCTS will repeat this prompt until /E is entered. When this phase is complete, ACCTS posts the new disk pool to memory and the Account File.

```
STATION CONFIGURATION (A(DD), D(ELETE), M(ODIFY) OR " " (NO CHANGE))?
```

If A is entered, ACCTS prompts for a station LU and associated device and default logical unit pairs.

If D is entered, ACCTS prompts for the station LU number to be deleted from the Configuration Table.

If M is entered, ACCTS prompts for the station LU number to be modified in the Configuration Table. ACCTS then prompts for the SST definitions to be associated with this station LU.

Example:

Suppose the Configuration Table contains an entry for station LU 30 as shown below:

30	1	station (terminal) LU
34	4	Default LU 34 to 4
35	5	Default LU 35 to 5

To modify the entry so that default LU 5 is directed to LU 39 instead of LU 35 and to include a new entry association LU 38 with default LU 6, enter the following commands:

```
ALTER,A
SESSION LIMIT?
    :
```

ACCTS prompts with:

```
CONFIGURATION TABLE (A(DD), D(ELETE), M(ODIFY) OR " " (NO CHANGE))?
```

Enter M. ACCTS then prompts for the station LU to modify:

```
STATION LU?
```

Enter 30, the station logical unit. ACCTS then prompts (and continues prompting until /E is entered) for each pair of device/default logical units to be associated with station LU 30:

```
SESSION LU, SYSTEM LU?  Enter "4,34".
SESSION LU, SYSTEM LU?  Enter "5,39".
SESSION LU, SYSTEM LU?  Enter "6,38".
SESSION LU, SYSTEM LU?  Enter "/E".
```

ACCTS makes the modifications and returns with the NEXT? prompt.

The modified Configuration Table entry for Station LU 30 now looks like:

30	1	station (terminal) LU
34	4	Default LU 34 to LU 4
39	5	Default LU 39 to LU 5
38	6	Default LU 38 to LU 6

PURGE,ACCT

The PURGE,ACCT command is used to purge the entire Session Monitor account structure including all user and group accounts, the Configuration Table and the spare disk pool. ACCTS accepts this command only if there are no active sessions. The command format is:

```
PU[RGE],A[CCT]
```

The following prompt is always issued to verify the purge request:

```
DO YOU REALLY WANT TO PURGE THE ACCOUNT STRUCTURE (YES OR NO)?
```

A YES response will purge the account structure. ACCTS must be run to create another account file. To reconstruct the account file from a backup file created by the UNLOAD command, use the LOAD command.

UNLOAD

The UNLOAD command is used to write the contents of the account file to a logical unit or another file. This provides a backup copy of the account system for use with the LOAD command in the event that the account file is destroyed. The command format is:

```
UN[LOAD],namr
```

where:

namr — is the logical unit or new file name to which the account file is to be dumped. The default is LU 8.

Note UNLOAD compresses empty spaces out of Account File. Also, prior to unloading the Account File, it is recommended that all disk cartridges be dismounted. This eliminates the chance of errors in the user cartridge list if user accounts are purged and added between UNLOAD and LOAD operations.

LOAD

The LOAD command is used to restore the account system using the backup file produced by the UNLOAD command. The command can be used to restore user and group accounts or the entire account file including the header information, the Configuration Table, the spare disk pool, the directory and all accounts. The command format is:

```
                [,ACCTS]
LO[AD],namr<
                [,ALL]
```

where:

namr — is the logical unit or name of the account backup file. A 0 parameter entered specifies that a new account file is to be constructed from the current file to expand the Account File.

ACCTS — indicates that the accounts in the account file are purged and those on the backup file will be loaded into the existing file. This is the default parameter.

ALL — is similar to ACCTS. This parameter also rebuilds the header information, the Configuration Table, and the spare cartridge from the backup file.

The accounts system must be shut down before loading a new accounts file except when loading from the current file (i.e., LOAD, 0 command). If there are active sessions when any other LOAD command is entered, ACCTS prints the number of the sessions, jobs, and spools, and asks if these processes can be aborted:

```
TO SHUT DOWN "NOW" WE MUST ABORT
THE ABOVE PROCESSES!!
ABORT THE ABOVE PROCESSES (YES OR NO)?
```

Enter NO to allow a soft shut down to occur. The sessions, jobs and spools currently active are allowed to proceed until their normal termination; however, new users attempting to log on see this message at their terminals:

```
SESSION SHUT DOWN
```

ACCTS then responds with a NEXT? prompt. Once the current session activity has completed, you can enter a new LOAD command to restore your account system.

Enter YES to allow ACCTS to log all active users off and terminate and clean-up any session related batch jobs and spools. ACCTS displays a shut down message and then proceed with the LOAD operation.

ACCTS first requests the DISK LU on which the accounts file is to be located.

```
ENTER DISK LU FOR ACCTS FILE:
```

Caution For all session subsystems to operate properly the DISK cartridge must be mounted as a system disk.

ACCTS then reports the total number of accounts:

```
XXXX ACCOUNTS REQUIRED
```

ACCTS then allows changing of the account file contents by prompting with:

```
NUMBER OF USER ACCOUNTS?  <number> or <space>
NUMBER OF GROUP ACCOUNTS? <number> or <space>
```

If no changes are required, enter " " (space) to both questions. Otherwise, enter the new maximum number of user and group accounts. ACCTS uses the following algorithm for calculating account directory size and the number of records allocated in the account file for account definitions.

(# USERS + # USERS/5 + # GROUPS)*8 / 8 + 7

Note UNLOAD compresses the Account File to an absolute minimum. Therefore, the default would not allow the addition of any new accounts.

Next ACCTS reports the size of the current Configuration Table:

```
STATION TABLE REQUIRES XXXXX WORDS
```

ACCTS next prompts for an estimate of the new Configuration Table size:

```
ENTER <number of stations>, <average size>
```

Enter the number of stations in your system and the estimated average number of Configuration Table SST definitions for each station (plus two for the entry length word and station LU word). If you wish to use the minimum length necessary to accommodate the Configuration Table defined in the backup file, enter " " (space).

Account Command File Formats

The account command file format is shown below:

For new user accounts:

Contents	**Comments**
NE,U	
user name	(1-10 ASCII characters)
group name	(1-10 ASCII characters)
Y or N	(group SST definition)
password or “ ”	(1-10 ASCII characters)
primary program	(FMGR or CI)
#UDSPs, depth	(2 integers, 0-8)
hello file	(filename:sc:crn)
capability	(integer, 1-63)
disk limit	(integer)
session LU, system LU	(user SST definition)
/E	(terminate SST definition)
SST spares	
link	(user,group/password or “ ”)
group name	(1-10 ASCII characters)
Y or N	(group SST definition)
:	
:	
/E	

For new group accounts:

Contents	**Comments**
NE,G	
group name	(l-10 ASCII characters)
session LU, system LU	(group SST definition)
/E	(terminate SST definition)

ACCTS and LGON Error Messages

Error conditions encountered during the execution of the ACCTS program result in the display of numbered error codes in the format:

```
ACCT nnn
```

where:

nnn   is the error number

A list of the common error messages is provided in Table 8-3. When an error is detected, a transfer to the operator console occurs, allowing the operator to enter the correct response in order to continue ACCTS execution. Refer to the Quick Reference Guide for a complete list of the ACCTS program errors. Complete ACCTS error descriptions follow Table 8-3.

LGON error messages follow the ACCTS error messages.

Table 8-3. ACCTS Error Summary

Error Code	Meaning
ACCT 004	ILLEGAL LU
ACCT 012	LU NOT IN SESSION SWITCH TABLE
ACCT 013	TRANSFER STACK OVERFLOW
ACCT 046	INSUFFICIENT CAPABILITY
ACCT-200	ACCOUNT NOT FOUND
ACCT-201	NO FREE ACCOUNTS
ACCT-202	ACCOUNT WITH THIS NAME ALREADY EXISTS
ACCT-203	INVALID ACCOUNT NAME
ACCT-204	INVALID PASSWORD
ACCT-205	INVALID COMMAND
ACCT-206	INVALID FILE NAME
ACCT-207	INVALID CAPABILITY
ACCT-208	INVALID DISK LIMIT
ACCT-209	INVALID SST ENTRY
ACCT-210	CONFLICT IN SST DEFINITION
ACCT-211	USER OR GROUP ID NOT AVAILABLE
ACCT-212	INVALID NUMBER OF SST SPARES
ACCT-213	INVALID MEMORY REQUEST
ACCT-215	LIST NAMR IN TRANSFER STACK
ACCT-216	ILLEGAL RESPONSE FOR PRIMARY PROGRAM
ACCT-217	ILLEGAL VALUE ENTERED FOR NUMBER OF UDSPS OR DEPTH
ACCT-218	SESSION NOT SHUTDOWN

Table 8-3. ACCTS Error Summary (Cont.)

Error Code	Meaning
ACCT-219	NOT ENOUGH ROOM IN FILE FOR NEW TABLE
ACCT-220	CORRUPT STATION TABLE SPARES
ACCT-221	NOT AN ACTIVE SESSION
ACCT-222	ILLEGAL SYSTEM LU
ACCT-223	ILLEGAL SHUT DOWN PARAMETER
ACCT-225	SESSION MEMORY CAN NOT BE RETURNED TO SYSTEM (REBOOT).
ACCT-046	GREATER THAN 255 EXTENTS
ACCT-099	DIRECTORY MANAGER EXEC REQUEST WAS ABORTED
ACCT-041	NO ROOM IN SST
ACCT-040	LU NOT FOUND IN SST
ACCT-039	CONFLICT IN SST DEFINITION
ACCT-035	63 DISKS ALREADY MOUNTED TO SYSTEM
ACCT-034	DISK ALREADY MOUNTED
ACCT-033	NOT ENOUGH ROOM ON CARTRIDGE
ACCT-032	CARTRIDGE NOT FOUND
ACCT-030	VALUE TOO LARGE FOR PARAMETER
ACCT-026	QUEUE FULL OR MAX PENDING SPOOLS EXCEEDED
ACCT-025	NO SPLCON ROOM
ACCT-024	NO MORE BATCH SWITCHES
ACCT-023	NO AVAILABLE SPOOL FILES
ACCT-022	NO AVAILABLE SPOOL LU'S
ACCT-021	ILLEGAL DESTINATION LU
ACCT-020	ILLEGAL ACCESS LU
ACCT-019	ILLEGAL ACCESS ON A SYSTEM DISK
ACCT-018	ILLEGAL LU; LU NOT ASSIGNED TO SYSTEM
ACCT-017	ILLEGAL READ/WRITE ON TYPE 0 FILE
ACCT-016	ILLEGAL TYPE 0, OR FILE BLOCKSSIZE=0
ACCT-015	ILLEGAL NAME
ACCT-014	DIRECTORY FULL
ACCT-013	DISK LOCKED
ACCT-012	EOF OR SOF ERROR
ACCT-011	DCB NOT OPEN
ACCT-010	NOT ENOUGH PARAMETERS
ACCT-009	ATTEMPT TO USE APOSN OR FORCE A TYPE 0 FILE TO TYPE 1
ACCT-008	FILE OPEN OR LOCK REJECTED
ACCT-007	ILLEGAL SECURITY CODE OR ILLEGAL WRITE ON LU 2 LU 3
ACCT-006	FILE NOT FOUND
ACCT-005	RECORD LENGTH ILLEGAL
ACCT-004	MORE THAN 32767 RECORDS IN A TYPE 2 FILE
ACCT-003	BACKSPACE ILLEGAL
ACCT-002	DUPLICATE FILE NAME
ACCT-001	DISK ERROR

ACCTS Help File Entries

```
""
ACCT 004
ILLEGAL LU
AN LU WAS SPECIFIED WHICH:
1) CANNOT HANDLE BINARY DATA
2) IS NOT AN INPUT DEVICE
3) IS NOT AN OUTPUT DEVICE
4) THE DEVICE IS WRITE PROTECTED

""
ACCT 012
LU NOT IN SESSION SWITCH TABLE
ENTER THE CORRECT LU OR EXIT ACCOUNTS AND PUT LU IN SST WITH SL
COMMAND.

""
ACCT 013
TRANSFER STACK OVERFLOW
THE TRANSFER STACK IS ONLY 10 DEEP
TR,-11 CLEARS THE TRANSFER STACK.

""
ACCT 046
INSUFFICIENT CAPABILITY
ATTEMPT TO EXECUTE A COMMAND WHICH IS RESERVED FOR GROUP
MANAGERS OR THE SYSTEM MANAGER.

""
ACCT-200
ACCOUNT NOT FOUND
GROUP ACCOUNT MUST BE DEFINED BEFORE A USER CAN BE ASSIGNED TO IT.

""
ACCT-201
NO FREE ACCOUNTS
THE "LO,0" COMMAND CAN BE USED TO EXPAND THE ACCOUNTS FILE.

""
ACCT-202
ACCOUNT WITH THIS NAME ALREADY EXISTS.

""
ACCT-203
INVALID ACCOUNT NAME

   1)  ONLY 10 ALPHANUMERIC CHARACTERS ARE ALLOWED FOR A NAME.
   2)  WHEN LINKING TO AN ACCOUNT WHICH BELONGS TO GROUP GENERAL
       ".GENERAL" MUST BE SPECIFIED.
```

```
""
ACCT-204
INVALID PASSWORD

   1) THE PASSWORD OF THE SYSTEM MANAGER IS REQUIRED TO RUN ACCTS
      FROM A NON SESSION CONSOLE.
   2) THE PASSWORD OF THE ACCOUNT TO WHICH THIS IS BEING LINKED IS
      REQUIRED.
   3) THE PASSWORD FOR CURRENT ACCOUNT IS INCORRECT.
   4) THE NEW PASSWORD CONTAINS AN ILLEGAL CHARACTER.  THE CHARACTER MUST
      BE PRINTABLE AND NOT A DELIMITER ( . , * / ).
""
ACCT-205
INVALID COMMAND
ENTER "HE" TO GET THE COMMANDS.

""
ACCT-206
INVALID FILE NAME.

""
ACCT-207
INVALID CAPABILITY
CAPABILITY MUST BE BETWEEN 1 AND 63.

""
ACCT-208
INVALID DISK LIMIT
ONLY 60 DISKS ARE ALLOWED.

""
ACCT-209
INVALID SST ENTRY
SESSION LU MUST BE GREATER THAN 3 AND LESS THAN 64
SYSTEM LU MUST BE GREATER THAN 0 AND LESS THAN 255
SESSION LU IS ALREADY DEFINED.

""
ACCT-210
CONFLICT IN SST DEFINITION
USER AND GROUP SST'S DISAGREE.

""
ACCT-211
USER OR GROUP ID NOT AVAILABLE
ENTER "LIST,USER,.,6,ID" TO FIND
LARGEST GROUP ID AND SMALLEST USER ID.
PURGE AND REBUILD THE CONFLICTING ACCOUNT(S).

""
ACCT-212
INVALID NUMBER OF SST SPARES
MUST BE BETWEEN 0 AND 60
SPARES PLUS DISK LIMIT MUST BE LESS THAN 68.
```

```
""
ACCT-213
INVALID MEMORY REQUEST
MEMORY REQUEST MUST BE BETWEEN 70 AND 7000 WORDS.

""
ACCT-215
LIST NAMR IN TRANSFER STACK
REISSUE TR COMMAND.

""
ACCT-216
ILLEGAL RESPONSE FOR PRIMARY PROGRAM.
MUST BE FMGR OF CI.

""
ACCT-217
THE NUMBER OF UDSPS OR DEPTH VALUE ENTERED IS ILLEGAL
(VALUES MUST BE BETWEEN 0 AND 8, AND IF ONE IS ZERO, THEY MUST BOTH
BE ZERO.)

""
ACCT-218
SESSION NOT SHUT DOWN
SESSION MUST SHUT DOWN FOR LOAD,<NAMR>

""
ACCT-219
NOT ENOUGH ROOM IN FILE FOR NEW TABLE
ENTER "LO,0" TO EXPAND FILE.

""
ACCT-220
CORRUPT STATION TABLE SPARES
MUST BUILD ACCOUNTS FILE FROM SCRATCH.

""
ACCT-221
NOT AN ACTIVE SESSION
THE SESSION ADDRESSED IS NOT ACTIVE.

""
ACCT-222
ILLEGAL SYSTEM LU
SYSTEM LU MUST BE BETWEEN 1 AND 255.

""
ACCT-223
ILLEGAL SHUT DOWN PARAMETER SHUT DOWN OPTIONS ARE:

   1)"SD"     SHUT DOWN THE SESSION SYSTEM
   2)"SD,RE"  SHUT DOWN THE SESSION SYSTEM AND RELEASE SESSION MEMORY
   3)"SD,
""
ACCT-225
SESSION MEMORY CAN NOT BE RETURNED TO SYSTEM (REBOOT).
```

```
""
ACCT-046
GREATER THAN 255 EXTENTS
ATTEMPT TO CREATE EXTENT 256.  MAKE FILE SIZE OF MAIN LARGER.

""
ACCT-099
DIRECTORY MANAGER EXEC REQUEST WAS ABORTED
EXEC REQUEST MADE BY D.RTR WAS ABORTED.  MAKE SURE THAT ALL DISKS
BEING ACCESSED ARE UP.  NOTIFY SYSTEM MANAGER.

""
ACCT-041
NO ROOM IN SST.

""
ACCT-040
LU NOT FOUND IN SST
ATTEMPT TO ACCESS AN LU THAT IS NOT IN YOUR SST.
USE THE SL COMMAND TO ADD THE LU TO THE SST.

""
ACCT-039
CONFLICT IN SST DEFINITION.

""
ACCT-035
63 DISKS ALREADY MOUNTED TO SYSTEM
ATTEMPT TO MOUNT A DISK WHEN THERE ARE ALREADY 63 DISKS MOUNTED.
A DISK WILL HAVE TO BE DISMOUNTED BEFORE A NEW ONE MAY BE MOUNTED.

""
ACCT-034
DISK ALREADY MOUNTED.
ATTEMPT TO MOUNT A DISK THAT IS ALREADY MOUNTED ON THE CARTRIDGE
LIST.
EITHER DISMOUNT THE DUPLICATE DISK OR MOUNT A DIFFERENT ONE.

""
ACCT-033
NOT ENOUGH ROOM ON CARTRIDGE
ATTEMPT TO ACCESS A CARTRIDGE WHICH HAS NO MORE ROOM.  TRY USING
ANOTHER CARTRIDGE OR DECREASE THE FILE SIZE.

""
ACCT-032
CARTRIDGE NOT FOUND
ATTEMPT TO ACCESS A CARTRIDGE THAT CANNOT BE FOUND IN THE CARTRIDGE
LIST.
CHECK THE CARTRIDGE NUMBER FOR CORRECTNESS.

""
ACCT-030
VALUE TOO LARGE FOR PARAMETER
CHECK THE COMMAND AND RE-SIZE THE PARAMETER.
```

""
ACCT-026
QUEUE FULL OR MAX PENDING SPOOLS EXCEEDED
THE SPOOL QUEUE IS FULL OR THE MAXIMUM NUMBER OF PENDING SPOOLS HAS BEEN EXCEEDED. THE JOB MUST BE RE-RUN WHEN THE SPACE BECOMES AVAILABLE.

""
ACCT-025
NO SPLCON ROOM THE SPLCON IS FULL.
THIS ERROR MAY OCCUR WHEN THE SPOOL SYSTEM IS COMPETING WITH PROGRAMS USING THEIR OWN SPOOLING FILE AND RUNNING OUTSIDE OF BATCH.

""
ACCT-024
NO MORE BATCH SWITCHES
THE LU SWITCH TABLE IS FULL. THE SIZE OF THE SWITCH TABLE SPECIFIED AT SYSTEM GENERATION IS INADEQUATE. NOTIFY THE SYSTEM MANAGER OF THIS CONDITION.

""
ACCT-023
NO AVAILABLE SPOOL FILES
ALL SPOOL FILES ARE CURRENTLY BEING USED. RE-RUN THE JOB WHEN A SPOOL FILE BECOMES AVAILABLE.

""
ACCT-022
NO AVAILABLE SPOOL LU'S
ALL SPOOL LOGICAL UNITS ARE CURRENTLY UNAVAILABLE. RE-RUN THE JOB WHEN A SPOOL LU BECOMES AVAILABLE.

""
ACCT-021
ILLEGAL DESTINATION LU
THE LU SPECIFIED WAS NOT ALLOCATED BY GASP. TRY AGAIN USING AN LU ALLOCATED BY GASP.

""
ACCT-020
ILLEGAL ACCESS LU

1. THE LOGICAL UNIT NUMBER SPECIFIED IN THE LU OR CS COMMAND WAS NOT A POSITIVE LOGICAL UNIT NUMBER. RE-ENTER THE CORRECTED COMMAND.

OR

2. THERE IS AN LU ENTRY IN THE CARTRIDGE LIST THAT DOES NOT POINT TO A DISK DEVICE. THE LU COMMAND WAS USED TO DO A LOGICAL UNIT SWITCH ON THE DEVICE AFTER THE DISK WAS MOUNTED. SWITCH THE LU BACK TO ITS DISK DEFINITION. IF DESIRED, DISMOUNT THE DISK. THE LU CAN THEN BE SWITCHED TO A NON-DISK DEVICE.

""
ACCT-019
ILLEGAL ACCESS ON A SYSTEM DISK
ATTEMPT TO WRITE ON A SYSTEM DISK. THE SYSTEM MANAGER IS THE ONLY USER WITH THIS CAPABILITY.

""
ACCT-018
ILLEGAL LU; LU NOT ASSIGNED TO SYSTEM
ATTEMPT TO ACCESS AN LU THAT IS NOT ASSIGNED TO THE SYSTEM.

""
ACCT-017
ILLEGAL READ/WRITE ON TYPE 0 FILE
ATTEMPT TO READ, WRITE, OR POSITION A TYPE 0 FILE THAT DOES NOT SUPPORT THE OPERATION. CHECK THE FILE PARAMETERS OR THE NAMR.

""
ACCT-016
ILLEGAL TYPE 0, OR FILE BLOCKSSIZE=0
ONE OF THE FOLLOWING OCCURRED:

1) THE WRONG FILE TYPE WAS SPECIFIED,
2) ATTEMPT TO CREATE OR PURGE A TYPE 0 FILE, OR
3) THE SIZE SPECIFIED WAS ZERO BLOCKS. CHECK THE SIZE AND TYPE PARAMETERS.

""
ACCT-015
ILLEGAL NAME
THE FILE NAME DOES NOT CONFORM TO THE SYNTAX RULES. CORRECT THE NAME AND RE-ENTER THE COMMAND.

""
ACCT-014
DIRECTORY FULL
THERE IS NO MORE ROOM IN THE FILE DIRECTORY. PURGE UNUSED FILES AND PACK THE DISK IF POSSIBLE. OTHERWISE, TRY ANOTHER CARTRIDGE

""
ACCT-013
DISK LOCKED
THE CARTRIDGE SPECIFIED IS LOCKED. INITIALIZE THE CARTRIDGE IF IT WAS NOT INITIALIZED, OTHERWISE KEEP TRYING.

""
ACCT-012
EOF OR SOF ERROR
ATTEMPT TO READ, WRITE, OR POSITION A FILE BEYOND THE FILE BOUNDARIES. CHECK THE RECORD POSITION PARAMETERS. THE RESULTS DEPENDS ON THE FILE TYPE AND THE CALL.

""
ACCT-011
DCB NOT OPEN
ATTEMPT TO ACCESS AN UNOPENED DCB. USE THE CREATE OR OPEN CALL TO OPEN THE DCB AND CHECK FOR ERRORS.

""
ACCT-010
NOT ENOUGH PARAMETERS
ONE OR MORE OF THE REQUIRED PARAMETERS WERE OMITTED FROM THE CALL. ENTER THE REQUIRED PARAMETERS.

""
ACCT-009
ATTEMPT TO USE APOSN OR FORCE A TYPE 0 FILE TO TYPE 1
A TYPE 0 FILE CANNOT BE POSITIONED WITH APOSN OR BE FORCED TO A TYPE 1 FILE.

""
ACCT-008
FILE OPEN OR LOCK REJECTED
ATTEMPT TO OPEN A FILE THAT WAS ALREADY OPENED EXCLUSIVELY, WAS ALREADY OPENED TO EIGHT PROGRAMS, OR THE CARTRIDGE CONTAINING THE FILE IS LOCKED. USE THE CL OR DL COMMAND TO LOCATE THE LOCK. IF THE FILE IS BEING PACKED, CHECK TO SEE IF SPOOLING IS SHUT DOWN.

""
ACCT-007
ILLEGAL SECURITY CODE OR ILLEGAL WRITE ON LU2 OR LU 3

1. ATTEMPT TO ACCESS A FILE WITHOUT SPECIFYING THE SECURITY CODE OR WITH THE WRONG SECURITY CODE. USE THE CORRECT CODE OR DO NOT ACCESS THE FILE.

OR

2. ATTEMPT BY A SESSION USER TO WRITE ON LU 2 OR LU 3. ONLY THE FILE MANAGER HAS WRITE ACCESS TO LU 2 OR LU 3.

""
ACCT-006
FILE NOT FOUND
ATTEMPT TO ACCESS A FILE THAT CANNOT BE FOUND. CHECK THE FILE NAME.

""
ACCT-005
RECORD LENGTH ILLEGAL
ATTEMPT TO READ OR POSITION A FILE TO A RECORD THAT HAS NOT BEEN WRITTEN, OR TO WRITE AN ILLEGAL RECORD LENGTH ON AN UPDATE. CHECK THE FILE POSITION OR SIZE PARAMETER.

""
ACCT-004
MORE THAN 32767 RECORDS IN A TYPE 2 FILE
ATTEMPT TO CREATE A TYPE 2 FILE WITH TOO MANY RECORDS OR WITH A RECORD SIZE THAT IS TOO LARGE. CHECK THE SIZE PARAMETER.

""
ACCT-003
BACKSPACE ILLEGAL
ATTEMPT TO BACKSPACE A DEVICE (OR TYPE 0 FILE) THAT CANNOT BE BACKSPACED. CHECK THE DEVICE TYPE.

""
ACCT-002
DUPLICATE FILE NAME
A FILE ALREADY EXISTS WITH THE NAME SPECIFIED. REPEAT THE COMMAND WITH A NEW NAME OR PURGE THE EXISTING FILE.

```
""
ACCT-001
DISK ERROR THE DISK IS DOWN.
TRY AGAIN AND THEN REPORT THE PROBLEM TO THE SYSTEM MANAGER.
```

Table 8-4. LOGON Error Summary

Error Code	Meaning
LGON 00	SESSION ENVIRONMENT NOT INITIALIZED
LGON 01	FMP ERROR ON ACCOUNT FILE ACCESS
LGON 03	SESSION LIMIT EXCEEDED
LGON 04	NO SUCH USER
LGON 05	ILLEGAL ACCESS
LGON 06	CONFLICT IN DEFINITION OF SESSION LU
LGON 07	NO ROOM FOR SESSION CONTROL BLOCK
LGON 08	DUPLICATE SESSION IDENTIFIER
LGON 09	SST OVERFLOW
LGON 10	NO FREE ID SEGMENTS OR FMGR NOT FOUND
LGON 11	FMP ERROR ON DISK MOUNT ATTEMPT
LGON 12	ACCOUNT FILE CORRUPT
LGON 13	CONFLICT WITH SYSTEM DISC LU
LGON 14	BAD JOB LOG-ON REQUEST
LGON 15	SESSION PRIMARY PROGRAM NOT FOUND
LGON 16	NUMBER OF UDSP'S OR DEPTH INVALID

LGON Help File Entries

```
""
LGON 00
SESSION ENVIRONMENT NOT INITIALIZED.
THE CLASS GET FAILED ON LOGON'S CLASS NUMBER OR THE RESOURCE NUMBER
LOCK FAILED.  THESE CONDITIONS INDICATE THAT THE SESSION ENVIRONMENT
HAS NOT BEEN INITIALIZED BY THE ACCTS PROGRAM.

""
LGON 01
FMP ERROR - XXXXX ON ACCOUNT FILE ACCESS.
AN FMP ERROR HAS OCCURRED WHILE TRYING TO OPEN THE ACCOUNTS FILE.
DETERMINE THE SOURCE OF THE FMP ERROR AND CORRECT THE SITUATION.

""
LGON 03
SESSION LIMIT EXCEEDED.
THE SESSION LIMIT REPRESENTS THE MAXIMUM NUMBER OF ACTIVE SESSIONS
ALLOWED ON THE SYSTEM AT ANY ONE TIME.  CONTACT THE SYSTEM MANAGER
WHO MAY WISH TO INCREASE THE SESSION LIMIT.

""
LGON 04
NO SUCH USER.
THIS USER ACCOUNT DOES NOT EXIST.

""
LGON 05
ILLEGAL ACCESS.
AN INCORRECT PASSWORD WAS GIVEN WHEN ATTEMPTING TO LOG-ON TO AN
ACCOUNT.

""
LGON 06
CONFLICT IN DEFINITION OF SESSION LU.
THIS IS AN INFORMATIONAL DIAGNOSTIC. THE STATION (TERMINAL) BEING
LOGGED ONTO HAS A CONFIGURATION TABLE ENTRY WHICH IS A DUPLICATE OF
AN ENTRY IN YOUR ACCOUNT FILE ENTRY.  IF YOU HAVE THE CAPABILITY TO
MAKE CHANGES IN THE SESSION SWITCH TABLE (SL,X,Y), BOTH THE
CONFIGURATION TABLE AND YOUR ACCOUNT FILE DEFINITION (OF THE SESSION
LU) ARE REPORTED.  CONTACT THE SYSTEM MANAGER TO HAVE THE CONFLICT
REMOVED.

""
LGON 07
NO ROOM FOR SESSION CONTROL BLOCK.
THERE IS INSUFFICIENT SAM FOR YOUR SCB.  CONTACT YOUR SYSTEM MANAGER
WHO MAY NEED TO INCREASE THE SIZE OF SAM.

""
LGON 08
DUPLICATE SESSION IDENTIFIER.
```

""
LGON 09
SST OVERFLOW.
YOUR SESSION HAS EXCEEDED THE MAXIMUM SESSION SWITCH TABLE SIZE. THE OVERFLOW WAS DETECTED IN ONE OF THE FOLLOWING AREAS: BUILDING THE SST ENTRIES DEFINED BY THE USER'S ACCOUNT ENTRY, BUILDING SST ENTRIES DEFINED BY THE STATION CONFIGURATION TABLE OR MOUNTING SYSTEM GLOBAL DISKS. CONTACT YOUR SYSTEM MANAGER AS YOU MAY BE MISSING DEVICE DEFINITIONS.

""
LGON 10
NO FREE ID SEGMENTS OR FMGR NOT FOUND.
WHILE TRYING TO MAKE A COPY OF YOUR PRIMARY PROGRAM EITHER THERE WERE NO ID SEGMENTS AVAILABLE OR THE PRIMARY PROGRAM COULD NOT BE DUPLICATED FOR SOME REASON. THE FIRST CASE IS USUALLY THE PROBLEM, IT INDICATES THAT NOT ENOUGH FREE ID SEGMENTS ARE IN THE SYSTEM. EITHER RP LESS PROGRAMS OR REGENERATE YOUR SYSTEM WITH MORE ID SEGMENTS.

""
LGON 11
FMP ERROR ON DISK MOUNT ATTEMPT.
THE LOGON PROGRAM RECEIVED THE SPECIFIED ERROR WHEN ATTEMPTING TO MOUNT A PRIVATE OR GROUP DISK TO THIS SESSION. CHECK THE TERMINAL USER'S MANUAL (ERROR SUMMARY) FOR MORE INFORMATION.

""
LGON 13
CONFLICT WITH SYSTEM DISK LU.
THIS IS AN INFORMATIONAL DIAGNOSTIC. LOG ON DETECTED A USER SST WHICH ATTEMPTED TO REDEFINE A SYSTEM DISK'S LOGICAL UNIT NUMBER. DISK LU'S MUST BE DIRECT MAPS (SESSION LU=SYSTEM LU). CONTACT YOUR SYSTEM MANAGER TO CORRECT YOUR ACCOUNT.

""
LGON 14
BAD JOB LOG-ON REQUEST.

""
LGON 15
SESSION PRIMARY PROGRAM CI.XX NOT FOUND.
WHILE TRYING TO SET UP CI.XX AS THE PRIMARY PROGRAM FOR THIS SESSION, THE PROGRAM CI COULD NOT BE FOUND. IF CI WAS NOT LOADED PERMANENTLY, IT MAY HAVE BEEN REMOVED. EITHER RP CI OR LOAD CI PERMANENTLY.

```
""
LGON 16
LOGON HAS FOUND THE NUMBER OF AND/OR DEPTH OF THE UDSP'S TO BE TOO
LARGE.  INFORMATION IN THE ACCOUNTS FILE IS ERRONEOUS.  THIS CAN
OCCUR USING AN OLDER ACCOUNTS FILE (+@CCT!) FROM SYSTEMS THAT DID
NOT SUPPORT UDSP'S.  LOGON SETS THE NUMBER AND DEPTH TO 0/0 AND
PROCEEDS WITH THE LOGON.  HAVE YOUR SYSTEM MANAGER UPDATE YOUR USER
ACCOUNT ENTRY WITH THE ACCTS PROGRAM AND ALTER THE NUMBER/DEPTH OF
THE UDSP'S.
```

9

Adjusting System Parameters

Introduction

Certain system and FMGR commands can be used to enable your system to meet specific requirements of your installation. The overall effects of these commands are described below. For a discussion of command syntax and operation, refer to the RTE-6/VM Terminal User's Reference Manual.

Device Control

The TO command can be used to set EQT time-out values for device controllers. EQT time-out values are initially set during system generation. The TO command can be used to correct generation values and adjust time-outs after operating experience is gained with your system. EQT time-outs are used to place a time limit on an I/O request once it is sent to a device driver. When the time limit expires, the system either sets the device down or informs the driver of the time-out.

For example, time-outs are frequently associated with terminals in order to limit the amount of time programs may wait for commands or data from the keyboard. The Session Monitor automatically logs you off your terminal after five consecutive time-outs.

EQT time-out settings depend on both the device and the associated driver. Unless you have a good reason to do otherwise, it is recommended that EQT time-out be set according to the values given in this manual (refer to the Device Configuration section in Chapter 4) and other subsystem manuals and configuration guides.

The LU command can be used to alter system Logical Unit/EQT Subchannel relationships. LU commands are frequently used to:

- Correct generation errors. If device LU assignments were incorrectly specified during generation, you can fix these definitions at system startup by putting the appropriate LU commands in the WELCOM file.
- Configure new devices into the system. The LU command can assign unused LU numbers to new devices attached to an existing controller. It is recommended that you generate spare LU numbers in your system just for this purpose (refer to the I/O Structure Planning section in Chapter 4).

- Change device control parameters. Certain drivers (for example, DVR00) obtain device control information from the LU subchannel definition. The LU command can redefine the subchannel to accommodate changing application requirements.

The EQ command can be used to enable/disable automatic output buffering to selected devices. When this feature is enabled for a device, output data will be buffered in SAM before it is sent to the device. When a program does standard output to a device without output buffering, the program must wait for the output request to complete before resuming execution. In addition, if the program is partition resident, it is locked into that partition for the duration of the output request, that is, the partition will be unavailable for other higher priority tasks. On the other hand, if automatic output buffering is enabled, these restrictions are lifted. Therefore programs may perform output operations without waiting for device completion and they may be swapped at any time.

It is advantageous to enable output buffering on devices with very slow output rates relative to the rate of output requests. Typical peripherals in this category are line printers, terminals, paper tape punches, etc. Magnetic tape peripherals would also fall into this category for control operations such as rewind and file skip. By buffering these devices, you may significantly improve system throughput and resource utilization. Certain devices, such as disks, must NOT be buffered.

The decision whether to enable buffering on a device must be balanced with considerations of your systems SAM requirements. Buffering output requests through SAM will reduce the amount available for other operations (for example, class I/O, reentrant processing, scheduling strings, etc.). To keep a program from monopolizing SAM with buffered output requests, the system enforces upper and lower limits on the amount of memory queued on any I/O device. When a program makes an I/O request, the system sums up all output requests already on the device I/O request queue. If this sum exceeds the high buffer limit, the program is suspended. Suspended programs are not rescheduled until the queued memory drops below the lower limit.

The upper and lower buffer limits are initially set during system generation, but can be modified on-line with the BL command. For optimum operation, the differential between the high and low buffer limits should be set such that once a program is suspended on buffer limits, there will be enough time for lower priority programs to do useful work before the low limit is reached. The low limit should be set high enough so that there is sufficient data to keep the device busy until the rescheduled program can issue a new I/O request. For example, it might have to be swapped in from disk. Remember, these considerations must be tempered with the availability of SAM in your system.

It is recommended that your system high and low buffer limits be set initially to 100 and 300, respectively. After your system is operational with the intended mix of applications, run performance tests with different limits to find the optimum settings.

Timeslicing

The System Manager can control timeslicing in the following ways:

- Modify the system timeslice Quantum Multiplier (QU command).
- Modify the system timeslice Priority Fence (QU command).
- Modify a specific program's timeslice level (PR command).

All programs competing for the central processor (CPU) access it according to their order in the scheduled list. Programs are placed in the scheduled list in order of their priority. Within priorities, scheduling can be performed in a linear or circular fashion. (Refer to the RTE-6/VM Programmer's Reference Manual for a detailed description of linear and circular scheduling.) The scheduled list is divided into two parts. Those programs with priority numbers less than the priority fence will be scheduled in linear fashion. Programs with priority numbers greater than or equal to the fence will be scheduled in circular fashion.

Program priority levels should be set such that real time, response time critical tasks have priority numbers below the Priority Fence (i.e., they will be linear scheduled). Background tasks should have priority numbers above the fence. To make most effective use of timeslicing, programs performing similar types of functions should have identical priority numbers. This will allow them to compete more evenly for CPU time. You might want to establish several standard priority levels for different types of functions. For example, highly interactive programs such as editors and data entry processors might be assigned priority level 50, and less interactive programs, such as FMGR and BASIC, level 90; and highly compute bound tasks might be assigned level 300.

RTE-6/VM Operating System gives timesliced programs a full execution slice (quantum) when: they are initially scheduled; they are rescheduled after leaving the scheduled list (due to I/O suspend, buffer limit suspend, etc.); or they have exhausted their current quantum. In all three cases, the program is placed in the scheduled list after all programs of the same priority, thereby allowing programs to execute on a round-robin basis. The maximum quantum given to a program is calculated as follows:

Max Quantum = Quantum Multiplier * (prog priority/256+1)

For programs with a priority level of between 0 and 255, the maximum quantum is equal to the quantum multiplier. For programs with a priority level of 256 to 511, the maximum quantum is 2 * the quantum multiplier and so on. This algorithm gives lower priority (higher priority numbered) programs a longer execution slice as they are assumed to execute less frequently.

The primary advantage of timeslicing programs is to prevent programs from monopolizing available CPU time. If the Quantum Multiplier is set low enough (e.g., less than .25 seconds), it can give users the illusion of a dedicated processor. However, keep in mind that as the quantum multiplier is decreased, the system may dispatch programs for execution more often. If enough memory partitions exist to hold all active timesliced programs, this extra system overhead is minimal (it basically involves a switch in user maps). However, if there are more timesliced programs than available partitions, the system overhead involved to switch between scheduled programs can increase substantially due to disk swapping. In this case, the Quantum Multiplier should be relatively large (e.g., greater than 1 sec) or timeslicing can be turned off altogether (i.e., QU,0,32767). It is recommended that, after the system is operational, you run performance tests to find a quantum multiplier acceptable to system users. This value can be adjusted as your memory configuration changes.

Note The system makes no attempt to adjust the partition list so that swapping is evenly distributed over a set of programs competing for the same partitions. Therefore it is possible for timesliced programs to receive unequal shares of CPU time even though they have identical priority levels.

System Console as a Session Terminal

In many installations it may not be desirable to operate the system console as a session terminal since users on LU 1 will see system messages unrelated to their session. However, if you need session operation from LU 1, the system console can be enabled with the EN command. (It is recommended that the security code option be specified here, i.e., EN,sc,1).

After the system console is enabled as a session terminal, commands can be routed directly to the operating system (instead of the Session Monitor break mode processors) with OP commands. In addition, OP suppresses command checking on the supplied command. For example, to abort the current batch job, from the system console when enabled as a session terminal, enter:

```
S=01 COMMAND?  OP,sc,AB,1
```

Where sc is the master security code (required if EN,sc,1 was specified.) This command has the same effect as:

```
*AB,1
```

when entered in non-session mode. Note that OP commands can be entered from any capability level session. System security is maintained through specification of the master security code.

To convert the system console back to non-session operation run the ACCTS program and enter the SD,0 command (Chapter 8). Note that this command does not shut down any session that is currently active on LU 1. It merely converts break mode to non-session operation. Use SD,1 to shut down the session on LU 1.

Under certain circumstances, the user may receive the following break mode prompt on the system console when operating in session mode:

```
S=?? COMMAND? OP,
```

This prompt is issued when LOGON or RPN are already processing a break mode request from LU 1 and an additional break mode interrupt is made. This can occur if:

1. LOGON or RPN are busy processing other requests.
2. LOGON or RPN have been temporary shut down by the ACCTS program (for example, during a LOAD,0).
3. LOGON or RPN cannot process inputs due to unavailable system resources (for example, SAM, swap tracks, partitions, etc.).

4. LOGON or RPN have been aborted and permanently purged from the system.

For the first two cases, it is suggested that you wait a short period and try again. If you still receive the special prompt, there are probably more serious system problems. At this point it is suggested that the system console be changed back to non-session mode.

Example:

```
S=01  COMMAND? WH                (User issues WH command)
                                 (No response)
S=??  COMMAND? OP, <cr>          (Tries again – gets special prompt and types
                                 RETURN)
S=??  COMMAND? OP,sc,RU,ACCTS    (The System Manager runs ACCTS)
PASSWORD?                        (enters password)
NEXT?
SD,0                             (disables session break mode.
                                 Session 1 is still active however)
NEXT?
EXIT                             (terminates ACCTS)
END ACCTS

*RU,WHZAT,1                      (runs WHZAT in non-session mode)
    .
    .                            (Corrects problem)
    .
*EN,sc,1                         (reenables system console for session)
```

Partition Management

The AS command assigns programs to specific partitions. This may be done for a variety of reasons:

- To keep programs from contending for the same partitions, you can assign each to a different partition.
- For response time critical tasks, programs can be assigned to previously reserved partitions. If there is only one program assigned to a partition, the program is effectively made memory-resident.

The AG command allows high-priority suspended (state 3) programs to be swapped out in favor of lower priority programs that are scheduled but waiting for a partition.

The UR command can be used to release a previously reserved partition. This allows programs not specifically assigned to this partition to run in it. Partitions cannot be reserved on line. They may be reserved only during system generation or reconfiguration.

The SZ command can be used to change the minimum partition size a program can run in. Certain HP supported programs use the space between the end of the program and the partition for buffer areas (refer to the Utility Loading Considerations section in Chapter 5).

```
CHANGING THE MASTER SECURITY CODE
```

The system master security code can be changed with the following file manager command:

```
:IN,ol--nw
```

where:

ol	is the old master security code.
nw	is the new code.

In the session environment, this command requires a command capability level of 60. It is STRONGLY advised that the system master security code not be publicized, as it gives users access to all file security codes (which in turn gives access to all system files).

System Scratch and VMA Cartridge

When a program creates a file using CREAT, ECREA, or CRETS it can specify −32768 as the LU where the file will reside. This is a flag to the FMP create routines to create the file on the cartridge specified with a VL command or on the first cartridge in the user's cartridge list. The VMA system, languages and many system utilities use this method of creating scratch files.

Since scratch files are frequently used, the VL command can be used to designate a system scratch cartridge. This prevents private and group cartridges from being filled with scratch files and allows users with small private or group cartridges to run utilities that require large scratch files (VMA backing store files).

For example, a group of users may have a private cartridge at the top of their cartridge lists and they commonly use utilities that create temporary scratch files. To ensure that all users can run all of these utilities, each user must have enough room on their private cartridge for the largest scratch file created by any one of the utilities. This means you are effectively reserving enough space for the largest scratch file times the number of users. This much disk space is not usually needed since all users will not be running the utility requiring the largest scratch file at the same time.

To estimate the required size of the system scratch cartridge, consider the typical system activity and use of programs that require very large scratch files such as certain VMA programs. The scratch cartridge should be large enough for the typical system use and large enough for the largest scratch file that will ever reside on it. Also remember that frequent purging of scratch files will fragment the available disk space and that the file management system cannot utilize 100% of the disk space. Disk space utilization is improved by periodically packing the scratch cartridge. On a typical system, 200 tracks plus space for VMA files should suffice.

The following language processors and utilities place scratch files on a system scratch cartridge if one is specified.

Program	**Scratch File Size**
VMA	Default backing store file is created if needed. File size is 256 blocks plus extents that are created when needed.
Edit/1000	File to hold working copy of source. The file is 256 blocks and enlarged with extents to hold source. Refer to the Edit/1000 User's Guide for information concerning placement of edit scratch files.
Macro/1000	File to hold tables and source. One file is created equal to the size of the source and a table file of 92 blocks is created and enlarged with extents for approximately every 2000 lines of source code.

10

Memory and I/O Reconfiguration

General

The ability to reconfigure the I/O and memory assignments during system boot-up without going through a complete, new system generation is a feature of the RTE-6/VM operating system. The reconfiguration option is exercised during system boot-up through S-Register settings (described below) in order to postpone completion of the boot-up process and schedule an interactive Configurator program that performs the desired I/O and/or memory reconfiguration.

I/O reconfiguration is performed by user reassignment of I/O devices to octal select codes other than those assigned at system generation time.

Memory reconfiguration includes changing the size of the System Available Memory (SAM) extension, redefining user partitions, modifying program page requirements, defining shared EMA data area, and assigning programs to partitions. Defective pages in memory (pages with parity errors) can be avoided by using the Configurator to redefine the SAM extension and user partitions around the defective pages.

I/O and memory reconfigurations (either or both) can be made permanent by changing the system on the disk.

Scheduling the Configurator from Disk Loader ROM

If a disk loader ROM is used to load the boot-extension into memory during system boot-up, the Configurator can be scheduled by setting bit 5 of the S-Register, in addition to the S-Register settings for the disk loader ROM.

The disk loader ROM used for each type of disk is as follows:

MAC RPL compatible	– HP 12992B
ICD RPL compatible	– HP 12992H
CS/80 RPL compatible	– HP 12992J

The Boot Extension is assumed to reside on physical track 0, sector 0 of the system disk. Standard boot-up procedures can be found in the 12992 Loader ROM Installation Manual. In the following procedure, differences between CS/80 and other disks will be indicated. Otherwise, the information applies to all disks.

1. Select the S-Register for display on the computer front panel.

2. Press CLEAR DISPLAY

3. Set the S-Register as follows:

Bits	Enter
0-2	Surface number for MAC/ICD disk, and unit number for CS/80 disk.
3-4	0 (reserved).
5	1 for reconfiguration (0 for no reconfiguration).
6-11	Disk select code.
12	1 for MAC/ICD disk, 0 for CS/80 disk.
13	0 (reserved).
14-15	Loader ROM selection for disk.

   Press STORE

4. Press PRESET, IBL, PRESET (again, this resets parity error logic if set by IBL) and RUN to load the contents off the disk. A successful load will be indicated when the HLT 77B occurs. If bit 5 was set for reconfiguration, proceed to step 5. If bit 5 was not set, the system SET TIME message appears on the system console.

5. Select and CLEAR the S-Register, then set as follows:

Bits	Enter
0-5	Console select code, or 0 if the console select code has not changed from previously defined system.
6-11	Disk select code, or 0 if the disk select code has not changed from previously defined system.
12-14	0 (reserved).
15	1 for reconfiguration.

6. Press STORE, PRESET, and RUN to perform reconfiguration.

Scheduling the Configurator from Bootstrap Loader

A Bootstrap Loader can be used to load the Boot Extension into memory. For example, the boot file such as the 264x Cartridge Tape Loader ROM (12992C) may be used. To load from a bootstrap loader, proceed as follows:

1. Select the S-Register for display on the computer front panel.
2. Press CLEAR DISPLAY.
3. Set the S-Register as follows:

Bits	Enter
0-5	0 (reserved).
6-11	Input device select code.
12-13	0 (reserved).
14-15	Loader ROM selection for input devices.

4. Press STORE, PRESET, IBL, PRESET, and then RUN. Boot file will be loaded from the device specified.
5. When HLT 77B occurs, set the P-Register to 100B.
6. Press STORE.
7. For MAC/ICD disks, set the S-Register to zero to signify no change. For CS/80 disks, select and CLEAR the S-Register, then set as follows:

Bits	Enter
0-5	Console select code.
6-11	Select code of the disk.
12-14	0 (reserved).
15	1 for reconfiguration.

8. Press STORE, PRESET, and RUN to perform reconfiguration.

Configurator Program

The Configurator works interactively to change the current I/O and memory configurations. Reconfiguration is performed in accordance with user responses to a series of Configurator prompts and queries displayed on the system console. When reconfiguration is completed, the Configurator queries whether it is to be made permanent. Boot-up of the RTE-6/VM system is then completed in accordance with the proper entry.

The Configurator consists of two modules, $CNFG and $CNFX. $CNFG is located at the end of the system modules. After configuration has completed, the memory area occupied by $CNFG is allocated to SAM. $CNFX is used to reconfigure memory and is a Type 3 disk-resident program, brought into the user partition area from disk by the $CNFG program. $CNFG changes $CNFX's program name to ",,,,," and therefore $CNFX cannot be run online.

The Configurator program first checks the contents of the S-Register. If bit 15 is set, I/O and memory reconfiguration are performed. The system is reconfigured in accordance with any specified new disk and console select codes. Entering invalid disk and console select codes in the S-Register will cause the system not to function properly. The Configurator then loads the driver partitions, memory-resident library and memory-resident programs (if they are defined for the system) into memory.

If bit 15 is not set in the S-Register, control is given to the operating system.

Reconfiguration is performed interactively by using the system console and list device. Note that the standard method of getting system attention by pressing any key on the system console does not work during reconfiguration, since the system is not yet completely initialized. The bootup procedure must therefore be restarted if any equipment I/O errors occur (e.g., a device not ready or a parity error).

Configurator Halts and Error Messages

Various halts and Configurator error messages may occur during system boot-up or reconfiguration that require corrective action by the operator. Halts are displayed on the computer front panel. System boot-up and configuration halts, their meaning and required operator action are itemized in Table 10-1 at the end of this section.

Whenever an invalid response to a Configurator prompt or query is entered, the Configurator issues an error message in the form:

```
CONFIG ERR xx
```

where xx is a Configurator error code as defined in Table 10-2 at the end of this section. Following the error message, the Configurator usually repeats the prompt or query and the user need only enter the correct response. In the reconfiguration procedures given below, only error recovery procedures requiring further action are described in text.

Reconfiguration Procedures

Perform the following steps if reconfiguration of the system console is requested by entering the new select code in bits 0-5 of the switch register.

1. If the new system console has the same driver type as the old system console, then just point the old system console EQT entry to the new select code.
2. If the new system console needs a different driver type, then scan the EQTs to find a matching driver type and the new select code. Use this EQT number for the new system console. No change is made in the I/O configuration of the old system console.
3. If an EQT with matching driver type and the new select code is not found, then scan EQTs to find one with a matching driver type. First such EQT encountered is used for the new system console. The select code that this EQT previously pointed to is the old select code.

The Configurator begins the reconfiguration process by first displaying the message:

```
START  RECONFIGURATION
```

on the system console, and followed by a set of queries to which the user enters responses on the console keyboard. The Configurator will redisplay a query if the user response is not what was expected. The Configurator next displays the query:

```
LIST DEVICE LU#?
```

Enter a Logical Unit number to which the Configurator can direct listings or press the space bar and RETURN key on the console keyboard for the default case, which is the system console. Entering a list device other than the system console causes the Configurator to display the following message:

```
LIST DEVICE SELECT CODE#?
```

Enter a list device select code or press the space bar and RETURN key for the default case, where the default is the list device select code configured into the system.

If the entered list device was not the system console, the Configurator displays the query:

```
ECHO? (YES/NO)
```

Enter YES to have all output to the list device echoed on the system console.

An additional message is displayed after the list device select code is entered during reconfiguration. This message is displayed only if the disk, system console, or list device select codes were entered into the switch register:

```
I/O RECONFIGURATION ALREADY PERFORMED:
CURRENT SELECT CODE #, NEW SELECT CODE #
21, 13        *SYSTEM DISK
20, 20        *LIST DEVICE
```

Note: No entry was made in the switch register for system console select code.

When the system console is reconfigured, an attempt is made to reuse the old system console's EQT. If the driver types for the old and new system consoles are not different, then the EQT for the old system console is reused and the following message printed:

```
X,X   *SYSTEM CONSOLE
```

where X is the select code for the new console.

However, if the driver types are different, then a new EQT is used and the old select code is printed:

```
X,Y   *SYSTEM CONSOLE
```

where:

X	is the select code of the former system console,
Y	is the select code of the new console.

I/O Reconfiguration Steps

I/O reconfiguration is performed by assigning the Interrupt Table and trap cell values for the current select code to the corresponding entries for the new select code.

The Configurator first prompts for I/O reconfiguration by displaying a list of the current I/O configuration, beginning with octal select code 10 for the operating system, in the format:

```
CURRENT I/O CONFIGURATION:

                                                  PNAME
SELECT CODE xx = EQTyy            [,TYPE nn[   or   ]]
                                                  nnnnnn

SELECT CODE xx = TBG

                                                  PNAME
SELECT CODE xx = PRIV I/O         [,TYPE nn[   or   ]]
                                                  nnnnnn
```

where:

xx	=	octal select code number ranging from 10 to 77.
EQTyy	=	EQT entry number.
TBG	=	Time Base Generator.
PRIV I/O	=	privileged I/O card.
TYPE nn	=	equipment type code.
PNAME	=	name of program to be automatically scheduled.
nnnnnn	=	absolute instruction to be executed upon interrupt; for example, a JSB LINK,I where LINK contains the entry point address.

The CURRENT I/O CONFIGURATION data is automatically displayed to provide a basis on which to make decisions regarding reconfiguration. If the system disk, system console or the list device were assigned to a new select code, they have already been configured in memory and must NOT be reconfigured during I/O reconfiguration.

The list does not include the select codes previously configured to the system disk, system console, or list device that have been reconfigured via the SWITCH register at bootup. However, these previously-occupied select codes are still available for reassignment. Also, those devices formerly occupying the select codes now reconfigured to the system disk, console, or list device may be reassigned if referenced by their old select code.

Following display of the current configuration, the Configurator then displays the query:

```
I/O  RECONFIGURATION?(YES/NO)
```

Enter NO to bypass I/O reconfiguration. The Configurator skips all further I/O reconfiguration prompts and begins prompting for memory configuration entries (see below).

Enter YES if I/O is to be reconfigured. The Configurator program then displays the message:

```
CURRENT SELECT CODE#,NEW SELECT CODE#?(/E TO END)
-
```

where the hyphen (–) prompts entry of the current and new select code pairs. The current and new select codes response must be in octal and must vary between 10 and 77 octal, in the form:

```
xx,yy
```

followed by a carriage return, where xx is the current select code number and yy is the new select code number. The Configurator hyphen prompt is repeated after each successful entry until a /E is entered to terminate the list.

A privileged I/O card assignment can be removed by entering the current select code number of the privileged I/O card followed by zero, in the form:

```
xx,0
```

where select code 0 is only used to remove the privileged I/O card assignment. A new value of 0 is assigned to the privileged I/O card.

Caution A privileged driver does not work correctly if the privileged I/O card has been removed from the system.

A privileged I/O card can be added to a system that does not have one by entering the specification:

```
xx,PI
```

where xx is the specified select code in octal, and PI assigns the privileged I/O card to select code xx.

If a /R is entered, I/O reconfiguration is restarted with display of the CURRENT SELECT CODE#, NEW SELECT CODE#?(/E TO END) query.

If the current select code number entry is repeated in more than one response, the last entry is taken as valid and the previous entries are ignored.

Following entry of a /E to terminate select code changes, the Configurator prints a list of the NEW I/O CONFIGURATION. The next query displayed is:

```
NEW I/O CONFIGURATION PERMANENT?  (YES/NO)
```

Enter YES to modify the system on the disk to the new I/O configuration. Enter NO otherwise. The format switch on the system disk drive (7905/06/20/25) must be in the ON position if the new I/O configuration has to be made permanent. If it is desirable to restart I/O reconfiguration for any reason, enter the request:

```
/R
```

and I/O reconfiguration will restart by another display of the list:

```
CURRENT I/O RECONFIGURATION:
```

The list contains what the I/O configuration was changed to during the reconfiguration just completed.

Caution

1. It is strongly recommended that the system subchannel of the disk be backed up before making I/O reconfiguration permanent.

2. If a select code has been given a new assignment and its current I/O device has not been reassigned, the I/O device cannot be added to the system at a later date if the new I/O configuration is made permanent.

3. If a device has multiple select codes, make sure that all select codes are moved and kept in the same relative order.

4. Reassigning some devices to empty I/O slots may cause unexpected results.

Memory Reconfiguration Procedures

After the I/O reconfiguration phase is either bypassed or terminated, the Configurator displays the following statement and query:

```
CURRENT PHYSICAL MEM SIZE: xxxx PAGES
MEM  RECONFIGURATION?(YES/NO)
```

Enter NO if memory reconfiguration is not desired. The Configurator then transfers control to the operating system after displaying the message:

```
RECONFIGURATION  COMPLETED
```

Enter YES if memory is to be reconfigured. The Configurator then displays the query:

```
PHYSICAL MEM SIZE?(#PAGES)
```

Enter the desired total number of memory pages, between 48 and 1024 (decimal).

Excluding Bad Pages

The Configurator program can be used to redefine the SAM extension and user partitions to exclude any bad pages (pages containing parity errors) within these areas. Each user partition must be a contiguous block of memory; therefore, user partitions must be defined on blocks of memory between the bad pages. Bad pages in the system area, driver partitions and the memory-resident area cannot be avoided.

The Configurator displays the query:

```
DEFINE BAD PAGES BEGINNING AT PAGE xxxx (/E TO END)
-
```

where the hyphen (−) prompts for the decimal number of a bad memory page. The hyphen is repeated after acceptance of each entry until a /E or 100 bad page numbers are entered, terminating the list. (The Configurator accepts up to 100 bad memory page entries.) The bad page specifications entered can range from xxxx to the maximum page number in physical memory and must be entered in an increasing order.

If /R is entered in response to the hyphen prompt, the Configurator redisplays the query:

```
DEFINE BAD PAGES BEGINNING AT PAGE xxxx (/E TO END)
-
```

and the entire list of bad pages must be re-entered.

When a /E is entered either to terminate bad page entries or bypass the entire phase, the Configurator displays the following information:

```
CURRENT SIZE OF SAM
DEFAULT: xxxxx WORDS
EXTENSION: yy PAGES
SAM EXTENSION STARTS AT PHYSICAL PAGE xx
MAX PAGES AVAIL FOR SAM EXTENSION: xx
```

The number of words displayed for default SAM are the decimal number of words assigned to the first block of SAM.

SAM Extension Reconfiguration

The Configurator next prompts for any desired change in the size of SAM extension by displaying the query:

```
CHANGE SAM EXTENSION?(# PAGES/" " CR)
```

Press the space bar and RETURN key (the default case) if no change is desired.

Enter the decimal number of pages desired if the SAM extension is to be changed. The number of pages can vary from 0 (which removes SAM extension) to the maximum pages available for the SAM extension. Note that this count must not include any bad pages that fall within the SAM extension (see above).

The Configurator sets up the System Map to avoid bad pages in the SAM extension regardless of whether or not a change was requested.

If the specified SAM extension extends beyond the size of physical memory because of bad pages within this area, the Configurator displays the message:

```
CONFIG ERR 12
CHANGE SAM EXTENSION?(# PAGES/" " CR)
```

Enter a smaller number of pages for SAM extension size. The Configurator allows SAM extension to be divided up into a maximum of five blocks of memory between bad pages. If the number of pages in SAM extension requires division into more than five blocks, the Configurator displays the message:

```
CONFIG ERR 22
```

and the query is redisplayed. Enter a smaller size of SAM extension.

Changing Partition Definitions

The Configurator next displays a list of current partition definitions in the format:

```
CURRENT PART'N DEFINITIONS:

                         ,RT
                         ,BG        ,R
PART'N nn = pp PAGES     ,RTM
                         ,BGM
                         ,S
```

where:

nn	=	the partition number
pp	=	is the number of pages in partition nn
RT	=	a real-time partition
BG	=	a background partition
RTM	=	a real-time mother partition
BGM	=	a background mother partition
S	=	a subpartition
R	=	a reserved partition

Following the definition list, the Configurator next displays a list of shareable EMA partitions.

```
CURRENT SHAREABLE EMA PART'N DEFINITION:
PART #s LABEL
  :          :
  :          :
```

Next a list of current partition requirements is displayed:

```
CURRENT PART'N REQMTS:
REALTIME
PNAME   XX PAGES [E][PART'N=nn]
         :
         :
         :
BACKGROUND
PNAME   XX PAGES [*][E][PART'N=nn]
         :
         .
         .
```

where:

PNAME	the real-time or background program name
E	indicates an EMA (Extended Memory Area) program
*	indicates the background program does not include Table Area II (that is, a Type 4 program)
nn	is the number of the partition into which program PNAME is assigned.

The Configurator then displays the following information:

```
MAX PROGRAM SIZE:
W/OUT COMMON:          xx PAGES
W/COMMON:              xx PAGES
W/TABLE II:            xx PAGES
MAX MLS PATH LENGTH:   xx
MAX # OF PART'NS:      xx
PAGES REMAINING:       xx
DEFINE PART'NS FOR xxxx PAGES
#PAGES,RT(M)/BG(M)/S(,R)
PART'N x,pppp(,mmmm) PAGES?
```

where:

MAX PROGRAM SIZE is the maximum logical space a program may occupy. However, the partition size may be larger than the stated maximum if the partition will be used for EMA program execution.

MAX # OF PART'NS is the decimal number of partitions that can be defined in memory.

PAGES REMAINING is the decimal number of pages available for defining user partitions (including bad pages that may have been listed earlier).

#PAGES,RT(M)/BG(M)/S(,R) indicates the required format for user entries in response to the partition definition prompt described below.

PART'N x,pppp(,mmmm) PAGES? asks the user for the size (in pages) and format for the next partition to be defined.

x	is the partition number.
pppp	is the number of contiguous pages to be defined before the next page.
mmmm	is the number of pages remaining to be defined in the mother partition.

If the maximum number of partitions was defined as 0 during generation time, the Configurator skips the rest of memory reconfiguration and displays the query:

```
NEW MEMORY CONFIGURATION PERMANENT?
```

Since partitions must be defined contiguously, they must be within the section of memory between the bad pages. If a section of memory between bad pages has a size of one page, it is skipped by the Configurator. The Configurator prompts for a partition definition after each accepted entry until partitions have been defined for all xxxx pages in this section of memory.

As each entry is accepted, the Configurator reissues the prompt with an incrementally increased partition number for the next partition. If the number of pages entered for a partition is greater than the maximum logical address space, and RT or BG was specified, the Configurator displays the message:

```
SUBPARTITIONS?(YES/NO)
```

Enter NO if the configurator is to ignore subpartition considerations and proceed with the normal partition definitions.

Enter YES if subpartitions are to be defined. Subpartition definitions are specified by using the following format in response to the prompt:

```
#PAGES,S(,R)
```

where S specifies a subpartition and the optional R specifies the subpartition is to be reserved.

If RTM or BGM is specified, subpartition definition phase is automatically entered. If no subpartitions are to be defined, enter a /E or define the next partition of RT, BG, RTM, or BGM type.

The memory space allocated for subpartitions is the same area occupied by the mother partition. Subpartition definition will end as soon as an RT(M) or BG(M) partition is defined, or can be terminated by entering a /E.

When an attempt is made to end the subpartition definition phase by defining an RT or BG partition and there are no more pages left in this section of memory, an ERR 13 will be displayed. In this case, either enter a /E to terminate subpartition definitions and continue partition definitions for the next block of memory or enter /R to restart the partition definition phase.

The total number of pages defined for subpartitions must not exceed the size of the mother partition or an error code is issued and the last subpartition must be redefined.

The Configurator analyzes each partition definition for possible errors as soon as it is entered. Any error code issued will be followed by a prompt to redefine the last partition displayed. If /R is entered instead of a partition description, the partition definition phase is restarted from the first partition definition.

Partitions defined for each section of memory between bad pages must be defined for all pages available within the section. A running total is maintained of the number of pages currently defined within a section of good memory. The Configurator then takes one of five possible courses of action, depending upon the prevailing memory structure and size:

1. If the remaining total equals the number of pages available, the Configurator automatically requests partition definitions for the next section of good memory.
2. If the number of pages remaining to be defined is one, the Configurator increments the last defined partition by one page and then requests partition definitions for the next block of good memory.
3. If the running total exceeds the number of available pages defined within the memory block, the Configurator displays an error message and prompts for the last partition to be redefined.
4. If the number of partitions already defined is equal to the maximum number of partitions allowed and more undefined good pages remain, the Configurator displays an error message and all user partitions must be redefined. The Configurator then prompts for new partition definitions and repeat the prompt after each accepted entry.
5. If the running total is less than the number of pages in the block of memory, definition for next partition is requested.

   A list of NEW PART'N DEFINITIONS is issued to the list device when all partitions have been defined.

Assigning Shareable EMA Partitions

After new partitions are displayed, the Configurator asks that shareable EMA partitions be defined. The maximum number of partitions that can be defined is displayed:

```
MAX NUMBER SHAREABLE EMA PART'NS =  x
SHAREABLE EMA PARTITIONS
LABEL,PART'N #    (/E TO END)
2,DATA        *  EMA LABELED "DATA" IN PART'N 2
5,SHEMA
```

Changing Program Partition Assignments

The Configurator performs a check to ensure that every program assigned to a partition fits in its partition size and is not assigned to a shareable EMA partition. A program is unassigned if the program size is larger than the partition size, the partition number does not exist, or the partition is a shareable EMA partition. Following the check, the Configurator issues a list under the heading:

```
UNASSIGNED PROGRAMS
       :
```

Next, for every shareable EMA program, the Configurator prompts for the shareable EMA label used by the program:

```
SHAREABLE EMA PROGRAMS
PROGRAM xxxxx? yyyyy    *PROGRAM xxxxx USES SHAREABLE EMA yyyyy
PROGRAM aaaaa? bbbbb
       :
```

If /R is entered in the shareable EMA program phase, then the Configurator returns to the shareable EMA partition definition phase. After all shareable EMA programs have been assigned labels, the Configurator prompts:

```
MODIFY PROGRAM PAGE REQMTS?(/E TO END)
PNAME,#PAGES
-
```

Enter the specifications for any disk-resident programs whose page requirements must be changed, using the format:

```
program name,xx
```

where the number of pages entered for each program must include the base page. The number of pages must be greater than or equal to the program relocation size, and less than or equal to the maximum address space for the program. The program can only be Type 2, 3 or 4.

The hyphen prompt is repeated after acceptance of each entry until a /E is entered to terminate the list.

Note that the page requirements for an EMA program cannot be modified.

Program Partition Assignments

The Configurator now asks if any programs need to be assigned to partitions by displaying the query and prompt:

```
ASSIGN PROG PART'NS?(/E TO END)
PNAME, PART'N#
-
```

where the hyphen prompt is repeated after each accepted entry until a /E or /D is entered to terminate the list. The use of /D forces any remaining pages into one background partition.

Enter each desired program partition assignment in the form:

```
program name,xx
```

where xx is the partition number to which the program is to be assigned. If xx is 0, the program is unassigned and can be dispatched to any partition of the proper type large enough to run the program. The program must be a Type 2, 3 or 4. Note that a program cannot be assigned to a shareable EMA partition. When a /E is entered to terminate the list, the Configurator issues the query:

```
NEW MEMORY CONFIGURATION PERMANENT? (YES/NO)
```

Enter a YES to a change the appropriate tables and locations on the disk-resident system. The format switch on the system disk drive must be in the on position (7905/06/20/25) if the new memory configuration has to be made permanent. The Configurator then issues the message:

```
RECONFIGURATION COMPLETED
```

and turns control over to the operating system.

If a /R is entered in response to the prompt instead of YES, memory reconfiguration is restarted from the query:

```
PHYSICAL MEM SIZE?(#PAGES)
```

and the system is in the state it was changed to during the earlier reconfiguration.

Reconfiguration Example

The sample reconfiguration illustrated below assumes that reconfiguration was requested by setting the switch register as described at the beginning of this chapter.

```
START RECONFIGURATION
LIST DEVICE LU#?
20                             *SPECIFY A LIST DEVICE.
LIST DEVICE SELECT CODE#?
20                             *SPECIFY LIST DEVICE'S SELECT CODE.
ECHO?(YES/NO)
YES                            *ECHO OUTPUT ON LIST DEVICE.
I/O RECONFIGURATION ALREADY PERFORMED:
CURRENT SELECT CODE#, NEW SELECT CODE#
21,13                          *SYSTEM DISK 1
25,25                          *SYSTEM CONSOLE
20,20                          *LIST DEVICE
CURRENT I/O CONFIGURATION:     *CURRENT I/O CONFIGURATION
SELECT CODE 10= TBG            * IS DISPLAYED.
SELECT CODE 13= EQT  1,TYPE 32
SELECT CODE 14= EQT  6,TYPE  0
SELECT CODE 15= EQT  7,TYPE  1
SELECT CODE 16= EQT  3,TYPE 23
SELECT CODE 17= EQT  3,TYPE 23
SELECT CODE 20= EQT  5,TYPE 12
SELECT CODE 22= EQT  4,TYPE  2
SELECT CODE 25= EQT  2,TYPE  5
I/O RECONFIGURATION?(YES/NO)
YES                            *SPECIFY I/O RECONFIGURATION.
CURRENT SELECT CODE#,NEW SELECT CODE#?(/E TO END)
-
10,14                          *RECONFIGURE SELECT CODES.
-
14,15
-
15,16
-
16,23
-
17,24
-
22,17
-
/E

NEW I/O CONFIGURATION:         *NEW I/O CONFIGURATION

SELECT CODE 13= EQT  1,TYPE 32 *  IS DISPLAYED.
SELECT CODE 14= TBG
SELECT CODE 15= EQT  6,TYPE  0
SELECT CODE 16= EQT  7,TYPE  1
SELECT CODE 17= EQT  4,TYPE  2
SELECT CODE 20= EQT  5,TYPE 12
SELECT CODE 23= EQT  3,TYPE 23
SELECT CODE 24= EQT  3,TYPE 23
SELECT CODE 25= EQT  2,TYPE  5
NEW I/O CONFIGURATION PERMANENT?(YES/NO)
NO                             *SPECIFY NONPERMANENT.
CURRENT PHYSICAL MEM SIZE:     48 PAGES
MEM RECONFIGURATION?(YES/NO)
```

```
YES                               *SPECIFY MEMORY RECONFIGURATION.
PHYSICAL MEM SIZE?(#PAGES)
256                               *SPECIFY AN INCREASE IN MEMORY SIZE.
DEFINE BAD PAGES BEGINNING AT PAGE  28 (/E TO END)
-
/E CURRENT SIZE OF SAM:
DEFAULT:  3802 WORDS
EXTENSION:   0 PAGES
SAM EXTENSION STARTS AT PHYSICAL PAGE   28
MAX PAGES AVAIL FOR SAM EXTENSION:  12
CHANGE SAM EXTENSION?(#PAGES/" "CR)
6                                 *INCREASE SIZE OF SAM.
CURRENT PART'N DEFINITIONS:       *CURRENT PARTITION DEFINITIONS
PART'N   1 =  20  PAGES,BG        *  ARE DISPLAYED.
CURRENT SHAREABLE EMA PART'N DEFINITION:
PART# LABEL
===== =====
2     DATA
5     SHEMA
CURRENT PART'N REQMTS:            *CURRENT PARTITION REQUIREMENTS
REALTIME                          *  FOR VARIOUS PROGRAMS ARE
BACKGROUND                        *    DISPLAYED.
$CNFX    3 PAGES
ASMB    16 PAGES
XREF    16 PAGES
LOADR   16 PAGES
WHZAT    3 PAGES
FMGR     7 PAGES
RT6GN   20 PAGES
SWTCH   11 PAGES
MAX PROGRAM SIZE:                 *MAXIMUM PARTITION SIZES FOR
W/OUT COMMON: 29 PAGES            *  VARIOUS PROGRAM TYPES ARE
W/ COMMON:    29 PAGES            *    DISPLAYED.
W/ TABLE II:  27 PAGES MAX MLS PATH LENGTH: 29
MAX # OF PART'NS: 15
PAGES REMAINING:   222
DEFINE PART'NS FOR  32 PAGES:
#PAGES,RT(M)/BG(M)/S(,R)
PART'N   1,  32 PAGES?
32,BG,R                           *BG PARTITION.
DEFINE PART'NS FOR  79 PAGES:
#PAGES,RT(M)/BG(M)/S(,R)
PART'N   2,  79 PAGES?
49,RT                             *RT PARTITION WITH NO SUBPARTITIONS.
SUBPARTITIONS?(YES/NO)
NO
PART'N   3,  30 PAGES?
27,RT,R                           *RT PARTITION WHICH IS RESERVED.
PART'N   4,  3 PAGES?
3,RT,R                            *RT PARTITION WHICH IS RESERVED.
DEFINE PART'NS FOR 131 PAGES:
#PAGES,RT(M)/BG(M)/S(,R)
PART'N   5,  131 PAGES?
115,BGM                           *BG MOTHER PARTITION.
PART'N 6, 16,(115) PAGES?
48,S                              *SUBPARTITION LARGER THAN 32K WORDS.
PART'N   7,  16,(67) PAGES?
29,S                              *SECOND SUBPARTITION.
PART'N  8, 16,(38) PAGES?
29,S                              *THIRD SUBPARTITION.
PART'N   9,  16,(9) PAGES?
```

```
9,S                               *FOURTH SUBPARTITION.
PART'N  10,  16 PAGES?
16,BG                             *BG PARTITION.
NEW PART'N DEFINITIONS:           *NEW PARTITION DEFINITIONS
PART'N   1 =  32  PAGES,BG,R      *  ARE DISPLAYED.
PART'N   2 =  49  PAGES,RT
PART'N   3 =  27  PAGES,RT,R
PART'N   4 =   3  PAGES,RT,R
PART'N   5 = 115  PAGES,BG
PART'N   6 =  48  PAGES,S
PART'N   7 =  29  PAGES,S
PART'N   8 =  29  PAGES,S
PART'N   9 =   9  PAGES,S
PART'N  10 =  16  PAGES,BG
MAX NUMBER SHAREABLE EMA PARTITIONS = 8

SHAREABLE EMA PARTITIONS:
LABEL,PART'N #   (/E TO END)
2,DATA                            *EMA LABELED "DATA" IN PART'N 2
5,SHEMA                           *EMA LABELED "SHEMA" IN PART'N 5
/E

UNASSIGNED PROGRAMS:
LOADR
SAVE

SHAREABLE EMA PROGRAMS:
PROGRAM TESTX? SHEMA              *PROGRAM TESTX USES SHAREABLE EMA SHEMA
PROGRAM ABC  ? DATA

MODIFY PROG PAGE REQMTS?(/E TO END)
PNAME,#PAGES                      *SPECIFY NEW PROGRAM PAGE REQUIREMENTS.
-
RT6GN,27
-
MACRO
-
/E
ASSIGN PROG PART'NS?(/E TO END)
PNAME,PART'N#                     *ASSIGN PROGRAMS TO PARTITIONS.
-
RT6GN,3
-
WHZAT,4
-
D.RTR,1
-
/E
NEW MEM CONFIGURATION PERMANENT?(YES/NO)
NO                                *DO NOT MAKE MEMORY CHANGES PERMANENT.

                                  *END OF I/O AND MEMORY RECONFIGURATION.
                                  *SYSTEM WILL NOW ATTEMPT TO BOOTUP.
SET TIME
:SV,4
TE,*****
TE,*****  92084A RTE-6/VM 7905 7906 7920 7925 DISK CARTRIDGE
TE,*****          HP 92084-13XXX (7905/7906)
TE,*****          HP 92084-13XXX (7920)
TE,*****          HP 92084-13XXX (7925)
TE,*****
:
```

Boot-up and Reconfiguration Halts

During either system boot-up or reconfiguration, various HLTS (of the form 1020xx) may be issued on the computer front panel. The meaning of these halts and any required operator action are given in Table 10-1.

Configurator Error Messages

Whenever a user response to a Configurator prompt is illegal or inappropriate, the Configurator issues a CONFIG ERR message and prompts for a correct entry. All possible Configurator error codes are listed sequentially in Table 10-2. Locate the appropriate code and take the described action.

Table 10-1. System Boot-up and Reconfiguration Halts

Halt		Meaning and User Action Required
4	Meaning:	Powerfail occurred and powerfail automatic restart is enabled.
	Action:	Restart system boot-up procedure.
5	Meaning:	Memory protect switch was set and memory parity error occurred.
	Action:	Restart system boot-up procedure.
6	Meaning:	A partition was found not properly linked into an operating system partition list. The operating system may be corrupt or a bug may exist.
	Action:	Reboot the system and if the problem persist, call your HP representative.
10B	Meaning:	FMGR or D.RTR cannot be scheduled at startup because there is not a large enough partition (issued by the system).
	Action:	Restart system boot-up and redefine memory to include a partition large enough for FMGR and D.RTR.
11B	Meaning:	Attempt was made to re-execute a non-RPL compatible ROM Loader 12992A, or Bootstrap Loader.
	Action:	Reload the ROM Loader or Bootstrap Loader before re-executing. For CS/80 disk: check that the correct unit was specified is the S-Register at boot-up. If so, run diagnostics to isolate disk problem.
20B	Meaning:	Uninstalled memory halt. The system halts if memory has been defined but is not installed. The system checks pages from the beginning of the user partitions through the end of the memory stack. The first page it finds defined by the user but not physically installed causes the HLT 20 instruction to be executed. The first uninstalled page number is displayed in the A- and B-Registers.
	Action:	Reconfigure memory such that the amount of memory specified in the reconfiguration is less than or equal to the amount of physically installed memory.
21B	Meaning:	Bad VMA/OS firmware. The firmware does not pass the self-test.

Table 10-1. System Boot-up and Reconfiguration Halts (Cont.)

Halt	Meaning and User Action Required
22B	Meaning: One of the following conditions was encountered: 1. $CNFX is not a Type 3 program. 2. A contiguous memory block of three good pages cannot be found in the user partition area. Action: Restart system boot-up procedure. If memory reconfiguration is needed, $CNFX must be permanently loaded as a Type 3 program and there must be at least three good pages of contiguous memory in the user partition area. If $CNFG cannot find the ID segment for $CNFX, the system will boot-up with no memory reconfiguration.
30B	Meaning: Error was encountered in the disk I/O process by one of the RPL-compatible ROM Loaders Part Nos. 12992B and 12992F. If the disk is a 7905/20, the status word 1 is displayed in the B-Register and disk status word 2 in the A-Register. Action: Retry the system boot-up procedure.
31B	Meaning: Error was encountered in the disk I/O process by the Boot Extension. If the disk is a 7905 or 7920, the disk status word 1 is displayed in the B-Register and disk status word 2 is displayed in the A-Register. For CS/80 disks: A-Register contains the failed status word number and B-Register contains the status indicated by A. A=1 implies reject error, A=3 is access error. Action: Retry the system boot-up procedure. For CS/80 disks: retry or run diagnostics to isolate disk error.
47B	Meaning: An attempt was made to call an OS overlay before the OS overlay mapper was set up. This usually means that the I/O configuration for the system disk is bad, causing a disk error condition at boot-up. I/O errors are processed in an OS overlay that is not yet loaded. Action: Check the disk configuration on the Generator listing for correct select code on the EQT, and correct interrupt table entry.
55B, 56B or 57B	Meaning: While dispatching a program the operating system encountered an unexplainable condition. The operating system may be corrupt or a bug may exist. Action: Reboot the system and if the problem persists, call your HP representative.

Table 10-2. I/O and Memory Reconfiguration Error Codes

Halt	Meaning and User Action Required
1	Meaning: Invalid LU number or a bit bucket LU.
2	Meaning: Illegal select code number. Action: Enter valid number that must be between 10 and 77 octal.
3	Meaning: New select code entered is identical to new select code assigned to disk, system console or list device, or else the current select code entered is identical to the old select code for disk, system console or list device (i.e., do not reconfigure that which was already done via the SWITCH register). Action: Enter a different select code.
10	Meaning: Specified total number of pages outside the range. Action: Enter valid number in the range 48-1024 for physical memory size and between 0 and maximum pages available for SAM extension.
11	Meaning: Invalid bad page number. Action: Enter valid number greater than the previous entry and less than the physical memory size, or enter /E to terminate the list.
12	Meaning: Specified SAM extension entry beyond physical memory size due to bad pages. Action: Enter smaller number of pages for SAM extension.
13	Meaning: Current running total exceeds available pages in block of good memory or exceeds size of mother partition. Action: Redefine last partition or subpartition size. If there are no more pages available in the block of memory to be defined, /E or /R are the only responses accepted.
14	Meaning: Second parameter of partition definition entry other than RT, BG, or S, or else S was entered when a subpartition definition was not expected. Action: Reenter definition with correct parameter.
15	Meaning: Third parameter of partition definition entry other than R. Action: Reenter definition with R as third parameter if partition is to be reserved.
16	Meaning: No such program, or the name of a segment was entered or invalid type was entered for partition assignment. Action: Reenter assignment with correct program name or type or /E to end this sequence.

Table 10-2. I/O and Memory Reconfiguration Error Codes (cont.)

Halt	Meaning and User Action Required
17	Meaning: Invalid partition number. Action: Enter valid number or /E to end this sequence.
18	Meaning: Program does not fit in the assigned partition. Action: Assign program to larger partition if available, or continue without assigning the program.
19	Meaning: Invalid number of pages was entered for program size. Action: Enter valid number of pages for program, between the size of the program at load time and the maximum logical address space for the program.
20	Meaning: Number of defined partitions already equal to allowed maximum number and more undefined pages remain. Action: Redefine all partitions.
21	Meaning: Page requirements of an EMA program cannot be modified. Action: Entry is skipped.
22	Meaning: Number of pages in SAM extension requires division into more than 5 blocks. Action: Enter a smaller size of SAM extension.
23	Meaning: An illegal label was entered. Action: Enter valid label beginning with a letter followed by one to five alphanumeric characters.
24	Meaning: Too many shareable EMA partitions. Action: Modify source code of $EMTB to allow more shareable EMA partitions.
25	Meaning: An undefined label was entered during the shareable EMA program phase of reconfiguration. Action: Enter one of the labels defined earlier during shareable EMA partition definition.
26	Meaning: Shareable EMA partition specified is too small. Action: Assign a larger shareable EMA partition.
27	Meaning: Program assigned to a shareable EMA partition. Action: Assign program to a different partition.

Table 10-2. I/O and Memory Reconfiguration Error Codes (cont.)

Halt	Meaning and User Action Required
28	Meaning: Assigning more than one label to the same partition. Action: Assign a unique label to that partition. Note that if a mother partition is declared to be a shareable EMA partition, then the subpartitions of that mother partition may not be declared to be shareable EMA, or vice versa.
29	Meaning: A duplicate label was entered. Action: Redefine the partition using a different label.
30	Meaning: Tried to assign a program to a shareable EMA partition when no shareable EMA partitions have been defined. Action: Shareable EMA partition definition is restarted by the configurator. Define a shareable EMA partition.
31	Meaning: No ID-EXTENSIONS left. Shareable EMA partitions are being reassigned so that more ID-EXTENSIONS are required. Action: Shareable EMA partition definition is restarted by the configurator. Reallocate shareable EMA programs so that they share more ID-EXTENSIONS.

A

Real-Time Disk Usage

This appendix covers the following subjects:

- Disk Parity Errors
- Track Configuration
- Multiple Disk Controller Operation
- Multiple Interface Card Operation
- Multiple CPU – MAC System Operation

Disk Parity Errors

When a program tries to write to a track with either a track number greater than the number of tracks assigned to a given subchannel, or with a track number equal to −1, the driver for the disk sets bit 5 in the status word (end-of-disk) and exits with the transmission log (B-Reg) set to the number of tracks assigned to the subchannel. If the request is a read, the driver will also return the number of 64-word sectors per track for the subchannel in the first word of the buffer. To obtain this information, a program can request an impossible track number once and thereafter stay within the bounds on the subchannel.

Further information on disk I/O requests and error returns can be obtained from the DVA32/DVR32 Driver Manual and DVM33 Driver Manual.

If a parity error occurs during disk transfer, a special error message is printed:

```
TR nnnnn EQT eqt, U pp S (or U)
```

where:

nnnnn	is the logical track number within the subchannel pp.
eqt	is the EQT entry number.
pp	is the subchannel or unit number.

This is an irrecoverable disk transfer parity error. If the transfer is to a system or auxiliary disk, the following results apply:

a. If user request (U), then the program is abnormally terminated and the track is made unavailable for further operations. If the user request was an on-line modification with the RTE loader, the parity error could be the result of failing to turn off the hardware disk protect switch. The loader should be executed again with the protect switch off and the format switch in the on position.

b. If system request (S), the program transfer terminates.

For peripheral disk transfers, a parity error causes the transmission log to be returned to the calling program as a −1.

Track Configuration

The configuration of disk tracks is normally done through the interactive generation process described in Chapter 3 of the *RTE-6/VM Online Generator Manual*, part number 92084-90010. However, when more than one type of disk controller/interface is needed, the generator dialogue cannot be used and a track map table for the additional controller/interface must be defined in a user module. Because the track map tables for MAC disks, ICD disks, and CS/80 are different, these processes are described separately.

Configuration For Disks

The track map table for the CS/80 disks (HP 7908, 7911, 7912, 7914, 7933, 7935, 7937, and 7958) contains the following information:

- HP-IB address of disk.
- Unit and Volume Number of disk.
- Starting block number of subchannel.
- Number of tracks included in subchannel.
- Number of blocks/track of subchannel.

The configuration of CS/80 disks is identical except for the number of blocks contained on each type of disk. The specifications are given below:

HP 7908	64750	blocks
HP 7911	109824	blocks
HP 7912	256256	blocks
HP 7914	516096	blocks
HP 7933	1579916	blocks
HP 7935	1579916	blocks
HP 7937	2232204	blocks
HP 7958	510552	blocks
HP 7958B	593872	blocks
HP C2200	1309840	blocks
HP C2203	2619792	blocks

The HP 12821A Disk Interface can address up to four disk controllers. Any combination of CS/80 drives can be used.

Each subchannel is a contiguous group of blocks, which have been grouped into tracks, on a single drive. There may be more than one subchannel per drive, but subchannels cannot cross drive boundaries. The exact number of subchannels is specified by the user. There may be as many as 64 subchannels per controller. Subchannels are numbered sequentially from zero; no numbers may be skipped.

CS/80 Disk Unit Number

The unit number is a number associated with a portion of each drive. The disk is unit 0 and the Cartridge Tape Drive (CTD) is unit 1, except for 9144 tape drives, when the CTD is unit 0.

Defining the CS/80 Track Map Table

When an extra disk interface is needed, tracks are mapped in a table defined as follows:

```
MACRO,R,B,L,
      NAM  $TM33,15       ($TM33 must be a type 8 or type 15 program.)
      ENT  $TM33
$TM33 DEC  -n             n is the total number of subchannels
SC0   word 0              See entry format below.
      word 1
      word 2
      word 3
      word 4
      word 5
      word 6
      word 7
SC1   word 0              Repeat for next subchannel
      :
SCn-1 word 0              Until all subchannels are defined
      :
      END
```

When defining your own $TM33, you are restricted to 13 subchannels if you use a type 8 program.

To solve this problem you can make $TM33 a type 15 program. This places $TM33 in Table Area I and eliminate the 13 subchannel limitation.

Disk Track Map Table Format

15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

<table>
<tr><td>word 0</td><td>Reserved</td><td>HP-IB Address</td></tr>
<tr><td>word 1</td><td>Unit Number</td><td>Volume Number</td></tr>
<tr><td>word 2</td><td>High Word</td><td rowspan="3">of starting
block number</td></tr>
<tr><td>word 3</td><td>Middle Word</td></tr>
<tr><td>word 4</td><td>Low Word</td></tr>
<tr><td>word 5</td><td colspan="2"># of tracks</td></tr>
<tr><td>word 6</td><td colspan="2"># blocks/track</td></tr>
<tr><td>word 7</td><td colspan="2">Reserved</td></tr>
</table>

CTD Track Map Table Format

15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

<table>
<tr><td>word 0</td><td>T</td><td>I</td><td>Reserved</td><td>HP-IB Address</td></tr>
<tr><td>word 1</td><td colspan="3">CTD Unit Number</td><td>CTD Volume Number</td></tr>
<tr><td>word 2</td><td>c</td><td colspan="2">Disk cache unit #</td><td>Disk Volume Number</td></tr>
<tr><td>word 3</td><td colspan="2">high</td><td colspan="2" rowspan="2">Start block of cache
on Disk</td></tr>
<tr><td>word 4</td><td colspan="2">low</td></tr>
<tr><td>word 5</td><td>d</td><td colspan="3">Reserved</td></tr>
<tr><td>word 6</td><td colspan="4">Reserved</td></tr>
<tr><td>word 7</td><td colspan="4">Reserved</td></tr>
</table>

T – Set indicates a CTD.
c – Set indicates a cached CTD. 0 for noncached CTD.
d – dirty bit set by driver. Reserved
I – CTD supports immediate reporting.

Reserved words must be set to zero.

Only one entry is needed in the Track Map Table to define the CTD. This entry declares the CTD, assigns it to a subchannel and allocates disk cache.

Disk cache is a 64-kilobyte disk buffer that can be defined in the CTD Track Map Table if the CTD shares a controller with the disk drive. Disk cache buffering greatly improves transfer time to tape for transfers of more than eight kilobytes.

For cartridge tapes that have their own dedicated controller, set the I bit (14) in word 0 of the CTD Track Map Table to specify the immediate reporting option. Immediate reporting allows the controller and driver to process a pending request without waiting for completion of the current write to CTD, thereby improving transfer time.

Example:

```
ASMB,Q,C
     NAM $TM33,15 92084-16652 REV.2440 < 840925.1057 >

*****************************************************************
*                                                               *
*  NAME:         $TM33                                          *
*  SOURCE:       92084-18652                                    *
*  RELOC:        92084-16652                                    *
*  PGMR:         J.S., W.J.A., K.P.J., T.F.                     *
*                                                               *
*****************************************************************

      ENT $TM33
$TM33 DEC -4

SC0   OCT 0              hpib addr0
      OCT 0              unit 0 / vol 0
      OCT 0              ! 3 word
      OCT 0              ! start block  (0)      disk subchannel 0
      DEC 0              ! value
      DEC 400            number of tracks
      DEC 48             blocks per track
      OCT 0              reserved word

SC1   OCT 0              hpib addr 0
      OCT 0              unit 0 / vol 0
      OCT 0              ! 3 word
      OCT 0              ! start block (19,200)  disk subchannel 1
      DEC 19200          ! value
      DEC 443            number of tracks
      DEC 48             blocks per track
      OCT 0              reserved word

SC2   OCT 0              hpib addr 0
      OCT 0              unit 0 / vol 0
      OCT 0              ! 3 word
      OCT 0              ! start block (40,464)  disk subchannel 2
      OCT 117020         ! value
      DEC 500            number of tracks
      DEC 48             blocks per track
      OCT 0              reserved word

SC3   OCT 100000         ctd flag / immediate reporting / hpib addr 0
      OCT 400            ctd unit 1 / ctd volume 0
```

```
      OCT 100000        cached flag / disk cache - unit 0 / volume 0
      OCT 0             ! 2 word starting block
      OCT 175720        ! of cache on disk (64,464)
      OCT 0             reserved
      OCT 0             reserved                 tape subchannel 3
      OCT 0             reserved

*    There is a 30 block hole between the end of cache and the
*    physical end of the 7908 disk.  In the system generation phase for
*    a CS/80 system disk, this is created by specifying a negative
*    number of blocks (ie. a 'hole' is created).  Purpose of this hole is
*    to align physical backups of disks.  To create a 'hole' using the
*    track map table simply increase the start block address for a sub-
*    channel by the number of blocks you would like skipped past the end
*    of the previous subchannel.

*    Below is an example of using a non-cached tape device.
*    Note setting of bit 14, word #0 to 1.
*SC3 OCT 140001   ctd flag / immediate reporting / hpib addr 1
*    OCT 0        ctd unit 0 / ctd volume 0
*    OCT 0        cached flag / disk cache - unit 0 / volume 0
*    OCT 0        !  2 word starting block address ignored
*    OCT 0        !  No cache word
*    OCT 0        reserved
*    OCT 0        reserved                  tape subchannel 3
*    OCT 0        reserved

     END
```

Multiple HP 12821A Interface Card Operation

In order to increase disk throughput, you may wish to include more than one HP 12821A interface card in the system generation. This involves relocating a second copy of the DVM33 driver (named %DVN33) during the generation, and including entries in the Equipment Table (EQT), the Device Reference Table (DRT), and the Interrupt Table for all disks on the second 12821A interface card. In addition to this, you must supply a track map table ($TN33) to describe the subchannel configuration. The format of this table is identical with $TM33, but the name must be changed to satisfy the driver, DVN33. You may wish to take the source (&$TM33) of the HP supplied CS/80 track map table and modify it to meet your particular requirements. See the DVM33 Driver Manual for additional information.

The track map table for DVN33 would appear as follows:

```
MACRO,R,B,L,
      NAM  $TN33,15
      ENT  $TN33
$TN33 DEC  -n          (n is the total number of subchannels.)
      DEC  0
        :
        :
      END
```

Track Configuration for HP 13037B/C MAC Disk

The track map table for the HP 13037B/C Multiple Access Controller disks (HP 7905, 7906, 7920, and/or 7925) contains the total number of subchannels on controller and the following information for each subchannel:

- Number of 64-word sectors per track
- Cylinder number of track 0
- Number of surfaces included in subchannel
- Head number of track 0
- Unit number of disk drive
- Number of tracks on subchannel
- Number of spares allocated to each subchannel

Information that is required to properly configure a track on an HP 7905, 7906, 7920, or 7925 disk is given below (a full description of track configuration can be found in Chapter 3).

The drive specifications are:

HP 7905	HP 7906
64 words per sector	64 words per sector
96 sectors per track	96 sectors per track
411 tracks per surface	411 tracks per surface
3 surfaces per drive	4 surfaces per drive

HP 7920	HP 7925
64 words per sector	64 words per sector
96 sectors per track	128 sectors per track
823 tracks per surface	823 tracks per surface
5 surfaces per drive	9 surfaces per drive

Note The RTE MAC Disk Driver DVR32 treats a logical track as 64 words per sector, with the number of sectors per track dependent upon the subchannel definition. Therefore, HP 7905/7906/7920 disks would have 96 logical sectors per track, and an HP 7925 disk would have 128 logical sectors per track.

MAC Disk Subchannels

The HP MAC disk system can control up to eight disk drives connected to one HP 13037B/C controller. Any combination of HP 7905, 7906, 7920, and 7925 disk drives can be used. The MAC subchannels are not directly related (one per platter) to the disk drive and they are not restricted to eight subchannels.

Each subchannel is a contiguous group of tracks on a single drive. There may be more than one subchannel per drive, but subchannels cannot cross drive boundaries. The exact number of subchannels is specified by the user. There may be as many as 64 subchannels per controller. Subchannels are numbered sequentially from zero; no numbers may be skipped.

MAC Disk Sectors

The following paragraphs describe how to optimally read from or write to a MAC disk (using sector organization):

READ DATA – The drivers divide each track into 64-word sectors. Whenever more than 64 words are transmitted, the READ request is fastest when begun on an even sector.

WRITE DATA – WRITE requests starting on an odd sector or ending in an even sector require more time; thus, the fastest transfers are WRITE requests that start on an even sector and end in an odd sector. The system always organizes programs and swaps them out in such a way that transfers start on an even sector and end on an odd sector, thereby minimizing program load and swap times. The WRITE request data can be checked for recoverability by setting bit 10 in the control word (ICNWD). This check on all data written slows the WRITE process.

MAC Disk Tracks

The number of tracks on a disk drive is determined by multiplying the cylinders (or head positions) by the number of surfaces on the drive.

Disk Drive	Cylinders or Head Positions	# Surfaces	Maximum # of Tracks
HP 7905	411	3	1233
HP 7906	411	4	1644
HP 7920	823	5	4115
HP 7925	823	9	7407

Theoretically, the number of tracks could all be assigned to one subchannel; however, there are limitations. Peripheral disk subchannels must not have more than 32,767 tracks (excluding spares) per subchannel. The combined size of the system and auxiliary subchannel (LU 2 and LU 3) must not exceed 1600 tracks, excluding spares.

Head positions or cylinders are numbered sequentially starting from 0. Heads are numbered sequentially starting from 0, one for each surface.

MAC Disk Surface Organization

Subchannels on an HP 7905 may be on one, two, or three surfaces. Subchannels on an HP 7906 may be on from one to four surfaces. Subchannels on an HP 7920 may be on from one to five surfaces. Subchannels on an HP 7925 may be on from one to nine surfaces.

It is best to alternate surfaces (to minimize head movement) when more than one surface is used. For example, if track 0 is at cylinder 10 on head 0, then track 1 should be at cylinder 10 on head 1 and track 2 at cylinder 11 on head 0. The implications of splitting a subchannel between HP 7905/7906 fixed and removable platters are discussed in Chapter 2 of this manual under Disk Planning.

MAC Disk Unit Number

The unit number is a number associated with each HP 7905, 7906, 7920, or 7925 disk drive. The unit number is set (by the user) behind the front panel of the drive, and is always displayed on the front panel. There may be eight units, numbered 0 through 7. Do not change the unit specification while the drive is being accessed.

Defining the MAC Disk Track Map Table

When the HP 13037B/C controller is not the system disk controller, tracks are mapped in a table defined as follows:

```
MACRO,R,B,L,O
      NAM   $TB32,15     ($TB32 must be a type 8 or type 15 program)
      ENT   $TB32
$TB32 DEC   -n           n is the total number of subchannels
SC0   DEC   word 0       See entry format below
      DEC   word 1
      OCT   word 2
      DEC   word 3
      DEC   word 4
SC1   DEC   word 0       Repeat for next subchannel
      DEC   word 1
      OCT   word 2
      DEC   word 3
      DEC   word 4
SCn-1 DEC   word 0       Until all subchannels are defined
      :
END
```

There is a 13 subchannel restriction if you make $TB32 a type 8 program. Therefore, if the total number of subchannels is greater than 13, you need to make $TB32 a type 15 program.

$TB32 Entry Format

	15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0		
word 0	#64-word sectors/track		
word 1	first cylinder #		
word 2	#surfaces	starting head#	unit#
word 3	#tracks		
word 4		#spares	

Where the # of sectors per track is based on 64-word sectors. The following apply:

HP 7905/7906/7920	96 sectors/track
HP 7925	128 sectors/track

All unused fields should be set to zero.

Example: Define five HP 7905 subchannels using two surfaces of the removable disk cartridge. Each subchannel starts at head 0.

```
MACRO,R,B,L,O
      NAM   $TB32,15
      ENT   $TB32
$TB32 DEC   -05         Total of five subchannels
SCO   DEC   96          # 64 word sectors/track
      DEC   0           Subchannel 0 starts at cylinder 0
      OCT   04005       Two surfaces, head 0, unit 5
      DEC   150         150 tracks for subchannel 0
      DEC   4           4 spare tracks
SC1   DEC   96
      DEC   77          Subchannel 1 starts at cylinder 77
      OCT   04005
      DEC   200         200 tracks for subchannel 1
      DEC   6           6 Spare tracks
SC2   DEC   96
      DEC   180         Subchannel 2 starts at cylinder 180
      OCT   04005
      DEC   200         200 tracks for subchannel 2
      DEC   6           6 spare tracks
SC3   DEC   96
      DEC 283           Subchannel 3 starts at cylinder 283
      OCT 04005
      DEC 150           150 tracks for subchannel 3
      DEC 4             4 spare tracks
SC4   DEC   96
      DEC   360         Subchannel 4 starts at cylinder 360
      OCT   04005
      DEC   99          99 tracks for subchannel 4
      DEC   3           3 spare tracks
END
```

NOTE: Use approximately 6 spare tracks per 200 data tracks.

Multiple HP 13037B/C Controller Operation

In order to increase disk throughput, the user may wish to include more than one HP 13037B/C controller in the system generation. This involves relocating a second copy of DVR32 (named %DVP32) during the generation, and including entries in the equipment table , the device reference table, and the interrupt table for the disks on the second 13037B/C controller. In addition to this, the user must supply the track map table ($TP32) to describe the subchannel configuration. The format of this table is identical with $TB32, but the name must be changed to satisfy the driver, DVP32. The user may wish to take the HP supplied source for $TB32 (filename &$TB32), and modify it to meet his particular requirements. See the DVR32/DVA32 Driver Manual for additional information. The track map table for DVP32 should appear as follows:

```
MACRO,R,B,L,
      NAM  $TP32,15
      ENT  $TP32
$TP32 DEC  -n  (n is the total number of subchannels)
      DEC  word 0
      DEC  word 1
       :
      END
```

Track Configuration for ICD Disk

The track map table for the ICD (Integrated Controller) Disks (HP 9895, 7906H, 7920H, and/or 7925H) contains the following information:

- Total number of subchannels defined.

The following information must also be specified for each subchannel:

- Number of sectors per track
- Cylinder number of track 0
- Number of surfaces included in subchannel
- Head number of track 0
- Address select number of disk drive
- Number of tracks on subchannel
- Number of spares allocated to subchannel
- Unit number for HP 9895 drives only.

Information that is required to properly configure a track on these disk models is given below (a full description of track configuration can be found in Chapter 3).

HP 9895	HP 7906H
64 words per sector 60 sectors per track 77 tracks per surface 2 surfaces per drive	64 words per sector 96 sectors per track 411 tracks per surface 4 surfaces per drive
HP 7920H	**HP 7925H**
64 words per sector 96 sectors per track 823 tracks per surface 5 surfaces per drive	64 words per sector 128 sectors per track 823 tracks per surface 9 surfaces per drive

Note The RTE ICD Disk Driver DVA32 treats a logical track as 64 words per sector, with the number of sectors per track dependent upon the subchannel definition.

ICD Disk Subchannels

The HP 12821A ICD Interface can address up to two disk controllers. Any combination of HP 9895, 7906H, 7920H and 7925H disk drives can be used. These subchannels are not directly related (one per platter) to the disk drive, and they are not restricted to eight subchannels.

Each subchannel is a contiguous group of tracks on a single drive. There may be more than one subchannel per drive, but subchannels cannot cross drive boundaries. The exact number of subchannels is specified by the user. There may be as many as 64 subchannels per interface card. Subchannels are numbered sequentially from zero; no numbers may be skipped.

ICD Disk Sectors

The discussion of sectors for the MAC Disks is also true for the HP 9895, 7906H, 7920H, and 7925H.

ICD Disk Tracks

The number of tracks on a disk drive is determined by multiplying the cylinders (or head positions) by the number of surfaces on the drive.

Disk Drive	Cylinders or Head Positions	# Surfaces	Maximum # of Tracks
HP 9895	77	2	154
HP 7906H	411	4	1644
HP 7920H	823	5	4115
HP 7925H	823	9	7407

Theoretically, the number of tracks could all be assigned to one subchannel; however, there are limitations. Peripheral disk subchannels must not have more than 32767 tracks (excluding spares) per subchannel. The combined size of the system and auxiliary subchannel (LU 2 and LU 3) must not exceed 1600 tracks, excluding spares.

Head positions (or cylinders) are numbered sequentially starting from 0. There is one head for each surface numbered sequentially from 0.

ICD Disk Surface Organization

Subchannels on a 9895 should match the anticipated floppy media in use, single or double sided (or both). Subchannels on a 7906H may be on from one to four surfaces. Subchannels on a 7920H may be on from one to five surfaces. Those on a 7925H may be from one to nine surfaces.

It is best to alternate surfaces (to minimize head movement) when more than one surface is used. For example, if track 0 is at cylinder (head position) 10 on head 0, then track 1 should be at cylinder 10 on head 1 and track 2 at cylinder 11 on head 0.

ICD Disk Address and Unit Numbers

The ICD (Integrated Controller Disk) address is a number associated with each 9895, 7906H, 7920H and 7925H disk drive. The ICD Address number is set (by the user) behind the front panel of the drive, and is always displayed on the front panel. They may be numbered 0 through 7. Do not change the ICD address specifications while the heads are loaded.

In addition the two drives on a 9895 controller are addressed by their respective unit numbers (0 or 1) which refers to the left and right drives, respectively.

Defining the ICD Track Map Table

When an extra disk interface is needed, tracks are mapped in a table defined as follows:

```
MACRO,R,B,L,O
      NAM   $TA32,15     ($TA32 must be type 8 or type 15 program)
      ENT   $TA32
$TA32 DEC   -n           n is the total number of subchannels
SC0   Word 0             See entry format below
      Word 1
      Word 2
      Word 3
      Word 4
SC1   Word 0             Repeat for next subchannel
        :
SCn-1 Word 0             Until all subchannels are defined
        :
END
```

There is a 13 subchannel restriction if you make $TA32 a type 8 program. Therefore, if the total number of subchannels is greater than 13, you will need to make $TA32 a type 15 program.

$TA32 Entry Format

	15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0			
word 0	#64-word sectors/track			
word 1	first cylinder #			
word 2	#surfaces	starting head#	addr num	
word 3	#tracks			
word 4	100	unit #		#spares

Where the # of sectors per track is based on 64-word sectors. The following apply:

HP 7906H/7920H	96	sectors/track
HP 7925H	128	sectors/track
HP 9895	60	sectors/track

Unit # is used for the HP 9895 (0 for the left drive, 1 for the right drive).

All unused fields should be set to zero.

Example:

Define three HP 7906H subchannels using two surfaces of the removable disk cartridge, and two subchannels for an HP 9895 using double-sided flexible media. The number of tracks on each HP 7906H subchannel is 76 plus 4 spare tracks per subchannel. Each subchannel starts at head 0.

```
MACRO,R,B,L,
      NAM   $TA32,15      ($TA32 must be a type 8 or type 15 program)
      ENT   $TA32
$TA32 DEC   -10           Total of ten subchannels
SC0   DEC   96            Number of 64-word sectors/track
      DEC   0             Subchannel 0 starts at cylinder 0
      OCT   04005         Two surfaces, head 0, select address 5
      DEC   76            76 tracks for subchannel 0
      OCT   100004        # spares=4, ICD controller indicator
SC1   DEC   96
      DEC   40            Second subchannel starts at cylinder 40 (4 sparet racks)
      OCT   04005
      DEC   76
      OCT   100004
SC2   DEC   96
      DEC   80            Third subchannel starts at cylinder 80 (4 sparet racks)
      OCT   04005
      DEC   76
      OCT   100004
      .     .
      .     .
SC3   DEC   60            Number of 64−word sectors/track
      DEC   0             Subchannel 3 starts of cylinder 0 on the 9895.
```

```
       OCT   04006          Two surfaces, head 0, select address 6.
       DEC   134            134 tracks (RTE standard for double−side media)
       OCT   100024         #spares = 20 controller indicator, unit 0 (left side)
SC4    DEC   60
       DEC   0
       OCT   04006
       DEC   134            134 tracks (RTE standard)
       OCT   102024         # spares = 20 ICD Controller indicator, Unit 1
                            (right side)
       END
```

Multiple HP 12821A Interface Card Operation

In order to increase disk throughput, the user may wish to include more than one HP 12821A interface card in the system generation. Performance can also be improved by putting LUs 2 and 3 on separate disks, each having their own HP 12821A interface card. This involves relocating a second copy of the DVA32 driver (named %DVC32) during the generation, and including entries in the equipment table, the device reference table, and the interrupt table for all disks on the second HP 12821A interface card. In addition to this, the user must supply a track map table ($TC32) to describe the subchannel configuration. The format of this table is identical with $TA32, but the name must be changed to satisfy the driver, DVC32. The user may wish to take the source (&$TA32) of the HP supplied ICD track map table and modify it to meet his particular requirements. Refer to the DVR32/DVA32 Driver Manual for additional information.

The track map table for DVC32 should appear as follows:

```
MACRO,R,B,L,
      NAM   $TC32,15      ($TC32 must be type 8 or type 15 program)
      ENT   $TC32
$TC32 DEC   -n            (n is the total number of subchannels)
      DEC   word 0
      DEC   word 1
        .
        .
      END
```

Multiple CPU Operation with 13037 Controller

Note The file system on RTE-6/VM does not support multiple CPU operation.

In a multiple CPU system environment, the MAC disk driver, DVR32, and the HP 13037B/C controller prevent destructive interference during transfers of data to and from the disk. The drivers and controller provide adequate protection if a CPU is not to share access to the same physical disk addresses with any other CPU.

If a file or set of files is to be shared by more than one CPU, a procedure is needed to prevent the following possible events:

a. CPU A reads a sector to update it.

b. CPU B reads the same sector to update it.

c. CPU A writes its updated sector back to the disk.

d. CPU B writes its updated sector back to the disk, destroying the effect of CPU A access.

To allow software to be written to effect multiple CPU – MAC system operation without destructive interference, the HP 13037B/C disk controller driver (DVR32) services a lock/unlock function call. This call can be issued from one CPU to lock the disk during an I/O operation or set of I/O operations. No other CPU can access the locked disk until an unlock function call is issued by the original CPU.

DVR32 Lock/Unlock Function Call

The I/O Control request is used to hold a Resource Number (RN) and, subsequently, to release the RN. The RN must be allocated and set as a global RN prior to issuing the I/O Control request. For a description of the I/O Control request and Resource Numbering, see the appropriate RTE Software System Programming and Operating Manual.

The FORTRAN calling sequence for an I/O Control request containing a lock/unlock function call is:

```
ICODE=3
ICNWD=control word
IRNUM=resource number
CALL EXEC (ICODE,ICNWD,IRNUM)
```

ICNWD defines a one-word octal value containing control information. For DVR32, control word bits 12-6 contain a function code for the following control states:

Function Code (Bits 12-6)	Meaning
15	Lock
00	Unlock

IRNUM is specified only for function code 15. IRNUM contains the RN to be cleared when the lock function call is executed. If a lock is currently in effect from another CPU, the calling program is suspended until the disk is available. If the lock is obtained immediately, the I/O Control request completes immediately. If a lock is already in force by this disk controller, the request completes with the RN cleared.

The lock/unlock function codes are provided to alleviate any CPU contention problem. If a CPU wishes to modify the same disk area as another CPU, the following code sequence could be executed from both units to prevent their interfering with each other:

```
ICODE=12B                          Allocate and set global RN
CALL  RNRQ(ICODE,IRNUM,ISTAT)

CALL  EXEC(3,IDLU+1500B,IRNUM)     Issue lock call, function code=15
  :
CALL  RNRQ(5,IRNUM,ISTAT)          Set/clear the RN
 :                                 Lock is granted by this point
  :
CALL  EXEC(1,IDLU,....)            Next, read the disk and modify data
  :
CALL  EXEC(2,IDLU,....)            Then, write it back
  :
CALL  EXEC(3,IDLU)                 Now, issue unlock call, function code=0
  :
  :
```

To use the lock/unlock function, each CPU operating system must support this function.

The sequence described previously for CPU A and CPU B using the lock/unlock function would now be:

Step 1. CPU A requests a lock from the driver and it is granted (no other CPU has a lock in force).

Step 2. CPU A reads a sector to update it.

Step 3. CPU B requests a lock from its driver. Because CPU A has a lock, CPU B must wait.

Step 4. CPU A writes its updated sector back to the disk.

Step 5. CPU A releases its lock.

Step 6. CPU B disk driver gets an interrupt from the disk controller informing it that the lock is now available and completes the lock requested by b at Step 3.

Step 7. CPU B reads the same sector to update it.

Step 8. CPU B writes its updated sector back to the disk. The sector now has both updates.

Step 9. CPU B releases its lock.

B

System Tables

This Appendix contains information about the following topics:

- SYSTEM COMMUNICATIONS AREA – Base page locations of area used for system communications.
- PROGRAM ID SEGMENT MAP – Format of ID segments kept in system area for user programs, ID segment extension, and short ID segments.
- DISK LAYOUT – Allocation of disk space for an RTE-6/VM system.

Other system tables relating to I/O considerations, such as the Equipment Table, Device Reference Table, and Driver Mapping Table are contained in the RTE-6/VM Index to the Operating Systems Manuals.

System Communication Area

This area is a block of storage in the system base page, starting at location 1645, that is used by RTE-6/VM to define request parameters, I/O tables, scheduling lists, operating parameters, memory bounds, etc. The Macroassembler allows relocatable programs to reference this area by absolute addresses 1645 through 1777 octal. User programs can read information from this area but cannot alter it because of the memory protect feature.

The contents and description of each location in this area are listed in Table B-1.

Table B-1. System Communications Area Locations

Octal Loc	Contents	Description
SYSTEM TABLE DEFINITION		
01645	XIDEX	Address of current program's ID extension
01646	XMATA	Address of current program's MAT entry
01647	XI	Address of index register save area
01650	EQTA	FWA of Equipment Table
01651	EQT#	Number of EQT entries
01652	DRT	FWA of Device Reference Table, word 1
01653	LUMAX	Number of logical units in DRT
01654	INTBA	FWA of Interrupt Table
01655	INTLG	Number of Interrupt Table Entries
01656	TAT	FWA of Track Assignment Table
01657	KEYWD	FWA of keyword block
I/O MODULE/DRIVER COMMUNICATION		
01660	EQT1	Addresses of first 11 words of current EQT entry (see 01771 for last four words)
01661	EQT2	
01662	EQT3	
01663	EQT4	
01664	EQT5	
01665	EQT6	
01666	EQT7	
01667	EQT8	
01670	EQT9	
01671	EQT10	
01672	EQT11	
01673	CHAN	Current DCPC channel number I/O address
01674	TBG	of time-base card EQT entry address of system TTY
01675	SYSTY	
SYSTEM REQUEST PROCESSOR/EXEC COMMUNICATION		
01676	RQCNT	Number of request parameters −1
01677	RQRTN	Return point address
01700	RQP1	Addresses of request parameters (set for a maximum of nine parameters
01701	RQP2	
01702	RQP3	
01703	RQP4	
01704	RQP5	
01705	RQP6	
01706	RQP7	
01707	RQP8	
01710	RQP9	
SYSTEM LISTS ADDRESSES		
01711	SKEDD	Schedule list
01712	PVCN	Privileged nest counter
01713	SUSP2	Wait Suspend list
01714	SUSP3	Available Memory list
01715	SUSP4	Disk Allocation list
01716	SUSP5	Operator Suspend list

Table B-1. System Communications Area Locations (Cont.)

Octal Loc	Contents	Description
PROGRAM ID SEGMENT DEFINITION		
01717	XEQT	ID segment address of current program
01720	XLINK	Linkage
01721	XTEMP	Temporary (five words)
01726	XPRIO	Priority word
01727	XPENT	Primary entry point
01730	XSUSP	Point of suspension
01731	XA	A-Register at suspension
01732	XB	B-Register at suspension
01733	XEO	E and overflow register suspension
SYSTEM MODULE COMMUNICATION FLAGS		
01734	OPATN	Operator/keyboard attention flag
01735	OPFLG	Operator communication flag
01736	SWAP	RT disk-resident swapping flag
01737	DUMMY	I/O address of dummy interface flag
01740	IDSDA	Reserved
01741	IDSDP	Reserved
MEMORY ALLOCATION BASES DEFINITION		
01742	BPA1	FWA user base page link area
01743	BPA2	LWA user base page link area
01744	BPA3	FWA user base page link
01745	LBORG	FWA of resident library area
01746	RTORG	FWA of real-time COMMON
01747	RTCOM	Length of real-time COMMON
01750 D	RTDRA	FWA of real-time partition
01751 D	AVMEM	LWA—1 of real-time partition
01752	BGORG	FWA of background COMMON
01753	BGCOM	Length of background COMMON
01754 D	BGDRA	FWA of background partition
UTILITY PARAMETERS		
01755	TATLG	Negative length of track assignment table
01756	TATSD	Number of tracks on system disk
01757	SECT2	Number of sectors/track on LU 2 (system)
01760	SECT3	Number of sectors/track on LU 3 (aux.)
01761	DSCLB	Disk address of library entry points
01762	DSCLN	Number of user available library entry points
01763	DSCUT	Disk address of relocatable disk-resident library
01764	SYSLN	Number of system library entry points
01765	LGOTK	LGO: LU#, starting track, number of tracks (same format at ID segment word 26)
01766	LGOC	Current LGO track/sector address (same format at ID segment word 26)
01767	SFCUN	LS: LU# and disk address (same format at ID segment word 26)
01770	MPTFL	Memory protect ON/OFF flag (0/1)
01771	EQT12	Address of last four words of current EQT
01772	EQT13	
01773	EQT14	
01774	EQT15	
01775 D	FENCE	Memory protect fence address
01776	VMASWP	VMA swap flag
01777	BGLWA	LWA memory background partition
The letter D indicates the contents of the location are set dynamically by the dispatcher.		

Program ID Segment

Each user program has a 36-word ID segment located in memory that contains static and dynamic information defining the properties of the program. The static information is set during generation time or when the program is loaded online. The dynamic information is maintained by the operating system.

The number of ID segments contained in a system is established during system generation and is directly related to the number of programs that can be in main memory at any given time. If all the ID segments are in use, no more programs can be added online unless some other existing program is first offed (removed from the system) to recover an ID segment.

The format of the ID segment is illustrated in Figure B-1. Each ID segment's address is located in the Keyword Table (see location 01657).

Cathy will install this figure

8100-2A

Figure B-1. ID Segment Format

where:

*	=	words used in short ID segments for program segments
TM	=	temporary load (copy of ID segment is not on the disk)
ML	=	memory lock (program may not be swapped)
DS	=	for use by distributed systems software
SS	=	short segment (indicates a nine-word segment)
Type	=	specified program type (1-5)
NA	=	no abort (instead, pass abort errors to programs)
NP	=	no parameters allowed on reschedule
NS	=	no suspension on I/O requests (instead, pass control to program)
W	=	wait bit (waiting for program whose ID segment address is in word 2)
A	=	abort on next list entry for this program
FS	=	file system bit (this program has files open)
O	=	operator suspend on next schedule attempt
LP	=	load in progress; program is being dispatched from disk
R	=	resource save (save resources when setting dormant)
D	=	dormant bit (set dormant on next schedule attempt)
Status	=	current program status
T	=	time list entry bit (program is in the time list)
BA	=	batch (program is running under batch)
FW	=	father is waiting (father scheduled with wait)
M	=	Multi-Terminal Monitor bit
AT	=	attention bit (operator has requested attention)
RM	=	reentrant memory must be moved before dispatching program
RE	=	reentrant routine now has control
PW	=	program wait (other program wants to schedule this one)
RN	=	Resource Number either owned or locked by this program
RP	=	reserved partition (only for programs that request it)
MPFI	=	memory protect fence index
DE	=	defer EXEC 6 (terminate program) request
LU	=	0 if LU 2, 1 if LU 3
SD	=	clear this ID segment on program termination
DEBUG	=	Program debug state: 00=not being debugged, 01=being debugged, 10=being monitored, 11=being debugged and waiting for initial load or $LIBZ call.
SEQCNT	=	sequence counter
SH	=	shareable EMA flag (program or progeny uses shareable EMA)
DC	=	don’t copy flag
CP	=	copy flag
DS	=	DS program
Session ID	=	system LU of terminal where program was loaded
MS	=	multi-level segmentation flag
MP	=	program is using modified maps for I/O
E	=	EXEC 4 (track allocation) request was made by program
DB	=	Reserved

High Main Address +1 (23)

This is the address of the first word after the root for MLS programs. If there are any memory-resident nodes, this is the first word address of the page where the memory-resident node starts.

High Base Page Address +1 (25)

For MLS programs, the contents of this word are the same as the contents of location 1743B, which is the address of the last user available link, that is, the last available link address in RTE.

Open Flag Word (31)

SEQCNT	=	sequence counter. Each time a program is aborted or terminates (unless saving resources) the counter is incremented. The counter value is used to build FMP open Flags.
SH	=	Shareable EMA flag. Indicates that the program or one of the ancestors (that is, father, grandfather, etc., program) uses shareable EMA.
DC	=	Don't copy flag. Set by the generator (if 128 is added to program type) or the loader (using Don't copy op-code).
CP	=	Copy Flag. Indicates that the program is a copy.
Session ID	=	System LU of terminal that program was scheduled from.
MD	=	Memory/Disk-resident node in control flag.

Session Word (32)

The session word identifies the owner of a program.

A negative value represents the logical unit number of the terminal from which the program was invoked (not under session).

A positive value represents the address of the SST length word of the Session Control Block (SCB) for the session currently using this program (under session).

Programs scheduled by interrupt have a zero in this word.

Multilevel Segmentation Word 1 (33)

MS = Multilevel segmentation flag.

#pages disk-res = Number of pages of the longest disk-resident node path not including the root.

#pages memory-res = Number of pages of all the memory-resident nodes and the root.

Multilevel Segmentation Word 2 (34)

SM = Save maps bit. It informs the operating system that the user map has been modified and must be saved on a context switch.

#pages dynamic buf = Number of pages of dynamic buffer area. This is the space between the last word +1 of the longest path and the start of the EMA area.

E = Exec 4 track allocation request was made by program.

# of swap tracks = Number of swap tracks used to swap the code area (not the EMA area) of the program.

ID Segment Extensions

Each EMA program requires a 5-word ID segment extension in addition to its 36-word ID segment. The number of ID extensions contained in the system is also set at generation time, and if all are in use, no more EMA programs can be added online. SHEMA programs can share ID segment extensions. The format of the ID segment is illustrated in Figure B-2.

Cathy will install this figure also

8100-14

Figure B-2. ID Segment Extension

where:

SW = 0 PTE table does not contain valid data.
= 1 PTE table is still intact (working set has not been moved).

DE = 0 if the EMA size was specified by the user.
= 1 if the EMA size is allowed to default to the maximum size available to the system.

S = Shareable EMA.

Short ID Segments

Short ID segments require ten words and are used only for program segments. A short ID segment is required for each segment of a segmented program. If no empty short ID segments are available during an online load, a standard 36-word ID segment will be used. The information contained in a short ID segment is illustrated in Figure B-1.

RTE-6/VM System Disk Layout

Figure B-3 illustrates how disk space is allocated when a RTE-6/VM system is generated.

Cathy will install this figure also

8100-8

Figure B-3. RTE-6/VM System Disk Layout

C

Session Monitor Tables

This appendix contains information on the following:

- Session Control Block (SCB)
- Session Switch Table (SST) and Station Configuration Table
- Account File Structure

Session Control Block (SCB)

A Session Control Block (SCB) is established for each user who has successfully logged-on to the system. The SCB contains the information necessary to identify the user to the system and describe his capabilities in terms of command processing and I/O addressing space.

The format of the SCB is shown in Figure C-1.

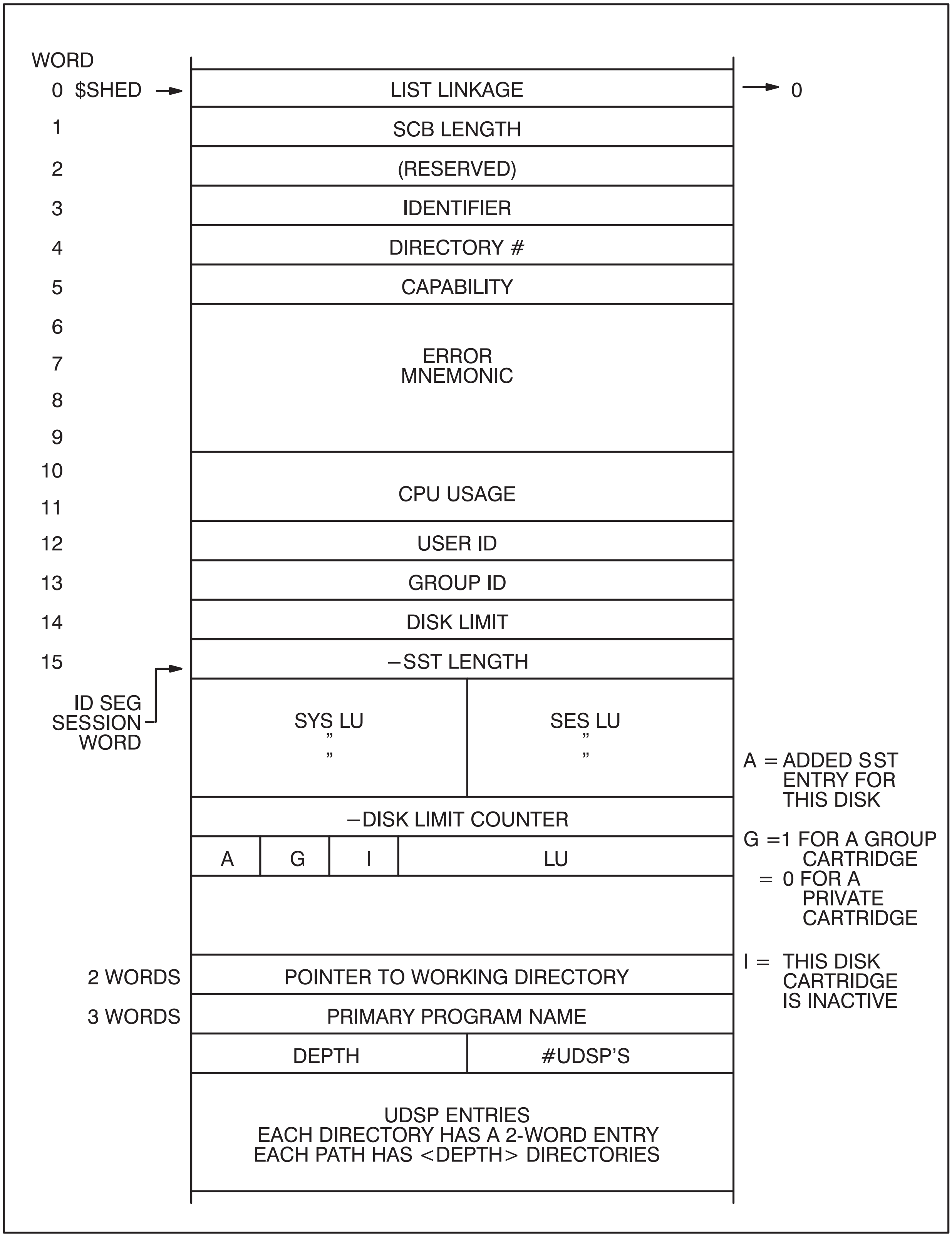


Figure C-1. Session Control Block (SCB)

Session Switch Table (SST) and Station Configuration Table (CT)

When operating in the session environment, every I/O request is routed to the appropriate I/O device via the Session Switch Table (SST). Each SST entry describes a session LU, which the user addresses, and associated system LU where the I/O request will actually be directed. The SST describes the session user's I/O addressing capabilities by defining the system LUs the user has access to and the associated session LUs by which the user accesses them.

When the user makes an I/O request, the SST is searched for the specified session LU. If the requested LU is found, it is switched to the associated system LU as specified in the SST entry and the I/O request is processed. Zero-offset addressing makes the internal representation of the LU in the Session Control Block (SCB) one less than the assigned number. If the requested LU is not found, an error is returned (IO12, LU not defined for this session).

The Session Switch Table is maintained in memory as part of the Session Control Block (SCB). The format of the SST is shown in Figure C-2. System LUs can be integer numbers between 1 and 254. Session LUs can be integer numbers between 1 and 63. Session LUs are assigned:

- at log-on, via user and group account file entries.
- online using the SL command.
- at log-on, via Station Configuration Table (CT) entries.

The Station Configuration Table contains the default logical units to be used for the specific device logical units at this station. Each station (terminal) logical unit defined in the Station Configuration Table has associated with it a set of device logical units that are assigned default logical units to be used when a user logs on at that station (terminal). The default logical unit associated with the station itself is always 1.

At log-on, these default values are written from the Station Configuration Table in the account file into the user's Session Control Block (SCB), unless overridden by entries in this particular user's SST.

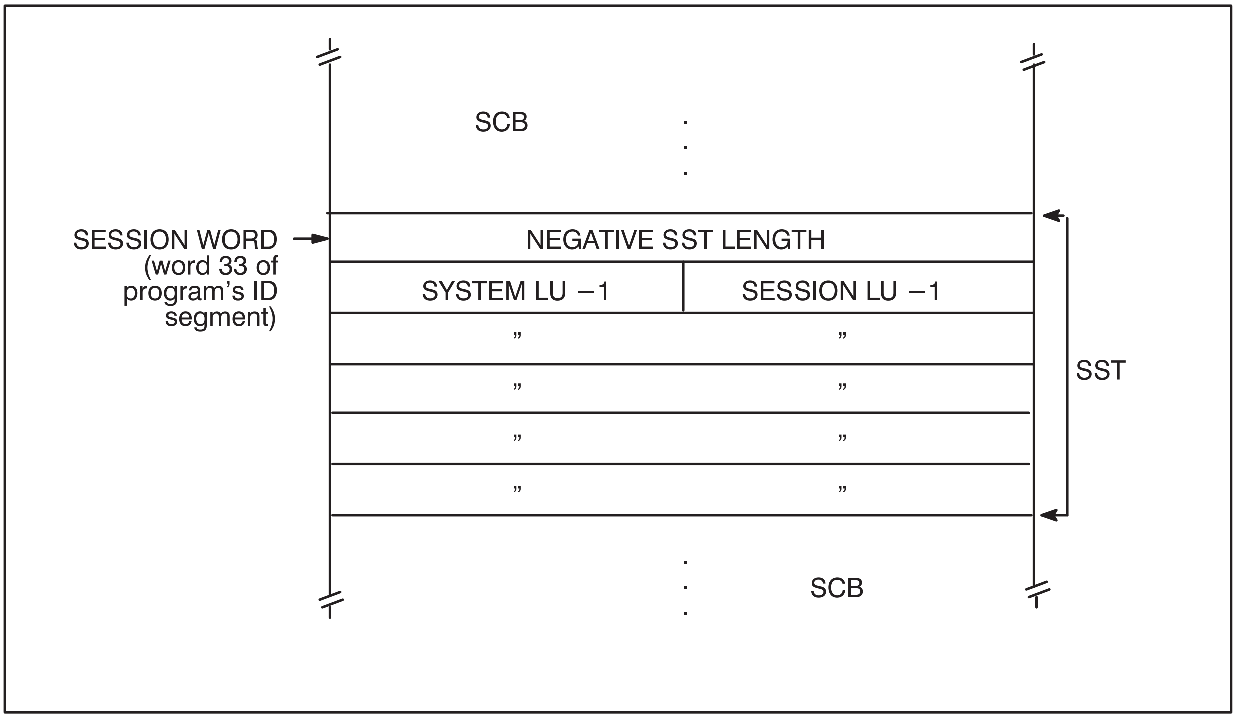


Figure C-2. Session Switch Table (SST) Format

Account File Structure

The account file contains information necessary to maintain the Session Monitor Accounts System. This section illustrates the structure of the accounts file and then presents each part in more detail.

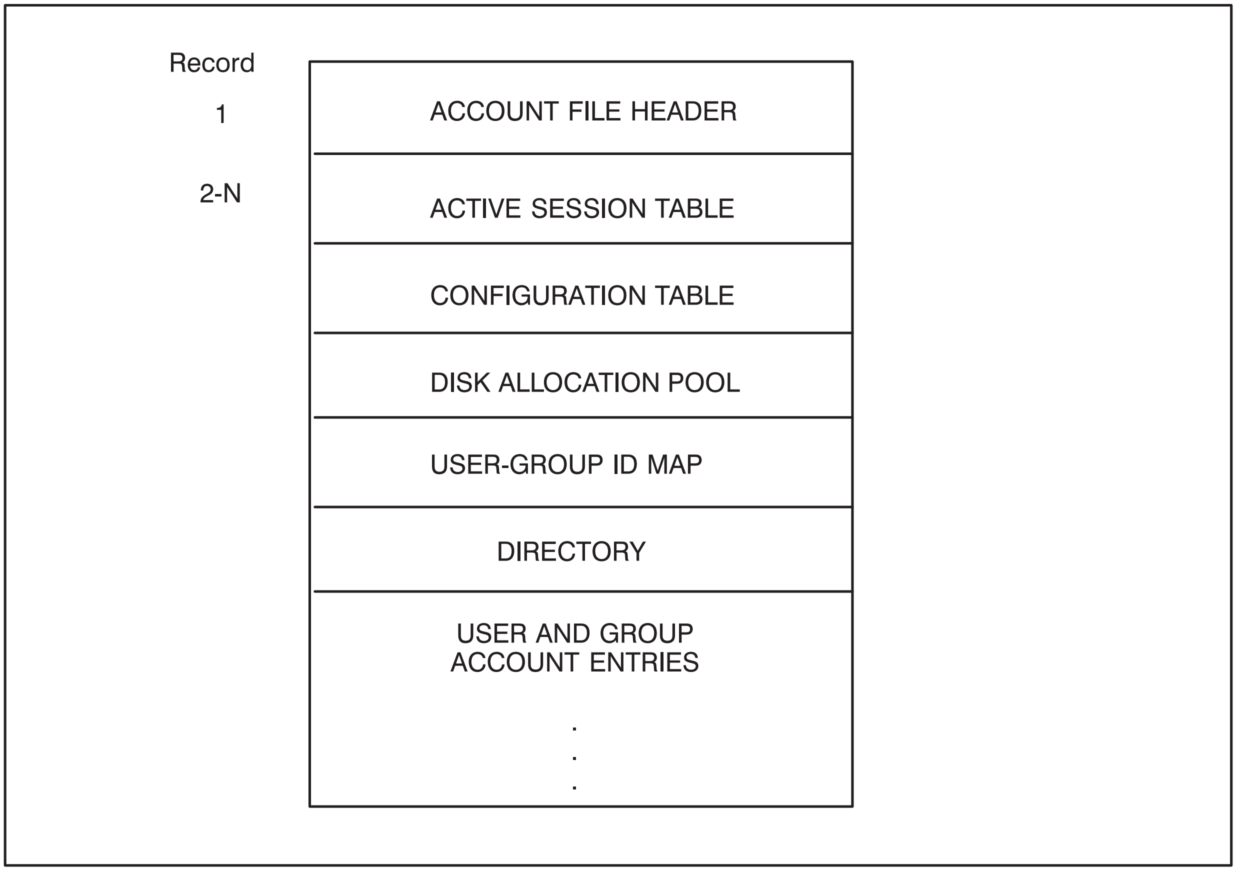


Figure C-3. Account File Structure

WORD		
1	LOCATION OF ACTIVE SESSION TABLE	
2	LOCATION OF CONFIGURATION TABLE	
3	LOCATION OF DISK POOL	
4	LOCATION OF USER/GROUP ID MAP	
5	LOCATION OF DIRECTORY	
6	LOCATION OF 1ST ACCOUNT ENTRY	
7-9	SYSTEM MESSAGE FILE	
10	SECURITY CODE	
11	CARTRIDGE	
12	# OF CHARS IN PROMPT STRING	← 0 if using default prompt
13-22	PROMPT STRING	
23	LOWEST PRIVATE ID USED	
24	HIGHEST GROUP ID USED	
25	RESOURCE NO.	
26	LU # OF MSG. FILES	
27	MEMORY ALLOCATION SIZE (WDS)	← If bit 15=1, use session monitor memory allocation
28	− SESSION LIMIT	
29	NUMBER OF ACTIVE SESSIONS	
30	SHUT DOWN FLAG	
31	COPY OF SESSION LIMIT	
32	CLASS NUMBER	
33	LENGTH OF CONFIGURATION TABLE	
34	IRN2	
35	DISK POOL LENGTH	

Figure C-4. Account File Header

Active Session Table

WORD	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	LOGICAL UNIT (0 IF FREE ASB)															
2	YEAR*					MINUTES					SECONDS					
3	REV		JULIAN DAY									HOURS				
4	DIRECTORY ENTRY NUMBER															
	. . .															

Active Session Block (ASB)

* NOTE, the year is an offset from 1978:

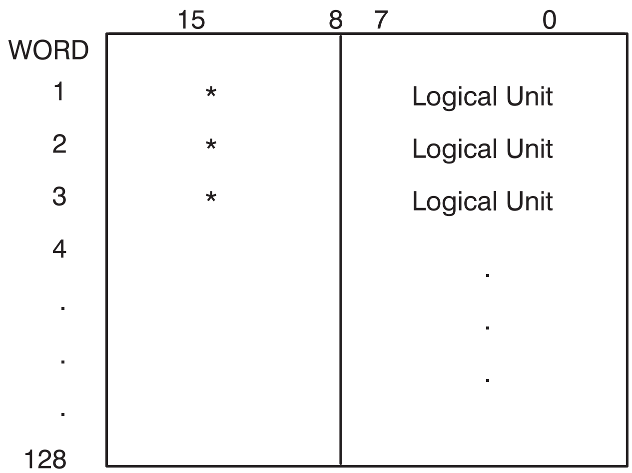
YEAR 0 = 1978
YEAR 1 = 1979
.
.

Configuration Table

WORD			
1	N		LENGTH OF ENTRY
2	LU	1	DEFAULT LU, TERMINAL
3	LU	4	DEFAULT LU, LEFT CTU
4	LU	5	DEFAULT LU, RIGHT CTR
5	LU	6	DEFAULT LU, LINE PRINTER
. . .	N . .		LENGTH OF NEXT ENTRY

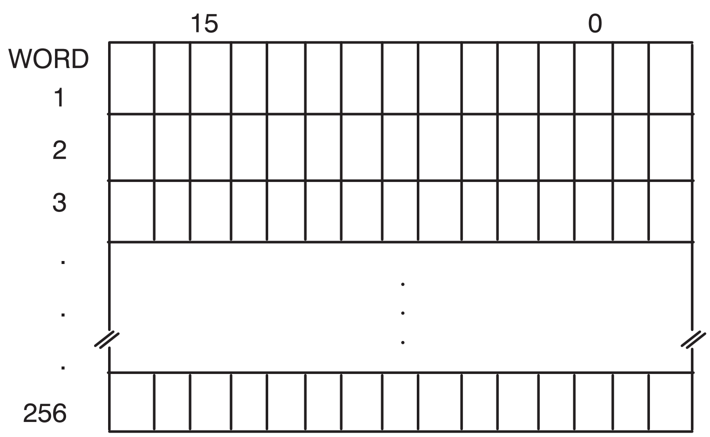
Figure C-5. Active Session Table and Configuration Table

Disk Allocation Pool



* Reserved for future use

User/Group ID Map



Bit=1 Indicates ID is assigned to an account

Figure C-6. Disk Allocation Pool and User/Group ID Map

Account File Directory

WORD	15 — 8	7 — 0	
1	# Chars User	# Chars Group	0 = End of Directory −1 = Free Entry −2 = Extension
2-6	User Name		
7-11	Group Name		
12	User ID		(0 if entry is for a group account)
13	Group ID		
14	Group Acct Record #		If Bit 15=1, Account is in 2nd 64 word (0 if entry is for a group account)
15	User Acct Record #		
16	*		

* Reserved for future use

Figure C-7. Account File Directory

User Account Entry

WORD	15 — 8	7 — 0	
1		Chars in Passwd	Bit 15 = 1 indicates account extends to 2nd block
2-6	Password		
7-9	User HELLO File		
10	Security Code		
11	Cartridge		
12-14	Primary Program Name		
15	UDSP Depth	#UDSP's	
16			
17-19	User Message File		
20-21			
22	Capability		
23-24	Last Log-Off Time		
25-26	Cumulative Time (Seconds)		2 Words
27-28	CPU Usage (Tens of Milliseconds)		2 Words
29	User ID		
30	Group ID		
31	Disk Limit		
32	Grp.SST Length	#SST Spares	
33	User/Group SST Length (Total)		
.	System LU " "	Session LU " "	User SST
.	System LU " "	Session LU " "	Group SST
.			
64			If bit 15 of word 1 is a 1, then this word is the record number of 2nd block of account

Figure C-8. User Account Entry

Group Account Entry

WORD	15	0	
1	Group ID		Bit 15 = 1 indicates account extends to 2nd half of block
2 3	Cumulative Time		
4 5	Cumulative CPU Usage		
6	− Group SST Length		
. . .	System LU . . .	Session LU . . .	

Figure C-9. Group Account Entry

D

DCB and Directory Formats

This Appendix contains information on the following:

- Data Control Block (DCB) Format
- Cartridge Directory Format
- File Directory Format

Data Control Block Format

Cathy will install this illustration.

8100-4A

Figure D-1. Data Control Block Format

Legend for Data Control Block

Word		Contents	
0	File Directory Address:		
	Sector offset:	bits 15-13	= entry offset from the beginning of the block (origin 0)
	Block offset	bits 12-7	= block offset from the beginning of the file directory track
	(DS) Opened by DS:	bit 6	= 1 – opened by DS transparency software
	LU number	bits 5-0	= LU number of the file directory or of the file if on disk
5	Spacing Code:	bit 15	= 1 – backspace legal
	(type 0 file)	bit 0	= 1 – forward space legal
6	End-of-File Code:	01 lu	= EOF on magnetic tape
	(type 0 file)	10 lu	= EOF on paper tape
		11 lu	= EOF on line printer
7	Status Information		
	(WA) Write Allowed:	bit 15	= 1 – write to file allowed
			= 0 – write to file not allowed
	(PF) Partially Full:	bit 14	= 1 – DCB is only partially full
			= 0 – DCB is full
	DCB Buffer:	bits 13-7	= number of blocks in the DCB buffer
	(FM) File Modify:	bit 6	= 1 – file has been modified
			= 0 – file has not been modified
	(EX) Extendable:	bit 5	= 1 – file is not extendable
			= 0 – file is extendable
	(RA) Read Allowed:	bit 4	= 1 – read from file allowed
			= 0 – read from file not allowed
	(OM) Open Mode:	bit 3	= 1 – update open
			= 0 – standard open
	(IB) In Buffer Flag:	bit 2	= 1 – data in DCB buffer
			= 0 – data not in DCB buffer
	(EF) EOF Read Flag:	bit 0	= 1 – EOF has been read
			= 0 – EOF has not been read
	(WR) To Be Written:	bit 0	= 1 – data in DCB buffer to be written
			= 0 – data in DCB buffer not to be written

9 Open/Close Indicator: If open, contains ID segment location of program performing open. If closed, set to zero.

Cartridge Directory Format

The cartridge directory is located in the system area on LU 2. Its length is two blocks.

Cathy will install this illustration.

8100-45A

Figure D-2. Cartridge Directory Format

File Directory

The first entry in each File Directory is the specification entry for the cartridge itself. The directory starts on the last FMP track of each cartridge in sector zero on all disks. The directory blocks are written using sector skipping. The directory sector address can be obtained from the block address by the following formula:

```
sector address = (block *14) modulo S/T
```

where S/T is the number of sectors per track. Directory blocks are 128 words long. Each Directory entry is l6 words long.

Cathy will install this illustration.

8100-13

Figure D-3. File Directory

Disk File Directory

word	bit 15–8	bit 7–0	CR Parameters:
0–2	6-character file name		file name
3	file type (1 thru 32767)		
4	starting track		
5	extent #	starting sector	
6	file size in	+ sector or − chunks	
7	record length (Type 2 only)		
8	security code		
9–15			Open Flags 15 = 1 for exclusive open 14-12 = reserved 11-8 = sequence counter 7-0 = keyword offset of opening program's ID segment

word 0 = 0 if the last entry in directory;
= −1 if file is purged

Type 0 File Directory Entry

The entries for non-disk (type 0) files differ from those for disk files in words 3 through 7:

word	bit 15 … 0	CR Parameters:
3	0 (file type default)	
4	logical unit number	
5	end of file subfunction	← EO, LE, PA or control
6	spacing code	← BS, FS, or BO
7	input/output code	← RE, WR, or BO

Word 5-7 are octal codes:

end-of-file subfunction = 01LU for MT(EO)
10LU for paper tape (LE)
11LU for line printer (PA)
or subfunction code

spacing code = bit 15 = 1 backspace legal (BS)
bit 0 = 1 forward space legal (FS)

input/output code = bit 15 = 1 input legal (RE)
bit 0 = 1 output legal (WR)

E

Modifying System Tables

This appendix contains information on modifying the following:

- Command Capability Table ($CMND).
- File Manager Command Table (C.TAB).
- Shareable EMA Partition Table ($EMTB).

Command Capability Table

Note Hewlett Packard does not support modified command capability tables or modified shareable EMA partition tables.

The capability level of each command is defined in the operating system capability table, $CMND. In $CMND, each command is defined by a two-word entry of the form:

<table>
<tr><td>15</td><td>14</td><td>8</td><td>6</td><td>0</td></tr>
<tr><td colspan="3">CHAR1</td><td colspan="2">CHAR2</td></tr>
<tr><td>P</td><td>R</td><td colspan="2"></td><td>NUM</td></tr>
</table>

where:

CHAR1 and CHAR2 = the two character ASCII command.

P = 0 If any number of parameters allowed.

= 1 If a limitation is placed on the number of parameters allowed.

NUM = The maximum number of parameters allowed with this command (specified when P=1).

R = 0 No reference check required.

= 1 Program specified for first parameter of command must be attached to this session (ID segment word 33 of program must equal word 33 of caller) or program must be non-session (word 33 equals zero).

The command capability level associated with a command is determined by the position of the command entry relative to level pointers located at the head of the table. Refer to the following listing for details.

If you wish to substitute your own command table for the HP supplied table, it must be specified AFTER the operating system relocatables during generation.

```
ASMB,R,L,C,Q
*     NAME:   $CMND
*     SOURCE: 92084-18463
*     RELOC:  PART OF 92084-12022
*     PGMR:   G.L.M
*
*  ****************************************************************
*  * (C) COPYRIGHT HEWLETT-PACKARD COMPANY 1981.  ALL RIGHTS      *
*  * RESERVED.  NO PART OF THIS PROGRAM MAY BE PHOTOCOPIED,       *
*  * REPRODUCED OR TRANSLATED TO ANOTHER PROGRAM LANGUAGE WITHOUT*
*  * THE PRIOR WRITTEN CONSENT OF HEWLETT-PACKARD COMPANY.        *
*  ****************************************************************
*
       NAM $CMND,0 92084-1X463 REV.2121
       ENT $CMND
*
*
$CMND  DEF EINDX     DEFINE THE ADDRESS OF HIGHEST CAPABILITY
       DEF BEGIN     DEFINE BEGINNING OF TABLE
       DEF END       DEFINE END OF TABLE
*
L60    DEC -60       LEVEL 60
L60A   DEF BEGIN     DEFINE START OF THIS CAPABILITY
*
L50    DEC -50
L50A   DEF L.50
*
L30    DEC -30
L30A   DEF L.30
*
L10    DEC -10
L10A   DEF L.10
*
L00    NOP
L00A   DEF L.00
*
EINDX  EQU *-2
*
*
*
       ORG $CMND
       BSS L10A-L30A
       BSS L30A-L50A
       BSS L50A-L60A
       ORR
*
       SKP
L.60   EQU *
BEGIN  ASC 1,QU
       OCT 0
       ASC 1,DN
       OCT 0
       ASC 1,LU
       OCT 0
       ASC 1,EQ
       OCT 0
```

```
       ASC 1,TO
       OCT 0
       ASC 1,BL
       OCT 0
       ASC 1,TM
       OCT 0
       ASC 1,OF
       OCT 0
       ASC 1,BR
       OCT 0
       ASC 1,GO
       OCT 0
       ASC 1,SS
       OCT 0
       ASC 1,DB
       OCT 0
       ASC 1,AG                                                  CLC810127
       OCT 0                                                     CLC810127
       ASC 1,SN                                                  CLC810127
       OCT 0                                                     CLC810127
       ASC 1,CU
       OCT 0
       ASC 1,DB                                                  CLC810201
       OCT 0                                                     CLC810201
       ASC 1,UL                                                  CLC810318
       OCT 0                                                     CLC810318
*
L.50   ASC 1,IT
       OCT 0
       ASC 1,L3       ABILITY TO ADD AN ENTRY IN SST -- SL CMND
       OCT 0
       ASC 1,AS
       OCT 0
       ASC 1,UR
       OCT 0
       ASC 1,ON
       OCT 0
       ASC 1,PR
       OCT 0
*
L.30   ASC 1,RU
       OCT 0
       ASC 1,OF
       OCT 40000
       ASC 1,SS
       OCT 40000
       ASC 1,GO
       OCT 40000
       ASC 1,RT
       OCT 0
       ASC 1,SZ
       OCT 0
       ASC 1,WS                                                  CLC810127
       OCT 0                                                     CLC810127
       ASC 1,VS                                                  CLC810127
       OCT 0                                                     CLC810127
       ASC 1,VL                                                  CLC810310
       OCT 0                                                     CLC810310
       ASC 1,L2      LEVEL 2 SL CMND -- SPOOL AN LU
```

```
        OCT 0
*
L.10    ASC 1,FL
        OCT 0
        ASC 1,RS
        OCT 0
        ASC 1,QU
        OCT 100000
        ASC 1,BL
        OCT 100000
        ASC 1,ST
        OCT 0
        ASC 1,BR
        OCT 40000
        ASC 1,EQ
        OCT 100001
        ASC 1,SL
        OCT 0
        ASC 1,TO
        OCT 100001
        ASC 1,TE
        OCT 0
        ASC 1,WH
        OCT 0
        ASC 1,TI
        OCT 0
        ASC 1,UP
        OCT 0
        ASC 1,EN
        OCT 0
*
L.00    ASC 1,OP
        OCT 0
        ASC 1,HE
        OCT 0
*
END     EQU *-2
        END $CMND
```

File Manager Command Table

The file manager command table follows on subsequent pages. The capability levels assigned to various commands depends on their position within the table relative to table pointers located at the front of the command table. Each command is defined by a two-word entry. To change the capability level of a command, relocate the two-word entry to the appropriate table section for the desired capability level. Do not modify the two-word entry.

Then reassemble the modified capability table and relocate it after the file manager modules (that is, %BMPG1,...) during generation. (You can ignore GEN05 and GEN08 errors here.)

There is one special consideration when choosing a name for the &C*TAB relocatable. Since the RTE-6/VM generator considers all *'s as the start of a comment, the relocatable cannot contain an asterisk.

Note Hewlett Packard does not support modified command capability tables.

```
 ASMB,R,L,C
*     NAME:   C.TAB
*     SOURCE: 92084-18135
*     RELOC:  PART OF 92084-12003
*     PGMR:   G.A.A., B.L., D.C.L., S.P.K., E.D.B.
*
*  **************************************************************
*  * (C) COPYRIGHT HEWLETT-PACKARD COMPANY 1981.  ALL RIGHTS     *
*  * RESERVED.  NO PART OF THIS PROGRAM MAY BE PHOTOCOPIED,      *
*  * REPRODUCED OR TRANSLATED TO ANOTHER PROGRAM LANGUAGE WITHOUT*
*  * THE PRIOR WRITTEN CONSENT OF HEWLETT-PACKARD COMPANY.       *
*  **************************************************************
*
      NAM C.TAB,8 92084-1X135 REV.2121 811113
      ENT C.TAB
*
*     SET UP SEGMENT AND ROUTINE NUMBERS.
*
R0    EQU 0
R1    EQU 400B
R2    EQU R1+R1
R3    EQU R2+R1
R4    EQU R3+R1
R5    EQU R4+R1
R6    EQU R5+R1
R7    EQU R6+R1
R8    EQU R7+R1
R9    EQU R8+R1
R10   EQU R9+R1
      SPC 1
S0    EQU 60B
S1    EQU S0+1
S2    EQU S0+2
S3    EQU S0+3
S4    EQU S0+4
S5    EQU S0+5
```

```
S6    EQU S0+6
S7    EQU S0+7
S8    EQU S0+8
S9    EQU S0+9
SA    EQU 101B
SB    EQU SA+1
*
*     THIS IS THE COMMAND DISPATCH TABLE FOR THE FMGR PROGRAM.
*     EACH COMMAND ID IS FOLLOWED BY ITS ADDRESS.
*     FOR ROUTINES IN THE HOME SEGMENT THIS IS AN ADDRESS (DEF XX).
*     FOR ROUTINES IN OTHER SEGMENTS IT IS THE ASCII SEGMENT
*     SUFFIX IN THE LOW HALF OF THE WORD AND THE ROUTINE
*     NUMBER IN THAT SEGMENT IN THE HIGH HALF OF THE WORD.
*     .PARS BREAKS THESE APART BY THE ADDRESS BEING 0<ADDh10000B
*     FOR SEGMENT ADDRESS.
*
*     COMMANDS WITH THE SIGN BIT SET INDICATE THAT THE COMMAND
*     NEED NOT SATISFY ALL THE SYNTAX RESTRICTIONS IMPOSED ON
*     OTHER COMMANDS.
*
      SPC 1
*
*     SESSION MONITOR COMMAND CAPABILITY LEVELS
*
C.TAB DEF BEGIN
      DEF ENDS
      DEF SCMD
L1    DEC 1
L1A   DEF LV10
L10   DEC 10
L10A  DEF LV20
L20   DEC 20
L20A  DEF LV30
L30   DEC 30
L30A  DEF LV40
L40   DEC 40
L40A  DEF LV50
L50   DEC 50
L50A  DEF LV60
L60   DEC 60
L60A  DEF SCMD
ENDS  DEF NONSM
ENDT  DEF END
      SPC 1
*
*     STRUCTURE CHECKS
*
      ORG C.TAB
      BSS ENDT-ENDS
      BSS ENDS-L60A
      BSS L60A-L50A
      BSS L50A-L40A
      BSS L40A-L30A
      BSS L30A-L20A
      BSS L20A-L10A
      BSS L10A-L1A
      ORR
      SPC 1
BEGIN EQU *
```

```
      NOP             NULL COMMAND (TR)
      DEF TR..
      ASC 1,TR
      EXT TR..
      DEF TR..
      ASC 1,EX
      EXT EE..
      DEF EE..
      OCT 151531      ”SY” WITH SIGN BIT SET
      ABS S7+R2
LV10  ASC 1,??        <<CAPABILITY LEVEL 10 COMMANDS>>
      ABS S7+R1
      OCT 125052      ”**” WITH SIGN BIT SET
      DEF COMM
      OCT 125000      ”*<NULL>” WITH SIGN BIT SET
      DEF COMM
      OCT 125040      ”*<BLANK>” WITH SIGN BIT SET
      DEF COMM
      ASC 1,LI
      ABS S9+R1
      ASC 1,CL
      ABS S9+R0
      ASC 1,DL
      ABS S3+R1
      ASC 1,MC
      ABS S4+R3
      ASC 1,DC
      ABS S4+R4
      ASC 1,WH
      ABS SB+R3                                          800221
      OCT 151515      ”SM” WITH SIGN BIT SET
      ABS SA+R0
      ASC 1,ME
      ABS SA+R1
      ASC 1,AC
      ABS S4+R5
LV20  ASC 1,CR        <<CAPABILITY LEVEL 20 COMMANDS>>
      ABS S8+R1
      ASC 1,ST
      ABS S0+R2
      ASC 1,DU
      ABS S0+R3
      ASC 1,PU
      ABS S2+R2
      ASC 1,RN
      ABS S6+R4
      ASC 1,CO
      ABS S0+R1
      ASC 1,PK
      ABS S0+R0
      ASC 1,CN
      ABS S5+R4                                          800221
      ASC 1,LL
      ABS S4+R0
      ASC 1,SV
      ABS S4+R2
      OCT 142120      ”DP” WITH SIGN BIT SET
      EXT DP..
      DEF DP..
```

```
      OCT 140516    "AN" WITH SIGN BIT SET
      ABS S5+R3                                         800221
      OCT 141524    "CT" WITH SIGN BIT SET
      ABS S5+R5                                         800221
LV30  ASC 1,SP      <<CAPABILITY LEVEL 30 COMMANDS>>
      ABS S8+R0
      OCT 151125    "RU" WITH SIGN BIT SET
      ABS SB+R1                                         800221
      ASC 1,RP
      ABS SB+R0                                         800221
      ASC 1,OF
      ABS S6+R3
      ASC 1,RT
      ABS S6+R2
      ASC 1,JO
      ABS S6+R0
      ASC 1,EO
      ABS S6+R1
      ASC 1,CS
      ABS S3+R0
      ASC 1,AB
      EXT AB..
      DEF AB..
      ASC 1,TL
      ABS S5+R0                                         800221
LV40  ASC 1,SE      <<CAPABILITY LEVEL 40 COMMANDS>>
      EXT SE..
      DEF SE..
      ASC 1,IF
      EXT IF..
      DEF IF..
      ASC 1,CA
      EXT CA..
      DEF CA..
      OCT 150101    "PA" WITH SIGN BIT SET
      ABS S5+R1                                         800221
LV50  ASC 1,LO      <<CAPABILITY LEVEL 50 COMMANDS>>
      ABS S4+R1
LV60  ASC 1,IN      <<CAPABILITY LEVEL 60 COMMANDS>>
      ABS S2+R1
      ASC 1,VL
      ABS S4+R7
SCMD  ASC 1,SL      <<SPECIAL SESSION COMMANDS>>
      ABS S6+R5
      OCT 144105    "HE" WITH SIGN BIT SET
      ABS SB+R2                                         800221
      OCT 152105    "TE" WITH SIGN BIT SET
      ABS S5+R2                                         800221
NONSM ASC 1,LU      <<NON-SESSION COMMANDS>>
      ABS S6+R5
      ASC 1,LS
      ABS S6+R2
      ASC 1,LG
      ABS S6+R2
      ASC 1,MS
      ABS S4+R6
      ASC 1,MR
      EXT MR..
      DEF MR..
```

```
      ASC 1,SA
      ABS S8+R2
END   NOP             <<END OF COMMAND TABLE>>
*
*
COMM  NOP
      LDA COMM,I
      JMP 0,I
      END
```

Shareable EMA Partition Table

The $EMTB table in Table Area II contains the information for shareable EMA partitions. This table is provided as a separate source which can be modified to allow more than eight shareable EMA partitions. Each shareable EMA partition has an entry of five words in this table. The first word contains the negative number of shareable EMA entries possible in the table. To modify the number of shareable EMA partitions, the negative number of entries, and the amount of space allocated for the partition information must be changed. This modified table must then be reassembled and relocated after the operating system during generation.

The source code for $EMTB is:

```
      NAM   $EMTB,13
      ENT   $EMTB
*
$EMTB DEC-8
      BSS   40
      END   $EMTB
```

Index

E

F

G

H

I

T

U

W