RTE-M System Generation

Reference Manual

HEWLETT hp PACKARD

HEWLETT-PACKARD COMPANY
11000 WOLFE ROAD, CUPERTINO, CALIFORNIA, 95014

Library Index Number
2RTEM.320.92064-90003

PART NO. 92064-90003

Printed in U.S.A. 3/77

LIST OF EFFECTIVE PAGES

Changed pages are identified by a change number adjacent to the page number. Changed information is indicated by a vertical line in the outer margin of the page. Original pages do not include a change number and are indicated as change number 0 on this page. Insert latest changed pages and destroy superseded pages.

Change 0 (Original) ..... Mar 1977

HP Computer Museum
www.hpmuseum.net

For research and education purposes only.

PREFACE

This manual is addressed to on-site system managers and programmers who will be planning, generating and maintaining an RTE-M operating system. The manual describes the sequential procedures and other considerations required to plan, generate and load a site-specific RTE-M system that is tailored to meet your installation's on-going requirements.

To use this manual effectively, you should have a working knowledge of the appropriate 21MX-series computer and a good understanding of the structure and functions of the RTE-M operating system software as described in the *RTE-M Programmer's Reference Manual* (briefly outlined later in this Preface). Of particular interest in that manual is the section describing the Relocating Loader and its control command syntax. A subset of these commands is used to relocate system and user software modules during generation.

The RTE-M System Generation Manual is organized as follows:

Section I — Summary descriptions of the functions, features and requirements for MI, MII and MIII versions of the RTE-M system. It also lists the relocatable system software modules required to generate each of the three RTE-M versions. The RTE-M Generator and its loader are described, and the steps you will be going through to plan and generate a system are summarized.

Section II — Describes the installation planning and layout for MI, MII and MIII configurations. Information is given regarding input/output planning and memory organization. Instructions for filling output the I/O Configuration Worksheet are included and a sample is provided.

Section III — Describes system generation planning. It also gives the planning considerations for generating a system using different hardware configurations, such as flexible disc or tape cartridge subsystems. It provides the procedures and worksheets for defining and preparing your responses to prompts output by the RTE-M Generator (RTMGN) during the generation process described in Section IV. Sample worksheet entries are included.

Section IV — Describes how to boot the minimum absolute MI system and RTE-M Generator into main memory to begin generation. Boot-in procedures from both flexible disc and cartridge tape units are given. Procedures for creating and editing a command input file (answer file) are described. Generation error codes and subsequent operator action are provided. Procedures for booting in your generated site-specific RTE-M system are also provided.

Section V — Describes how to plan, relocate and maintain the RTE-M Real-Time BASIC processor in the system.

Appendices — HP Character Set for computer systems, system generation examples, System Tables and Communications Area, Flexible Disc track and Sector Conversion Table, Relocatable Record format, and blank Generation Worksheets suitable for reproduction.

Other manuals offering information relevant to generating and using an RTE-M operating system are briefly summarized below:

- *RTE-M Programmer's Reference Manual,* Manual Part No. 92064-90002

This manual is required for those involved with RTE-M system generations. It describes the functions of RTE-M and the procedures for utilizing system services by both executing programs and programs being developed. The format and function of all operator commands used at the console for system control are also described, and typical examples of program use of system services are provided.

- *RTE-M Software Numbering Catalog,* Manual Part No. 92064-90001

This manual is required for generating systems and provides the module name, description, module part no. and part no. of distribution media (flexible disc, tape cartridge, etc.) of every available RTE-M system software module.

- *RTE Operating System Drivers and Device Subroutines, Programming and Operating Manual,* Manual Part No. 92200-93005

This manual describes real-time input/output considerations common to all RTE operating systems, and the procedures for creating and using site-specific I/O drivers.

- *RTE-M Editor Reference Manual,* Manual Part No. 92064-90004

This manual describes the format and function of the RTE-M Editor (EDITM) commands and the procedures for utilizing all on-line editing services (off-line editing is described in the appropriate 2644/5 Display Station User's Manual). The EDITM user should also read the section describing the File Manager (FMGR) commands in the *RTE-M Programmer's Reference Manual.*

- *RTE-M BASIC Language Reference Manual,* Manual Part No. 92065-90001

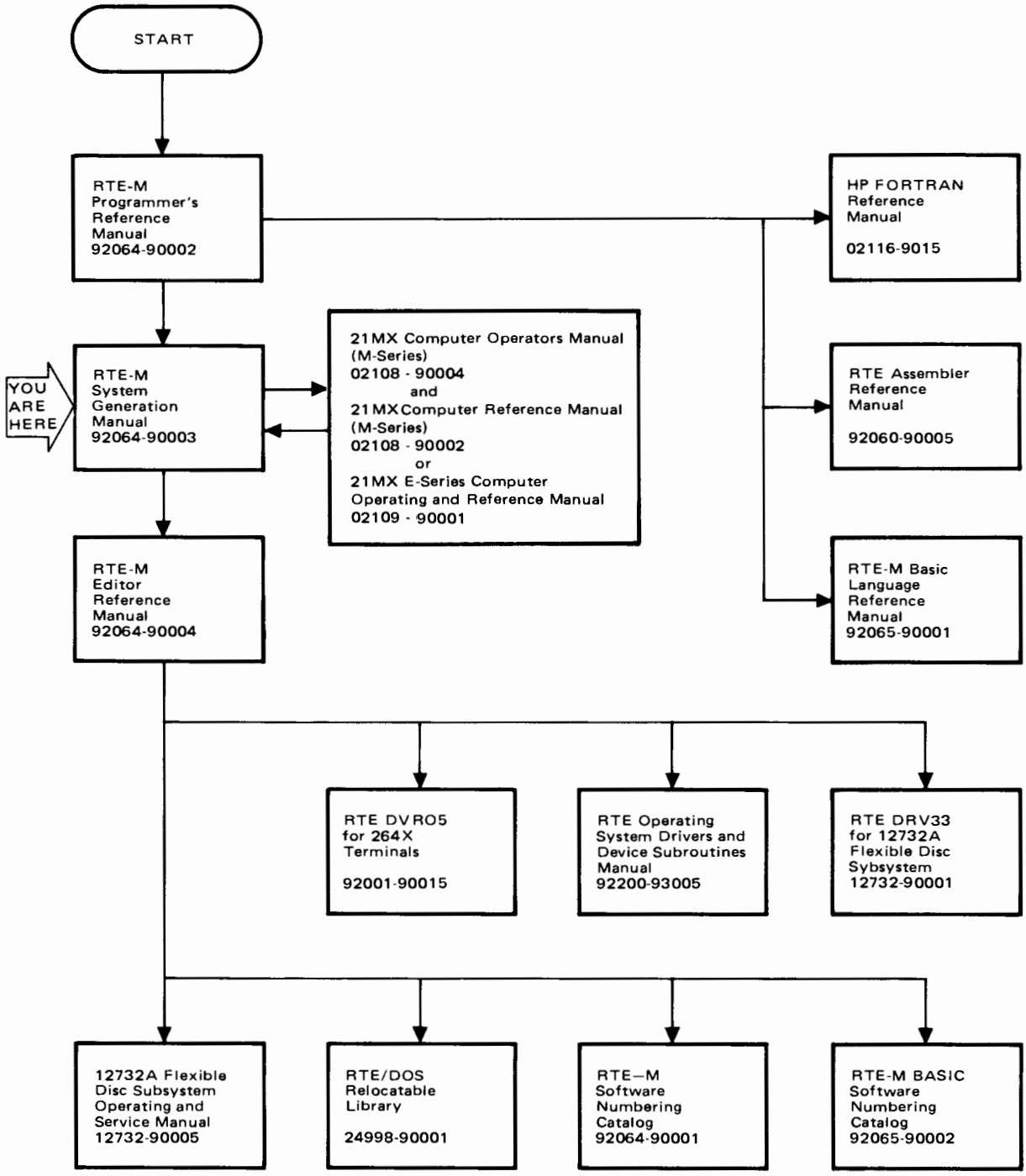
This manual describes the syntax and procedures for writing programs in RTE-M BASIC programming language.

- *RTE-M BASIC Software Numbering Catalog,* Manual Part No. 92065-90002

This manual is required for generating systems that will include BASIC. It provides the module name, description, part no. and distribution media for every module comprising the RTE-M BASIC Subsystem.

The manual part numbers for other manuals describing appropriate driver and language reference manuals are given on the Documentation Map following this Preface.

DOCUMENTATION MAP



CONTENTS

CONTENTS (continued)

ILLUSTRATIONS

TABLES

GLOSSARY OF TERMS FOR RTE-M SYSTEMS

ABSOLUTE PROGRAM - a program that has been relocated through use of the Relocating Loader and can be loaded into CPU memory for execution. An "absolute program" is synonymous with a "relocated program."

ABSOLUTE PROGRAM LOADER (APLDR) - a memory resident program that loads relocated (absolute) user programs into their pre-defined main memory locations, assigns an available ID segment to a program, and makes suitable entries into the ID segment so that the program has access to specific system services. APLDR is automatically called for program loading when using the LO operator command.

ASYNCHRONOUS DEVICE - a device that can perform I/O operations that are independent of time considerations but operates simultaneously with program execution. Interaction with the computer is through request/response circuitry.

BASE PAGE - a 1024-word area of memory containing the system's communication area, system and library links, resident program links and trap cells for interrupt processing. In MIII systems, each partitioned user program has one page for its own base page linking.

BLOCK - two logical disc sectors of 64 words each, totalling a block of 128 words.

CLASS I/O - a method of communication between a set of programs or devices that may be synchronous or asynchronous with respect to each other, in order to provide parallel processing of information. Class I/O allows a program to continue processing after initiating the operation, without requiring that it wait for completion (I/O without wait).

CLOSE FILE - a method of terminating access to a file so that no further read/write instructions may be performed on the file. (See also OPEN file.)

DATA CONTROL BLOCK (DCB) - a table within an executable program that contains information used by the File Management Package (FMP) in performing disc accesses.

DIRECT MEMORY ACCESS - see Dual Channel Port Controller.

DEVICE DOWN - relates to the state of a peripheral device. When the device is down, it is no longer operable. Also (noun), refers to the DN operator command, which sets the device down to make it unavailable for system use.

DEVICE UP - relates to the state of a peripheral device. When the device is up, it is operable. Also (noun), refers to the UP operator command, which sets the device up after it has been set down.

DEVICE REFERENCE TABLE - a table created during system generation consisting of logical unit numbers that provides logical addressing of peripheral devices. Each LU references an EQT entry and contains subchannel information. The table may be modified by using the operator LU command if desired. (See also Equipment Table and Interrrupt Table.)

DUAL CHANNEL PORT CONTROLLER (DCPC) - a hardware capability that permits an I/O process to by-pass the operating system and directly access memory, thus providing a much faster transfer of data. The operating system controls access to the DCPC channels.

DYNAMIC MAPPING SYSTEM (DMS) - a hardware option allowing MIII systems (only) of addressing memory configurations larger than 32K of physical memory.

ECHO DEVICE - if specified, a device that receives a copy of all user command input entered during system generation or on-line program relocation. RTMGN and RTMLD prompts are also echoed, but have a preceding asterisk (*) to define them as non-executable comments. Therefore, the echo file may be used as a command input file (answer file) for repetitive, automatic (non-interactive) generations or relocations. To use the echo file as a following command input file, the echo file must NOT be assigned to the same device as the MAP output, since the mix will contain data not preceded by an asterisk to define it as non-executable.

EQUIPMENT TABLE (EQT) - a table in memory associating each I/O interrupt location with a particular software processing routine (driver). For a given device, the EQT provides status information , temporary storage and parameter passing services. (See also Device Reference Table and Interrupt Table.)

EXTENDED EQT - a method for increasing the size of an Equipment Table entry, during system generation, that gives the specified I/O driver more words of buffer space than is available in the EQT temporary storage area.

EXTENDABLE FILE - is a file (type 3 and up) that is automatically extended in reponse to a write request to points beyond the range of the currently defined file. The extent is created with the same name and size, and the access is continued. Extensions will be of the same size as the base file, and all extensions of any given file will be on the same disc. No flags or pointers relating to the next extent are kept in the file area. Open flags are kept only in the base file directory entry.

EXTERNAL REFERENCE - a reference to a declared symbolic name not defined in the software module in which the reference occurs. An external reference is satisfied by searching another object module that defines the reference name by an entry point definition (i.e., a library search).

FILE CLOSED - see CLOSED FILE

FILE EXTENTS - see EXTENDABLE FILES

FILE LOCKED - see LOCKED FILE

FILE OPEN - see OPEN FILE

FILE MANAGER (FMGR) - an RTE-M program that provides file creation, access and manipulation services through FMGR commands entered by the user. FMGR exists in two different versions: one for flexible disc subsystems and one for tape cartridge subsystems.

ID SEGMENT - a block of 31 words associated with each resident program, that is used by the system to keep track of the program's name, software priority field, current scheduling status and other characteristics. Each resident program must have its own ID segment.

INTERRUPT TABLE - a table that interprets a received interrupt and causes the operating system to take one of three possible actions: (1) call an I/O driver, (2) schedule a program, (3) if nothing is found in the table, issue an illegal interrupt message.

LOCAL COMMON - an area of COMMON appended to a program and accessible only by that program, its subroutines or segments.

LOCKED DEVICE - see logical unit lock

LOCKED FILE - a file exclusively opened for one program and therefore not currently accessible to another program. In tape cartridge systems, opening one file on a tape cartridge effectively locks all other files on the tape from use by the program or other programs.

LOGICAL MEMORY - a 32K (maximum) address space described by a currently enabled memory map. If the System Map is enabled, it describes those areas of physical memory necessary for the operation of the system and does not change during system operation. When the User Map is enabled, it is updated to describe those areas needed by programs when it is to be executed. Port A and Port B Maps describe buffers during DCPC transfers. Logical memory is meaningful only for MIII versions of RTE-M.

LOGICAL UNIT LOCK - a mechanism for acquiring temporary exclusive use of an I/O device or devices by a lower priority program, to ensure it's I/O completion before being preempted by a higher priority program.

LOGICAL UNIT NUMBER (LU) - a number used by a program to refer to an I/O device. Programs do not refer directly to the physical I/O device channel number, but through the LU number which has a cross-reference to the device. This allows I/O devices to be changed through operator intervention without having to change the programs or operating system.

MASTER SECURITY CODE - a flexible disc file protection mechanism, established during system generation or changed by the IN command, by which users possesing the master security code may gain access to files, security codes and track and sector addresses of individually protected files. Access is through DL and IN commands in the flexible disc version of the File Manager. Files may be read-only access, no access or unlimited access, depending upon the security code.

MEMORY BASED SYSTEM - an operating system (such as RTE-M) in which neither system modules nor user programs are swapped in and out of memory for execution. Typically, such systems do not permit on-line program development or relocation. Although RTE-M is memory based, it can be configured for on-line loading (tape cartridge system) and/or program development (flexible disc system).

MEMORY PROTECT FENCES - a method for protecting system areas, library areas and (optionally) COMMON areas by preventing stores and jumps to locations below a specified address. For resident programs not using COMMON, the memory protect fence is set at the base of the entire memory area. For programs using COMMON, all of logical memory is mapped and the fence is set at one of three possible locations, depending upon the the portion of COMMON being used.

MEMORY RESIDENT LIBRARY - a collection of reentrant or privileged library routines available to all programs, thus eliminating the need to append the needed routines to each program. All of these routines are fetched from relocatable libraries.

MEMORY RESIDENT PROGRAM - a program that executes from a dedicated area in main memory.

MULTI-TERMINAL MONITOR - a system software module that provides for interactive program development and editing in a multi-terminal environment.

OFF-LINE - refers to use of I/O devices not currently recognized by the main operating system and therefore do not receive system supervision or services.

OFF-LINE GENERATOR PROGRAM (RTMGN) - a combined minimum absolute MI system and RTE-M Generator program used to configure the initial user-defined RTE-M operating system. RTE-M configures and outputs a site-specific absolute RTE-M system in conformance with user specifications received through RTMGN queries and a subset of RTE-M Relocating Loader commands.

ON-LINE - refers to I/O devices recognized and controlled by the main operating system at the time they are being used. Some devices, such as the 2644/5A Display Station terminals, have their own memory module and can be switched from on-line to off-line mode or vice versa at the user's option.

ON-LINE LOADING - refers to the loading of a relocated (absolute) user program into its pre-determined location in main memory at some point in time after the operating system is generated and running. On-line loading requires use of the memory resident Absolute Program Loader (APLDR).

OPEN FILE - a method of gaining access to a specific file to perform a read/write instruction

PARTITION - a block of memory with a fixed size (in pages) and identification number, and located in the User Program area of main memory. A program partition must have a minimum size of two pages; one page for a base page and one page for the user program. The user program area may be divided up into as many as 64 partitions during system generation. The partitioned memory scheme is available only to MIII versions of RTE-M.

PHYSICAL MEMORY - all of memory that is available to the user. Physical memory includes the operating system, libraries, COMMON, System Available Memory (SAM) and user program area or (for MIII systems) partitioned memory.

PRIVILEGED DRIVERS - I/O drivers whose interrupts are not processed by the central interrupt controller of the standard I/O processor (RTIOC). Such drivers offer improved response time, but must perform their own internal housekeeping; i.e., saving machine status.

PRIVILEGED INTERRUPTS - interrupts that by-pass normal interrupt processing to achieve optimum response time for interrupts having the greatest urgency. Privileged interrupts are handled by I/O privileged drivers

PROGRAM STATE - refers to the status of an executable program at any given time. A memory resident user program is always in one of four possible states: executing, scheduled, suspended or dormant.

REAL-TIME EXECUTIVE - a collection of software modules comprising the total operating system; e.g., EXEC, SCHED, RTIOC, I/O drivers and various tables. For all practical purposes, Real-Time Executive, operating system and RTE-M are synonymous terms.

RELOCATABLE LIBRARIES - consists of the following libraries:

System Library - subroutines that can be appended to each user program and are accessed by system entry points.

DOS/RTE Relocatable Library (RLIB1 and RLIB2) - a collection of utility subroutines that are primarily accessed by FORTRAN programs.

FORTRAN Formatters (FF.N AND FF4.N) - format subroutines for FORTRAN I/O operations.

RELOCATING LOADER (RTMLD) - a user program that sets the communications links and forms an absolute load module from a relocatable program. RTMLD creates the relocated program in conformance with the system snapshot and Relocating Loader commands entered by the user.

RESPONSE TIME - the total amount of time (system overhead) required to bring a real-time program or routine into execution in response to an interrupt, interval timer, call from another program or operator call. Response time is usually measured in milliseconds.

RTE-M EDITOR (EDITM) - a user utility program that obtains a source file and permits editing and manipulation of lines of text in response to EDITM control commands entered by the user. Use of the EDITM program also requires presence of the FMP routines in the system.

RTE-M GENERATOR (RTMGN) - a system generation program that is delivered to customer sites in two formats: an absolute version for initially configuring a site-specific operating system (see OFF-LINE GENERATOR PROGRAM), and a relocatable version used for successive reconfigurations.

RESOURCE MANAGEMENT - an optional feature that allows the user to manage a specific resource shared by a particular set of programs, so that no two of these program access the resource at the same time.

SEGMENTED PROGRAM - a program consisting of a memory resident main segment that brings in segments from a flexible disc file to overlay one another in the same area of memory as they are executed. Each segment and the main operate as a single program. In RTE-M, the segments share the ID segment of the main.

SESSION CONSOLE - in a multi-terminal environment, the specific terminal from which a program is scheduled for execution.

SNAPSHOT - a file containing entry points of a program just relocated. A snapshot is required for programs with segments. (See also SYSTEM SNAPSHOT.)

SYNCHRONOUS DEVICE - devices that perform I/O operations in a fixed timing sequence, regardless of the readiness of the computer.

SELECT CODE - an octal value that specifies the address of an I/O device card.

SUBCHANNEL NUMBERS - decimal numbers associated with the LU numbers of devices with multiple functions on the same device. Each subchannel number is associated with a specific subchannel; e.g., a 2645A terminal could have four subchannels: one for the keyboard, one each for the right and left tape channels, and one for an optional line printer.

SUBSYSTEM GLOBAL AREA (SSGA) - an area of memory providing multiple communications and buffering for HP susbsystems. It is accessed by entry point (using EXT statements) rather than COMMON declarations. Programs using SSGA access include the COMMON area in their maps and have the memory protect fence set below SSGA.

SYSTEM AVAILABLE MEMORY (SAM) - a temporary storage area used by the system for Class I/O, reentrant I/O and I/O buffering.

SYSTEM COMMON - an area of memory that is sharable by programs operating in different partitions of main memory. Memory resident programs always include System COMMON in their map to allow inter-program communication.

SYSTEM SNAPSHOT - a file containing a memory map that gives the entry points of the operating system, memory resident programs and address bounds of the space used. A snapshot is required input for programs to be relocated on-line.

TAPE CARTRIDGE SYSTEM - a mix of HP-supplied hardware/software consisting of one or more HP 2644/45A terminals equipped with tape cartridge subchannels as bulk storage devices, and a cartridge tape version of the RTE-M File Manager program to provide file creation, access and manipulation services.

TIME BASE GENERATOR (TBG) - an optional hardware module (real-time clock) that generates an interrupt in 10 millisecond increments. It is used to trigger execution of time-scheduled user programs at predetermined intervals.

TIME OUT - relates to the state of a peripheral device. When the amount of time the system will wait for an I/O transfer for a given device is exceeded (the time is determined at system generation), the device becomes inoperable. The device must then be restored to system use through operator command.

TIME SCHEDULING - the process of automatically scheduling a program for execution at pre-determined time intervals. Program scheduling is established through use of the IT command, and requires the presence of the Time Base Generator in the system.

GENERAL DESCRIPTION

SECTION I

1-1. USING THIS MANUAL

The RTE-M System Generation Manual provides a set of basic procedures for planning, creating and loading an RTE-M system to your specifications. RTE-M is a highly flexible system, and there are several places in the described generation process where alternate procedures or parameter options could be used to achieve successful results. These alternate approaches were deliberately not presented in the manual. The intent was to avoid possible confusion for readers who are performing their first generation. The procedures given are relatively straightforward and should be followed by all new users. Some of the alternate modes will no doubt suggest themselves to more experienced users during the generation planning stage.

Two other RTE-M manuals should be readily at hand for fast reference when filling out the worksheets included in this manual: the *RTE-M Programmer's Reference Manual* and the *RTE-M Software Numbering Catalog* (see the Preface in this manual for the Manual Part Numbers and summary descriptions).

1-2. TAILORING AN RTE-M SYSTEM

During system generation, you create one of six different RTE-M operating system environments to meet your installation's exact requirements and available hardware resources. You first define the system as being either an RTE-MI, MII or MIII environment. The MI through MIII designators represent three general classes of hardware and software resources that define the scope of system features and services available to user applications.

You further define the selected environment as either an execute-only system or as having a concurrent program development capability. (An entirely practical seventh option is to have an RTE-M system that is dedicated to development programmming.) Both execute-only and concurrent program development RTE-M systems allow on-line loading of absolute programs if the Absolute Program Loader (APLDR) is present in the system. APDLR and the procedures for using it are described in Sections I and II of the *RTE-M Programmer's Reference Manual.*

RTE-M is sometimes user-designed to execute within a memory area smaller than what is initially required for generation. Generation for such a system must then be performed on a larger RTE-M configuration.

Installations with multiple RTE-M configurations typically have one system dedicated to generations and on-going program development. The absolute output from the "grandfather" system is then loaded into other RTE-M stations that are dedicated to executing specific applications.

A flexible disc is a highly recommended option for generations. Use of the disc significantly reduces both generation time and operator handling of the relocatable software modules. It also decreases the possibility of operator errors during the load process.

The RTE-M software package delivered to your site includes an Off-Line Generator program (RTMGN), and a complete set of relocatable software modules labeled %MSY1, %MSY2 and %MSY3. The %MSY1, %MSY2 and %MSY3 designators represent RTE-MI, MII and MIII versions of the system. These relocatable groups contain all the required and optional software modules available for the RTE-M version you decide to specify.

After the Generator program is loaded into main memory, you call RTMGN to create your site-specific RTE-M system from the relocatable %MSY1, %MSY2 or %MSY3 and optional software. RTMGN relocates the selected software and configures the system according to your responses to its queries. It outputs the user-defined absolute RTE-M system on the specified device or file.

1-3. FLEXIBLE DISC SYSTEMS

If your RTE-M system is to be configured on a flexible disc configuration, the Off-Line Generator and all relocatable software modules are delivered on two flexible disc platters. The general process of creating a site-specific system with a flexible disc is illustrated in Figure 1-1.

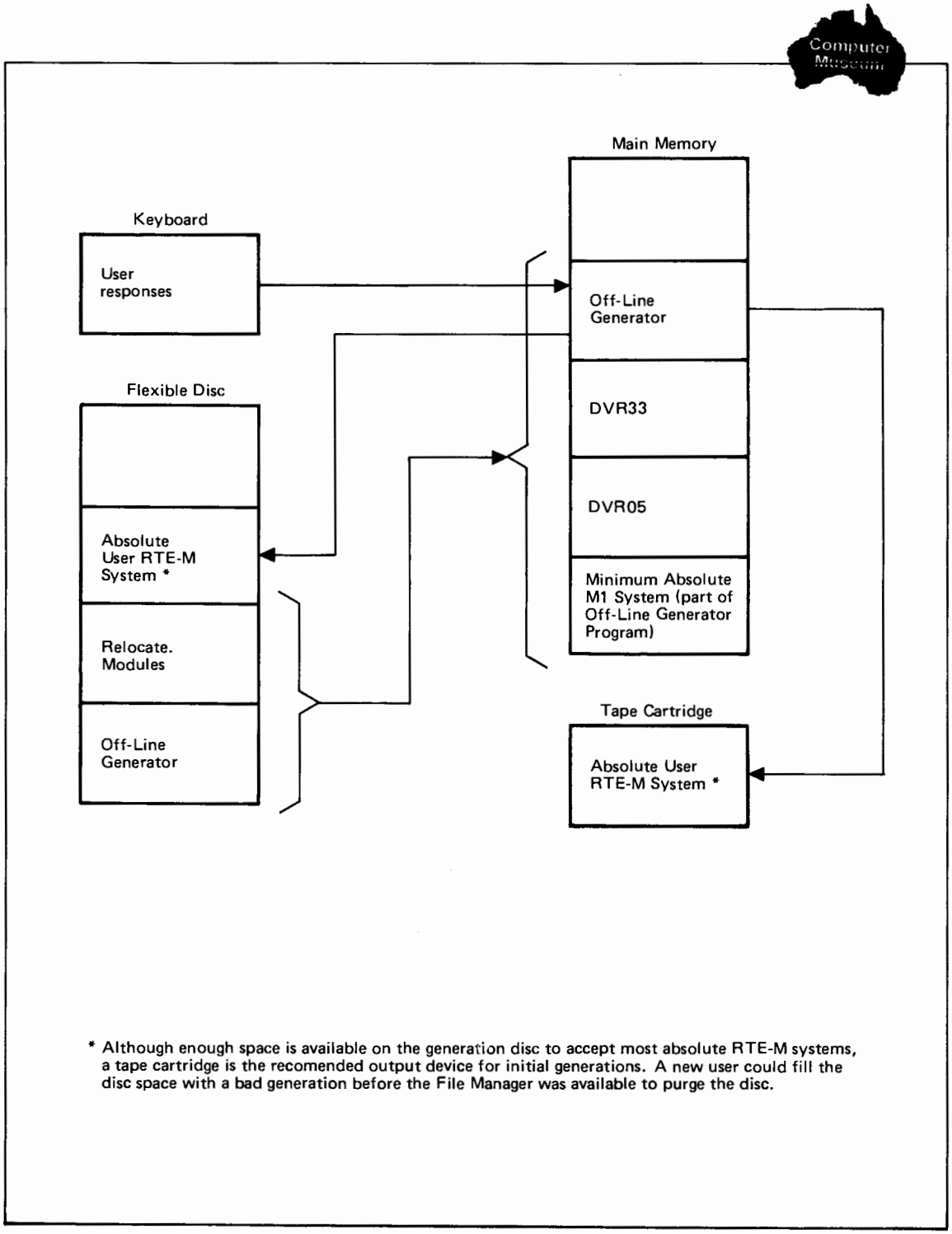


Figure 1-1. Generating With Flexible Disc

1-4. TAPE CARTRIDGE SYSTEMS

If your RTE-M system is to be configured using the tape cartridge system, the Off-Line Generator and all relocatable modules are delivered on a set of labeled tape cartridges. On the 2644/5A terminal, you will load the Generator and relocatable modules from the RIGHT tape channel (LU5). The relocatable modules are loaded in the sequence requested by RTMGN. The generated user RTE-M operating system is output on the LEFT tape channel (LU4), and subsequently loaded into its absolute addresses in main memory.

The general process of creating a system using tape cartridges is illustrated in Figure 1-2.

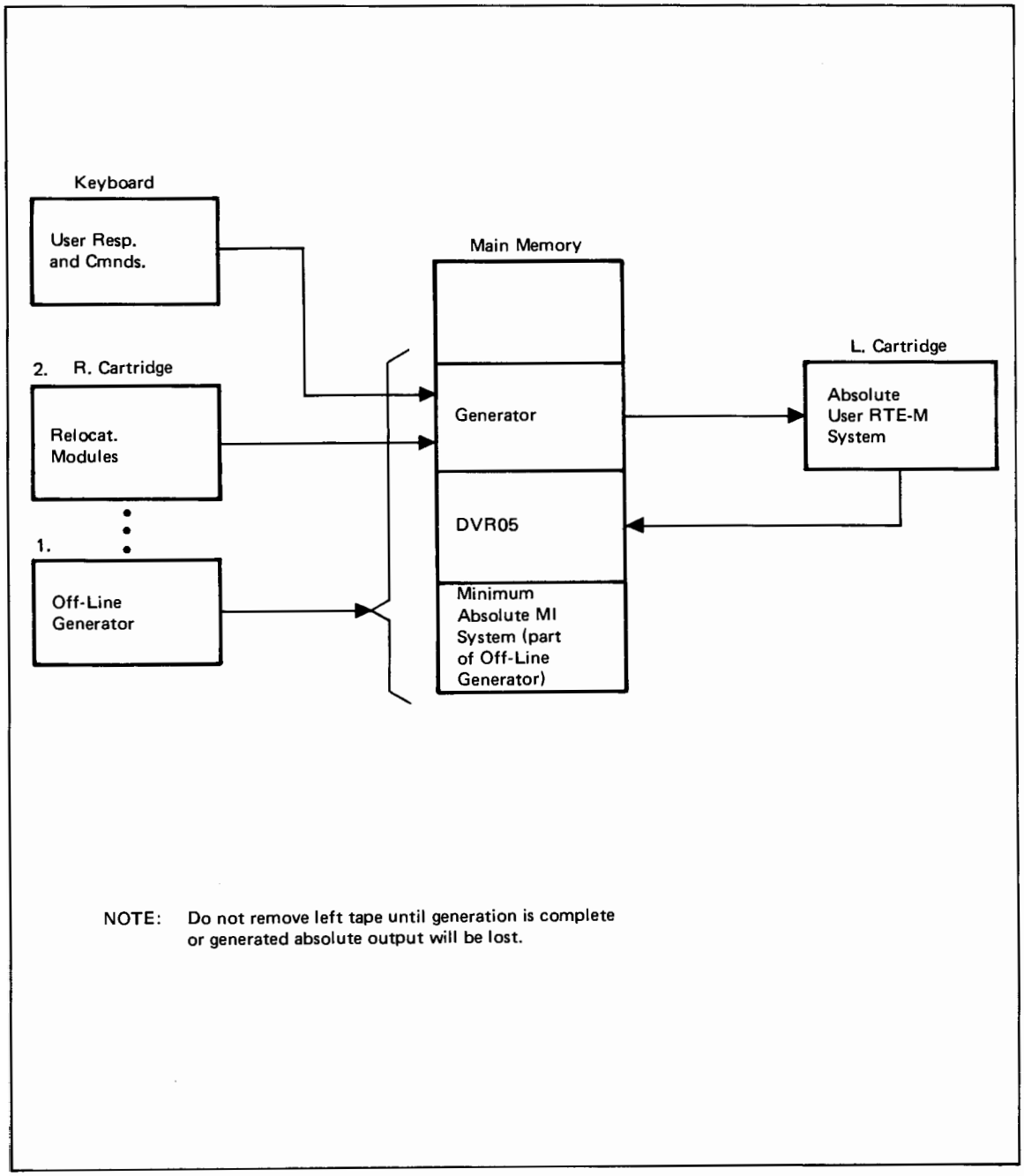


Figure 1-2. Generating With Tape Cartridges

1-5. SELECTING AN RTE-M VERSION

The primary reason for having three different version of RTE-M is to provide you with an operating system that matches on-site requirements without the overhead of supporting unnecessary hardware and software features. The general intent is offer the following environments:

- MI- primarily intended to provide an execute-only mode for one or more user programs.
- MII- Intended to provide a multi-programming environment, with both program execution and concurrent program development capability.
- MIII- Intended to provide a concurrent multi-programming/multi-user environment and the capability of accessing more than 32K of memory.

1-6. MI CAPABILITIES AND REQUIREMENTS

You create a site-specific MI system from the relocatable software labeled %MSY1 and optional modules, using the Off-Line Generator program to create it. A minimum MI environment is the most basic RTE-M system, and provides for execution (only) of one or more user programs. For such a system, the hardware requirements are as follows:

1. A miminum of 8K of main memory (MI supports up to 32K of memory). A minimum of 16K of memory is required for execute-only operations if an optional flexible disc is present on the system. 24K of main is required if generation is to be performed.
2. A 2644/45 Display Station with cartridge tape units, flexible disc unit or paper tape I/O device. A flexible disc is recommended for generations because of the increased speed and reduced operator handling.

The services provided for your executable user programs in such a system will be as follows:

- Centralized I/O control and reentrant subroutine processing.
- Operating system services available through EXEC function calls or other system subroutines. Direct calls to the EXEC are made without the overhead of going through a memory protect violation.
- Program control by operator through commands entered at the system console. Operator commands may also be coded into user programs via EXEC calls.
- Error handling and diagnostics.

1-7. MI OPTIONS. An MI system requires 24K of main memory if system generation and relocation are to be performed on the target system. Program development capability will also require a flexible disc unit (except BASIC programs). Memory protection is not available for MI systems. The following options supported with additional hardware/software modules:

- On-line loading of absolute modules into main memory (requires APLDR in main memory)
- Device time-out processing and Time Base Generator (TBG clock)
- Time-of-day scheduling (requires TBG above)
- Privileged interrupt I/O
- Fast FORTRAN Processor (FFP)
- Output buffering
- Powerfail auto-restart capability
- Additional operator commands, including: logical unit display modification (LU), equipment table display or buffering assignment (EQ), and buffering limits display or modification (BL).

1-8. MII CAPABILITIES AND REQUIREMENTS

You create a site-specific MII system from the relocatable %MSY2 software and optional software modules. Memory protection and 16K of main memory are required for an execute-only MII system. Up to 32K of memory will be supported. If system generation is to be performed on the system, a minimum of 24K of main memory is required. Program development also requires a flexible disc device.

In addition to all of the services and optional features listed for MI, MII provides the following services for user programs:

- Multi-programming capability with full scheduling for user programs, both by program priority and time scheduling.
- I/O scheduling. I/O processing can take place concurrent with, and independent of, program execution.

1-9. MII OPTIONAL FEATURES. The optional services available to MII configurations are as follows:

- Multi-Terminal Monitor (MTM) if more than one terminal is supported, with a practical limit of up to four user terminals.

NOTE

Support of more than two tape cartridge units by the File Management Package (FMP) requires modification (reassembly) of the HP-supplied source code for the $TBLCR routine. See Appendix E.

- Class I/O that permits program-to-program communication and allows a program to continue execution without waiting for a requested I/O device to become available (I/O without wait)

- Resource management, including resource numbers (RN's) and logical unit lock/unlock.

1-10. MIII CAPABILITIES AND REQUIREMENTS

You create an MIII system from the relocatable software (labeled %MSY3) using the Off-Line Generator program. For execution-only systems, requirements include: 32K of memory, Dynamic Mapping System (DMS), Dual Channel Port Controller (DCPC) and Absolute Program Loader (APLDR). APLDR displays all current ID segments of programs in memory, defines or redefines a partition and performs on-line loading of a absolute programs. MIII provides all the services and features listed above for MI and MII configurations. In addition, it offers the following features for user programs:

- Supports up to 304K of main memory, or whatever the current hardware is capable of supporting.

- Provides full memory protection for all system software, resident user programs and development programs.

- Dynamic Mapping System (DMS) and up to 64 partitions in main memory (each partition requires a minimum of two pages of memory; a base page and a user program page).

1-11. PROGRAM DEVELOPMENT SOFTWARE REQUIREMENTS

For all three RTE-M configurations, the following software modules should be present in the system if program development is to be provided:

1. The desired language processors; those available are Real-Time BASIC, FORTRAN and HP Assembly Language.

2. The File Manager Program (FMGR), which is part of the File Management Package (FMP).

3. The RTE-M Editor (EDITM) for editing source programs.

4. The Relocating Loader to create the absolute load modules from the relocatable programs.

5. The Absolute Program Loader to load an absolute program into main memory and assign and fill in an ID segment to be associated with the executable program. (The required number of ID segments is user-defined during generation; one for each intended user program.)

1-12. OFF-LINE GENERATOR (RTMGN)

The Off-Line Generator program consists of a generator, any required drivers and a minimum MI system that are in absolute format. They are booted into main memory as a single load module through sense switch settings on the computer control panel. When called for service, RTMGN identifies itself and begins a series of sequential questions to the operator regarding the desired RTE-M configuration. After each prompt or question, it waits until the operator responds with the requested data. The loader portion of the Generator uses a subset of the Relocating Loader commands to relocate the required and selected software modules.

RTMGN is delivered in two formats with each RTE-M system; an Off-Line version used for initial generations as described above, and a relocatable version that is relocated and loaded by the user for succeeding generations.

The services provided by the Generator are as follows:

1. Supervises system generation.

2. Establishes a dialog with you by issuing a series of queries regarding system structure. It configures and relocates the system according to your responses and commands.

3. Performs the relocation of the selected modules and outputs an absolute user-defined RTE-M operating system.

During system building and user program relocation, you will direct the relocation of modules or programs through RTMGN commands. These commands are listed below, and are a subset of the Relocating Loader commands described in the *RTE-M Programmer's Reference Manual* (see that manual for the exact format and function):

BOUNDS	defines upper and lower limits of memory.
DISPLAY	UNDEFS displays all undefined DEF's; TABLE displays all or part of current symbol table definitions.
END	specifies end of user program input phase.
MAP	controls format of memory map listing options and lists certain information.
RELOCATE	causes the relocation of next module from specified file or LU.
SEARCH	causes selective relocation of modules such as library routines. Another SEARCH may be used to satisfy any remaining undefined external references.
TRANSFER (TR)	causes a transfer to some other command file or LU.
SET	causes the setting of either BPLOCC or LOCC to an octal value, creates a new symbol table entry with specified value, or changes value of current entry.
LINKS START AT	sets the link portion of symbol table entry to specified link address.
LINKS IN	specifies whether linkages are to be made through base page or current page. If not specified, LINKS IN defaults to the base page.
SNAP	causes a snapshot to be output at any time during the generation process. A system snapshot is required if program relocation is to be performed.
OUTPUT ON	permits sending the absolute output to some LU or file other than the standard device (LU4). OUTPUT ON must be specified prior to relocation of any modules; there is no default case.
EX (EXIT)	terminates execution of the generation process.
ECHO	specifies where the input commands should be echoed. A complete listing of input commands, modules names and entry points can be received by setting the echo command file equal to MAP MODULE command. However, such a mix prevents use of the echo file as a command input file (answer file) for any succeeding generations.

1-13. ANSWER FILE

If a generation is performed on an RTE-M configuration with a flexible disc (other than the initial generation), it is sometimes convenient to generate from a previously created answer file that contains the correct responses to RTMGN queries. Additional parameters on the RU,RTMGN command causes RTMGN to read directly from the answer file for your responses. This simplifies and greatly speeds up the generation process and reduces the possibility of keyboard error. Use of an answer file is not practical for RTE-M configurations limited to a tape cartridge system, because the required output tape channels are generally not available. Procedures for using an echo file as a command input file are described in Section IV of this manual.

1-14. GENERATION PROCESSES

There are seven major steps in the generation process, called "phases." Within each phase, the Off-Line Generator outputs queries about how you wish it to structure some portion of the system. Your responses are derived from generation worksheets you previously prepared during system planning. The phases are summarized below (detailed descriptions are given in later sections of this manual):

1. Initialization Phase: you define the file name or logical unit number of the desired output and echo devices. You then define the system type desired; either an MI, MII or MIII by entering a 1, 2 or 3 respectively. Thereafter, all following RTMGN queries are only those appropriate to the selected system type. You also define various memory areas and system parameters.

2. Resident Program Input Phase: you create the absolute RTE-M resident by relocating the system modules through use of RTMGN commands that are a subset of Relocating Loader commands (see Section VII, *RTE-M Programmer's Reference Manual*). Creating the absolute Resident consists of relocating:

   a. %MSY1, %MSY2 or %MSY3

   b. Optional system modules, including powerfail and (for MII and MIII only) class I/O and a source management.

   c. I/O Drivers

3. Table Generation Phase: you build the Equipment Table, Device Reference Table and Interrupt Table, and define a number of other system parameters such as buffer limits and (for MII and MIII only) number of I/O classes and resource numbers. For MIII systems, you also define the the number of program partitions you intend to have after system available memory (maximum 64).

4. Program Relocation Phase: you first relocate optional system library routines, followed by (for MII and MIII only) the Subsystem Global Area. You then relocate the desired user programs and an optional "startup" program. You will also define the number of ID segments desired. There must be one ID segment for every program relocated, PLUS an ID segment for every program to be loaded on-line at a later date. A "startup" program is optional user program that is part of the generated absolute operating system, and begins execution as soon as the system is loaded into main memory.

5. Change Memory Bounds Phase: you define any desired changes in the size of the resident user program and system available memory areas.

6. Partition Definition Phase: you define the size and "name" of each program partition to reside in remaining physical memory. A partition name is some number between 1 and the maximum number of partitions you previously entered during the Table Generation Phase. RTMGN will only request this information if you are generating an MIII system.

7. Snapshot Phase: the RTE-M Snapshot is output at the end of the generation process. It is used in conjunction with the Relocating Loader for setting up the memory bounds and symbol table while forming the absolute load module to be loaded on-line.

SYSTEM PLANNING AND LAYOUT

SECTION II

2-1. RTE-M SYSTEM PLANNING INSTRUCTIONS

System planning is divided into two major subsections as follows:

- I/O Planning--I/O interface cards for peripheral devices are identified, logical unit numbers are assigned, and tables are planned that effect communication between the devices and the system. 2644/45 cartridge tape systems and flexible disc considerations are also discussed. Considerations and "cookbook" procedures for filling out the I/O Configuration Worksheet are provided.

- RTE-M Memory Configuration--The physical and logical organization of system memory is discussed and planned. Memory protection for MII and MIII systems is covered. Finally, memory partitioning for MIII systems is considered.

2-2. INPUT/OUTPUT PLANNING

Input/output locations in all HP 21MX series computers have the same sequence of priority addresses; the highest priority address is the lowest numbered select code (I/O location). The octal select codes start at octal 10 and continue upward toward octal 77, limited by the I/O capacity of the computer and any attached extenders.

Interface cards are assigned to priority addresses according to the speed of the interrupt response required by the I/O device. Interface cards for high-speed devices are assigned higher priority addresses than low-speed devices. Devices requiring privileged interrupts are always assigned to the highest priority addresses, while direct memory access (DCPC) devices are assigned to the lowest.

2-3. USING THE I/O CONFIGURATION WORKSHEET

The first step in planning a specific RTE-M system is to define the hardware to be used. You assign the select code (in octal) associated with each I/O device, device name, I/O driver name, logical unit number, table entries, etc.

The above information is entered in the I/O Configuration Worksheet, which is part of the worksheet set supplied in Appendix D of this manual. It is recommended that you duplicate all the blank worksheets and fill in your entries on the reproduced copies. In particular, you should make several copies of the I/O sheet and page 2 of 2 of the Program Partition/Program Parameter Input sheet.

Once the I/O Configuration Worksheet(s) is filled in, it becomes an accurate, fast and highly convenient reference when filling in the correct parameter entries in the other worksheets. These in turn, supply the data you type in when responding to RTMGN queries during the actual generation.

Figure 2-1 illustrates a sample I/O Configuration Worksheet filled in for a typical RTE-M system. It is recommended that you first examine Figure 2-1 and then refer back to it when studying the instructions and considerations for the steps following the figure.

> NOTE
>
> Beginning with Figure 1, all examples given in the sample Generation Worksheets in this manual reflect the actual input used to generate a single, "typical" MIII system. A mix of the resulting echo file and system map actually produced at the end of the sample generation is illustrated in Figure H-3 of Appendix H.

STEP 1. I/O Locations

Considering the factors given in preceding paragraphs and the instructions below, write the select code (in octal) for the first I/O card in the "I/O Slot No." column.

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 interrupts in order of decreasing response time requirements (i.e., time from interrupt to service).

b. Assign the privileged interrupt I/O card if privileged interrupt devices are to be used.

c. Assign the TBG I/O card if the Time Base Generator option is to be implemented. Caution should be exercised in the number and type of privileged devices assigned to lower I/O select codes (higher priority) than TBG if the Time Base Generater is present. Privileged interrupts have the potential of offsetting execution time for time-scheduled programs. The interaction between privileged devices and programs scheduled by time increments should be carefully planned.

d. Assign all devices that do not use direct memory access (DCPC) in order of decreasing interrupt rate.

RTE-M I/O CONFIGURATION WORKSHEET PAGE of

I/O SLOT NO. (octal)	DEVICE NAME	DRIVER NAME	LU NO. (dec.)	EQT SUBCH. NO. (dec.)	INTERRUPT TABLE ENTRY	BUF. REQ. (B)	DMA (D)	TIME OUT (dec. no. of 10 ms)	EQT EXT. (dec. no. words)
4	PWRFL	DVP43	12	4	ENT ,$POWR				
10	TBG				,				
11	FLX.DISC	DVR33	2	1 Ø	EQT , 1		D	5Ø	
12	" "	" "	3	1 1	EQT , 1				
13	SYS.CONSOLE	DVRØ5	1	2 Ø	EQT , 2	B			13
	LEFT CTU	" "	4	2 1	,				
	RIGHT CTU	" "	5	2 2	,				
14	HP-IB (Line Printer)	DVR37	6	3 1	,				
	HP-IB BUS		7	3 Ø	EQT , 3		D		5Ø
	SPARE		8		,				
	"		9		,				
	"		10		,				
	"		11		,				
					,				

Figure 2-1. MIII I/O Configuration Worksheet Example

NOTE

If a device uses direct memory access for data transfer and still generates an interrupt for end-of-record (EOR) processing, the hardware priority of the device should be treated as a non-DMA device, with the interrupt rate of the EOR condition determining its priority location. You should give some consideration to the priority of a data transfer vs. the priority of a record termination. Data transfers should normally be given priority over EOR interrupts of equivalent or even slightly slower interrupt rates.

e. If an I/O extender is required and the extender does not have DCPC capability, the order of steps "d" and "e" 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.

Although each column of the worksheet is being discussed separately here, it is recommended that you do NOT fill in any entire column at one time. Move horizontally across the worksheet, filling in relevant entries in each column for a given device. This is done because some devices, such as the 2644/5 Display Station, require multiple entries in the EQT/SUBCH column (see Figure 2-1).

STEP 2: Device Names

Fill in the device name associated with the device. Note that there may be multiple entries for identical devices, each with a different I/O card select code entry. (See the entries for the different flexible discs in Figure 2-1.)

STEP 3: Driver Name

Write in the driver name for each device; e.g., the flexible disc driver is DVR33. For devices or subsystems that have more than one I/O card, refer to the I/O card or subsystem documentation covering that device and driver. Most (but not all) driver names have a preceding "DVR" followed by a two-digit number.

NOTE

Caution must be used when filling out this column, so that you do not accidently enter a driver's relocation file name instead of the required actual driver program name. Although the file name of the relocatable module is frequently identical (except for its preceding % sign) to the driver's program name, this is not always true. For example, the file name of the 2644/5 Display Station is %4DV05, while the terminal's driver program name is DVR05, with the latter being the correct entry for the "Driver Name" column. See the "Generation Requirements" or "Configuration Information" section in the appropriate I/O driver manuals to get the correct names to be entered in this column.

STEP 4-A. Standard Logical Unit Assignments

In the "LU NO." column, make standard logical unit number assignments (1 through 6) for the appropriate devices. The standard LU assignments are as follows:

1. system console (keyboard device)
2. system peripheral (flexible disc if present on system). LU2 must be assigned to the flexible disc if the flexible disc version of FMP is to be used.
3. system peripheral (second flexible disc)
4. standard output device (e.g., left tape cartridge unit)
5. standard input device (e.g., right tape cartridge unit)
6. standard list device

STEP 4-B. Additional Logical Unit Assignments

Starting with decimal 7, write in a logical unit number for all other devices or subchannel numbers as applicable. A typical example might be 7 for an HP-IB BUS for a line printer. These numbers can be arbitrarily assigned to I/O devices. If more than one 2644/5A terminal is included, the LU numbers of the CTU's must be greater than the LU numbers of the associated keyboard device.

> NOTE
>
> If a device has two I/O cards, use only the highest priority (lowest select code) I/O card for Step 5.

STEP 5. Equipment Table and Subchannel Numbers

In the EQT part (leftmost side) of the EQT/SUBCH NOS. column, write in the EQT number for each device. A device that has subchannels is assigned the same EQT number for each subchannel. Next, enter the subchannel number(s) for the device if appropriate. For instance, a 2644/5 terminal may have four subchannels; one each for the right and left tape cartridges, one for the keyboard, and one for an optional printer (see Figure 2-1).

STEP 6. Interrupt Table Entry

For devices, this column is filled in from data derived from the I/O Slot and EQT columns in the I/O Worksheet. Although there is only one comma in the column, indicating two parameter entries, there is an implicit select code parameter that precedes the other two entries. When you later fill out the Interrupt Table Entry Worksheet, you first enter the select code for the device as given in I/ O Worksheet, followed by the appropriate parameters from the Interrupt Table Entry column. Add an extra comma to the column in the I/O Worksheet and include the the select code as the first parameter if you wish.

In Figure 2-1, the $POWR parameter is the entry point for the powerfail routine.

STEP 7. Buffering

In the "BUF REQ." column, write in a "B" for each device that will use output buffering. Buffering means that a copy of data that is to be output to a device (e.g., a line printer) is first written into a system buffer area in main memory. The system allows the user program to continue processing after issuing a WRITE request to such a device, rather than suspending the program while it waits for the user program buffer to be emptied.

> NOTE
>
> I/O buffering is optional in MI operating systems. Therefore, errors may result if buffering is specified without the option being included in the system.

STEP 8. Direct Memory Access (DMA)

Write a "D" in the DMA column for every device that uses direct memory access (recheck the NOTE in Step 1-d before making entries in this column). Some drivers, such as DVR62 for the HP 2313 subsystem, are capable of dynamically assigning a Dual Channel Port Controller (DCPC) to themselves when required. In those cases, do not assign DMA. Refer to the individual driver manuals for more information regarding this capability.

STEP 9. Time-Out

In the "TIME OUT" column, enter the decimal number of milliseconds for each device that will use the time-out parameter. The value entered represents increments of 10 ms. Note that the time processing modules are optional for all three versions of RTE-M.

STEP 10. Extended EQT

In the "EQT EXT" column, enter the decimal number of words needed by the specified driver to increase the size of the Equipment Table. The extension gives the driver more buffer space than is available in the system buffer area. A typical driver that uses extended EQT is DVR05 for the 2644/5 terminal (see Figure 2-1).

2-4. MEMORY CONFIGURATION PLANNING

This subsection of the planning phase describes most of the elements you must consider when dividing up physical memory, loading programs, and (for MII and MIII only) establishing memory protect fences and setting up partitions (MIII). The material is provided for both reference and planning purposes. Some of the actual inputs to RTMGN will depend upon your on-the-spot analysis of the data output by the generator at a given point. You will then make a decision based on that hard data using the considerations presented here.

2-5. Physical Memory. The basic purpose of performing a generation is to create an operating system structured as illustrated in Figure 2-2 (for M1 and MII systems) and Figure 2-3 (for MIII systems). Although the system organization is fixed, the relative sizes of the areas will depend upon the needs of your installation. Some areas, such as COMMON, may not necessarily exist in all systems. You will make the following determinations, as relevant:

- The size of System Available Memory (SAM)
- The size of COMMON
- The size and composition of the resident library
- The size of the resident user program area
- The size of each partition (MIII system only).

2-6. Memory Size. The size of physical memory depends upon the the hardware available. RTMGN can configure an MI system from 8 to 32 pages long, an MII system from 16 to 32 pages, and an MIII system from 32 to 304 pages. A page consists of 1K (1024 words) of memory.

2-7. System Base Page. The system base page contains the system communication area and is used by the system to define requested paramaters, I/O tables, scheduling lists, operating parameters, memory bounds, etc. System and library links, resident program links, and trap cells are also located on the system base page. System and library links and the system communication are available to all programs for read-only access.

The system communication area is fixed (see Figures 2-2/2-3). The size of the system links area varies with the number of page crossings that cause indirect links to be generated on the base page (current page linking can reduce the number of base page links used).

Linkages are allocated as needed during generation, starting from the interrupt links area. If the base page linkage area overflows, a "BPG OV" error message is output and you must then delete one or more programs from the resident user program area of the system. To aid you during generation, RTMGN will print the allocation of links on a module-by-module basis if the LINKS option of the MAP command is specified.

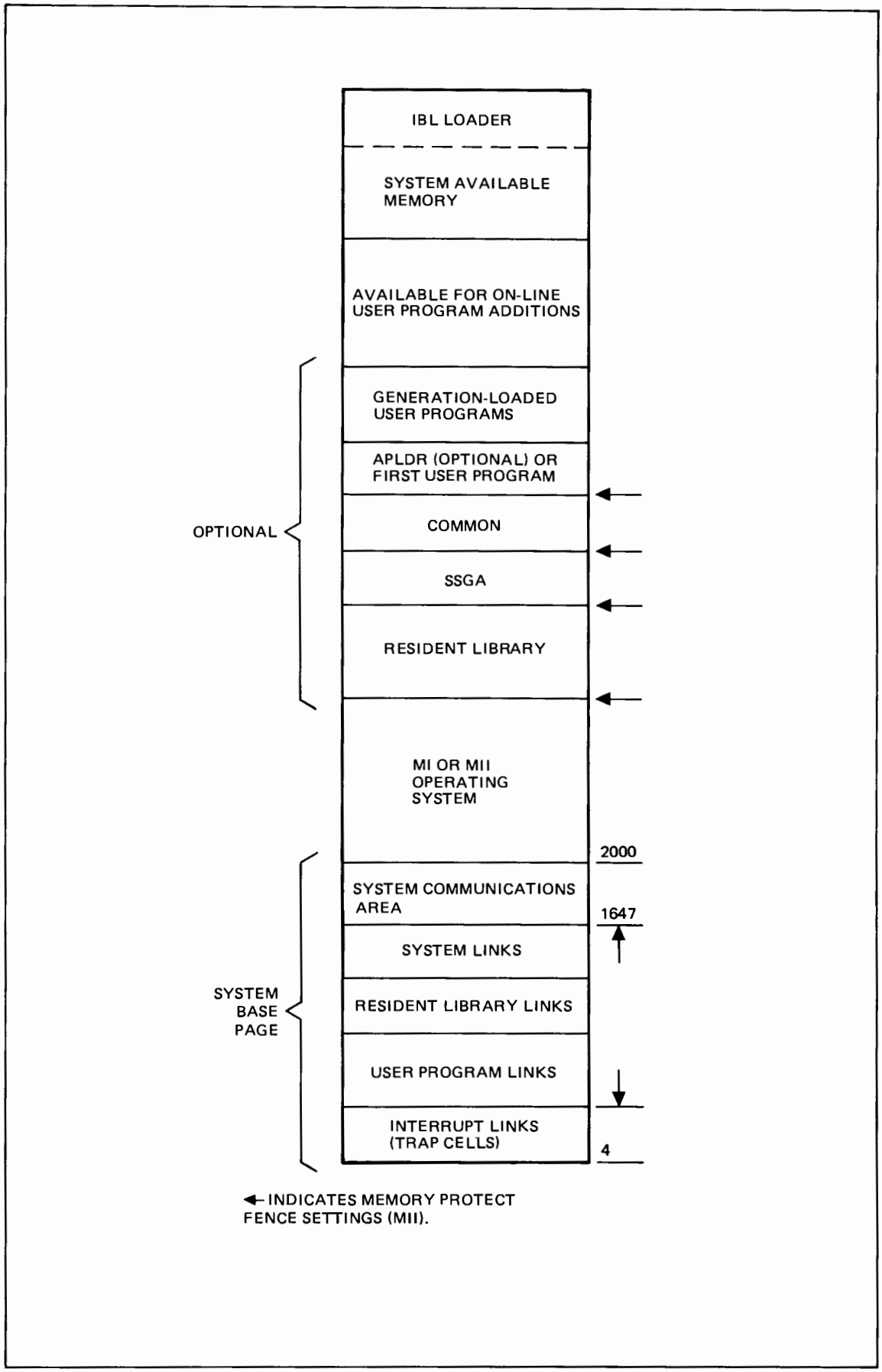


Figure 2-2. MI and MII Memory Map

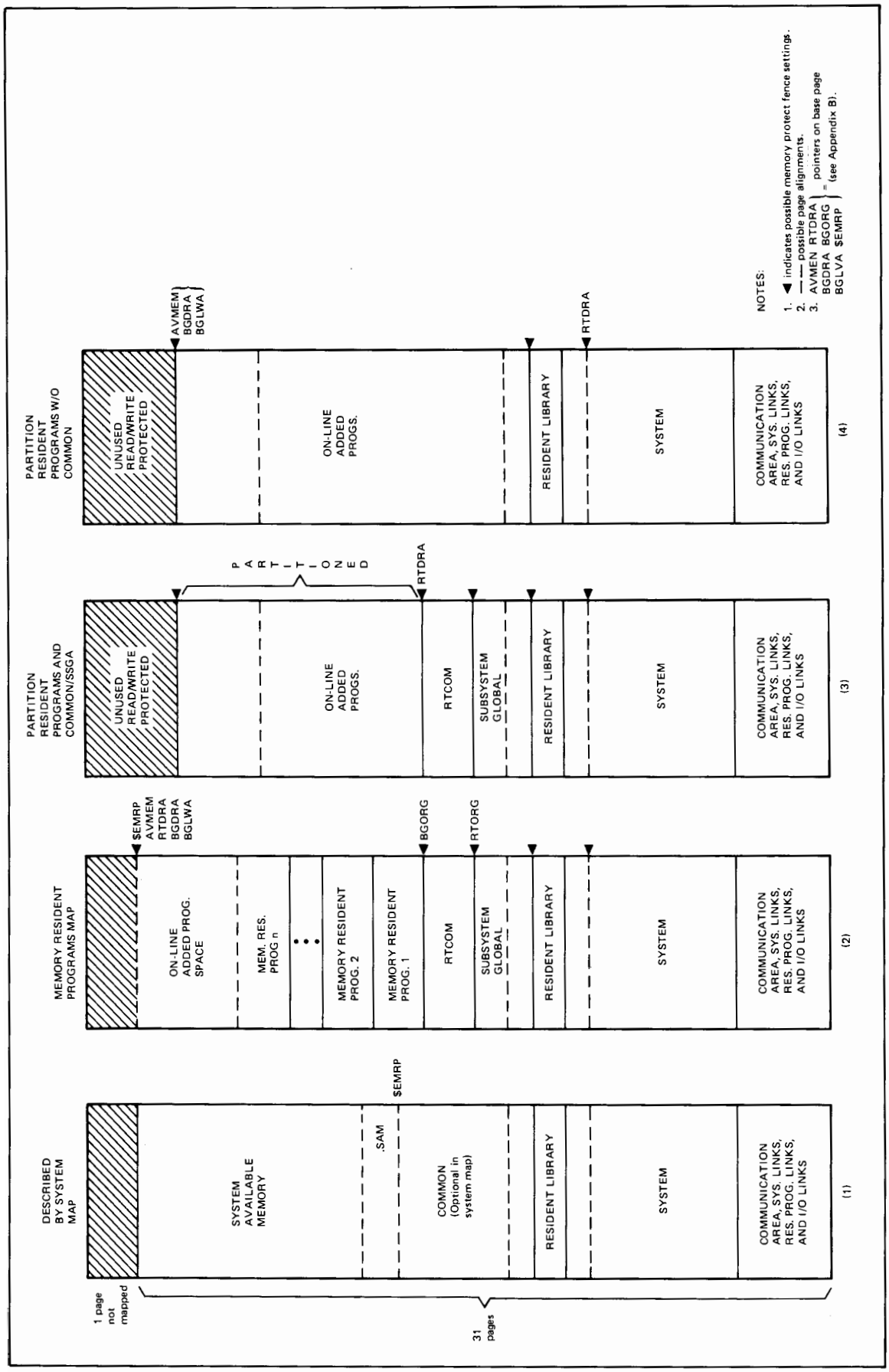


Figure 2-3. MIII Memory Map

2-8. System and Library Areas. These two areas become a part of every user program's logical address (see Figure 2-3. The system area contains type 0 system modules (e.g., RTIOC, SCHED EXEC) and drivers plus tables. The size of the system area is indirectly controlled by the number of I/O devices configured (i.e., table sizes and drivers).

The resident library area contains the reentrant or privileged library routines (type 6) that are specified to go in that area upon operator command. A module placed in this area need no be appended to a program that calls it. However, the module is subject to certain design constraints to prevent two programs from inadvertantly gaining concurrent access.

> NOTE
>
> For MIII partitioned memory systems, each user program is limited to 32K addressability. Therefore, the size of the system area directly reduces the area available for each user program in a partition.

2-9. COMMON AREA. The area is divided into two sub-areas: the Subsystem Global Area (SSGA) and the Real-Time COMMON Area. The Subsystem Global Area is used by HP subsystems and contains type 30 modules (SSGA modules) loaded sequentially. The SSGA modules are accessed by their entry point and not through common declarations.

The size of Real-Time COMMON Area is user declared. During generation, the size of Real-Time COMMON can be declared large enough to accomodate programs to be loaded.

The COMMON area may optionally be included in the System Map to aid privileged drivers. This makes COMMON immediately accessible at privileged driver interrupt (MIII only). If no system COMMON is defined or there is not enough system COMMON, any user program may use local COMMON.

2-10. USER PROGRAM AREA. This area can be optionally divided into two separate sub-areas: a generation-loaded user program area, and available space for adding new resident programs on-line at a later date. On-line additions capability requires that APLDR be resident in main memory.

The first user programs loaded should be those that will never be removed from the system. D.RC, D.RF and APLDR (all HP-supplied) are such typical user programs.

All resident user programs (including programs loaded on -line) must fall within the first 32K of physical memory or less, depending upon the constraints of available memory size.

It naturally follows that you must have a good idea as to the number and expected size of any programs to be loaded on-line. The on-line user area is also used for development programming, except in MIII systems, where partitions may be used.

2-11. SYSTEM AVAILABLE MEMORY (SAM). This area is used by the system as a temporary storage area for class I/O, reentrant I/O and automatic buffering. The amount of SAM depends upon specific applications. The lack of enough buffer space can cause temporary suspension of a program or program abort.

SAM may start immediately after the user program area(s), be aligned at the next page, or may be set at the end of memory so other user programs may be loaded on-line. In MIII systems, alignment prevents accidental destruction of critical data by a resident user program accessing the same page. Any words skipped due to alignment are added to the on-line "available" memory resident program area.

For MIII systems, SAM always ends at a page boundary where the first program partition starts. Therefore, its size defaults to the number of words between its starting address and the next page (between 1 and 1024 words). The recommended minimum is 1024 words.

System Available Memory can be increased in 1024-word (1K) increments by increasing the page number where the user program partitions start.

2-12. MEMORY PROTECTION. Memory protection applies only to MII and MIII systems. For these two, memory protect fences provide protection to the System, Library, and (optionally) SSGA and System COMMON areas. The memory protect fences are automatically set by the operating system and provide their protection by preventing stores and jumps to locations below a specified address. All possible fence positions are indicated by arrows in Figure 2-3.

The memory protect fence applies to the logical address space, and addresses are compared to the fence before translation. If a user program does not use any of the COMMON areas, the memory protect fence is set at the base of the entire resident user program area.

For programs using COMMON, all of COMMON is mapped and the fence is set at one of two possible locations, depending upon whether SSGA or System COMMON is being used (see Figure 2-3).

2-13. MIII MEMORY PARTITIONING. In MIII systems, the number of pages remaining after SAM may be divided into partitions (maximum 64). Each partition must be at least two pages long, with one page used as the base page and the remainder for the user program.

The size of a given partition depends upon program needs. A partition resident program, out of its 32K address space, has 10 or more pages taken up by the system and library area. Some programs use a COMMON area that must be mapped. This may result in less address space for the programs, depending upon the size and location of the COMMON area. A useful partition therefore will normally be between 4 and 22 pages long.

The RTE-M Generator reports the the largest useful partition sizes for programs with and without COMMON (including a base page for the program) to aid you in determining partition sizes. Partition sizes for each program relocated are also reported; however, some programs may require additional pages for buffer area. It may not be possible for you to completely plan partition sizes until this information is reported by RTMGN. Figure 2-3 shows two typical MIII user maps; one showing COMMON and one without.

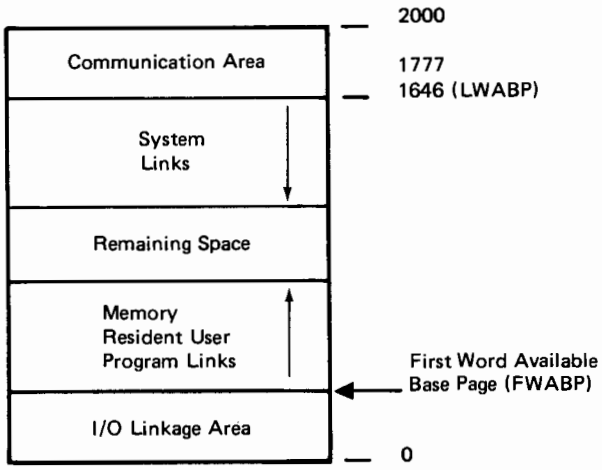
2-14. CHANGING BOUNDARY ADDRESSES. During the memory configuration phase of generation, RTMGN has loaded the system and reports the actual size of the COMMON areas in decimal number of words and the octal boundary addresses. As each area is reported, you have the opportunity to either increase the size or address or leave it as it is. Your responses to RTMGN depend upon your analysis of the reported data.

Some of the boundary address changes are for the convenience of your site-specific RTE-M system. That is, you change a reported boundary to allow the associated area to begin at the start of a memory page. If there is to be no change, you enter a zero (0). Refer to Figure 2-4 in locating the areas referenced in the following paragraphs.

1 refers to memory size in decimal number of pages in the system, and is relevant only for MIII systems.

The first boundary is A, and concerns the last word address of available m;emory (LWAM) in MI and MII systems (only). It is the last address that resident user programs can be loaded to. Increase the size if desired.

The next area that be changed is B, which is the base page first word address. The octal address specifies where RTMGN may begin allocating base page links for the resident user programs. You should make this number higher than the highest location needed for I/O linkages.



C,D, the Real-Time COMMON area,is the next boundary that can be changed. The decimal number of words you wish to be allocated for Real-Time COMMON is entered. The number you specify may be 0 for no change, or some positive number to change the number of words allocated. The first word available (LWA) of Real-Time COMMON is then printed. Closely associated with this boundary change is the alignment (D) of the end of COMMON at the next page boundary to protect user programs (MII and MIII only). You align the end of COMMON to the next page boundary by entering a YES when queried by RTMGN. An updated last word address of COMMON will be issued.

The next area is E,F and concerns the resident user program area. RTMGN issues the LWA of the user program area and asks if you want to increase it. You enter a desired new address in octal, or 0 if no increase is wanted. For MII and MIII systems, RTMGN will ask if you want the user program ;area to end the current page boundary. If you reply with YES, System Available Memory will be aligned to the next page. The size of SAM will then be issued.

The last area that can be changed (MIII systems only) is G, and concerns additional pages for SAM. In MI and MII systems, SAM extends to the end of available memory and RTMGN will query these systems. If you have an MIII system, RTMGN will ask for the number of additional pages desired for SAM. You enter up to the allowable maximum number of pages in decimal (a number greater than the allowable maximum causes an error code to be output).

This concludes the boundary addresses that can be changed. They were discussed separately here because, to some extent, you cannot fully plan your final memory configuration until RTMGN provides some of the data necessary to make your decisions. The full range of RTMGN prompts and queries, and your reponses that comprise generation are covered in succeeding sections of this manual.

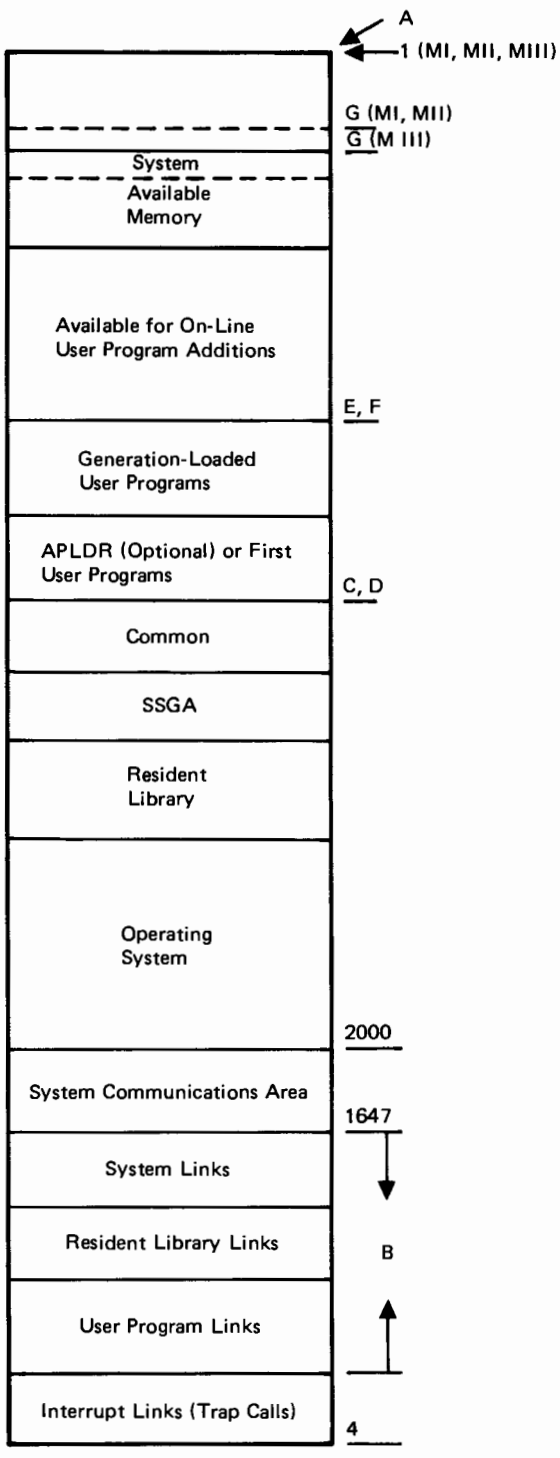


Figure 2-4. User Changeable Boundary Addresses

GENERATION PLANNING

SECTION III

Generation planning involves determining what software resources and services are to be incorporated into your resident operating system, and how those resources are to be allocated to best meet your installation's intended purpose. You define the decisions in a series of reponses in dialog with the RTE-M Generator at generation time. All your responses are first entered in the Generation Worksheets supplied in Appendix D (sample worksheets with input examples for each generation "phase" are included in this section).

The data entered in the worksheets is partly derived from the I/O Configuration Worksheet and other considerations described in Section II, plus the requirements and desired optional resources described in this section. When all the Generation Worksheets are filled out, you will be ready to establish a dialog with RTMGN, using the worksheets for fast and accurate reponses to the Generator's queries.

3-1. SYSTEM INITIALIZATION PLANNING

During generation, RTMGN identifies itself and begins outputting a series of queries and prompts in the sequence given below. Following each RTMGN prompt or query listed are the requirements and other items you must take into account when planning your responses on the worksheets. All data output by RTMGN given below is shown in **BOLDFACE** type.

When you first call the Generator it responds by issuing the following messages:

RTMGN
DEFINE OUTPUT DEVICES
–

where the hyphen (-) is an RTMGN prompt character and your respond immediately below the prompt. Your only required response to the above message is

OUTPUT ON file name or LU number
END

However, you can also specify an echo device and the MAP option at this time, or you can specify them later during other phases of generation (see other Generation Worksheets in Appendix D).

A sample Initialization Phase Worksheet is illustrated in Figure 3-1. New users should study the example and material in this subsection before filling in the actual worksheet supplied in Appendix D. In Figure 3-1, note that the optional echo file is assigned to LU6 and later (Figure 3-2), the MAP file is assigned to the same LU during System Module Relocation. This is legal but produces a mixed output that will be unusable as an Answer File, in case you want to use it for this purpose in succeeding generations.

TYPE OF SYSTEM?

Enter a decimal 1,2 or 3 to designate whether RTMGN is to configure an MI, MII or MIII system. Following your response, RTMGN will delete or add some of the queries appearing on the worksheets as appropriate for the specified system.

TBG CHNL?

Enter the two octal digits you specified on the I/O Configuration Worksheet for the Time Base Generator. Enter 0 if TBG is not to be implemented.

PRIV INT CARD ADDR?

Enter the octal select code specified on your I/O Configuration Worksheet. Enter a 0 if there is no such card.

PRIVILEGED DRIVERS ACCESS COMMON?

The question is asked only for MIII systems. Specify whether or not the common area is to be included in the system map for access by privileged drivers. Enter NO to deny privileged drivers access to the common area through the system map or if there are no privileged drivers.

MEM SIZE?

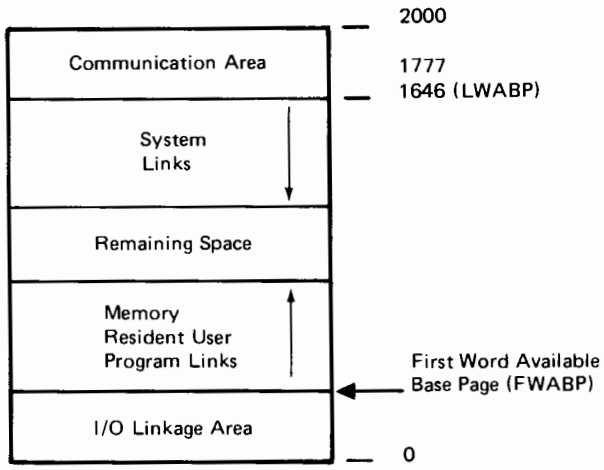
The question is asked only for MIII systems. Specify a decimal value indicating the total number of pages of memory (i.e., 32 for 32K; 64 for 64K, etc.).

LWAM?

The question is asked only for MI and MII systems. Enter the octal value representing the last word of available memory. This is the last address that memory resident programs may be loaded to.

FWA BP?

Enter the octal address where RTMGN may begin allocating base page links for the memory resident user programs. The octal number must be higher than the highest location needed for the I/O linkages as shown in the diagram below:



The first word available for base page linkages is derived from your I/O Configuration Worksheet as the last I/O select code location entry, plus 1. For instance, if the last priority (highest octal select code number) on your I/O Worksheet is 20, you would enter the FWABP linkage on the Initialization Worksheet as 21 or greater. This location is used by RTMGN as the starting point in building the system.

CHANGE ENTS? -

Enter the entry records to be changed, if any, and their octal values, or enter END to indicate no changes are desired. If you have microcoded routines available that your software is not utilizing but could be, you should consider changing the appropriate entry record addresses (see the relevant I/O driver manual). Although an entry of END without changes will not impact successful generation, you thereafter may not be using your system at its full potential and maximum efficiency. RTMGN will issue a hyphen prompt after each entry point change. Each ENT to be changed has the following form:

entry,type, value

where:

entry is the entry point name

type is the entry point type, where

AB= absolute
RP= replace

value is the entry point instruction value in octal.

When you declare an entry point as absolute (AB), its value is added to the referencing instruction to obtain the final instruction value.

RTE-M INITIALIZATION WORKSHEET

BOLDFACE terms are output to user by RTMGN. All other data (except comments) is user input. Fill in blanks with desired parameters.

RTMGN
DEFINE OUTPUT DEVICES

ECHO ON 6 (optional-enter LU or file name)
OUTPUT ON !MSYS (required-enter LU or file name)
MAP _____ , _____ , _____ , ON _____ (optional-enter LU or file name)
END (required)

TYPE OF SYSTEM?
3 (enter 1, 2 or 3 for MI, MII or MIII)

TBG CHNL?
10 (octal select code from I/O Worksheet)

PRIV INT CARD ADDR?
0 (octal select code, or 0 if none)

PRIVILEGED DRIVERS ACCESS COMMON? (MIII systems only)
NO (YES or NO)

MEM SIZE? (MIII only)
48 (decimal no. pages)

LWAM? (MI and MII only)
_____ (last word of available memory in octal)

FWA BP?
20 (octal address-higher than location for I/O links)

CHANGE ENTS? (repeated until END encountered — optional octal entries-see "Initialization Phase in Section III before changing entry points)

.MPY , RP , 100200	.FSB , RP , 105020	.MBT , RP , 105765
.DIV , RP , 100400	.FMP , RP , 105040	.MVW , RP , 105777
.DLD , RP , 104200	.FDV , RP , 105060	_____ , _____ , _____
.DST , RP , 104400	.IFIX , RP , 105100	_____ , _____ , _____
.FAD , RP , 105000	FLOAT , RP , 105120	END (required)

Figure 3-1. MIII Initialization Example

When you declare an entry point as replace (RP), the Loader will replace each reference to it with the octal value; for example:

.FMP,RP,105040

would cause each JSB .FMP instruction (Floating Point Multiply) to be changed to the microcode version of the floating point multiply instruction (105040). Other floating point or fixed point EAU type instructions that could be entered are as follows:

Floating Point	Fixed Point
.FAD,RP,105000 — Add	.MPY,RP,100200
.FSB,RP,105020 — Subtract	.DIV,RP,100400
.FMP,RP,105040 — Multiply	.DLD,RP,104200
.FDV,RP,105060 — Divide	.DST,RP,104400
IFIX,RP,105100 — Fix	.MBT,RP,105765
FLOAT,RP,105120 — Float	.MVW,RP,105777

At this point in generation, a master security code may be established to provide a full file protection system for flexible disc files (only). The method for defining and accessing protected files is described in Section III (NAMR) of the RTE-M Programmer's Reference Manual. The CHANGE ENTS format for setting up a master security code is

$SECM,AB,xxxxxx

where:

$SECM — is the entry point to the security code module

xxxxxx — is the octal equivalent for two ASCII characters used for the security code (see also File Management Package considerations under "Resident Library Relocation" in Section 3-5 of this manual).

CAUTION

If implemented, make sure you maintain a private record of your master security code for later file access and file directory listings of protected files.

Other uses of entry point changes include I/O configuration at load time, and configuring tables that are assembled as DEF statements to externals. Enter an END after the last desired CHANGE ENTS entry to terminate the Initialization Phase.

3-2. SYSTEM RELOCATION PLANNING

In the next generation "phase", RTMGN relocates the required and optional system modules and outputs an absolute RTE-M resident system. You direct RTMGN in the building of the absolute resident through a subset of the Relocating Loader commands recognized by RTMGN (see the Relocating Loader section in the *RTE-M Programmer's Reference Manual* for a complete description of the format and function of Loader parameters).

REL. SYS MOD

Relocate the required system modules for your RTE-M version and perform the necessary library searches.

It is recommended that you refer to the sample entries in the System Module Relocation Worksheet illustrated in Figure 3-2 when studying the considerations given below. The names and part numbers of all required and HP-supplied optional system modules can be found in Appendix E of this manual. Driver names and interrupt subroutine names are given in the RTE-M Software Numbering Catalog.

Table 3-1 below defines some of the requirements, considerations and suggested loading order for the system software. You should study it before filling out the System Module Relocation Worksheet supplied in Appendix D. Note that all modules loaded during this phase are type 0, 7 or 8 only. The first modules to be loaded are the main system relocatables, which will be one of three: %MSY1, %MSY2 or %MSY3, depending upon whether you are configuring an MI, MII or MIII system. RTMGN will issue a hyphen prompt character at the conclusion of each module relocation.

Table 3-1. System Module Relocation Considerations

REL SYS MODS		
–		
REL { %MSY1 / %MSY2 / %MSY3 }	(MI) / (MII) / (MIII)	Main System Modules
REL %MBU	(MI only)	Optional Output Buffering
REL %MMP	(MI only)	Optional Multi-program Scheduling
REL %MTI	(MI, MII, MIII)	Optional Timer
REL %MTS	(MI, MII, MIII)	Optional Timer, Operator/Program Functions
REL %MOP	(MI, MII, MIII)	Optional LU, EQ, RL Operator Requests
NOTE: %MTI and %MTS require that TBG be installed.		
REL { %MCL / %MCL3 }	(MII) / (MIII)	Optional Class I/O (Note: requires %MRN)
REL %MRN	(MII, MIII)	Optional Resource Numbers
SEARCH %MDMLB	(MI, MII, MIII)	System Dummy Entry Point Libriary (Note: Satisfies external REF's to optional modules not loaded)
REL { %DVRxx / %. / . / . / %DVxyz }		File names of all required drivers
REL %MPF	(MI, MII)	Optional Power Fail Recovery Driver
REL %MPF3	(MIII)	Optional Power Fail Recovery Driver
Note: %MPF or %MPF3 require that DVR43 be defined on I/O Configuration Worksheet.		
END	Terminates System Modules Relocation Phase	

RTE-M SYSTEM MODULES RELOCATION WORKSHEET

BOLDFACE terms are output to user by RTMGN.

– Hyphens are RTMGN prompts. All other data (except comments) is entered by user. Fill in blanks with desired parameters.

REL SYS MODS

LINKS IN CURRENT (optional; enter BASE or CURRENT)

– MAP MODULES , _____ , _____ , ON 6 (not necessary if specified previously)

– RELOCATE %MSY3 (system modules name - see RTE-M Software Numbering Catalog)

– RELOCATE %MTI (optional system modules; class I/O, resource numbers, powerfail, etc.)

– RELOCATE %MTS

– RELOCATE %MOP

– RELOCATE %MCL3

– RELOCATE %MRN

– SEARCH %MDML3 (Dummy library to satisfy unwanted external REFS)

– RELOCATE %DVR33

– RELOCATE %4DV05 (enter all I/O driver names and interrupt subroutine names - refer to appropriate I/O Driver manuals for names)

– RELOCATE _____

– RELOCATE %DV37

– RELOCATE %MPF3

– RELOCATE END

– RELOCATE _____

– RELOCATE _____

– DISPLAY UNDEFS (optional)

– END (required to terminate system module relocation)

STARTING ADDRESS 0002
NO UNDEFS

SEARCH _____ (any additional required library searches; indicate by line and arrow where they should be made.)

SEARCH _____

Figure 3-2. MIII Module Relocation Example

3-3. TABLE GENERATION PLANNING

During this phase of generation, you will build the EQT, LU, and Interrupt Tables. For MII and MIII systems only, RTMGN first queries you as to the number of I/O class numbers and resource numbers desired, if any. For all RTE-M systems, RTMGN then asks for the buffer limits of devices for which buffering was specified on your I/O Configuration Worksheet. It then queries you regarding the EQT, Device Reference and Interrupt Table entries. See Figures 3-3 through 3-5 for examples of the three Table Building Worksheets. The sequence of RTMGN queries is as follows:

 # OF I/O CLASSES?

Enter the desired decimal number of I/O classes (between 0-255). Multi-terminal operation requires one class number,and one for each class GET call that is simultaneously outstanding (see Section IV of the *RTE-M Programmer's Reference Manual*).

For example, if you specify 10 class numbers, user programs may simultaneously process class requests. A response of 0 means that no class requests can be made. If you make any response other than 0, you must have relocated %MCL for MII systems or %MCL3 for MIII systems, plus %MRN during the System Module Relocation phase (see Table 3-1).

 # OF RESOURCE NUMBERS?

Enter the decimal number of resource numbers desired (0-255). There must be one number for each resource to be controlled (see Section IV of the *RTE-M Programmer's Reference Manual*). For example, if 10 RN's are specified, 10 resources (devices or files) can be managed and used by cooperating user programs.

If you specify any number other than 0, you must have previously loaded the Resource Management module (%MRN) during the System Module Relocation phase.

BUFFER LIMITS (LOW,HIGH)?

Enter the decimal number of words for upper and lower buffer limits (see example in Figure 3-3). Enter 0 if no buffer limits are desired. Setting upper and lower limits during generation can prevent an inoperative or slow I/O device from monopolizing system available memory.

Each time a buffered I/O request is made (all class I/O requests are buffered), the system totals the lengths of all buffers for I/O requests queued to that EQT entry and compares the number to the upper limit set either during generation or later using the BL command. If the sum is less than the upper limit, the new buffered request is added to the queue. If the sum is larger than the upper limit, the requesting program is suspended in the general wait list (STATUS=3).

When a buffered I/O request completes, the system adds up the remaining words in I/O requests queued to that EQT entry and compares the number to the specified lower limit. When the sum is less than the lower limit, any program suspended for exceeding the buffer limits on this EQT are rescheduled and may again attempt their request. A suggested entry of 100 to 400 words can be entered during generation and changed on-line later with a BL command, if desired.

EQT TBL
EQT n=?

Transfer the EQT entry data from your I/O Configuration Worksheet into the correct sequenced entry on page 1 of the two Table Building Worksheets given in Appendix D (see also Figure 3-3). For each EQT parameter entry, enter the corresponding octal select code, driver number "name", B for buffering if it is specified, D for DMA if specified, T=n if timeout is specified, and X=n for decimal number of words for extended memory. Leave out any parameters not needed (e.g., T= and X=) in a given EQT entry.

> NOTE
>
> Buffering must NOT be specified for any flexible discs (DVR33) on the system, either during generation or on-line with an EQ command.

In the example in Figure 3-3, note that only the highest priority (lowest numbered) select code is entered for the two flexible discs that were specified in Figure 3-1.

Any driver of the following format can be entered:

DVxyz — where xyz are the ASCII characters specified for the driver in in the appropriate driver manual.

Terminate the EQT Table Building phase by entering an END command after your last desired EQT entry. RTMGN will then automatically go into the Device Reference Table prompt.

DRT TBL
LU#: n = EQT#?

Enter the correct EQT number and (if appropriate) subchannel number for each logical unit number in the sequence requested by RTMGN. You transfer the data from the "EQT/SUBC" column of your I/O Configuration Worksheet into page 2 of 3 of the Table Building Worksheets in Appendix D.

An example of prepared entries for a DRT Table Worksheet is illustrated in Figure 3-4, which in turn, was derived from the sample I/O worksheet illustrated in Figure 3-1.

EQT TABLE BUILDING WORKSHEET

BOLDFACE terms are output to user by RTMGN.
– Hyphens are RTMGN prompts. Fill in blanks with desired parameters.

#OF CLASSES? (MII and MIII only)
10 (decimal no. between 0 - 255)

#OF RESOURCE NUMBERS? (MII and MIII only)
10 (decimal no. between 0 - 255)

BUFFER LIMITS (LOW, HIGH?
100 , 400 (decimal no. of words. Enter 0, 0 for no limits)

EQT TBL (Equipment Table entries; get data from I/O Configuration Worksheet)

EQT 1=?
11 , DV 233 , D , _____ , T = 50 , X = _____

EQT 2 = ?
13 , DV 005 , B , _____ , T = _____ , X = 13

EQT 3 = ?
14 , DV 237 , D , _____ , T = _____ , X = 50

EQT 4 = ?
4 , DV 243 , _____ , _____ , T = _____ , X = _____

EQT 5 = ?
END , DV _____ , _____ , _____ , T = _____ , X = _____

EQT 6 = ?
_____ , DV _____ , _____ , _____ , T = _____ , X = _____

EQT 7 = ?
_____ , DV _____ , _____ , _____ , T = _____ , X = _____

EQT 8 = ?
_____ , DV _____ , _____ , _____ , T = _____ , X = _____

EQT 9 = ?
_____ , DV _____ , _____ , _____ , T = _____ , X = _____

EQT 10 = ?
_____ , DV _____ , _____ , _____ , T = _____ , X = _____

EQT n = ?
END (required-terminates EQT Table entries where desired)

See Section III, "EQT Tables" for format and input considerations.

Figure 3-3. MIII EQT Table Building Example

DRT TABLE BUILDING WORKSHEET

BOLDFACE terms indicate RTMGN prompts output to user. All other data (except comments) is user input.

DRT TBL (Device Reference Table)
LU #: (Logical Unit Nos.)

1 = EQT # ?
2 , Ø

2 = EQT # ?
1 , Ø

3 = EQT # ?
Ø , _____

4 = EQT # ?
2 , 1

5 = EQT # ?
2 , 2

6 = EQT # ?
3 , 1

7 = EQT # ?
3 , Ø

8 = EQT # ?
Ø , _____

9 = EQT # ?
Ø , _____

10 = EQT # ?
Ø , _____

11 = EQT # ?
Ø , _____

12 = EQT # ?
4 , _____

(Enter EQT entry nos. and subchannel nos. if applicable from I/O Worksheet)

13 = EQT # ?
END , _____

14 = EQT # ?
_____ , _____

15 = EQT # ?
_____ , _____

16 = EQT # ?
_____ , _____

17 = EQT # ?
_____ , _____

18 = EQT # ?
_____ , _____

19 = EQT # ?
_____ , _____

20 = EQT # ?

END (required to terminate DRT Table entries where desired)

Figure 3-4. MIII DRT Table Building Example

The first seven LU numbers are reserved for system devices as follows:

LU number 0 bit bucket (no prompt is issued for zero)
1 system console
2 system peripheral (tape cartridge/flexible disc, etc.)
3 " " "
4 standard output unit (left CTU)
5 standard input unit (right CTU)
6 standard list unit

LU 0 (bit bucket) is a system mechanism that allows immediate I/O completion; that is, the data buffer is written to or read from a non-existant device.

Extra LU numbers can be assigned using EQT number 0, and may be changed on-line to reference other EQT numbers if desired. If LU 2 and 3 are not used, you should reserve them by entering a zero (0) when RTMGN prompts you for their entries. Terminate the DRT entries with an END when you have entered all the LU's in sequence from your I/O Configuration Worksheet.

INT TBL
?

Transfer the select codes for each I/O location and matching parameters from the Interrupt Table Entry column of your I/O Configuration Worksheet into the Interrupt Table Building Worksheet as illustrated in Figure 3-5. RTMGN will prompt with a new question mark after you make each entry during generation. Make your entries in the ascending order of the octal select codes; that is, the lowest numbered (highest priority) select code and its Interrupt Table parameters is entered first.

NOTE

Do not make an Interrupt Table entry for the Time Base Generator (TBG) if it was specified during initialization. RTMGN automatically sets all parameters for TBG during initialization. All other I/O cards (select code entries) MUST have an Interrupt Table entry.

The Interrupt Table contains the address of the program ID segment or I/O driver EQT. There is an entry for each I/O channel requiring interrupt processing. All I/O locations from 5B to FWA BP-1 linkage are initially set to contain "JSB $CIC". Select code 4 is set to a HLT. The entries in the Interrupt Table include one word for each channel, from 6B to the I/O location specified by the last INT record entered.

INTERRUPT TABLE WORKSHEET

BOLDFACE terms are output to user by RTMGN

Question marks are RTMGN prompts. All other data (except comments) is entered by user. Fill in blanks with desired parameters.

INT TBL (Interrupt Table)

? 4, ENT, $POWR	? , ,
? 11, EQT, 1	? , ,
? 12, EQT, 1	? , ,
? 13, EQT, 2	? , ,
? 14, EQT, 3	? , ,
? , END ,	? , ,
? , ,	? , ,
? , ,	? , ,
? , ,	?
? , ,	END (required-terminates Interrupt Table entries where desired)
? , ,	
? , ,	

Get input data from I/O Configuration Worksheet. See "Interrupt Table" in Section III regarding four different forms of parameters.

Figure 3-5. MIII Interrupt Table Building Example

There are four possible options for constructing Interrupt Table entries:

select code,EQT,nn	relates I/O channel to EQT entry
select code,PRG,name	causes the named program to be scheduled for execution upon interrupt. The interrupt location will contain "JSB $CIC". The named program must be loaded during the user program relocation phase (described later in this section), either during generation or on-line at later time. If the interrupt programs are not relocated during generation and a request is given to terminate RTMGN, a reminder is output and you are asked whether or not to ignore relocating the programs. If the answer is YES, the program(s) is expected to be loaded on-line; if it is not, a SC03 error code is issued if the interrupt occurs while the system is running. A typical entry for PRG entry is PRG,ALARM (see Figures 3-1 and 3-5). ALARM can be the one supplied by HP or user written.
select code,ENT,entry	causes control to transfer to the entry point for either the HP-supplied powerfail program ($POWR) or a user-written system program. The interrupt location associated with the select code is set to contain "JSB x,I" where x is the base page link to the named entry point. The program that contains the named entry point MUST have been relocated in the RTE-M resident during system module relocation and before the Interrupt Table is built. Otherwise, an error (AD) occurs, designating an invalid entry point in record message. The entry point is entered directly when an interrupt occurs in the interrupt location associated with specified select code. Note that interrupt location 4 (powerfail) may be changed from its HLT 4 to an ENT entry if a powerfail routine was included, as was done in the example in Figures 3-1 and 3-5.
select code,ABS, xxxxx	places an absolute octal value xxxxx (instruction code) in the interrupt location. It may be a microcode call, but may not be I/O for HP 21MX-E computer systems and is therefore not recommended.

Enter an END to terminate the Interrupt Table Building phase after your last entry is made.

3-4. TABLE BUILDING CHANGE PROCEDURES

You are allowed to repeat any of the EQT, DRT and INT table building sessions to make changes. To repeat a given session, you must either presently be in that session or have already executed the table building session to be repeated. Within any one session, only the repeat requests specified below are acceptable:

Table Building Session	Legal Repeat Request	Resulting Action
EQT	RE	RE will go to build EQT, DRT, INT
DRT	RE or RD	RD will go to build DRT, INT
INT	RE or RD or RI	RI will go to build INT

The tables are built in the above order. Any time a table building sequence is to be repeated, the subsequent table building will also be repeated or completed. The repeat requests are only valid when performing the table building portion of generation, and cannot be used to repeat any other generation phase. An attempt to do so will result in an error.

3-5. PROGRAM PARTITION/PARAMETER PLANNING

For this phase of generation, you will plan the maximum number of program partitions (MIII only), number of ID segments to be allocated, Resident Library relocation, and user program relocation. You should refer to the sample worksheet illustrated in Figure 3-6 while studying the requirements and options described for this phase. RTMGN outputs the following prompts and queries:

MAX NUMBER OF PARTITIONS?

The question is only asked for MIII systems. Enter the decimal number of program partitions desired. You determine this number by dividing up the remaining pages after system available memory. You may specify up to 64 partitions, but you must remember that each partition requires a minimum of two pages; one for a base page and one for the program.

#ID SEG?

Enter the decimal number of blank ID segments to be allocated for the system (at least one must be allocated). One ID segment is required for each program to be relocated on-line by the Relocating Loader. The parameters for the ID segment are filled in by APLDR when it loads the absolute program into main memory. If five ID segments are allocated , then only five programs can be built in or loaded on-line into the system at any one time. The total number of ID segments that may be allocated is 99.

A program's ID segment can be recovered for system use by other programs by deleting the program with an OF name,8 operator command.

RTE-M PROGRAM PARTITION/PROGRAM PARAMETER INPUT WORKSHEET (1 of 2 pages)

BOLDFACE terms are output to user by RTMGN.

– Hyphens are RTMGN prompts. All other data (except comments) is entered by user. Fill in blanks with desired parameters.

MAX NUMBER OF PARTITIONS? (MIII only)

5 (decimal no. - up to 64)

#ID SEG?

10 (decimal no - at least 1 must be allocated)

START-UP PROG?

0 (program name - enter 0 if no start-up program.

REL RES LIB

– LINKS IN _____ (optional)

– MAP _____ , _____ , _____ , ON _____ (optional)

– RELOCATE %TBLFP(_____) There are two format options to load library routines:

– RELOCATE _____ (_____)

– RELOCATE _____ (_____) RELOCATE { file name / file name (module) }

– RELOCATE _____ (_____)

– SEARCH %FMPF (optional as needed)

– DISPLAY UNDEFS (optional)

– END (required-terminates relocation of library routines)

STARTING ADDRESS 0002
NO UNDEFS
RELOCATE SSGA

– RELOCATE _____ (optional - these are type 30 modules)

– RELOCATE _____

– RELOCATE _____

– END (required to terminate phase)

#WDS IN COMM?

50 (decimal no. of words to allocate for common)

LWA OF COMMON = xxxx

ALIGN AT NEXT PAGE?

YES (enter YES or NO)

LWA OF COMMON = xxxx (only output if above answer is YES)

WORKSHEET IS CONTINUED ON NEXT PAGE

Figure 3-6. MIII Program Partition/Resident Library Relocation Example

START-UP PROG?

Enter either 0 for no start-up program, or a program name for the start-up program. When the RTE-M resident is loaded into main memory and executed, the start-up program is automatically scheduled for execution. If specified, the program must be relocated as a user program. If you enter a start-up program name but the program is not relocated or another program's name was not changed to the start-up name, RTMGN will not be allowed to end. RTMGN will issue a reminder and make a request for another relocation. Any parameters entered after the program name will be ignored.

REL RES LIB
–

Enter the names of the library routines to be relocated to create the the Resident Library. The correct file names of the routines to be relocated can be found in the RTE-M Software Numbering Catalog. RTMGN will prompt with a hyphen character at the end of each relocation. There are two format options for making resident library entries:

REL file name

or

REL file name (module)

The LINKS IN and MAP options may be inserted prior to library relocation if they have not already been specified or if you wish to change their parameters. The SEARCH command is inserted wherever needed to satisfy external references.

It is good practice during generation to enter DISPLAY UNDEFS after every library search. The goal is to resolve every unsatisfied external reference and get the NO UNDEFS message from RTMGN at the end of the library routine relocation session.

Other reentrant/privileged library or user routines can be put into the Resident Library at this time.

If the File Management Package (FMP) is to be used on your system, the following routines must be relocated in the Resident Library:

REL %TBLCR	(MI, MII, MIII)	Tape Cartridge FMP Directory Table
REL %DIRD	(MI, MII, MIII)	Tape Cartridge Directory Read Routine
or		
REL %TBLFT	(MI, MII, MIII)	FLEXIBLE Disc FMP Directory Table
SEARCH %FMPF	(MI, MII, MIII)	Only necessary to satisfy the SECM security EQU if $SECM, AB, xxxxx was not defined during CHANGE ENTS (see Section 3-1).

REL %DRC1	(MI w/o %MMP)	Tape Cartridge Directory Handling Subr.
or		
REL %DRF1	(MI w/o %MMP)	Flexible Disc Directory Handling Subr.

> NOTE
>
> Use %DRC1 or %DRF1 only for MI systems without the scheduling option. If scheduling (%MMP) is included, load user program %DRC or %DRF into the user program area.

SEARCH %RLIB1 (only necessary for the .ENTP external reference in %DRC1
SEARCH %RLIB2 or %DRF1)

Enter an END to terminate Resident Library relocation at the desired point, or immediately after RTMGN outputs the REL RES LIB prompt if no Resident Library is to be created.

REL SSGA
–

Enter the module names of all type 30 modules that will comprise the Subsystem Global Area, if implemented. Enter an END following the last module relocated, or immediately following the REL SSGA prompt if SSGA is not to be used. RTMGN will issue a hyphen prompt character after each SSGA module is relocated. The SSGA area is accessed by entry point rather than COMMON declarations. It provides multiple communication and buffer areas for HP susbsystems.

#WDS IN COMM?

Enter the decimal number of words to be allocated for COMMON storage. Enter a zero (0) if no COMMON is to be allocated. Following your response, RTMGN will issue the message and query:

LWA OF COMMON = xxxx
ALIGN AT NEXT PAGE?

Enter YES or NO. A YES response causes the end of COMMON to be aligned at the next page boundary to protect memory resident programs. Regardless of your response, RTMGN will then output an updated version of the LWA OF COMMON = xxxx message, giving the last word address of COMMON. A YES response is recommended for MIII systems because of possible loss of protection for the p;rogram on the next page following COMMON.

3-6. USER PROGRAM PLANNING

For this phase of generation planning, you decide which HP-supplied user programs and programs you may write yourself are to be relocated as part of the RTE-M resident system. This will include all user programs that will NOT be residing in the User Program area or (for MIII systems) in partitioned memory. Your primary reference source for the correct file names of all HP-supplied relocatable user modules is the *RTE-M Software Numbering Catalog*. A convenient list of file names for the relocatable modules is also given in Appendix E this manual.

CAUTION

It is extremely important that you use the correct file name when searching the tape cartridge or flexible disc libraries. For example, a tape-oriented library such as %FMPC will satisfy external references in a flexible disc system generation/ relocation that will appear to complete successfully. However, severe system problems will occur when the programs are run. You should check the file names used against the described library version, listed in Appendix E or in the RTE-M Software Numbering Catalog, before actually entering them.

You make user program relocation entries on page 2 of 2 of the Program Partition/Program Parameter Worksheets given in Appendix D. Duplicate this worksheet on an office copier for as many copies as you have programs to be relocated, since you will be filling out a separate sheet for each user program. There is a convenient box at the bottom right-hand corner of the sheet for inserting a sequence number to inicate the planned relocation order.

Sample user program worksheets are illustrated in Figure 3-7. It is recommended that you refer to the figure while studying the considerations given in this section.

User programs can be relocated as part of the resident RTE-M system at this point in generation, or loaded on-line at a later period when the system is up and running. The full procedure for on-line loading of user programs is given in Sections VI and VII of the RTE-M Programmer's Reference Manual. However, note that the same considerations and constraints described below for program relocation at generation time will also exist for on-line loading. Also note that APLDR and the Relocating Loader MUST be relocated as user programs during generation if on-line loading is to be performed, with the Relocating Loader being the last module relocated.

RTE-M PROGRAM PARTITION/PROGRAM PARAMETER INPUT WORKSHEET (2 of 2 pages)

NOTE: Reproduce one copy of this worksheet for each user program to be relocated.

REL USER PROGS (See Relocating Loader in RTE-M Programmer's Reference Manual)

- LINK IN _____ (unnecessary if specified in Initialization Worksheet)
- MAP _____ , _____ , _____ , ON _____ (see comment above)
- RELOCATE %DRF (use program name)
- RELOCATE _____
- RELOCATE _____
- RELOCATE _____
- RELOCATE _____
- RELOCATE _____

(any subroutine names)

- SEARCH %FMPF
- SEARCH %MSYLB
- SEARCH %RLIB1
- SEARCH %RLIB2

(optional - library searches as required)

- DISPLAY UNDEFS (optional
- END (required after each user program relocation)

STARTING ADDRESS xxxx
NO UNDEFS
ENTER PRAMS

0 _____ , _____ , _____ , _____ , _____ , _____ , _____ , _____ , _____ , _____

(See "Program Parameter Input" in Section III for format - enter a 0 if no changes are to be made and defaults are wanted)

REL USER PROGS

- END (enter only if last program has been relocated; otherwise, enter next program name. END terminates all user program relocation.)

NOTE: Any user program that is a Loader or Generator should be relocated last.

ENTER RELOCATION SEQUENCE NO. HERE: 1

Figure 3-7. MIII User Program Relocation Example (Sheet 1 of 3)

RTE-M PROGRAM PARTITION/PROGRAM PARAMETER INPUT WORKSHEET (2 of 2 pages)

NOTE: Reproduce one copy of this worksheet for each user program to be relocated.

REL USER PROGS (See Relocating Loader in RTE-M Programmer's Reference Manual)

- LINK IN _____ (unnecessary if specified in Initialization Worksheet)
- MAP _____ , _____ , _____ , ON _____ (see comment above)
- RELOCATE %MAUTO (use program name)
- RELOCATE _____
- RELOCATE _____
- RELOCATE _____
- RELOCATE _____
- RELOCATE _____

(any subroutine names)

- SEARCH _____
- SEARCH _____
- SEARCH _____
- SEARCH _____

(optional - library searches as required)

- DISPLAY UNDEFS (optional
- END (required after each user program relocation)

STARTING ADDRESS xxxx
NO UNDEFS
ENTER PRAMS

0 _____ , _____ , _____ , _____ , _____ , _____ , _____ , _____ , _____ , _____

(See "Program Parameter Input" in Section III for format - enter a 0 if no changes are to be made and defaults are wanted)

REL USER PROGS

- END (enter only if last program has been relocated; otherwise, enter next program name. END terminates all user program relocation.)

NOTE: Any user program that is a Loader or Generator should be relocated last.

ENTER RELOCATION SEQUENCE NO. HERE: 2

Figure 3-7. MIII User Program Relocation Example (Sheet 2 of 3)

RTE-M PROGRAM PARTITION/PROGRAM PARAMETER INPUT WORKSHEET (2 of 2 pages)

NOTE: Reproduce one copy of this worksheet for each user program to be relocated.

REL USER PROGS (See Relocating Loader in RTE-M Programmer's Reference Manual)

- LINK IN _____ (unnecessary if specified in Initialization Worksheet)
- MAP _____ , _____ , _____ , ON _____ (see comment above)
- RELOCATE %MAP3 (use program name)
- RELOCATE _____
- RELOCATE _____
- RELOCATE _____
- RELOCATE _____
- RELOCATE _____

(any subroutine names)

- SEARCH %FMPF
- SEARCH %MSYLB
- SEARCH %RLIB1
- SEARCH %RLIB2

(optional - library searches as required)

- DISPLAY UNDEFS (optional
- END (required after each user program relocation)

STARTING ADDRESS xxxx
NO UNDEFS
ENTER PRAMS

0 _____ , _____ , _____ , _____ , _____ , _____ , _____ , _____ , _____ , _____

(See "Program Parameter Input" in Section III for format - enter a 0 if no changes are to be made and defaults are wanted)

REL USER PROGS

- END (enter only if last program has been relocated; otherwise, enter next program name. END terminates all user program relocation.)

NOTE: Any user program that is a Loader or Generator should be relocated last.

ENTER RELOCATION SEQUENCE NO. HERE: 3

Figure 3-7. MIII User Program Relocation Example (Sheet 3 of 3)

The following types of user programs must be relocated and loaded on-line rather than during generation:

- User-written segmented programs.
- HP-supplied segmented programs (e.g., FORTRAN AND Assembly Language procressors).
- Programs that will reside in partitioned memory (MIII systems only).

Relocating the RTE-M Real-Time BASIC processor as a user program requires special considerations. It can be loaded during generation or on-line. The procedures are fully described in Section V of this manual.

At the end of each program relocation during generation, RTMGN places the program name, memory bounds and transfer address into one of the empty ID segments you previously allocated following the "# ID SEG?" prompt (see Figure 3-6). Remember that each program relocated at generation requires one ID segment (as do programs loaded on-line). If all available ID segments have been used up, RTMGN issues an automatic END and no further relocation can take place.

Some HP-supplied user programs require extra memory space behind themselves for buffer area or symbol table space, and extra space is recommended for one other (FMGR). For all MI or MII systems (or MIII systems that will have one or more user programs in the resident system), you set the start point of the memory space either with a BOUNDS command at the start of program relocation, or with a SET LOCC AT xxxxx command, where xxxxx is an integer value in octal. See Section VI of the *RTE-M Programmer's Reference Manual* for the command formats and parameter options. Either command is entered BEFORE the REL command for the user program is entered, if you wish to change the starting address of the module to other than the end of the last module plus 1.

For MIII systems where the user program will reside in partitioned memory, the extra space is the remaining space in the program partition after the program is loaded (see the DEFINE PARTITIONS session in Section 3-6 of this manual).

The extra memory space requirements or recommendations for HP-supplied user programs are as follows:

User Program	**Additional Space**
RTE-M Editor	1K words
File Manager	1K words see - paragraph below
Relocating Loader	2K words (varies)
RTE-M Generator	3K words (varies)
Real-Time FORTRAN	1K words (varies-eight words per symbol)
RTE-M Assembler	varies - see paragraph below

The Editor uses available memory for I/O buffering. I/O transefer time can be reduced by increasing the available memory.

Although the File Manager (flexible disc version only) does not actually require added space behind itself, it is recommended for flexible disc PACK operations because it decreases the amount of time required by FMGR to fulfill PK requests.

The additional amount of available memory required for the Assembler symbol table varies, according to the number of characters per symbol. For instance, a symbol one character long requires two words for a symbol table entry; if the symbol is two or three characters long, it needs three words; if the symbol is four or five characters long, four words are required for an entry in the symbol table.

Enter the LINKS IN and MAP options if not entered previously or if parameter changes are desired. Once invoked, the commands remain in effect for all subsequent relocations unless reentered to change their options for a given program relocation.

Enter the desired user program file name, followed by other RELOCATE commands, as required, giving the file names for any included subroutines called by the main program.

Enter any required SEARCH commands to search the named libraries for external references in the user module. It is good practice to follow library searches with a DISPLAY UNDEFS command to see whether all external references have been satisfied, or if one or more other libraries must also be searched.

> NOTE
>
> It is occasionally necessary to re-SEARCH the same library several times to satisfy all external references. This is because some of the referenced subroutines themselves may reference other subroutines in the library.

Some of the considerations in relocating HP-supplied user programs, generation requirements, and preferred loading sequence are given in Table 3-2 below.

Table 3-2. User Program Relocation Considerations

D. R. — Directory Management Program

REL { %DRC } (system w/scheduling) Tape Cart. file name of main program
{ %DRF } (system w/sched.) Flexible Disc file name of main program

SEARCH { %FMPF } (Flexible Disc FMP Library) Required only for %DRF
{ %FMPC } (Tape Catr. FMP Library) Required only for %DRC

SEARCH %MSYLB (RTE-M System Library)

Table 3-2. User Program Relocation Considerations (Continued)

SEARCH %RLIB1 (Floating Point Library, module 1)

SEARCH %RLIB2 (Floating Point Library, module 2)

Note: %DRC or %DRF is required for all systems using the File Management Package (FMP) that do NOT use %DRC1 or %DRF1.

FMGR — RTE-M File Manager Program

REL { %FMGC0 (Tape Cartridge) / %FMGF0 (Flexible Disc) } FMGR main program file name

SEARCH { %FMPC (Tape Cartridge) / %FMPF (Flexible Disc) } FMP program file name

SEARCH %MSYLB (RTE-M System Library)
SEARCH %RLIB1 (Floating Point Library, module 1)
SEARCH %RLIB2 (Floating Point Library, module 2)

AUTOR — Auto-Restart Program

REL %MAUTO (file name for AUTOR program)

Note: If %MAUTO is loaded, either %MPF (MI, MII systems) or %MPF3 (MIII systems) must have been relocated with the I/O drivers during the system module relocation phase.

PRMPT — Multi-Terminal Prompt Program

REL %MPRMP (file name for PRMPT program)
SEARCH %MSYLB (RTE-M System Library)
SEARCH %RLIB1 (Floating Point Library, module 1)
SEARCH %RLIB2 (Floating Point Library, module 2)

Note: Use of this module also requires relocation of %NMTM and %MRSPN.

ONMTM — Multi-Terminal Enable Program

REL %ONMTM (File name for ONMTM program)

Note: If more than two HP 2644/5A terminals are to be present on the system, the $TBLCR library routine must be modified through reassembly of the supplied source code. See Appendix G for details. Use of this module also requires relocation of %MPRMP and %MRSPN.

Table 3-2. User Program Relocation Considerations (Continued)

MRSPN — Multi-Terminal Response Program

```
REL %MRSPN          (file name for Response program)
SEARCH %MSYLB       (RTE-M System Library)
SEARCH %RLIB1       (Floating Point Library, module 1)
SEARCH %RLIB2       (Floating Point Library, module 2)
```

Note: Use of this module requires relocation of %ONMTM and %PRMP

Other desired user modules

:
:
:

APDLR — Absolute Program Loader

```
REL { %MAP  } (MI, MII)
    { %MAP3 } (MIII)        File name for APLDR program

          %FMPC  (Tape Cartridge)
SEARCH                              FMP Subroutine Library
          %FMPF  (Flexible Disc)

SEARCH %MSYLB       (RTE-M System Library)
SEARCH %RLIB1       (Floating Point Library, module 1)
SEARCH %RLIB2       (Floating Point Library, module 2)
```

RTMGN — RTE-M System Generator Program

```
REL %RTMSC          (Loader/Generator Subcontrol module)
REL %RTMGN          (main RTMGN program module)
```

Note: Base page links will be at their absolute minimum if %RTMSC is relocated first and if %RTMSC and %RTMGN are both started on a page boundary.

```
          { %FMPC } (Tape Cart. FMP Subroutine Library)
SEARCH    {       }
          { %FMPF } (Flexible Disc FMP Subroutine Library)

SEARCH %MSYLB       (RTE-M System Library)
SEARCH %RLIB1       (Floating Point Library, module 1)
SEARCH %RLIB2       (Floating Point Library, module 2)
```

Table 3-2. User Program Relocation Considerations (Continued)

RTMLD — RTE-M Relocating Loader Program

REL %RTMSC	(Loader/Generator Subcontrol module)
REL %RTMLD	(main Relocating Loader program module)

Note: base page links will be at their absolute minimum if %RTMSC is relocated first and if %RTMSC and %RTMLD are both started on a page boundary.

SEARCH	%FMPC	(Tape Cart. FMP Subroutine Library)
	%FMPF	(Flexible Disc FMP Subroutine Library)
SEARCH %MSYLB		(RTE-M System Library)
SEARCH %RLIB1		(Floating Point Library, module 1)
SEARCH %RLIB2		(Floating Point Library, module 2)

Note: If any loader or the Generator itself are to be relocated as user programs in the RTE-M resident system, they should be the LAST modules relocated, since they require space behind themselves.

At the termination of each user program relocation, RTMGN will output a hyphen prompt. Enter an END command. RTMGN will then output

STARTING ADDRESS xxxx
NO UNDEFS
or
UNDEFS xxxxxx

If the NO UNDEFS message appears, RTMGN will immediately follow it with an ENTER PRAMS message . If the UNDEFS xxxxx message appears, you have two choices for your response. If the displayed unsatisfied external reference is not needed, simply perform a carriage return on the keyboard and RTMGN will output the ENTER PRAMS prompt. If the function is desired, you will have to enter another SEARCH command to search the appropriate library (check the RTE-M Software Numbering Catalog for the correct library name). RTMGN will then issue the NO UNDEFS prompt if the search was successful .

ENTER PRAMS

The prompt is output at the termination of each module relocated. Enter a zero (0) if no changes are to be made in the program's parameters and the default case is to be used. Enter desired changes in the worksheet made up for the corresponding program. The format for the parameters is:

program name[,priority],[execution interval]

where:

program name
is the name of the specified program

priority
is the program's priority in the range 1 through 32767, with 1 as the highest priority

execution interval
is a list of six parameters specifying the times the program should be scheduled for execution once it is turned on. The first two values specify the execution interval, and the last four specify an initial absolute starting time (see also the IT command in Section II of the *RTE-M Programmer's Reference Manual*). The six parameters are:

[,res[,mult[,hour,min,sec,10msec]]]

where:

res
resolution code (0 to 4)

0 = no execution interval
1 = tens of milliseconds
2 = seconds
3 = minutes
4 = hours

mult
execution multiple that is a number from 0 to 4095, and is used with the resolution code to give the actual time interval for scheduling (see paragraph below).

Initial absolute starting time;four values:

hour
hours (0 to 23)

min
minutes (0 to 59)

sec
seconds (0 to 59)

10msec
tens of milliseconds (0 to 99).

The resolution code (res) is the units in time to be multiplied by the multiple execution interval value (mult) to get the total time interval. For example, if res= 2 and mult= 100, program name would be scheduled every 100 seconds. If hour, min,sec and 10msec are present, the first execution occurs at the initial start time specified by these parameters. If the parameters are not present, the program's time values are set to zero and the program is removed from the time list. However, the program can still be called by another program or started with the ON name or RU name command when the system is running.

If you specify time scheduling parameters for any of your user programs, this might be a good time to reexamine the number and type of any privileged interrupts entered in your I/O Configuration Worksheet. This is to make certain there will be no time offset from your program parameter specifications caused by the privileged interrupts. Should you decide to change any privileged interrupt entries on the I/O Worksheet, BE CAREFUL to carry over the changes into subsequent worksheets where required.

Alternately, you may decide to modify the program time interval parameters if there is an apparent or potential conflict. The interaction of any privileged drivers and one or more time scheduled programs deserves your close attention during generation planning.

Enter an END command after each set of program parameter entries, if any. RTMGN will prompt for another REL USER PROGS until you respond with two END commands. RTMGN will then go into the next generation phase.

3-7. MEMORY BOUNDS AND PARTITION PLANNING

During this phase, you decide whether you wish to change the size of the memory resident program area and (for MIII only) define the program partitions. Enter the data in the Memory Bounds and Partition Worksheet given in Appendix D. An example of this worksheet with typical entries is illustrated in Figure 3-8.

LWA MEM RES PROG = xxxxx CHANGE?

Enter a zero (0) if no change is wanted in the size of the resident program area. Enter a new address in octal to set the last word of the program area if a new size is desired. Note that if 0 is specified, there will be no room for on-line additions in the memory resident area.

RTE-M MEMORY BOUNDS AND PARTITION WORKSHEET

BOLDFACE terms are output to user by RTMGN.

Question Marks are RTMGN prompts; all other data (except comments) is user input. Fill in blanks with desired parameters.

LWA MEM RES PROG = xxxxx CHANGE?

_Ø_ (enter desired new address in OCTAL; enter 0 for no change)

ALIGN AT NEXT PAGE?

_NO_ (enter YES or NO; YES causes SAM to start on next page)

SAM = xxxxx WORDS (current size of SAM)

NO. ADD. PAGES FOR SAM? MAX = xxx (MIII only. SAM extends to end of memory for MI and MII)

_1_ (enter desired no. added pages or 0)

LARGEST ADDRESSABLE PARTITION
W/O COMM xx PAGES
W COMM xx Pages
} (output for MIII only)

PAGES REMAINING = xxxxx
DEFINE PARTITIONS

?
_1_ , _4_ (enter partition no. and decimal no. pages for partition)
?
_2_ , _4_
PAGES REMAINING = xxxxx
?
_3_ , _5_
PAGES REMAINING = xxxxx
?
_4_ , _5_
PAGES REMAINING = xxxxx
?
_5_ , _12_
PAGES REMAINING = xxxxx
?
END , _____

PAGES REMAINING = xxxxx
?
_____ , _____
PAGES REMAINING = xxxxx
?
_____ , _____
PAGES REMAINING = xxxxx
?
_____ , _____
PAGES REMAINING = xxxxx
?
_____ , _____
PAGES REMAINING = 0
END (required)

NOTE: Partitions do not have to be defined in sequence.

Figure 3-8. MIII Memory Bounds and Partitions Example

ALIGN AT NEXT PAGE?

After RTMGN outputs the last word address of the user program area (see previous prompt), it asks whether you wish to align the end of the area to the next page boundary to protect system available memory. Note that those words skipped will be lost. Enter YES to align the memory - resident program area to the next page boundary. If YES is entered, RTMGN automatically allocates the rest of that page or a new page of memory to the System Available Memory area (SAM).

If NO is entered, RTMGN allocates the area between the LWA of the resident program area and the next page boundary to SAM. The size of SAM could be from 1 to 1024 words.

SAM = xxxxx WORDS

The current size of system available memory is output in decimal number of words. No response is required.

NO. ADD. PAGES FOR SAM? MAX=xxxxx

The question is asked only for MIII systems, since SAM will extend to the end of memory for MI and MII systems. Enter the decimal number of extra pages desired if you wish to increase the size of system available memory. An updated SAM = xxxxx WORDS message will then be issued. Enter a zero (0) if no change is desired.

LARGEST ADDRESSABLE PARTITION
W/O COMMON xx PAGES
W/ COMMON xx PAGES
REMAINING = xx DEFINE PARTITIONS
?

The question is asked only for MIII systems. After each entry, RTMGN will prompt with a new question mark, asking for the next entry. You terminate the partition definition dialog by entering an END after the question mark that follows your final entry. The area available given in decimal number of pages in the "PAGES REMAINING = xx" message must be subdivided into the number of partitions you specified on page 1 of 2 of the Program Partition/Program Parameter Worksheets (see Section 3-5 and Figure 3-6).

NOTE

All user programs to reside in partitioned memory must be relocated and loaded on-line.

The total sum of the partition sizes must equal the number of remaining pages output by RTMGN above. After each definition, RTMGN will output a new PAGES REMAINING message, decremented by the number pages specified in the previous entry. Enter each

definition using the following format:

partition number, size

where:

partition number
is a number between 1 and the maximum number you specified previously (up to 64). This number is the partition "name."

size
is the partition size in decimal number of pages. A partition must include enough pages for a program, plus one page for the program's base page. See also the additional buffer or symbol table memory requirements for some HP-supplied user programs as described in Section 3-8.

The order in which partition definition is entered is optional. Partition numbers may be skipped if desired. However, pages will be assigned in order by partition number (lower numbered partitions get lower numbered pages).

The last PAGES REMAINING message should ideally equal zero. Enter an END command after the following question mark to terminate the dialog.

3-8. SYSTEM SNAPSHOT CONSIDERATIONS

Creation of the system snapshot is the last phase of generation. The snapshot is an ASCII file or tape of the system entry points of your system, and the bounds of its modules. The snapshot is required input for any subsequent on-line program relocation and loading. A sample System Snapshot Worksheet is illustrated in Figure 3-9.

```
SNAPSHOT?
-
```

Enter an END if no snapshot is desired. If the snapshot is required for on-line loading, make your entry in the following format:

```
            file name
SNAP ON
            LU#
```

where:

file name
is the name of a file on a flexible disc or tape cartridge

LU#
is the logical unit number of some device.

Enter an END following your snapshot entry. For all later on-line loading, you perform a transfer to the snapshot after the RTMLD command is entered and before the RELOCATE command is encountered.

SYSTEM STARTS AT TRACKxxxxx,SECTORxxxxx

The message is only issued if you defined your output device as a disc file at the beginning of generation. The message will be displayed on the system console screen and echoed on the echo file if echo was specified.

> NOTE
>
> If echo was NOT specified to a hard-copy device, make certain you copy the displayed track and sector address before clearing the screen. The octal equivalent of the track and sector address must be set in B register to boot in your generated system. See Section 4-7 for the procedure.

RTMGN FINISHED

This completes your site-specific RTMGN generation planning. You are now ready to load your minimum absolute MI system into main memory and configure your own RTE-M system, using the data entered on your generation worksheets. The actual generation and loading procedures are described in the next section.

RTE-M SNAPSHOT WORKSHEET

BOLDFACE terms are output to user by RTMGN.

– Hyphens are RTMGN prompts; all other data (except comments) is user input. Fill in blanks with desired parameters.

SNAPSHOT? (used for on-line loading of user programs)

– SNAP ON &SNAP1 (enter file name or LU number, or enter 0 if no snapshot wanted.)

END (required)

SYSTEM STARTS AT TRACK xxxxx SECTOR xxxxx

(Message only appears when generated output was assigned to a flexible disc. Save a copy of the track and sector address!)

RTMGN FINISHED

Generation is completed, and absolute user-defined RTE-M operating system is on file or LU specified on Initialization Worksheet.

Figure 3-9. MIII Snapshot Example

GENERATION PROCEDURES

SECTION IV

Once your planned RTE-M system is completely defined on the Worksheets, the actual generation process is largely mechanical. This section of the manual contains the following information:

- Generation procedures using a flexible disc system. Loading the HP-supplied generation system into main memory and executing RTMGN to begin the generation dialog.
- Using the echo file as an answer file for following generations (flexible disc systems only). Editing an echo file.
- Procedures for booting in your generated system from a flexible disc.
- Generation procedures for tape cartridge systems and tape handling considerations. Booting in your generated system from tape cartridge.
- Generation error codes, their meaning and possible operator action.

4-1. GENERATING WITH FLEXIBLE DISC

The procedures described below assume the presence of a flexible disc drive, a 2644/5 Display Station terminal with tape cartridges. A line printer is a highly recommended option for a list device.

For flexible disc subsystems, all HP-supplied software is delivered on two disc platters:

1. The "generator" disc platter containing the RTE-M Off-Line Generator program, plus the following relocatable modules: Absolute Program Loader, Relocatable Loader, system modules (MI, MII or MIII), drivers and library modules necessary for generating a system.

2. The "program preparation" disc contains the HP-supplied user programs for on-line program development; e.g., Multi-Terminal Monitor, RTE-M Editor (EDITM), File Manager, Relocating Loader, language processors,etc.

Note that the File Manager and Relocating Loader are delivered on both the "generation" and "program prep" disc platters. This is done because the memory resident APLDR and Relocating Loader both use the File Manager library internally for file manipulation. However, "user" programs that are relocated during generation cannot be saved. Only the Relocating Loader can reconstitiute them. It is therefore always desirable to have absolute versions of the Relocating Loader and File Manager stored out on some mass storage media. Doing so ensures that you will always have the required software tools for building programs.

4-2. GENERATION DISC FILE SPACE

The generation disc platter has 300 blocks (10 tracks) of available bulk storage if you wish to output your generated absolute operating system to this disc. This is adequate file space for most relocated user RTE-M systems. However, there are several constraints that should be considered before utilizing this space for generated output:

1. Inexperienced users could fill up the 300 tracks with a bad generation before a File Manager was available to purge the tracks. It is therefore recommended that such users assign their generated output to a tape cartridge.

2. If user-written functions or programs have been added to the system, and thus increased its size beyond the 300 block file limitation, some other device must be used for the output file.

4-3. BOOTING OFF-LINE GENERATOR PROGRAM FROM DISC

> NOTE
>
> On systems with multiple disc drives, the Generation disc platter must be located on drive 0 when booting in.

1. Insert the generation disc platter into the drive. Make sure the Driver Select screw at the rear of the disc drive is turned to 0. Make sure the terminal is in REMOTE mode and the CAPS LOCK key is in the down position. Press the RESET key twice to clear terminal memory.

2. At the computer control panel, make sure the panel is in OPERATE mode, and press the Register Select switch until the indicator light is over the S register label.

3. Press the CLEAR DISPLAY switch.

4. Enter the lower select code of the flexible disc interface card in bits 6-11. Enter the Flexible Disc ROM Loader address in bits 14 and 15. Set bit 0 to 1.

5. Press the STORE switch; then press the IBL switch.

6. Press the PRESET switch; then press RUN.

If you have followed the procedures correctly, the Generator program will be read into main memory from the flexible disc. A halt of 102070 should be displayed in the T register.

> NOTE
>
> A halt of 102011 displayed in the DISPLAY REGISTER indicates a checksum error. Begin the boot-in procedure over again.

7. Again display the S register by pressing the Register Select switch; then press the CLEAR DISPLAY switch.

8. In bits 0-5, enter your system console select code; then press the STORE switch.

9. Display the B register; again press the CLEAR DISPLAY switch. In bits 0-5, enter the lower select code of the flexible disc interface card.

10. Press STORE; press PRESET; then press RUN.

You are now ready to begin the generation dialog with RTMGN at the system console. At the console, press any key on the keyboard to get the system's attention and type

RU,RTMGN

The Generator will respond with the prompt:

RTMGN
DEFINE OUTPUT DEVICES
–

Start the interactive dialog with RTMGN, using the completed Worksheets for your responses. Terminate each of your entries by pressing the RETURN key.

4-4. CREATING AN ANSWER FILE

During your DEFINE OUTPUT DEVICES response, you can make use of the ECHO command to create an input command file (answer file) if desired. Although your initial generation should be performed interactively at the console, following generations can be performed by directing RTMGN to the answer file for its input. Generation is then automatic and free of potential operator error during entry.

During the initial generation, an answer file can only be created and used on configurations with both a disc and tape cartridge subsystem. Creating an answer file is not a practical option for tape cartridge-only systems, because not enough "machine readable" output devices are normally available in such configurations to accomodate the file.

The echo file is simply a copy of the RTMGN prompts and commands you type on the console keyboard during generation. The RTMGN prompts are automatically given a preceding asterisk (*) to define what follows as a non-executable comment. This prevents RTMGN from interpreting its own prompts as commands when generating from the answer (command input) file. If you intend using the file as a command input file for later generations, you should enter the

ECHO ON logcal unit number (tape cartridge)

command as your first response to the DEFINE OUTPUT DEVICES prompt. For instance, if you entered the sequence

OUTPUT ON { logical unit number / file name }

ECHO ON logical unit number

MAP MODULES,GLOBALS,ON logical unit number (line printer)

END

the required OUTPUT ON definition would not be written at the beginning of the command input file, since the echo file has not yet been created.

Conversely, if the above sequence was reversed, the ECHO ON command would be missing from the command stream. One solution to this difficulty is to insert any missing commands in the echo file by editing the tape cartridge off-line after generation, using the built-in editor in the 2644/5 Display Station terminals. See the appropriate Display Station User's Manual for instructions. This same off-line editing technique can be used to correct or change other commands in the input command stream if desired.

> NOTE
>
> Although the optional MAP command can be assigned to the device or file as the echo file, the resultant mixed output prevents its use as a command input file for subsequent generations.

Once your site-specific RTE-M system has been generated, you can assign the echo device to a flexible disc file, provided the File Manager is present on the system. Any necessary editing could then be performed on-line if the RTE-M Editor is also available.

To perform another generation using the echo file for the command input stream (and your site-specific system with relocated Generator program), run the Generator through the command

RU RTMGN { ,fi,le,nm / ,lu }

followed by the TR command to direct the Generator to the mounted answer file for its input. See Section VI of the *RTE-M Programmer's Reference Manual* for the format and description of the TR command.

4-5. GENERATION COMPLETION

Assuming that at the beginning of generation you defined the output device as a flexible disc file, completion of generation will be indicated with the message

SYSTEM STARTS AT TRACK xxxxx,SECTOR xxxxx
RTMGN FINISHED

displayed on the system console screen and output on the echo file if echo was specified. If an echo file was NOT specified, copy the track and sector addresses from the display screen and save them.

The track and sector message will not appear if you assigned your generated output to some other device. Although generation is now completed, the Generation Program is still running. You must therefore press the HALT switch on the computer control panel to stop execution before you can boot in your generated system.

If your generated system was output on tape cartridge, use the boot-in procedures described in Section 4-9.

4-6. BOOTING YOUR GENERATED SYSTEM FROM DISC

> NOTE
>
> When booting, the flexible disc from which the input will be read must be mounted on flexible disc drive 0.

To boot in your generated system, perform the following steps:

1. At the computer control panel, press the HALT switch if you have not already done so. Display the S register by pressing the Register Select switch until the indicator light is over the S register.

2. Press the CLEAR DISPLAY switch; then enter the lower select code of the flexible disc interface card into bit 6-11. Enter the Flexible Disc ROM Loader address in bits 14 and 15. Clear bit 0. Press the STORE switch.

3. Display the B register and press the CLEAR DISPLAY switch. Get the track and sector address from the SYSTEM STARTS AT TRACKxxxxx,SECTOR xxxxx message on your listed output. Refer to Appendix F, Flexible Disc Boot Conversion Table, and convert your track address to the value given in the "B-REG Octal Equivalent" column of the table. Locate your sector address in the "Logical Sector Address" column, and convert it to the value in the "B-Reg Physical Sector Addr. Octal Equivalent" column. Add the two values and enter the sum in the B register.

4. Press the STORE switch; then press the IBL switch.

5. Press the PRESET switch; then press RUN.

If these procedures have been followed correctly, your generated system should now be loaded in main memory and automatically started.

4-7. BOOTING THE OFF-LINE GENERATOR PROGRAM FROM TAPE CARTRIDGE

The procedures given below assume the presence of an HP 2644/5 Display Station terminal on the system, with a line printer as a highly recommended option. For RTE-M systems employing a tape cartridge system, all HP-supplied software is delivered on a set of cartridge tapes that are labeled with a Part Number and contents.

> NOTE
>
> When replacing one tape cartridge with another, you must enter an
>
> RC,{R / L} command and carraige return
>
> before continuing, where R or L specify right or left tape channels respectively. If the Generator was waiting, you must also enter a GO,RTMGN command.

Each tape has a File Directory as its first file.

1. Put the terminal in LOCAL mode by making sure the REMOTE key is not depressed, and press the CAPS LOCK key down.

2. Press the RESET key twice to clear terminal memory.

3. Locate the tape cartridge labeled MI SYSTEM AND GENERATOR and insert it in the left-hand tape channel.

4. Press the GOLD key, f1 function key (FROM:L.TAPE), f7 function key (TO:DISPLAY) and then the READ key. Choice of the left tape channel is purely arbitrary. If the right channel is used, press f2 instead of f1.

The Directory file at the beginning of tape will be displayed on the terminal screen. This step is necessary to get past the first file (Directory file), which cannot be read by the 12992C ROM Loader during boot-up. The tape is now positioned at the beginning-of-file for the absolute MI system and Generator program. Although the first file could be skipped in REMOTE mode, the off-line procedure is recommended for display to ensure that the correct tape is mounted.

5. Press the REMOTE key down to put the terminal in REMOTE mode.

6. At the computer control panel, display the S register by pressing the Register Select switch until the indicator light is over the S register label.

7. Press the CLEAR DISPLAY switch.

8. Enter the select code of the Tape Cartridge ROM Loader in bits 14 and 15. Enter the select code of the system console in bits 6-11.

9. Press the STORE switch; then press the IBL switch.

10. Press the PRESET switch; then press the RUN switch.

If you have followed the procedures correctly, the Off-Line Generator program will be read into main memory from the tape cartridge. The loading process will be indicated by the blinking of the green indicator light on the tape channel.

When loading is completed, a halt of 102077 will be displayed in the T register. The tape channel indicator light will emit a steady glow.

11. Again display the S register by pressing the Register Display switch.

12. In bits 0-5, enter your system console select code. Press the STORE switch.

13. Display the P register by pressing the Register Select switch until the indicator light is over the P register label.

14. Press the CLEAR DISPLAY switch; then set address 2 (octal) in the DISPLAY register.

15. Press the STORE switch; then the PRESET switch; then RUN.

A halt of 102070 should be displayed in the T register. The S register should contain the select codes of the system console and tape channels when displayed, and all other bits should be zero.

16. Press RUN again.

4-8. EXECUTING RTMGN IN TAPE CARTRIDGE SYSTEMS

To begin execution of generation, press any key on the system console to get the system's attention and enter the command

RU,RTMGN

The Generator will respond with the prompt

RTMGN
DEFINE OUTPUT DEVICES
–

You are ready to begin the generation dialog with RTMGN, using the completed Worksheets for your responses. The Generator cartridge may now be removed and other tapes containing the relocatable modules inserted in the sequence appropriate for the Generator's prompts.

At the conclusion of generation, the Generator will output the message

RTMGN FINISHED

Although generation is completed, the Off-Line Generator is still running in memory. You must press the HALT switch on the computer control panel before booting in your generated system.

4-9. BOOTING YOUR GENERATED SYSTEM FROM TAPE

To load your site-specific RTE-M system into main memory, perform the following steps:

1. Make sure the tape cartridge containing your generated system is write-protected and insert it in the left-hand tape channel. If there is a Directory at the beginning of tape, it must be skipped off-line (see Section 4-8, steps 1-5).

2. RESET the terminal, make sure the terminal is in REMOTE mode, and that the CAPS LOCK key is in the down position.

3. At the computer control panel, display the S register by pressing the Register Select switch until the indicator light is over the S register label. Press the CLEAR DISPLAY switch.

4. Enter the system console select code in bits 6-11. Enter the Tape Cartridge ROM Loader address in bits 14 and 15.

5. Press the STORE switch; then press the IBL switch.

6. Press the PRESET switch; then press RUN.

If you have followed these procedures correctly, your generated system will be read into main memory. The loading process will be indicated by the blinking of the green indicator light on the left tape channel.

When the loading is completed, a halt of 102077 will be displayed in the T register. The tape channel indicator light will also emit a steady glow.

To execute your generated system, perform the following steps:

1. Display the P register, the press the CLEAR DISPLAY switch.

2. Set address 2 (octal) in the DISPLAY register.

3. Press the STORE switch; press the PRESET switch; then press RUN.

Your generated RTE-M system is now executing and is ready to accept your commands.

4-10. GENERATION ERROR CODES

During generation, RTMGN will reject any of your commands that cannot be interpreted or executed. The Generator will output an error code and come to a halt to permit possible recovery procedures. Error codes output during the Initialization and Table Generation phases of generation are given in Table 4-1. Refer to Section 9-3 of the RTE-M Programmer's Reference Manual for the meaning of error messages output during the Relocation and Snapshot phases.

Table 4-1. Initialization and Table Generation Error Messages

MESSAGE	MEANING	OPERATOR ACTION
ERR-AD	Undefined externals from SYS MOD relocation or Invalid entry in INT Table definition	Start RTMGN over again Enter valid INT entry
ERR-CH	Invalid channel no. in EQT or in INT (>77B)	Enter correct channel no.
ERR-DR	Invalid driver name: 1. Not DV xxx 2. Driver not relocated as SYS. MOD.	Enter correct EQT entry
ERR-DU	Duplicate program name (from CHANGE PRAMS response)	Repeat CHANGE PRAMS response
ERR-EQ	Invalid EQT no. in INT Table: 1. >77B 2. =0 3. >EQT entry	Enter valid EQT response in INT Table
ERR-IN	Execution interval error (in CHANGE PRAMS response)	Enter correct CHANGE PRAMS
ERR-LU	Invalid Device Ref. No: 1. $>99_{10}$ for DRN 2. $>99_{10}$ for subchannel 3. No corresponding EQT entry 4. No DRT entries	Enter correct DRT entry
ERR-ON	Invalid ON, RTMGN parameters	Start RTMGN over again
ERR-NA	Invalid interrupt mnemonic—only following allowed: EQT PRG ENT ABS	Enter correct INT entry
ERR-PA	Invalid parameter. or Parameter name error in CHANGE PRAMS; first char. must be $\leq 132_8$ and $\geq 4_8$. Octal 47 to octal 55 not allowed	Input correct parameter
ERR-PD	Partition already defined	Enter next partition definition
ERR-PR	Parameter priority error >5-digit decimal number in CHANGE PRAMS. Priority	Enter correct CHANGE PRAMS entry
ERR-PS	Partition size error. Partition bigger than remaining memory	Enter next partition definition (must be less than previous one).
ERR-PT	Partition definition error. Partition greater than maximum allowed	Enter next partition definition

Table 4-1. Initialization and Table Generation Error Messages

MESSAGE	MEANING	OPERATOR ACTION
ERR-SO	System overflow loader symbol table and short fixup table overflow (used for EQT extensions)	Redefine all EQT entries
ERR-TB	Symbol table/ID segment overflow; or Entered programs exceed ID segs; or No ID segs. left for start-up program	Start RTMGN over again

INCLUDING BASIC SOFTWARE

SECTION V

RTE-M system planning and layout, generation planning, and generation procedures are described in Section II, III, and IV of this manual. You should be thoroughly familiar with the information contained in those sections before you attempt to generate your site-specific RTE-M system. This section of the RTE-M Generation Manual contains information you will need for including the RTE-M BASIC subsystem modules.

5-1. RTE-M SYSTEM REQUIREMENTS

The minimum hardware required for RTE-M support of HP Real-Time BASIC is shown in Table 5-1.

Table 5-1. RTE-M BASIC Subsystem Minimum Hardware Requirements

SYSTEM LEVEL	BASIC REQUIREMENTS (EXECUTE AND PROGRAM DEVELOPMENT)
RTE-MI	HP 21MX-series Computer 24K Words of Memory (BASIC occupies approximately 11K) Floating Point Arithmetic HP 2644/2645 Terminal with Cartridge Tape Units
RTE-MII	HP 21MX-series Computer 24K Words of Memory (BASIC occupies approximately 11K) Floating Point Arithmetic HP 2644/2645 Terminal with Cartridge Tape Units Memory Protect
RTE-MIII	HP 21MX-series Computer 32K Words of Memory (BASIC occupies approximately 11K) Floating Point Arithmetic HP 2644/2645 Terminal with Cartridge Tape Units Memory Protect Dynamic Memory Mapping

The minimum software required for RTE-M support of HP Real-Time BASIC is:

- RTE-MI, RTE-MII, or RTE-MIII Operating System
- BASIC Interpreter
- BASIC Library
- BASIC Task Scheduling Subroutines (SCHED, TTYEV, TTYS, TRAP)
- Standard Branch and Mnemonic Tables

Refer to the HP 92065A BASIC Subsystem Software Numbering Catalog for the appropriate software part numbers and module names.

5-2. LOADING BASIC SOFTWARE

You may configure the Real-Time BASIC software subsystem into your RTE-M Operating System using the system generation procedures described in the preceding sections of this manual. Information for including the BASIC subsystem within your RTE-M System at generation are given in this section.

Alternatively, you may add the BASIC subsystem to an existing RTE-M Operating System using the Absolute Program Loader (APLDR). Procedures for loading the BASIC software on-line are given in this section.

5-3. RTE-M BASIC SUBSYSTEM CONSIDERATIONS

For maximum utilization of the RTE-M system support of the BASIC subsystem, you should consider the following requirements and suggestions:

1. The BASIC subsystem requires a user-generated Branch and Mnemonic Table (see paragraph 5-11) to satisfy external subroutine references upon relocation of the BASIC subsystem. In addition, a standard Branch and Mnemonic Table which contains all system defined functions (see the HP 92065A RTE-M BASIC Language Reference Manual) is supplied by Hewlett-Packard.

2. If instrumentation subroutines (for example, HP 2313 and/or HP 6940 subsystems) are to be called, the BASIC subsystem requires an Instrument Table (see the RTE-M BASIC Language Reference Manual, Section XIII).

3. If TRAP and task scheduling statements are to be used (see the RTE-M BASIC Language Reference Manual, Section XI), the BASIC Memory Library module %BAMLB (TRAP) must be relocated during system generation to satisfy external subroutine references.

   If TRAP and task scheduling statements are not to be used, the BASIC subsystem requires the Dummy Trap module (%DTRAP) to satisfy external subroutine references.

   Within the Multiple Terminal Monitor environment (see paragraph 5-10) only one copy of BASIC may utilize TRAP and task scheduling at a time.

4. If the support of an auxiliary terminal is planned, the BASIC subsystem requires that you include the RTE-M system software module TTYS.

5. To replace any JSB .MBT instructions (move byte) with the microcode move byte instruction (105765), enter the following entry point change in response to the "CHANGE ENTS?" Generator prompt:

   .MBT,RP,105765

5-4. INCLUDING BASIC SOFTWARE IN A NEW SYSTEM

RTE-M system modules required in your RTE-M Operating System depend upon which level of RTE-M System you plan to install — RTE-MI, RTE-MII, or RTE-MIII. The appropriate module names and part numbers are given in the HP 92064A and HP 92065A Software Numbering Catalogs.

5-5. STEPS REQUIRED TO GENERATE A BASIC SYSTEM

1. Plan your system and fill out the RTE-M I/O Configuration Worksheet.

2. Plan your system generation and fill out the RTE-M Generation Worksheets. Refer to paragraph 5-3 in this section for BASIC subsystem considerations in filling out the worksheets.

3. Boot up the Off-Line Generation program.

4. Execute the Generator (RTMGN) and begin your system generation using the responses from your Generation Worksheets.

5. Once the generation is complete, boot up your system as described in Section IV, paragraph 4-10.

6. Use the procedures described in paragraph 5-8 to relocate the BASIC subsystem as an absolute program file.

5-6. GENERATING AN RTE-M BASIC SUBSYSTEM FROM TAPE CARTRIDGE

1. Use the procedures described in Sections II and III to plan your system and to plan your generation. The HP 92064A and HP 92065A Software Numbering Catalogs contain the Tape Cartridge part numbers and the names of the RTE-M system and BASIC subsystem modules you will need to generate your specific system.

2. Use the procedures described in Section IV, paragraph 4-8 to boot up an RTE-MI system from Tape Cartridge and execute the system Generator (RTMGN).

3. Following your entry of the RU,RTMGN command, the Generator will display the introductory heading followed by the first query and a hyphen prompt:

   RTMGN
   DEFINE OUTPUT DEVICES
   –

4. Respond to the Generator queries and prompts with the information from your Generation Worksheets.

5. Remember that when you are generating from tape cartridges, you must use the RTE-M operator command RC (Replace Cartridge) each time that you remove a cartridge from the input tape unit and replace it with another cartridge.

5-7. INCLUDING BASIC SOFTWARE IN AN EXISTING SYSTEM

You will be using the RTE-M Relocating Loader (RTMLD) and the system snapshot to create an absolute program module from tape cartridge files which, subsequently, you will load into memory using the RTE-M Absolute Program Loader (APLDR).

5-8. RELOCATING THE BASIC SUBSYSTEM. Once your RTE-M system is operational with all required modules included, you may relocate the BASIC subsystem using the Relocating Loader, RTMLD. This loader produces an absolute program file which you may cause to be written to a tape cartridge via the loader command, OUTPUT ON. A sample RTMLD session follows:

*ON,RTMLD	Execute RTMLD.
LOADER STARTED	Loader message.
–	Loader prompt.
TR,SNPFL	Transfer to system snapshot file.
–	
MAP MODULES,GLOBALS ON 6	Map output to LU 6.
–	
LINKS IN CURRENT	Assign links to current page.
–	
OUTPUT ON 4	Write absolute program file to LU 4.
–	
RELOCATE %MBASC	Relocate BASIC Interpreter module.
–	
RELOCATE %MESGA	Relocate BASIC error message module.
–	
RELOCATE %DTRAP	Relocate BASIC dummy TRAP module.
–	
RELOCATE UBMTBL	Relocate user-generated Branch and Mnemonic Table.
–	
SEARCH %FMPC	Search File Manager cartridge library.*
–	
SEARCH %MSYLB	Search RTE-M system library.
–	
SEARCH %RLIB1	Search RTE-M relocatable library, part 1.
–	
SEARCH %RLIB2	Search RTE-M relocatable library, part 2.
–	
END	End relocation.
RELOCATION FINISHED	Loader message.
11 PAGES REQUIRED	Loader message.
SNAPSHOT?	Loader query.
–	Loader prompt.
END	End session.
LOADER FINISHED	Loader message

*Possible other searches for req. subrs. in Branch and Mnemonic Table: %HPIB, %FF4.N, %ISKSC, %BASLM. (See Figure 5-1.)

In this case, when RTMLD is finished, you will have an absolute program file on a tape cartridge (LU 4). Now you may use the RTE-M Absolute Program Loader (APLDR) to load the program file into memory. A sample sequence of commands to do this follows:

< rewind LU 4 (cartridge tape unit)>

*LO,4	Load program from LU 4.
APLDR: DONE-BASIC	This message is displayed when the absolute program has been loaded.

The BASIC Interpreter is now in memory. You can execute BASIC using the following command:

```
*RU,BASIC

BASIC READY                    BASIC Interpreter message.
>                              BASIC Interpreter prompt.
```

You may now enter BASIC commands.

5-9. BASIC INTERPRETER NAME CONVENTIONS

Depending on your RTE-M System configuration, the BASIC Interpreter program may exist within your system under one or more names. You need to know the correct name of the BASIC Interpreter for specification in system commands such as RU and BR.

The name of the Interpreter program supplied by HP when you purchase your RTE-M System is BASIC.

In an RTE-MIII System that supports the Multi-Terminal Monitor (MTM), you may have the original Interpreter named BASIC and one uniquely named copy of the Interpreter program for each MTM terminal (see paragraph 5-10).

Further, for an RTE-M System that supports the flexible disc, if you load the Interpreter program into your system on-line using the system LO command, the program will be known to the system by the first five characters of its disc file name rather than by the program name (the disc file name and the program name may be the same).

5-10. MULTIPLE TERMINAL OPERATION

The RTE-M Programmer's Reference Manual, Section VIII, describes multiple terminal operation via the Multiple Terminal Monitor (MTM) package.

To use MTM in an RTE-M system supporting the RTE-M BASIC subsystem, we recommend that you allow a minimum of 48K words of memory. Therefore, support of the BASIC subsystem and MTM is practical only in an RTE-MIII system.

Assuming the conditions recommended above, your RTE-MIII system will support the BASIC subsystem and up to four terminal devices in addition to the system console. Users at these separate terminals can develop and execute BASIC programs simultaneously.

A separate copy of the BASIC Interpreter must be available for each terminal supported by MTM. You may create copies of the BASIC Interpreter using the techniques described in the RTE-M Programmer's Reference Manual, Section VIII.

> NOTE
>
> When you schedule a copy of BASIC in the MTM environment, you must provide your terminal's LU number in the RUN or ON command turn on parameters. Otherwise, communication will go to the system console (LU 1).

5-11. RTE-M TABLE GENERATOR (RTMTG)

The RTE-M Table Generator (RTMTG) assists you in producing tables that are required for communication between BASIC and external subroutines and functions. These are the BASIC Branch and Mnemonic Tables. BASIC uses these tables to transfer program execution to a subroutine or function and back.

5-12. RTMTG HARDWARE ENVIRONMENT

The hardware environment for the Table Generator is the same as that defined for the BASIC subsystem (see paragraph 5-1). In addition, RTMTG requires hardware capable of simultaneous support of both an input file and an output file. For example, an HP 2644 or 2645 Terminal equipped with cartridge tape unit devices.

5-13. RTMTG SOFTWARE ENVIRONMENT

The Table Generator executes within any RTE-MI, MII, or MIII Operating System that includes the BASIC subsystem. RTMTG is capable of operating in the same amount of memory space as the BASIC program, if this is necessary. Essentially, RTMTG is a utility program that should be loaded only when it is needed. Use the RTE-M Relocating Loader (RTMLD) and system snapshot to create an absolute program module containing RTMTG which can be loaded when needed via the RTE-M Absolute Program Loader (APLDR).

5-14. RTMTG DESCRIPTORS REQUIRED FOR LIBRARY SUBROUTINES

To call the various library subroutines from BASIC, you must create a Branch and Mnemonic Table using a descriptor entry for each subroutine desired. You may select from Figure 5-1 the descriptor entry for the specific subsystems and subroutines you want to use or, for user-written subroutines, you may construct the appropriate descriptor entry as shown in paragraph 5-16.

Data Conversion DCODE(RVA,RVA,RA) NUM(R),INTG CHRS(I,RVA)	requires FRMTR	Search: %BASLB %FF4.N
Bit Manipulation IBSET(I,I),INTG,ENT= BBSET IEOR(I,I),INTG,ENT= BEOR OR(I,I)INTGENT= BIOR AND(I,I),INTG,ENT= BAND NOT(I),INTG,ENT= BNOT ISHFT(I,I),INTG,ENT= BSHFT IBTST(I,I),INTG,ENT= BBTST IBCLR(I,I),INTG,ENT= BBCLR ISETC(RA),INTG,ENT= ISETC	Requires presence of ISA FTN Tables in system	Search: %BASLB
HP 6940 Subsystem DAC(I,R) MPNRM RDWRD(I,IV) WRWRD(I,I) RDBIT(I,I,IV) WRBIT(I,I,I) SENSE(I,I,I,I)		Search: %694BS
HP 2313/91000 Subsystems AISQV(I,I,RVA,IV) AIRDV(I,RA,RVANIVXZ PACER(I,I,I) NORM SGAIN(I,R) RGAIN(I,RV) AOV(I,RA,RA,IV)		Search: %A2313
Magnetic Tape Subsystem MTTRD(I,RVA,I,IV,IV) MTTRT(I,RA,I,IV,IV) MTTPT(I,I,I) MTTFS(I,I)		Search: %BASLB

Figure 5-1. Library Subroutine Descriptor Entries for RTMTG

Plotter Subsystem SFACT(R,R) FACT(R,R) WHERE(RV,RV) PLOT(R,R,I) LLEFT URITE PLTLU(I) AXIS(R,RNRA,R,R,R,R) NUMB(R,R,R,R,R,I) SYMB(R,R,R,RA,R,I) LINES(RA,RA,I,I,I,R) SCALE(RVANR,INI)	Search: %PTLIB
HP-IB Subsystem SRQSN(I,I) HPIB(I,I,I) SRQ(I,I,RA) CMDR(I,RA,RVA) CMDW(I,RA,RA) IBERR(I),INTG IBSTS(I),INTG	Search: %HPIB
Task Scheduling TIME(RV) SETP(I,I),ENT=SSETP START(I,R),ENT=SSTRT DSABL(I) ENABL(I) TRNON(I,R)	Search: %TSKSC

Figure 5-1. Library Subroutine Descriptor Entries for RTMTG (Continued)

5-15. RTMTG OPERATION

RTMTG operates interactively with you at the system console or at a terminal. The interactive dialog is initiated when the heading, "RTMTG", is displayed on your console or terminal. Then, RTMTG displays a prompt character, >, to indicate that it is ready to accept commands from you which define the mode of operation you desire. You may choose one of three modes - Edit, List, or Table.

In Edit Mode, you may edit the contents of the input file. When your editing is completed, the revised data resides in the output file.

In List Mode, RTMTG copies the contents of the input file to the output file with appended line numbers.

In Table Mode, the data from the input file is processed into a Branch Table and a Mnemonic Table. These tables then are written as relocatable binary records to the output file.

5-16. RTMTG INPUT FILE FORMAT. Your input file must contain a descriptor entry for each subroutine and function that will be called from a BASIC program. The descriptor entry has the following form:

name [(*parameters*)] [**INTG** / **,** / **REAL**] [,**ENT**=*entry point*]

where:

name — Specifies the subroutine or function name (1 to 5 characters).

(*parameters*) — Defines a list of parameter types to be passed to a subroutine or function (up to 15 parameters may be passed by BASIC). The parameter types specified must be separated with a comma and the parameter list must be enclosed in parentheses.

Each parameter type entry is a combination of the following elements:

[I / R] [V] [A]

where:

I indicates an integer variable parameter

R indicates a real variable parameter

V indicates a value is to be returned to BASIC from the subroutine or function

A indicates an array variable parameter.

Possible combinations are I, IV, IVA, IA, R, RV, RVA, or RA.

For example: SUB7(I) or SUB2(IV,RVA)

NOTE

A string variable should be declared as type R (real).

INTG — For a function, or a subroutine to be treated as a function, INTG indicates that the function value is returned as an integer; if not a function, do not use this entry.

REAL — For a function, or a subroutine to be treated as a function, REAL indicates that the function value is returned as a real (floating point) number; if not a function, do not use this entry.

ENT=*entry point* — Specifies the entry point name (1 to 5 characters). If the entry point name is the same as the subroutine or function name, this entry need not be specified. If specified, the keyword **ENT=** must precede the entry point name.

Examples:

```
SUB1(IV,RVA),ENT=SUBA

FUNC(I),INTG
```

See Figure 5-1 for more examples of descriptor entries.

5-17. RTMTG OUTPUT FILE FORMAT. Following completion of RTMTG execution, the contents of the output file depend on the operating mode you selected.

For Edit Mode, the output file content is an edited copy of the input file.

For List Mode, the content is a copy of the input file with appended line numbers.

For Table Mode, the output file contents are a relocatable Branch Table and Mnemonic Table.

5-18. Branch Table Format. RTMTG creates one Branch Table entry for each descriptor entry read from the input file. The entry point for the Branch Table is BRTBL.

The format of the Branch Table entry records is shown in Figure 5-2.

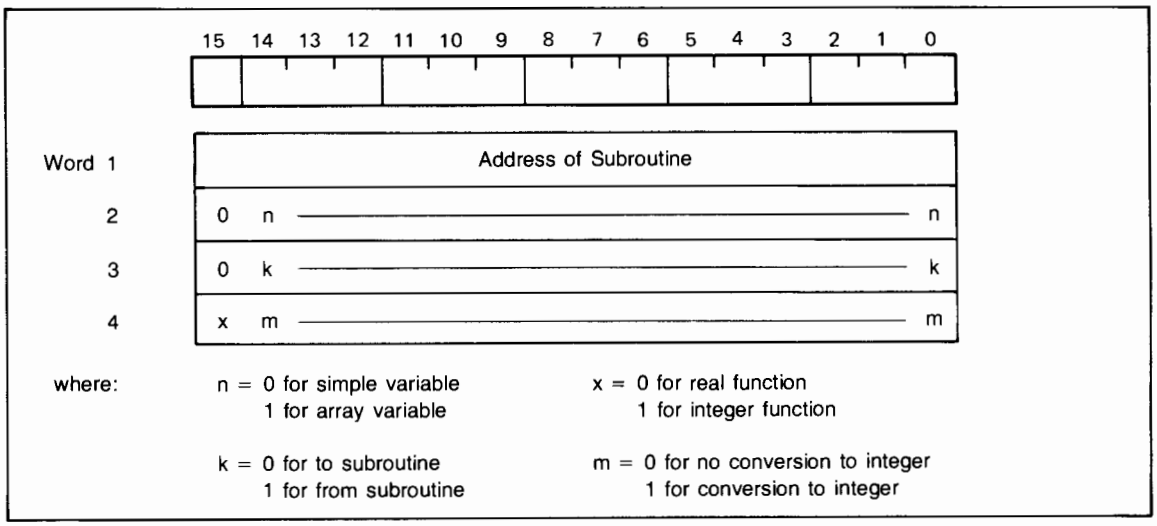
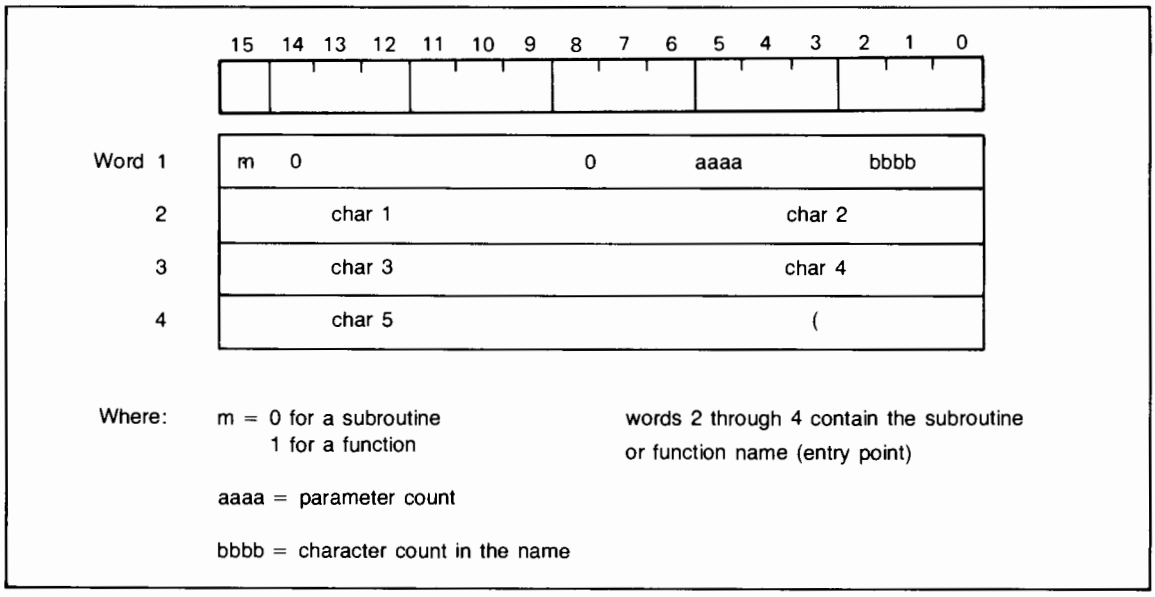


Figure 5-2. Branch Table Format

5-19. Mnemonic Table Format. Similar to the Branch Table format, RTMTG creates one Mnemonic Table entry for each descriptor entry read from the input file. The Mnemonic Table entry point is MNTBL and defines the number of table entries (expressed as a negative value).

The format of the Mnemonic Table entry records is shown in Figure 5-3.



5-3. Mnemonic Table Format

5-20. EXECUTING RTMTG

You execute the Table Generator by entering the RTE-M System command RU or ON (no turn on parameters are required), as follows:

RU,RTMTG

or,

ON,RTMTG

Table Generator execution begins with the display of a header message, followed by a prompt character (>):

```
"RTMTG"
>
```

You now may enter one of the following commands:

EDIT — to edit an input source file. The result is written to an output file. RTMTG will ask you for the location of these files before the Edit operation begins.

LIST — to copy an input file to an output file with appended line numbers. RTMTG will ask you for the location of these files before the List operation begins.

TABLE	to generate the Branch and Mnemonic Tables. Source is from an input file; relocatable binary records are written to an output file. RTMTG will ask you for the location of these files before the Table generation operation begins.
END	to terminate RTMTG execution. The message, "RTMTG FINISHED" is displayed on your session console following execution of the END command. You are returned to RTE-M System command mode.

The entry EDIT, LIST, or TABLE is followed by a request for the location of the input file:

INPUT ?

You enter the file name or the logical unit number where the input file resides.

The next RTMTG request is for the location of the output file:

OUTPUT ?

You enter the file name or the logical unit number where the Table Generator output will be written.

5-21. EDIT MODE OPERATING INSTRUCTIONS. When you enter the EDIT command, RTMTG displays the requests for the location of the input and output files. After you respond to these requests, RTMTG displays an edit header followed by a hyphen character (-) to prompt entry of an edit command:

"BRANCH AND MNEMONIC SOURCE EDIT"
-

At this point, you enter one of the following Edit Mode commands:

ABORT	terminates the Edit Mode immediately; you are returned to RTMTG command mode.
ADD	add a *descriptor* to the contents of the source file. If the source file is positioned at the beginning of information, the *descriptor* will be added before the first record. Otherwise, the *descriptor* will be added immediately after the current line. When you enter ADD followed by a carriage return, RTMTG displays a hyphen prompt (-). Following this prompt, you enter the *descriptor* (see Paragraph 5-16 for the *descriptor* format). Once a descriptor is added, it cannot be modified unless you rerun the Edit mode.
DELETE *nn*	delete *nn* lines starting with the current line. The value *nn* must be greater than zero or no operation occurs. Following each line deletion, a hyphen prompt (-) is displayed.
END	copies the remaining input file entries (without change) to the output file. You are then returned to RTMTG command mode.

FIND *nn*	find line number *nn*. The value *nn* must be greater than the current line number. Input file entries will be copied unchanged to the output file until line number *nn* is found. When the specified line number is found, that line number and its text is displayed on your session console. You may type a slash character (/) to cause the next line to be displayed.
REPLACE	replace the current line with a *descriptor*. When you enter REPLACE followed by a carriage return, RTMTG displays a hyphen prompt (-). Following this prompt, you enter a *descriptor* (see paragraph 5-16 for *descriptor* format). If you preceded the REPLACE command with a FIND command, the current line is replaced. Otherwise, the next line of the source file is displayed and then replaced.

When the source file is exhausted (at end-of-file point), the message "EOF" followed by a hyphen prompt (-) is displayed after execution of the DELETE, FIND, and REPLACE commands.

Editing operations must be performed by line number, sequentially, from the beginning of the input file. You can add, delete, replace, and find lines as long as you move forward through the file. Following each edit operation, the revised input entries (up to the current line number) are copied to the output file.

Edit operation error conditions result in an error message (see Table 5-2) and a return to Edit mode control. In these cases, you may continue your editing from the point where the error occurred (the current line number is displayed as part of the Edit command mode prompt).

If recovery from the error condition is not possible, you can enter the Edit command, ABORT. In this case, Edit mode is terminated and you are returned to RTMTG command mode. Enter the EDIT command and begin editing the contents of the input file from the beginning.

When you have completed your editing operations, enter the Edit command, END. Any unedited input file entries (that is, any entries with a line number value greater than the current line number) are copied unchanged to the output file. You are returned to RTMTG command mode.

When you specify the ADD command, you are prompted for a *descriptor* (see paragraph 5-16 for *descriptor* format).

For DELETE, the number of lines specified in the line count parameter will be deleted, beginning with the current line. If you attempt to delete lines past the end of the input file, the message, EOF, is displayed and control is returned to the Edit command mode.

The FIND command locates a specific line number. You must specify a line number value greater than the current line value. Input file entry lines will be copied unchanged to the output file until the specified line number is found. If the specified line number cannot be found, an error message results (see Table 5-2) and you are returned to Edit command mode.

The REPLACE command should be followed by a *descriptor* (see paragraph 5-16 for *descriptor* format).

5-22. LIST MODE OPERATING INSTRUCTIONS. When you enter the LIST command, RTMTG requests the file name or logical unit number for both the input and output file. RTMTG then displays the header message, "LIST", reads the descriptor entries from the input file, appends a line number to each entry, and copies the result to the output file.

5-23. TABLE MODE OPERATING INSTRUCTIONS. When you enter the TABLE command, RTMTG asks for the file name or logical unit number of both the input and output file. RTMTG then displays the header message, "BRANCH TABLE GENERATOR". Next, RTMTG reads the descriptor entries from the input file and creates a Branch Table and a Mnemonic Table which are written as relocatable binary records to the output file. The records are written in the following order:

Step 1. NAM record (with the compiler produced code bit set).

Step 2. ENT record for the Branch Table.

Step 3. EXT and DBL records for each descriptor entry (this is the Branch Table).

At this point, RTMTG displays the header message, "MNEMONIC TABLE GENERATOR", and continues as follows:

Step 4. ENT record for the Mnemonic Table.

Step 5. DBL records for each descriptor entry (this is the Mnemonic Table).

Step 6. END record.

Following Step 3, if the input file cannot be programmatically rewound, RTMTG displays the following message and suspends execution:

"REWIND SOURCE FILE"

To continue, you must rewind the input file and enter:

GO,RTMTG

5-24. TERMINATING RTMTG EXECUTION. Whenever RTMTG is in command mode (the command prompt, >, has been displayed and RTMTG is waiting for you to respond), you may terminate RTMTG execution by entering the command:

END

RTMTG execution terminates, the message "RTMTG FINISHED" is displayed on your session console, and you are returned to RTE-M System command mode.

5-25. RTMTG ERROR MESSAGES. Error conditions encountered during execution of the Table Generator result in the display of an appropriate error message. These messages are listed and described in Table 5-2.

Table 5-2. RTMTG Error Messages

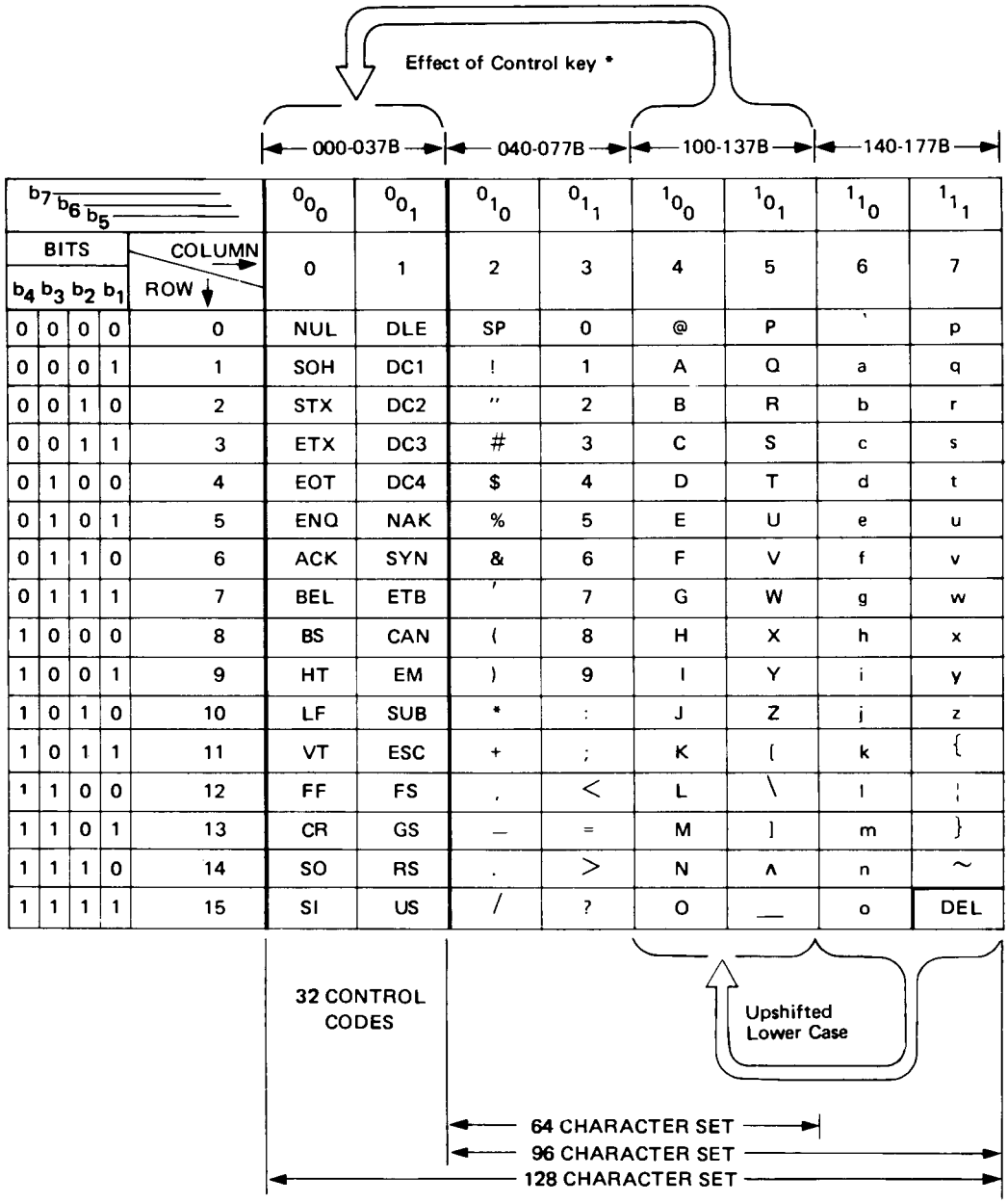
ERROR MESSAGE	MEANING	OPERATOR ACTION
RTMTG Command Mode Errors		
COMMAND ERR	Error in command string entry.	Correctly re-enter command.
File Manipulation Errors		
CLOSE ERR	File close failure on input or output file.	Control returns to RTE-M; determine problem and rerun RTMTG.
OPEN ERR	File open failure on input or output file.	Control returns to RTMTG; determine problem and re-enter RTMTG command.
READ ERR	File read error on input file.	Control returns to RTMTG; determine problem and re-enter RTMTG command.
WRITE ERR	File write error on output file.	Control returns to RTMTG; determine problem and re-enter RTMTG command.
GTFIL ERR	Subroutine GTFIL unable to access input or output file.	Control returns to RTE-M; determine problem and rerun RTMTG.
CREATE ERR	File creation failed on output file.	Control returns to RTMTG; determine problem and re-enter RTMTG command.
REWIND ERR	File rewind failure on input file.	Control returns to RTMTG; determine problem and re-enter RTMTG command.
Edit Mode Errors		
LINE ERR	Line Number specified cannot be found.	Control returns to Edit mode; check input file content.
SEQ ERR	Line number out of sequence.	Control returns to Edit mode; check input file content.
REPL ERR	No descriptor in REPLACE command string.	Control returns to Edit mode; re-enter command with a descriptor.
ADD ERR	No descriptor in ADD command string.	Control returns to Edit mode; re-enter command with a descriptor.
EOF	Attempt to delete lines past the end of the input file.	Control returns to Edit mode; re-enter command with correct line count value.

Table 5-2. RTMTG Error Messages (Continued)

ERROR MESSAGE	MEANING	OPERATOR ACTION
Table Mode Errors		
SYN ERR IN LINE *nn*	Syntax error encountered in descriptor entry at line number *nn*.	Descriptor entry processing continues to completion and any errors are listed. Control returns to RTMTG. Use LIST and EDIT to determine and correct erroneous entries and rerun Table Mode.

HP CHARACTER SET

APPENDIX A



b_7 b_6 b_5					$0\,0_0$	$0\,0_1$	$0\,1_0$	$0\,1_1$	$1\,0_0$	$1\,0_1$	$1\,1_0$	$1\,1_1$
BITS				COLUMN →	0	1	2	3	4	5	6	7
b_4	b_3	b_2	b_1	ROW ↓								
0	0	0	0	0	NUL	DLE	SP	0	@	P	`	p
0	0	0	1	1	SOH	DC1	!	1	A	Q	a	q
0	0	1	0	2	STX	DC2	"	2	B	R	b	r
0	0	1	1	3	ETX	DC3	#	3	C	S	c	s
0	1	0	0	4	EOT	DC4	$	4	D	T	d	t
0	1	0	1	5	ENQ	NAK	%	5	E	U	e	u
0	1	1	0	6	ACK	SYN	&	6	F	V	f	v
0	1	1	1	7	BEL	ETB	'	7	G	W	g	w
1	0	0	0	8	BS	CAN	(	8	H	X	h	x
1	0	0	1	9	HT	EM	)	9	I	Y	i	y
1	0	1	0	10	LF	SUB	*	:	J	Z	j	z
1	0	1	1	11	VT	ESC	+	;	K	[	k	{
1	1	0	0	12	FF	FS	,	<	L	\	l	¦
1	1	0	1	13	CR	GS	–	=	M	]	m	}
1	1	1	0	14	SO	RS	.	>	N	^	n	~
1	1	1	1	15	SI	US	/	?	O	_	o	DEL

EXAMPLE: The representation for the character "K" (column 4, row 11) is.

	b_7	b_6	b_5	b_4	b_3	b_2	b_1
BINARY	1	0	0	1	0	1	1
OCTAL	1		1			3	

* Depressing the Control key while typing an upper case letter produces the corresponding control code on most terminals. For example, Control-H is a backspace.

9206- 1A

HEWLETT-PACKARD CHARACTER SET FOR COMPUTER SYSTEMS

This table shows HP's implementation of ANS X3.4-1968 (USASCII) and ANS X3.32-1973. Some devices may substitute alternate characters from those shown in this chart (for example, Line Drawing Set or Scandanavian font). Consult the manual for your device.

The left and right byte columns show the octal patterns in a 16 bit word when the character occupies bits 8 to 14 (left byte) or 0 to 6 (right byte) and the rest of the bits are zero. To find the pattern of two characters in the same word, add the two values. For example, "AB" produces the octal pattern 040502. (The parity bits are zero in this chart.)

The octal values 0 through 37 and 177 are control codes. The octal values 40 through 176 are character codes.

Decimal Value	Octal Values		Mnemonic	Graphic[1]	Meaning
	Left Byte	Right Byte			
0	000000	000000	NUL	$^{N}_{U}$	Null
1	000400	000001	SOH	$^{S}_{H}$	Start of Heading
2	001000	000002	STX	$^{S}_{X}$	Start of Text
3	001400	000003	ETX	$^{E}_{X}$	End of Text
4	002000	000004	EOT	$^{E}_{T}$	End of Transmission
5	002400	000005	ENQ	$^{E}_{Q}$	Enquiry
6	003000	000006	ACK	$^{A}_{K}$	Acknowledge
7	003400	000007	BEL		Bell, Attention Signal
8	004000	000010	BS	$^{B}_{S}$	Backspace
9	004400	000011	HT	$^{H}_{T}$	Horizontal Tabulation
10	005000	000012	LF	$^{L}_{F}$	Line Feed
11	005400	000013	VT	$^{V}_{T}$	Vertical Tabulation
12	006000	000014	FF	$^{F}_{F}$	Form Feed
13	006400	000015	CR	$^{C}_{R}$	Carriage Return
14	007000	000016	SO	$^{S}_{O}$	Shift Out } Alternate
15	007400	000017	SI	$^{S}_{I}$	Shift In } Character Set
16	010000	000020	DLE	$^{D}_{L}$	Data Link Escape
17	010400	000021	DC1	$^{D}_{1}$	Device Control 1 (X-ON)
18	011000	000022	DC2	$^{D}_{2}$	Device Control 2 (TAPE)
19	011400	000023	DC3	$^{D}_{3}$	Device Control 3 (X-OFF)
20	012000	000024	DC4	$^{D}_{4}$	Device Control 4 ($\overline{\text{TAPE}}$)
21	012400	000025	NAK	$^{N}_{K}$	Negative Acknowledge
22	013000	000026	SYN	$^{S}_{Y}$	Synchronous Idle
23	013400	000027	ETB	$^{E}_{B}$	End of Transmission Block
24	014000	000030	CAN	$^{C}_{N}$	Cancel
25	014400	000031	EM	$^{E}_{M}$	End of Medium
26	015000	000032	SUB	$^{S}_{B}$	Substitute
27	015400	000033	ESC	$^{E}_{C}$	Escape[2]
28	016000	000034	FS	$^{F}_{S}$	File Separator
29	016400	000035	GS	$^{G}_{S}$	Group Separator
30	017000	000036	RS	$^{R}_{S}$	Record Separator
31	017400	000037	US	$^{U}_{S}$	Unit Separator
127	077400	000177	DEL	■	Delete, Rubout[3]

Decimal Value	Octal Values		Character	Meaning
	Left Byte	Right Byte		
32	020000	000040		Space, Blank
33	020400	000041	!	Exclamation Point
34	021000	000042	"	Quotation Mark
35	021400	000043	#	Number Sign, Pound Sign
36	022000	000044	$	Dollar Sign
37	022400	000045	%	Percent
38	023000	000046	&	Ampersand, And Sign
39	023400	000047	'	Apostrophe, Acute Accent
40	024000	000050	(	Left (opening) Parenthesis
41	024400	000051	)	Right (closing) Parenthesis
42	025000	000052	*	Asterisk, Star
43	025400	000053	+	Plus
44	026000	000054	,	Comma, Cedilla
45	026400	000055	-	Hyphen, Minus, Dash
46	027000	000056	.	Period, Decimal Point
47	027400	000057	/	Slash, Slant
48	030000	000060	0	} Digits, Numbers
49	030400	000061	1	
50	031000	000062	2	
51	031400	000063	3	
52	032000	000064	4	
53	032400	000065	5	
54	033000	000066	6	
55	033400	000067	7	
56	034000	000070	8	
57	034400	000071	9	
58	035000	000072	:	Colon
59	035400	000073	;	Semicolon
60	036000	000074	<	Less Than
61	036400	000075	=	Equals
62	037000	000076	>	Greater Than
63	037400	000077	?	Question Mark

Decimal Value	Octal Values		Character	Meaning
	Left Byte	Right Byte		
64	040000	000100	@	Commercial At
65	040400	000101	A	
66	041000	000102	B	
67	041400	000103	C	
68	042000	000104	D	
69	042400	000105	E	
70	043000	000106	F	
71	043400	000107	G	
72	044000	000110	H	
73	044400	000111	I	
74	045000	000112	J	
75	045400	000113	K	
76	046000	000114	L	
77	046400	000115	M	
78	047000	000116	N	Upper Case Alphabet, Capital Letters
79	047400	000117	O	
80	050000	000120	P	
81	050400	000121	Q	
82	051000	000122	R	
83	051400	000123	S	
84	052000	000124	T	
85	052400	000125	U	
86	053000	000126	V	
87	053400	000127	W	
88	054000	000130	X	
89	054400	000131	Y	
90	055000	000132	Z	
91	055400	000133	[	Left (opening) Bracket
92	056000	000134	\	Backslash, Reverse Slant
93	056400	000135	]	Right (closing) Bracket
94	057000	000136	^ ↑	Caret, Circumflex; Up Arrow[4]
95	057400	000137	_ ←	Underline; Back Arrow[4]

Decimal Value	Octal Values		Character	Meaning
	Left Byte	Right Byte		
96	060000	000140	`	Grave Accent[5]
97	060400	000141	a	
98	061000	000142	b	
99	061400	000143	c	
100	062000	000144	d	
101	062400	000145	e	
102	063000	000146	f	
103	063400	000147	g	
104	064000	000150	h	
105	064400	000151	i	
106	065000	000152	j	
107	065400	000153	k	
108	066000	000154	l	
109	066400	000155	m	
110	067000	000156	n	Lower Case Letters[5]
111	067400	000157	o	
112	070000	000160	p	
113	070400	000161	q	
114	071000	000162	r	
115	071400	000163	s	
116	072000	000164	t	
117	072400	000165	u	
118	073000	000166	v	
119	073400	000167	w	
120	074000	000170	x	
121	074400	000171	y	
122	075000	000172	z	
123	075400	000173	{	Left (opening) Brace[5]
124	076000	000174	¦	Vertical Line[5]
125	076400	000175	}	Right (closing) Brace[5]
126	077000	000176	~	Tilde, Overline[5]

9206-1C

Notes: [1]This is the standard display representation. The software and hardware in your system determine if the control code is displayed, executed, or ignored. Some devices display all control codes as "|", "@", or space.

[2]Escape is the first character of a special control sequence. For example, ESC followed by "J" clears the display on a 2640 terminal.

[3]Delete may be displayed as "__", "@", or space.

[4]Normally, the caret and underline are displayed. Some devices substitute the up arrow and back arrow.

[5]Some devices upshift lower case letters and symbols (` through ~) to the corresponding upper case character (@ through ^). For example, the left brace would be converted to a left bracket.

RTE SPECIAL CHARACTERS

Mnemonic	Octal Value	Use
SOH (Control A)	1	Backspace (TTY)
EM (Control Y)	31	Backspace (2600)
BS (Control H)	10	Backspace (TTY, 2615, 2640, 2644, 2645)
EOT (Control D)	4	End-of-file (TTY 2615, 2640, 2644, 2645)

9206-1D

SYSTEM TABLES AND COMMUNICATION AREA

APPENDIX B

BASE PAGE COMMUNICATION AREA

A block of storage in base page, starting at octal location 1647, contains the system communication area and is used by RTE-M to define request parameters, I/O tables, scheduling lists, operating parameters, memory bounds, etc. The Real-Time Assembler allows absolute references into this area (i.e., less than octal 2000) within relocatable programs, so that user programs can read information from this area, but cannot alter it because of the memory protect feature.

Octal Location	Contents	Description
SYSTEM TABLE DEFINITION		
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 word 1 table
01653	LUMAX	Number of logical units (in DRT)
01654	INTBA	FWA of interrupt table
01655	INTLG	Number of interrupt table entries
01656		Reserved for system
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 4 words)
01661	EQT2	
01662	EQT3	
01663	EQT4	
01664	EQT5	
01665	EQT6	
01666	EQT7	
01667	EQT8	
01670	EQT9	
01671	EQT10	
01672	EQT11	
01673	CHAN	Current DMA channel number
01674	TBG	I/O address of time-base card
01675	SYSTY	EQT entry address of system console

Octal Location	Contents	Description
SYSTEM REQUEST PROCESSOR/'EXEC' COMMUNICATION		
01676	RQCNT	Number of request parameters - 1
01677	RQRTN	Return point address
01700	RQP1	Addresses of
01701	RQP2	request
01702	RQP3	parameters
01703	RQP4	(set
01704	RQP5	for
01705	RQP6	maximum
01706	RQP7	of 9
01707	RQP8	parameters)
01710	RQP9	
ADDRESSES OF SYSTEM LISTS		
01711	SKEDD	'Schedule' list
01713	SUSP2	'Wait suspend' list
01714	SUSP3	'Available memory' list
01715	SUSP4	Reserved
01716	SUSP5	'Operator suspend' list
DEFINITION OF EXECUTING PROGRAM ID SEGMENT		
01717	XEQT	ID segment address of current program
01720	XLINK	'Linkage'
01721	XTEMP	'Temporary' (5-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' at suspension
SYSTEM MODULE COMMUNICATION FLAGS		
OPATN		
Operator/keyboard attention flag		
01735	OPFLG	Operator communication flag
01736		Reserved
01737	DUMMY	I/O address of dummy int. card
01740		Reserved
01741		Reserved

Octal Location	Contents	Description
DEFINITION OF MEMORY ALLOCATION BASES		
01742	BPA1	FWA RT BP link area for memory resident programs
01743	BPA2	LWA BP link user program area
01744		Reserved
01745	LBORG	FWA of resident library area
01746	RTORG	FWA of user program area
01747	RTCOM	Length of system common area
01750	RTDRA	FWA of RT area
01751	AVMEM	FWA of system available memory
01752		Reserved
01753		Reserved
01754	BKDRA	LWA of resident program area; plus 1.
UTILITY PARAMETERS		
01755		Reserved for system
01756		
01757		
01760		
01761		
01762		
01763		
01764		
01765		
01766		Reserved
01767		
01770	MPTFL	Memory protect on/off flag (0/1)
01771	EQT12	Address of
01772	EQT13	last 4
01773	EQT14	words of
01774	EQT15	current EQT
01775	FENCE	memory protect fence address
01777	BKLWA	LWA of memory

PROGRAM ID SEGMENT

Each user program has a 31 word ID segment located in the system area. The format of the ID segment is shown in Table B-1. The address of each ID segment is located in the Keyword Table (see location 01657 under "System Table Definition." The ID segment contains static and dynamic information defining the properties of a program. The static information is set during generation time or when a program is loaded on-line, and, the dynamic information is maintained by the operating system.

Table B-1. ID Segment Map

WORD	CONTENTS															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-1	X-REGISTER AT SUSPENSION (MEMORY RESIDENT PROGRAMS ONLY)															
0	Y-REGISTER AT SUSPENSION (MEMORY RESIDENT PROGRAMS ONLY)															
1	LIST LINKAGE															
2-6	5 WORD TEMPORARY AREA USED FOR SPECIAL FLAGS IN QUEUES (ETC)															
7	PRIORITY															
8	PRIMARY ENTRY POINT															
9	POINT OF SUSPENSION (XSUSP)															
10	A REGISTER AT SUSPENSION (XA)															
11	B REGISTER AT SUSPENSION (XB)															
12	E/O REGISTERS AT SUSPENSION (XEO)															
13	NAME (FIRST AND SECOND CHARACTERS)															
14	NAME (THIRD AND FOURTH CHARACTERS)															
15	NAME (FIFTH CHARACATER)								TM	CL	AM	SS	TYPE			
16	NA	*	NP	W	A	*	O	*	R	D	*		STATUS			
17	TIME LIST LINKAGE WORD															
18	RESOLUTION			T	MULTIPLE											
19	LOW ORDER 16 BITS OF EXECUTE TIME LESS 24 HRS. IN 10's MS															
20	HIGH ORDER 16 BITS OF EXECUTE TIME															
21	*	FW	*	AT	RM	RE	PW	RN	FATHER ID SEG. NUMBER							
22	RP	# OF PAGES						MPFI		*	PARTITION #					
23	LOW MAIN ADDRESS															
24	HI MAIN ADDRESS +1															
25	LOW BASE PAGE ADDRESS															
26	HI BASE PAGE ADDRESS +1															
27	CURRENT SEGMENT'S LAST ADDRESS +1 (SET BY SEGLD)															
28	(RESERVED)															
29	NRF										CONSOLE			LU		

WHERE:

- * = These bits are reserved for future improvements.
- AM = All memory (program uses all of its area).
- SS = Short segment)indicates a 9-word ID segment).
- NA = No abort (pass about errors to the program instead).
- NP = No parameters allowed on reschedule.
- W = Wait bit (waiting for program whose ID segment address is in word 2).
- A = Abort on next list entry for this program.
- O = Operator suspend on next schedule attempt.
- R = Resource save (same resources when setting dormant).
- D = Dormant bit (set dormant on next schedule attempt).
- T = Time list entry bit (program is in the time list).
- FW = Father is waiting (he scheduled with wait).
- AT = Attention bit (operator has requested attention).
- RM = Re-entrant memory must be moved before dispatching program.
- RE = Re-entrant routine in control now.
- PW = Program wait (some 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.
- MPF1= Memory protect fence index.
- NRF = New-Run flag is set each time the program is scheduled to run.

The number of ID segments contained in a system is set during generation time, and is directly related to the number of programs that can be in core at any given time. If all the ID segments are in use, no more programs can be added on-line.

Note that segments in a segmented program share the ID segment of the main (calling) program.

THE EQUIPMENT TABLE

The Equipment Table (EQT) has an entry for each I/O controller recognized by RTE-M (these entries are established by the user when the RTE-M System is generated). These 15-word EQT entries reside in the system, and have format as shown in Table B-2.

Table B-2. EQT Table Entries

Word	Contents															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	I/O Request List Pointer															
2	Driver "Initiation" Section Address															
3	Driver "Completion" Section Address															
4	D	B	P	S	T	Unit #							Channel #			
5	AV		EQ TYPE CODE					STATUS								
6	CONWD (Current I/O Request Word)															
7	Request Buffer Address															
8	Request Buffer Length															
9	Temporary Storage for Optional Parameter															
10	Temporary Storage for Optional Parameter															
11	Temporary Storage for Driver															
12	Temporary Storage for Driver															
13	Temporary Storage for Driver															
14	Device Time-Out Reset Value															
15	Device Time-Out Clock															

Where:

D = 1 if DMA required.

B = 1 if automatic output buffering used.

P = 1 if driver is to process power fail.

S = 1 if driver is to process time-out.

T = 1 if device timed out (system sets to zero before each I/O request).

Unit = Last sub-channel addressed.

Channel	=	I/O select code for the I/O controller (lower number if a multi-board interface).
AV	=	I/O controller availability indicator: 0 = available for use. 1 = disabled (down). 2 = busy (currently in operation). 3 = waiting for an available DMA channel.
STATUS	=	the actual physical status or simulated status at the end of each operation. For paper tape devices, two status conditions are simulated: Bit 5 = 1 means end-of-tape on input, or tape supply low on output.
EQ TYPE CODE	=	type of device. When this octal number is linked with "DV*x*," it identifies the devices's software driver routine as follows:
00 to 07	=	paper tape devices (or system control devices).
00	=	teleprinter (or system keyboard control device).
01	=	photoreader.
Y6	=	paper tape punch.
05 sub 0	=	console (or system keyboard control device).
05 sub 1 05 sub 2	=	mini cartridge devices
10 to 17	=	unit record devices.
11	=	card reader.
10	=	plotter.
12	=	line printer.
15	=	mark sense card reader.
20 to 37	=	magnetic tape/mass storage devices.
33	=	flexible disc
40 to 77	=	instruments
CONWD	=	user control. Word supplied in the I/O EXEC call (see Section III).

DEVICE REFERENCE TABLE

Logical unit numbers numbers from decimal 1 to 63 provide logical addressing of the physical devices defined in the EQT and the subchannels within the physical devices (if applicable). These numbers are maintained in the Device Reference Table (DRT), which is created by the generator, and can be modified by the LU operator request (see Figure B-1). Base page location 1652 contains the address of the DRT first word table. Base page location 1653 contains the number of LU entries (LUMAX).

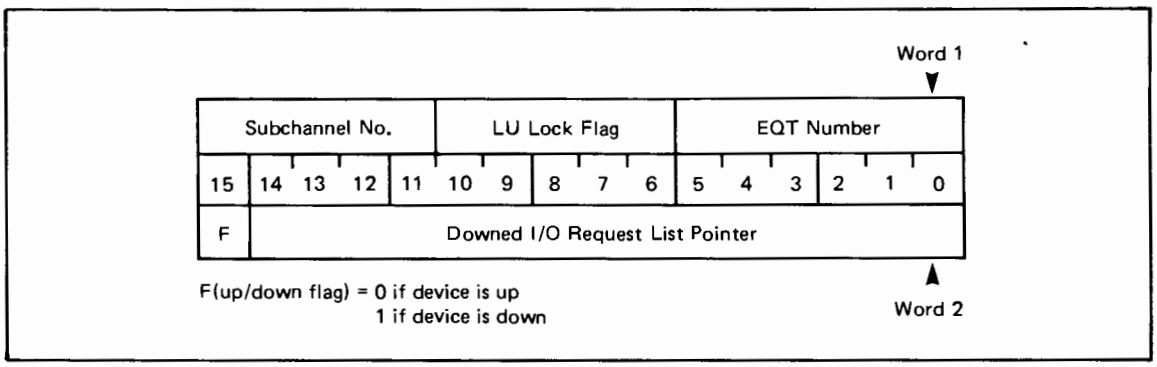


Figure B-1. Device Reference Table

The first DRT word contains the EQT entry number of the device assigned to the logical unit, and the subchannel number within the EQT entry. The second DRT word contains the logical unit's status (up or down) and a pointer to any downed I/O requests. If the pointer is less than 64, it is the LU number off of which the downed I/O requests are queued. If several LU's point to the same device, the requests are queued off the lowest LU number (the major LU). If the pointer is greater than 64, it points to the device's downed I/O request list. There are separate tables for words one and two, with the word two table located in memory immediately following the word one table. The functions of logical units 1 through 6 are predefined in the RTE-M System as:

1 — system console

2 — reserved for flexible disc if any

3 — reserved for system

4 — standard output unit left CTU

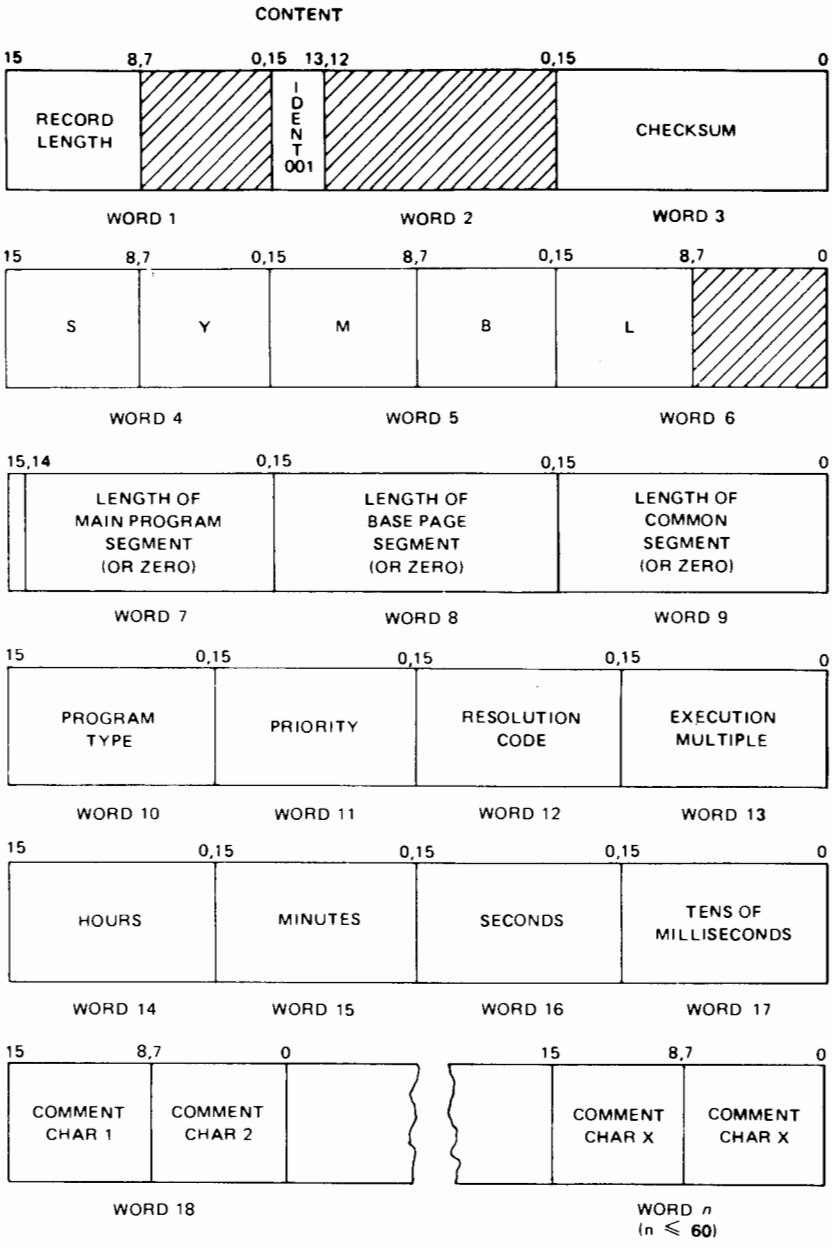
5 — standard input unit right CTU

6 — standard list unit

RELOCATABLE RECORD FORMATS

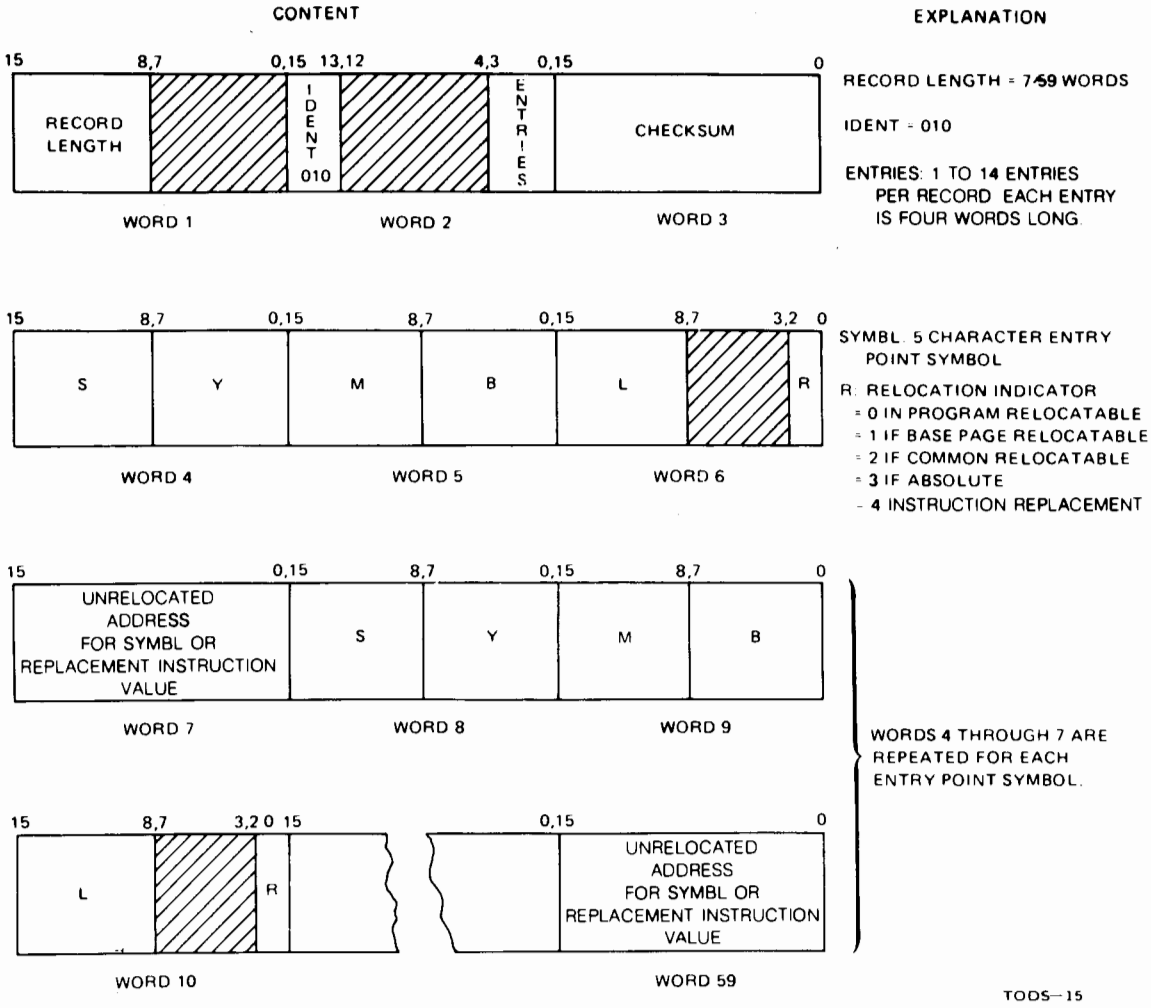
APPENDIX C

NAM RECORD

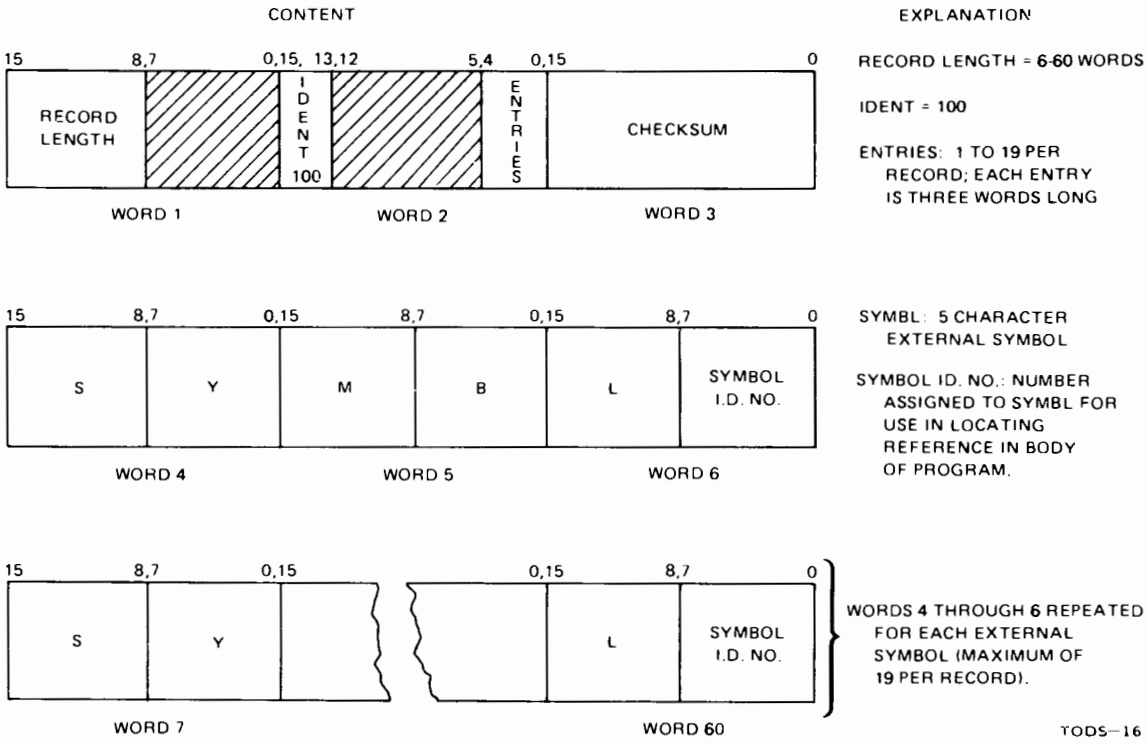


TODS-14

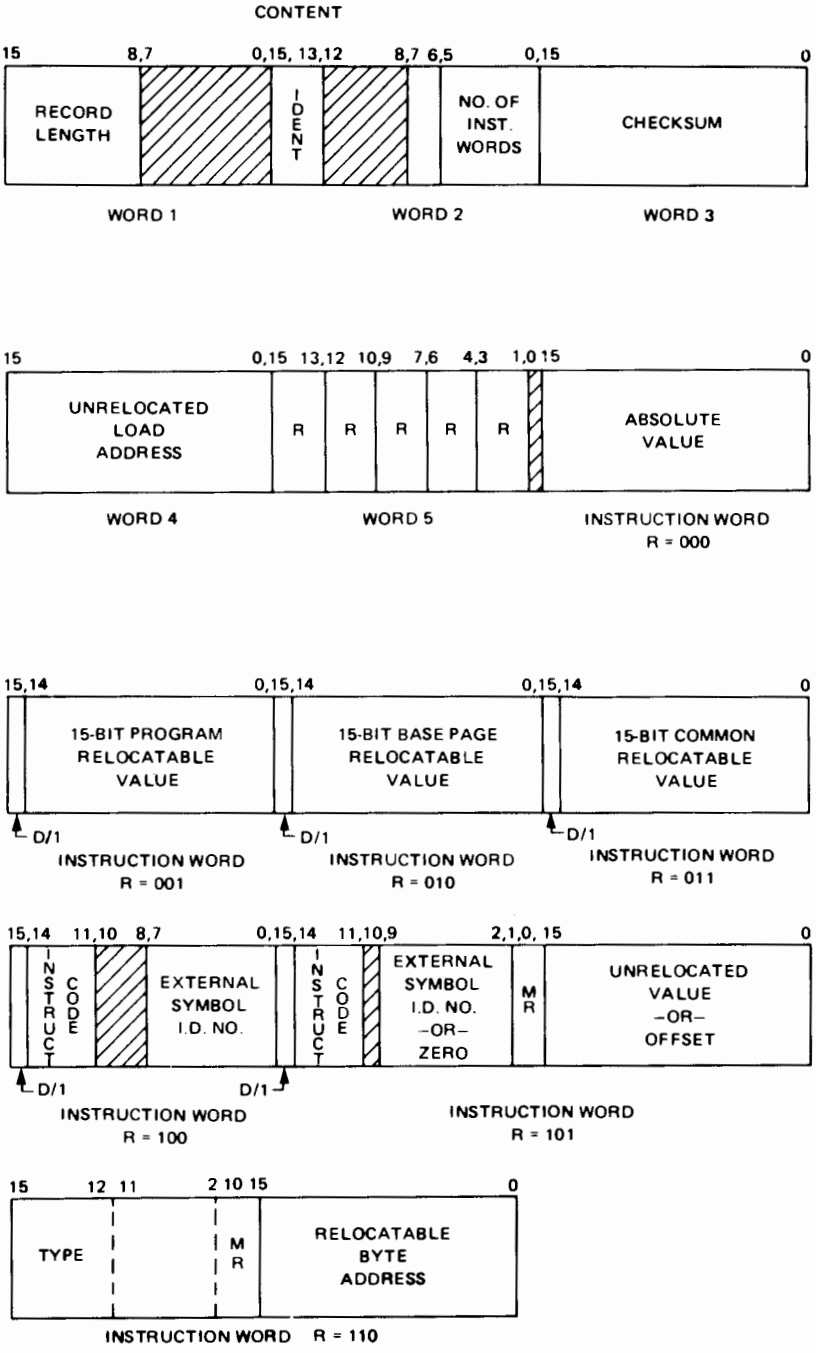
ENT RECORD


CONTENT
EXPLANATION
15
8,7
0,15
13,12
4,3
0,15
0
RECORD LENGTH
IDENT 010
ENTRIES
CHECKSUM
WORD 1
WORD 2
WORD 3
RECORD LENGTH = 7-59 WORDS
IDENT = 010
ENTRIES: 1 TO 14 ENTRIES PER RECORD EACH ENTRY IS FOUR WORDS LONG.
15
8,7
0,15
8,7
0,15
8,7
3,2
0
S
Y
M
B
L
R
WORD 4
WORD 5
WORD 6
SYMBL: 5 CHARACTER ENTRY POINT SYMBOL
R: RELOCATION INDICATOR
= 0 IN PROGRAM RELOCATABLE
= 1 IF BASE PAGE RELOCATABLE
= 2 IF COMMON RELOCATABLE
= 3 IF ABSOLUTE
= 4 INSTRUCTION REPLACEMENT
15
0,15
8,7
0,15
8,7
0
UNRELOCATED ADDRESS FOR SYMBL OR REPLACEMENT INSTRUCTION VALUE
S
Y
M
B
WORD 7
WORD 8
WORD 9
15
8,7
3,2
0
15
0,15
0
L
R
UNRELOCATED ADDRESS FOR SYMBL OR REPLACEMENT INSTRUCTION VALUE
WORD 10
WORD 59
WORDS 4 THROUGH 7 ARE REPEATED FOR EACH ENTRY POINT SYMBOL.
TODS-15

EXT RECORD

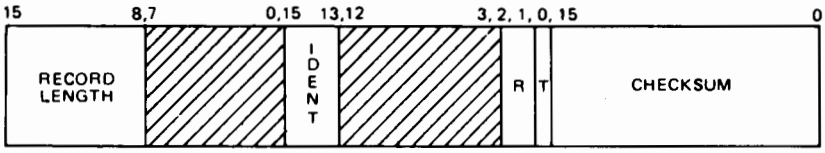

CONTENT
EXPLANATION
15
8,7
0,15, 13,12
5,4
0,15
0
RECORD LENGTH
IDENT 100
ENTRIES
CHECKSUM
WORD 1
WORD 2
WORD 3
RECORD LENGTH = 6-60 WORDS
IDENT = 100
ENTRIES: 1 TO 19 PER RECORD; EACH ENTRY IS THREE WORDS LONG
15
8,7
0,15
8,7
0,15
8,7
0
S
Y
M
B
L
SYMBOL I.D. NO.
WORD 4
WORD 5
WORD 6
SYMBL: 5 CHARACTER EXTERNAL SYMBOL
SYMBOL ID. NO.: NUMBER ASSIGNED TO SYMBL FOR USE IN LOCATING REFERENCE IN BODY OF PROGRAM.
15
8,7
0,15
0,15
8,7
0
S
Y
L
SYMBOL I.D. NO.
WORD 7
WORD 60
WORDS 4 THROUGH 6 REPEATED FOR EACH EXTERNAL SYMBOL (MAXIMUM OF 19 PER RECORD).
TODS-16

DBL RECORD


CONTENT
EXPLANATION
15
8,7
0,15, 13,12
8,7 6,5
0,15
0
RECORD LENGTH
IDENT
NO. OF INST. WORDS
CHECKSUM
WORD 1
WORD 2
WORD 3
RECORD LENGTH = 6-60 WORDS
IDENT = 011
Z/C: RELOCATION OF LOAD ADDRESS
= 0 FOR BASE PAGE
= 1 FOR PROGRAM
= 2 FOR ABSOLUTE
= 3 FOR COMMON
NO. OF INST. WORDS: 1 TO 45 LOADABLE INSTRUCTION WORDS PER RECORD
15
0,15 13,12 10,9 7,6 4,3 1,0 15
0
UNRELOCATED LOAD ADDRESS
R
R
R
R
R
ABSOLUTE VALUE
WORD 4
WORD 5
INSTRUCTION WORD R = 000
RELOCATABLE LOAD ADDRESS: STARTING ADDRESS FOR LOADING THE INSTRUCTIONS WHICH FOLLOW;
R's: RELOCATION INDICATORS:
000 = ABSOLUTE
001 = 15-BIT PROGRAM RELOCATABLE
010 = 15-BIT BASE PAGE RELOCATABLE
011 = 15-BIT COMMON RELOCATABLE
100 = EXTERNAL REFERENCE
101 = MEMORY REFERENCE
110 = BYTE REFERENCE
15,14
0,15,14
0,15,14
0
15-BIT PROGRAM RELOCATABLE VALUE
15-BIT BASE PAGE RELOCATABLE VALUE
15-BIT COMMON RELOCATABLE VALUE
D/I
D/I
D/I
INSTRUCTION WORD R = 001
INSTRUCTION WORD R = 010
INSTRUCTION WORD R = 011
R1 IS RELOCATION INDICATOR FOR INSTRUCTION WORD1; R2, FOR INSTRUCTION WORD2; ETC.
15,14
11,10
8,7
0,15,14
11,10,9
2,1,0, 15
0
INSTRUCT
CODE
EXTERNAL SYMBOL I.D. NO.
INSTRUCT
CODE
EXTERNAL SYMBOL I.D. NO. -OR- ZERO
MR
UNRELOCATED VALUE -OR- OFFSET
D/I
D/I
INSTRUCTION WORD R = 100
INSTRUCTION WORD R = 101
D/I: INDIRECT ADDRESSING
0 = DIRECT
1 = INDIRECT
MEMORY REFERENCE INSTRUCTIONS USE TWO WORDS, WITHIN THE TWO-WORD GROUP?, "MR" INDICATES RELOCATABILITY OF OPERAND SPECIFIED IN SECOND WORDS:
00 = PROGRAM RELOCATABLE
01 = BASE PAGE RELOCATABLE
10 = COMMON RELOCATABLE
11 = ABSOLUTE
15
12 11
2 10 15
0
TYPE
MR
RELOCATABLE BYTE ADDRESS
INSTRUCTION WORD R = 110
TODS—17

END RECORD

CONTENT



EXPLANATION

RECORD LENGTH = 4 WORDS
IDENT = 101

R: RELOCATION INDICATOR FOR TRANSFER ADDRESS

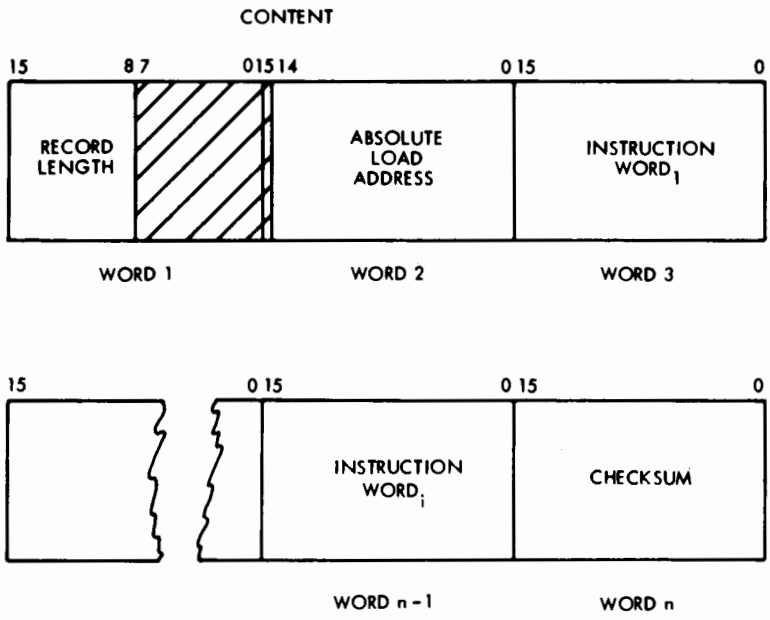
= 0 IF PROGRAM RELOCATABLE
= 1 IF BASE PAGE RELOCATABLE
= 2 IF COMMON RELOCATABLE
= 3 IF ABSOLUTE

T: TRANSFER ADDRESS INDICATOR

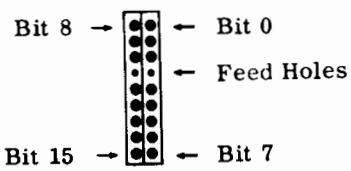
= 0 IF NO TRANSFER ADDRESS IN RECORD
= 1 IF TRANSFER ADDRESS PRESENT

TODS--18

ABSOLUTE TAPE FORMAT



†On paper tape, each word represents two frames arranged as follows:



RTE-M GENERATION WORKSHEET FORMS

APPENDIX D

This Appendix contains a complete set of worksheet forms for planning and configuring a site-specific RTE-M operating system. It is recommended that all forms be duplicated before using, and the blank original forms returned to the manual for possible later use.

In particular, multiple copies of page 2 of 2 of the Program Partition/Program Parameter Input Worksheet will probably be needed; one for each user program to be relocated during generation as part of the RTE-M resident operating system.

RTE-M I/O CONFIGURATION WORKSHEET PAGE of

I/O SLOT NO. (octal)	DEVICE NAME	DRIVER NAME	LU NO. (dec.)	EQT SUBCH. NO. (dec.)	INTERRUPT TABLE ENTRY	BUF. REQ. (B)	DMA (D)	TIME OUT (dec. no. of 10 ms)	EQT EXT. (dec. no. words)
					,				
					,				
					,				
					,				
					,				
					,				
					,				
					,				
					,				
					,				
					,				
					,				
					,				
					,				
					,				

RTE-M INITIALIZATION WORKSHEET

BOLDFACE terms are output to user by RTMGN. All other data (except comments) is user input. Fill in blanks with desired parameters.

RTMGN
DEFINE OUTPUT DEVICES

ECHO ON _____ (optional-enter LU or file name)
OUTPUT ON _____ (required-enter LU or file name)
MAP _____ , _____ , _____ , ON _____ (optional-enter LU or file name)
END (required)

TYPE OF SYSTEM?

_____ (enter 1, 2 or 3 for MI, MII or MIII)

TBG CHNL?

_____ (octal select code from I/O Worksheet)

PRIV INT CARD ADDR?

_____ (octal select code, or 0 if none)

PRIVILEGED DRIVERS ACCESS COMMON? (MIII systems only)

_____ (YES or NO)

MEM SIZE? (MIII only)

_____ (decimal no. pages)

LWAM? (MI and MII only)

_____ (last word of available memory in octal)

FWA BP?

_____ (octal address-higher than location for I/O links)

CHANGE ENTS? (repeated until END encountered — optional octal entries-see "Initialization Phase in Section III before changing entry points)

_____ , _____ , _____	_____ , _____ , _____	_____ , _____ , _____
_____ , _____ , _____	_____ , _____ , _____	_____ , _____ , _____
_____ , _____ , _____	_____ , _____ , _____	_____ , _____ , _____
_____ , _____ , _____	_____ , _____ , _____	_____ , _____ , _____
_____ , _____ , _____	_____ , _____ , _____	END (required)

RTE-M SYSTEM MODULES RELOCATION WORKSHEET

BOLDFACE terms are output to user by RTMGN.

– Hyphens are RTMGN prompts. All other data (except comments) is entered by user. Fill in blanks with desired parameters.

REL SYS MODS

LINKS IN _____ (optional; enter BASE or CURRENT)

– MAP _____ , _____ , _____ , ON _____ (not necessary if specified previously)

– RELOCATE _____ (system modules name - see RTE-M Software Numbering Catalog)

– RELOCATE _____ (optional system modules; class I/O, resource numbers, powerfail, etc.)

– RELOCATE _____

– RELOCATE _____

– RELOCATE _____

– RELOCATE _____

– SEARCH _____ (Dummy library to satisfy unwanted external REFS)

– RELOCATE _____

– RELOCATE _____ (enter all I/O driver names and interrupt subroutine names - refer to appropriate I/O Driver manuals for names)

– RELOCATE _____

– RELOCATE _____

– RELOCATE _____

– RELOCATE _____

– RELOCATE _____

– RELOCATE _____

– RELOCATE _____

– DISPLAY UNDEFS (optional)

– END (required to terminate system module relocation)

STARTING ADDRESS 0002
NO UNDEFS

SEARCH _____ (any additional required library searches; indicate by line and arrow where they should be made.)

SEARCH _____

EQT TABLE BUILDING WORKSHEET

BOLDFACE terms are output to user by RTMGN.
– Hyphens are RTMGN prompts. Fill in blanks with desired parameters.

 # OF I/O CLASSES? (MII and MIII only)

_____ (decimal no. between 0 - 255)

 # OF RESOURCE NUMBERS? (MII and MIII only)

_____ (decimal no. between 0 - 255)

BUFFER LIMITS (LOW, HIGH?

_____ , _____ (decimal no. of words. Enter 0, 0 for no limits)

EQT TBL (Equipment Table entries; get data from I/O
EQT 1=? Configuration Worksheet)
_____ , DV _____ , _____ , _____ , T = _____ , X = _____

EQT 2 = ?
_____ , DV _____ , _____ , _____ , T = _____ , X = _____

EQT 3 = ? See Section III,
_____ , DV _____ , _____ , _____ , T = _____ , X = _____

EQT 4 = ? "EQT Tables" for
_____ , DV _____ , _____ , _____ , T = _____ , X = _____

EQT 5 = ? format and input
_____ , DV _____ , _____ , _____ , T = _____ , X = _____

EQT 6 = ? considerations.
_____ , DV _____ , _____ , _____ , T = _____ , X = _____

EQT 7 = ?
_____ , DV _____ , _____ , _____ , T = _____ , X = _____

EQT 8 = ?
_____ , DV _____ , _____ , _____ , T = _____ , X = _____

EQT 9 = ?
_____ , DV _____ , _____ , _____ , T = _____ , X = _____

EQT 10 = ?
_____ , DV _____ , _____ , _____ , T = _____ , X = _____

EQT n = ?
END (required-terminates EQT Table entries where desired)

DRT TABLE BUILDING WORKSHEET

BOLDFACE terms indicate RTMGN prompts output to user. All other data (except comments) is user input.

DRT TBL (Device Reference Table)
LU #: (Logical Unit Nos.)

1 = EQT # ?
_____ , _____

2 = EQT # ?
_____ , _____

3 = EQT # ?
_____ , _____

4 = EQT # ?
_____ , _____

5 = EQT # ?
_____ , _____

6 = EQT # ?
_____ , _____

7 = EQT # ?
_____ , _____

8 = EQT # ?
_____ , _____

9 = EQT # ?
_____ , _____

10 = EQT # ?
_____ , _____

11 = EQT # ?
_____ , _____

12 = EQT # ?
_____ , _____

(Enter EQT entry nos. and subchannel nos. if applicable from I/O Worksheet)

13 = EQT # ?
_____ , _____

14 = EQT # ?
_____ , _____

15 = EQT # ?
_____ , _____

16 = EQT # ?
_____ , _____

17 = EQT # ?
_____ , _____

18 = EQT # ?
_____ , _____

19 = EQT # ?
_____ , _____

20 = EQT # ?

END (required to terminate DRT Table entries where desired)

INTERRUPT TABLE WORKSHEET

> **BOLDFACE** terms are output to user by RTMGN
>
> Question marks are RTMGN prompts. All other data (except comments) is entered by user. Fill in blanks with desired parameters.

INT TBL (Interrupt Table)

?

__________ , __________ , __________

?

__________ , __________ , __________

?

__________ , __________ , __________

?

__________ , __________ , __________

?

__________ , __________ , __________

?

__________ , __________ , __________

?

__________ , __________ , __________

?

__________ , __________ , __________

?

__________ , __________ , __________

?

__________ , __________ , __________

?

__________ , __________ , __________

?

__________ , __________ , __________

?

__________ , __________ , __________

?

__________ , __________ , __________

?

__________ , __________ , __________

?

__________ , __________ , __________

?

__________ , __________ , __________

?

__________ , __________ , __________

?

__________ , __________ , __________

?

END (required-terminates Interrupt Table entries where desired)

Get input data from I/O Configuration Worksheet. See "Interrupt Table" in Section III regarding four different forms of parameters.

RTE-M PROGRAM PARTITION/PROGRAM PARAMETER INPUT WORKSHEET (1 of 2 pages)

BOLDFACE terms are output to user by RTMGN.

– Hyphens are RTMGN prompts. All other data (except comments) is entered by user. Fill in blanks with desired parameters.

MAX NUMBER OF PARTITIONS? (MIII only)

_____ (decimal no. - up to 64)

#ID SEG?

_____ (decimal no - at least 1 must be allocated)

START-UP PROG?

_____ (program name - enter 0 if no start-up program.

REL RES LIB

– LINKS IN _____ (optional)

– MAP _____ , _____ , _____ , ON _____ (optional)

– RELOCATE _____ (_____)

– RELOCATE _____ (_____)

– RELOCATE _____ (_____)

– RELOCATE _____ (_____)

There are two format options to load library routines:

RELOCATE { file name / file name (module) }

– SEARCH _____ (optional as needed)

– DISPLAY UNDEFS (optional)

– END (required-terminates relocation of library routines)

STARTING ADDRESS 0002
NO UNDEFS
RELOCATE SSGA

– RELOCATE _____ (optional - these are type 30 modules)

– RELOCATE _____

– RELOCATE _____

– END (required to terminate phase)

#WDS IN COMM?

_____ (decimal no. of words to allocate for common)

LWA OF COMMON = xxxx

ALIGN AT NEXT PAGE?

_____ (enter YES or NO)

LWA OF COMMON = xxxx (only output if above answer is YES)

WORKSHEET IS CONTINUED ON NEXT PAGE

RTE-M PROGRAM PARTITION/PROGRAM PARAMETER INPUT WORKSHEET (2 of 2 pages)

NOTE: Reproduce one copy of this worksheet for each user program to be relocated.

REL USER PROGS (See Relocating Loader in RTE-M Programmer's Reference Manual)

- LINK IN _____ (unnecessary if specified in Initialization Worksheet)
- MAP _____ , _____ , _____ , ON _____ (see comment above)
- RELOCATE _____ (use program name)
- RELOCATE _____
- RELOCATE _____
- RELOCATE _____
- RELOCATE _____
- RELOCATE _____

(any subroutine names)

- SEARCH _____
- SEARCH _____
- SEARCH _____
- SEARCH _____

(optional - library searches as required)

- DISPLAY UNDEFS (optional
- END (required after each user program relocation)

STARTING ADDRESS xxxx
NO UNDEFS
ENTER PRAMS

_____ , _____ , _____ , _____ , _____ , _____ , _____ , _____ , _____ , _____

(See "Program Parameter Input" in Section III for format - enter a 0 if no changes are to be made and defaults are wanted)

REL USER PROGS

- END (enter only if last program has been relocated; otherwise, enter next program name. END terminates all user program relocation.)

NOTE: Any user program that is a Loader or Generator should be relocated last.

ENTER RELOCATION SEQUENCE NO. HERE: ☐

RTE-M MEMORY BOUNDS AND PARTITION WORKSHEET

BOLDFACE terms are output to user by RTMGN.

Question Marks are RTMGN prompts; all other data (except comments) is user input. Fill in blanks with desired parameters.

LWA MEM RES PROG = xxxxx CHANGE?

_____ (enter desired new address in OCTAL; enter 0 for no change)

ALIGN AT NEXT PAGE?

_____ (enter YES or NO; YES causes SAM to start on next page)

SAM = xxxxx WORDS (current size of SAM)

NO. ADD. PAGES FOR SAM? MAX = xxx (MIII only SAM extends to end of memory for MI and MII)

_____ (enter desired no. added pages or 0)

LARGEST ADDRESSABLE PARTITION
W/O COMM xx PAGES
W COMM xx Pages
} (output for MIII only)

PAGES REMAINING = xxxxx
DEFINE PARTITIONS

?

_____ , _____ (enter partition no. and decimal no. pages for partition)

?

_____ , _____

PAGES REMAINING = xxxxx

?

_____ , _____

PAGES REMAINING = xxxxx

?

_____ , _____

PAGES REMAINING = xxxxx

?

_____ , _____

PAGES REMAINING = xxxxx

?

_____ , _____

PAGES REMAINING = xxxxx

?

_____ , _____

PAGES REMAINING = xxxxx

?

_____ , _____

PAGES REMAINING = xxxxx

?

_____ , _____

PAGES REMAINING = xxxxx

?

_____ , _____

PAGES REMAINING = 0

END (required)

NOTE: Partitions do not have to be defined in sequence.

RTE-M SNAPSHOT WORKSHEET

BOLDFACE terms are output to user by RTMGN.

– Hyphens are RTMGN prompts; all other data (except comments) is user input. Fill in blanks with desired parameters.

SNAPSHOT? (used for on-line loading of user programs)

– SNAP ON _____ (enter file name or LU number, or enter 0 if no snapshot wanted.)

END (required)

SYSTEM STARTS AT TRACK xxxxx SECTOR xxxxx

(Message only appears when generated output was assigned to a flexible disc. Save a copy of the track and sector address!)

RTMGN FINISHED

Generation is completed, and absolute user-defined RTE-M operating system is on file or LU specified on Initialization Worksheet.

RELOCATABLE MODULE FILE NAMES

APPENDIX E

The file names of all HP-supplied required and optional relocatable modules (except drivers) used in generating RTE-M operating systems are given in Table E-1 below. See the RTE-M Software Numbering Catalog for a complete listing of available relocatable, absolute and source modules. The Catalog gives a description, Part Number and distribution media for each module.

Table E-1. RTE-M Relocatable Module File Names

FILE NAME	RELOCATABLE MODULE
%TSKSC	RTE-M Task Scheduler Library
%DRF	RTE-M MII/MIII Flexible Disc Directory Manager
%DRF1	RTE-M1 Flexible Disc Directory Manager
%TBLFP	RTE-M Flexible Disc Tables
%FMGF0	RTE-M Flexible Disc FMGR
%FMPF	RTE-M Flexible Disc FMP Library
%DRC	RTE-M MII/MIII Tape Cartridge Directory Manager
%DRC1	RTE-M MI Tape Cartridge Directory Manager
%TBLCR	RTE-M Tape Cartridge Tables
%DIRD	RTE-M Directory Read Subroutine
%FMGC0	RTE Tape Cartridge FMGR
%FMPC	RTE-M Tape Cartridge FMP Library
%MEDIT	RTE-M Editor (EDITM)
%MSYLB	RTE-M System Library
%MSY1	RTE-M MI Operating system
%MSY2	RTE-M MII Operating System
%MSY3	RTE-M MIII Operating System
%MBU	RTE/M I/O Buffering Option
%MMP	RTE-M MI Scheduling Option
%MRN	RTE-M MII/MIII Resource Management Package
%MTI	RTE-M Timer Option
%MTS	RTE-M Time Scheduling Option
%MOP	RTE-M Operator Command Options
%MCL	RTE-M MII Class I/O Option
%MCL3	RTE-M MIII Class I/O Option
%MAP	RTE-M MI/MII APLDR
%MAP3	RTE-M MIII APLDR
%MDMLB	RTE-M Dummy Library
%MPF	RTE-M MI/MII Power Fail Driver
%MPF3	RTE-M MIII Power Fail Driver
%MAUTO	RTE-M AUTOR Program
%ONMTM	RTE-M Multi-Terminal Monitor Start-Up Program
%MPRMP	RTE-M PRMPT Program
%MRSPN	RTE-M Response Program

Table E-1. RTE-M Relocatable Module File Names (Continued)

FILE NAME	RELOCATABLE MODULE
%RLIB1	RTE-M FORTRAN Relocatable Library #1
%RLIB2	RTE-M FORTRAN Relocatable Library #2
%SGPRP	RTE-M Segment Preparation Program
%RTMGN	RTE-M System Generator
%RTMLD	RTE-M Relocating Loader
%RTMSC	RTE-M Loader Subcontrol
%MBASC	RTE-M BASIC Interpreter Kit
%MESGA	RTE-M BASIC Error Message Module
%MESCD	RTE-M BASIC Alternate Error Module
%DTRAP	RTE-M BASIC Dummy Trap Subroutine
%MBTG	RTE-M BASIC Branch and Mnemonic Table Generator
%BASLB	RTE-M BASIC Subroutine Library
%BAMLB	RTE-M BASIC Memory Library
%MFTN0	RTE-M FORTRAN Main Program
%MFTN1	RTE-M FORTRAN Segment 1
%MFTN2	RTE-M FORTRAN Segment 2
%MASM0	RTE-M Assembler Main Program
%MASM1	RTE-M Assembler Seg 1
%MASM2	RTE-M Assembler Seg 2
%MASM3	RTE-M Assembler Seg 3
%MASM4	RTE-M Assembler Seg 4
%MASM5	RTE-M Assembler Seg D
%MASM6	RTE-M XREF Segment
%MXRF0	RTE-M XREF Main

FLEXIBLE DISC BOOT CONVERSION TABLE

APPENDIX F

An absolute program located on a 9885 Flexible Disc is booted into the system by using the following procedures:

1. Insert the disc platter containing the absolute program into disc drive 0 (if there is more than one disc drive on the system).

2. List the Flexible Disc Directory by using the File Manager DL command. The Master Security Code must be specified on the command so that the track and sector information for the absolute program will be listed.

> NOTE
>
> During system generation, the track and sector address will be specified by RTMGN at generation completion. It is not necessary (and will not be possible during the initial generation) to perform Step 2.

3. Find the track and sector addresses in Table F-1 below, and get the corresponding entries from their B-register Octal Equivalent columns. Add the two octal equivalents together and store the sum in the B register.

4. Enter the lower select code of the flexible disc interface card in bits 8-11 of the S register. Enter the Flexible Disc ROM Loader select code in bits 14 and 15.

5. Press the PRESET, IBL and RUN switches. The computer should halt and display 102070 in the T register.

If the computer halts with 102011 in the T register, or if the computer continues to run after the DRIVE SELECT light on the front of the disc drive goes out, either the data format for the file is incorrect or the disc device has malfunctioned.

Table F-1. Flexible Disc Track and Sector Conversion Table

Track Address	B-Reg. Octal Equiv. B-TR	Track Address	B-Reg. Octal Equiv. B-TR	Logical Sector Address	B-Reg. Octal Equiv. B-TR
00	000000	34	002100	00	000000
01	000040	35	002140	02	000001
02	000100	36	002200	04	000002
03	000140	37	002240	06	000003
04	000200	38	002300	08	000004
05	000240	39	002340	10	000005
06	000300	40	002400	12	000006
07	000340	41	002440	14	000007
08	000400	42	002500	16	000010
09	000440	43	002540	18	000011
10	000500	44	002600	20	000012
11	000540	45	002640	22	000013
12	000600	46	002700	24	000014
13	000640	47	002740	26	000015
14	000700	48	003000	28	000016
15	000740	49	003040	30	000017
16	001000	50	003100	32	000020
17	001040	51	003140	34	000021
18	001100	52	003200	36	000022
19	001140	53	003240	38	000023
20	001200	54	003300	40	000024
21	001240	55	003340	42	000025
22	001300	56	003400	44	000026
23	001340	57	003440	46	000027
24	001400	58	003500	48	000030
25	001440	59	003540	50	000031
26	001500	60	003600	52	000032
27	001540	61	003640	54	000033
28	001600	62	003700	56	000034
29	001640	63	003740	58	000035
30	001700	64	004000		
31	001740	65	004040	Note: 00036 and 00037 are illegal addresses.	
32	002000	66	004100		
33	002040				

INSTALLING ADDITIONAL HP 2644/45A DISPLAY STATIONS

APPENDIX G

If your RTE-M operating system is to have more than two HP 2644/5 Display Station terminals, the %TBLCR library routine must be modified through reassembly. The source code for %TBLCR routine is supplied on the Program Preparation flexible disc platter or tape cartridge, as appropriate for the installation (see the RTE-M Software Numbering Catalog for the correct Part Number).

The relevant instructions must be modified either on an RTE-M system that has the HP Assembly program installed, or on a compatible RTE system.

The procedure is as follows:

1. Change the instruction

   ```
   #ENT EQU 0
       to
   #ENT EQU x
   ```

   where y = the number of additional cartridge tape units to be mounted.

2. Add a four-word entry (as defined below) to the area indicated in the source for each additional cartridge tape unit to be mounted. Each unit of directory space has the following format:

   ```
   NOP
   NOP
   DEC *+41
   BSS 40
   ```

RTE-M GENERATION EXAMPLES

APPENDIX H

This appendix provides generation examples for the following system types:

H-1 — shows the echo file output for an MI system.
H-2 — shows the echo file output for an MII system
H-3 — shows a mixed echo file and map output for an MIII system.

```
* &MCGNT  761025
OUTPUT ON !MCGEN
END
*
* TYPE OF SYSTEM?
1
*
* TBG CHNL?
0
*
* PRIV. INT?
0
*
* LWAM?
57776
*
* FWA BP?
100
*
*
* CHANGE ENTS?
.MPY,RP,100200
* CHANGE ENTS?
.DIV,RP,100400
* CHANGE ENTS?
END
*
* REL SYS MODS
MAP MODULES ON &MCGNL
REL %MSY1
REL %MOP
SEARCH %MDMLB
REL %4DV05
SEARCH %MSYLB
END
*
* BUFFER LIMITS (LOW,HIGH)?
0,0
*
*
* EQT TBL
*
*
* EQT  1 =?
10,4DV05,X=13
*
* EQT  2 =?
END
*
* DRT TBL
*
* LU#:
*
*  1 = EQT #?
```

Figure H-1. RTE-M MI Generation Example

```
1,Ø
*
*  2 = EQT #?
Ø
*
*  3 = EQT #?
Ø
*
*  4 = EQT #?
1,1
*
*  5 = EQT #?
1,2
*
*  6 = EQT #?
1,Ø
*
*  7 = EQT #?
END
*
* INT TBL
*
* ?
1Ø,EQT,1
* ?
77,EQT,1
* ?
END
*
*
* #ID SEG?
2
*
* START-UP PROG?
Ø
*
* REL RES LIB
REL %TBLCR
REL %DIRD
REL %DRC1
SEARCH %FMPC
SEARCH %RLIB1
SEARCH %RLIB2
END
*
* REL SSGA
END
*
* # WDS IN COMM?
Ø
* LWA OF COMMON =  15730
*
* ALIGN AT NEXT PAGE?
NO
```

```
*
* REL USER PROGS
REL %RTMGN
SEARCH %RTMSC
SEARCH %FMPC
SEARCH %MSYLB
SEARCH %RL1B1
SEARCH %RL1B2
END
*
* ENTER PRAMS
0
*
* REL USER PROGS
BOUNDS FWAM=42000
END
*
* ENTER PRAMS
0
*
* NO ID SEGMENTS LEFT
*
*
* LWA MEM RES PROG = 42335  CHANGE?
57776
*
* ALIGN AT NEXT PAGE?
NO
* SAM =  00000  WORDS
*
* SNAPSHOT?
SNAP ON &MCGNS
END
*
* RTMGN FINISHED
```

```
-OUTPUT ON OUTY
END
*
* TYPE OF SYSTEM?
2
*
* TBG CHNL?
13
*
* PRIV. INT?
Ø
*
* LWAM?
67777
*
* FWA BP?
3Ø
*
*
* CHANGE ENTS?
.MVW,RP,1Ø5777
* CHANGE ENTS?
END
*
* REL SYS MODS
-MAP MODULES,GLOBALS ON LSTF
-RELOCATE %MSY2
REL %MTI
REL %4DVØ5
REL %DVR33
SEARCH %MDMLB
-END
*
* # OF I/O CLASSES?
Ø
*
* # OF RESOURCE NUMBERS?
Ø
*
* BUFFER LIMITS (LOW,HIGH)?
1ØØ,4ØØ
*
*
* EQT TBL
*
*
* EQT  1 =?
12,4DVØ5,B,X=13
*
* EQT  2 =?
21,DVR33,D
*
* EQT  3 =?
END
```


Figure H-2. RTE-MII Generation Example

```
*
* DRT TBL
*
* LU#:
*
*  1 = EQT #?
1,Ø
*
*  2 = EQT #?
2
*
*  3 = EQT #?
Ø
*
*  4 = EQT #?
1,1
*
*  5 = EQT #?
1,2
*
*  6 = EQT #?
1,Ø
*
*  7 = EQT #?
END
*
* INT TBL
*
* ?
12,EQT,1
* ?
21,EQT,2
* ?
END
*
*
* #ID SEG?
7
*
* START-UP PROG?
Ø
*
* REL RES LIB
END
*
* REL SSGA
END
*
* # WDS IN COMM?
2Ø
* LWA OF COMMON =  16375
*
* ALIGN AT NEXT PAGE?
NO
```

```
*
* REL USER PROGS
END
*
*
* LWA MEM RES PROG = 16375  CHANGE?
57777
*
* ALIGN AT NEXT PAGE?
YES
* SAM =  Ø4Ø96  WORDS
*
* SNAPSHOT?
-SNAP ON SNPF
END
*
* RTMGN FINISHED
```

```
*
* !MYSYS - RTE-M SYSTEM GENERATION (28 DEC 1976)
*
ECHO ON 6
OUTPUT ON !MYSYS
END
*
* TYPE OF SYSTEM?
3
*
* TBG CHNL?
10
*
* PRIV. INT?
0
*
* PRIV. DRIVERS ACCESS COMMON?
NO
*
* MEM SIZE?
48
*
* FWA BP?
20
*
*
* CHANGE ENTS?
.MPY,RP,100200
* CHANGE ENTS?
.DIV,RP,100400
* CHANGE ENTS?
.DLD,RP,104200
* CHANGE ENTS?
.DST,RP,104400
* CHANGE ENTS?
.FAD,RP,105000
* CHANGE ENTS?
.FSB,RP,105020
* CHANGE ENTS?
.FMP,RP,105040
* CHANGE ENTS?
.FDV,RP,105060
* CHANGE ENTS?
IFIX,RP,105100
* CHANGE ENTS?
FLOAT,RP,105120
* CHANGE ENTS?
.MBT,RP,105765
* CHANGE ENTS?
.MVW,RP,105777
* CHANGE ENTS?
END
*
* REL SYS MODS
LINKS IN CURRENT
MAP MODULES ON 6
 PROGRAM MODULE
          ENTRY     LOW      HIGH     LOW      HIGH       CP LINKS
          POINT     MAIN     MAIN     BASE     BASE      LOW     HIGH
  ----------------------------------------------------------------
```

Figure H-3. RTE-MIII Generation Example

```
-
REL %MSY3
  $MDI3       92064-16003  REV.1650  761020
  $MDI3            02000   02554    01637   01646   00000  177777

  $MIO3       92064-16003  REV.1650  761210
  $MIO3            02700   06426    01516   01636   02555   02613

  $MEX3       92064-16003  REV.1650  761020
  $MEX3            06441   10056    01477   01515   06427   06430

  $MSC3       92064-16003  REV.1650  761020
  $MSC3            10215   12724    01370   01476   10057   10074

  $MBU3       92064-16003  REV.1650  761020
  $MBU3            12725   13221    01370   01367   00000  177777

-
REL %MTI
  $MTI        92064-16008  REV.1650  761020
  $MTI             13222   13376    01366   01367   00000  177777

-
REL %MTS
  $MTS        92064-16009  REV.1650  761020
  $MTS             13440   14410    01325   01365   13377   13400

-
REL %MOP
  $MOP         92064-16010  REV.1650  761118
  $MOP             14411   15555    01313   01324   00000  177777

-
REL %MCL3
  $MCL3       92064-16015  REV.1650  761024
  $MCL3            15601   16447    01277   01312   15556   15600

-
REL %MRN
  $MRN        92064-16031  REV.1650  761020
  $MRN             16450   16613    01277   01276   00000  177777

-
SEARCH %MDMLB
  $MDMI       92064-16032  REV.1650  761020
  $MDMI            16614   16614    01277   01276   00000  177777

  $MER        92064-16013  REV.1650  761020
  $MER             16615   16664    01277   01276   00000  177777

-
REL %DVR33
  DVR33         12732-16001  REV 1650 11 16 76
  DVR33            16665   17766    01277   01276   00000  177777

-
REL %4DV05
  DVR05          92001-16027  1636 760820
  DVR05            17771   22545    01123   01276   17767   17770

-
REL %DV37
  DVR37       59310-16003 REV. 1614, 760329
  DVR37            22672   25056    01117   01122   22546   22663

-
```

```
REL %MPF3
  DVP43      92064-16029  REV.1650  761020
  DVP43            25057   25673   01113   01116   00000  177777

-
END
STARTING ADDRESS 00002
NO UNDEFS
*
* # OF I/O CLASSES?
10
*
* # OF RESOURCE NUMBERS?
10
*
* BUFFER LIMITS (LOW,HIGH)?
100,400
*
*
* EQT TBL
*
*
* EQT  1 =?
11,DVR33,D,T=50
*
* EQT  2 =?
13,DVR05,B,X=13
*
* EQT  3 =?
14,DVR37,D,X=50
*
* EQT  4 =?
4,DVP43
*
* EQT  5 =?
END
*
* DRT TBL
*
* LU#:
*
*  1 = EQT #?
2,0
*
*  2 = EQT #?
1,0
*
*  3 = EQT #?
0
*
*  4 = EQT #?
2,1
*
*  5 = EQT #?
2,2
*
*  6 = EQT #?
3,1
*
*  7 = EQT #?
3,0
*
*  8 = EQT #?
0
*
*  9 = EQT #?
```

```
0
*
* 10 = EQT #?
0
*
* 11 = EQT #?
0
*
* 12 = EQT #?
4
*
* 13 = EQT #?
END
*
* INT TBL
*
* ?
4,ENT,$POWR
* ?
11,EQT,1
  ?
 2,EQT,1
  ?
13,EQT,2
* ?
14,EQT,3
* ?
END
*
*
*
* MAX NUMBER OF PARTITIONS?
5
*
* #ID SEG?
10
*
* START-UP PROG?
0
*
* REL RES LIB
-
REL %TBLFP
  $TBLFP     92064-16057  REV.1650  761026
  $TBLFP           26763   27004   01110   01110   00000  177777

-
SEARCH %FMPF
  SECM       92064-16058  REV.1650  761005
  SECM             27005   27004   01110   01107   00000  177777

-
END
STARTING ADDRESS 00002
NO UNDEFS
*
* REL SSGA
END
*
* # WDS IN COMM?
50
* LWA OF COMMON =  27066
*
* ALIGN AT NEXT PAGE?
YES
* LWA OF COMMON =  27777
*
```

```
* REL USER PROGS
-
REL %DRF
  D.RFP       92064-16056  REV.1650  761020
  D.RFP            30000   31754   00020   00027   00000  177777

-
SEARCH %FMPF
  PMOVE       92064-16069  REV.1650  760512
  PMOVE            31755   31765   00030   00027   00000  177777

  P.PAS       92064-16059  REV.1650  740801
  P.PAS            31770   32016   00030   00030   31766   31767

-
SEARCH %MSYLB
  PRTN       92001-16005  741120
  PRTN             32017   32121   00031   00030   00000  177777

-
SEARCH %RLIB1
  RMPAR      750701  24998-16001
  RMPAR            32122   32145   00031   00030   00000  177777

-
SEARCH %RLIB2
-
DISPLAY UNDEFS
-
END
STARTING ADDRESS 30221
NO UNDEFS
*
* ENTER PRAMS
0
*
* REL USER PROGS
-
REL %MAUTO
  AUTOR       92064-16030  REV.1650  761027
  AUTOR            32146   32462   00031   00030   00000  177777

-
DISPLAY UNDEFS
-
END
STARTING ADDRESS 32146
NO UNDEFS
*
* ENTER PRAMS
0
*
* REL USER PROGS
-
REL %MAP3
  APLDR       92064-16016  REV.1650  761111
  APLDR            32615   35205   00031   00056   32463   32610

-
SEARCH %FMPF
  IMESS       92064-16064  REV.1650  760628
  IMESS            35206   35231   00057   00057   00000  177777

  OPEN        92064-16058  REV.1650  761116
```

```
  OPEN              35232   35705   00060   00070   00000  177777

  CLOSE       92064-16058  REV.1650  761019
  CLOSE             35716   36055   00071   00105   35706   35715

  CLD.R       92064-16058  REV.1650  761013
  CLD.R             36056   36215   00106   00107   00000  177777

  DD.RF       92064-16058  REV.1650  761010
  DD.RF             36216   36216   00110   00107   00000  177777

  READF       92064-16058  REV.1650  761115
  READF             36217   36761   00110   00114   00000  177777

  LOCF        92064-16059  REV.1650  760819
  LOCF              36762   37154   00115   00114   00000  177777

  $OPEN       92064-16059  REV.1650  740801
  $OPEN             37155   37363   00115   00115   00000  177777

  RW$UB       92064-16059  REV.1650  750422
  RW$UB             37364   37635   00116   00115   00000  177777

  R/W$        92064-16059  REV.1650  760801
  R/W$              37636   37767   00116   00115   00000  177777

  RWND$       92064-16059  REV.1650  760629
  RWND$             37771   40067   00116   00121   37770   37767

  P.PAS       92064-16059  REV.1650  740801
  P.PAS             40070   40116   00122   00121   00000  177777

  IDCB1       92064-16059  REV.1650  761214
  IDCB1             40117   40336   00122   00121   00000  177777

-
SEARCH %MSYLB
  .DRCT     92001-16005 741120
  .DRCT             40337   40345   00122   00121   00000  177777

-
SEARCH %RLIB1
  RMPAR     750701  24998-16001
  RMPAR             40346   40371   00122   00121   00000  177777

-
SEARCH %RLIB2
  .ENTR     750701  24998-16001
  .ENTR             40372   40461   00122   00121   00000  177777

-
DISPLAY UNDEFS
-
END
STARTING ADDRESS 32615
NO UNDEFS
*
* ENTER PRAMS
0
*
* REL USER PROGS
-
END
*
*
* LWA MEM RES PROG = 40461  CHANGE?
```

```
0
*
* ALIGN AT NEXT PAGE?
NO
* SAM =  00718  WORDS
*
* NO. ADD. PAGES FOR SAM? MAX =  00019
1
*
* LARGEST ADDRESSABLE PARTITION
*
* W/O COM  00021 PAGES
* W/  COM  00021 PAGES
*
* PAGES REMAINING =  00030
* DEFINE PARTITIONS
*
* ?
1,4
* PAGES REMAINING =  00026
* ?
2,4
* PAGES REMAINING =  00022
* ?
3,5
* PAGES REMAINING =  00017
* ?
4,5
* PAGES REMAINING =  00012
* ?
5,12
* PAGES REMAINING =  00000
* ?
END
*
* SNAPSHOT?
-
SNAP ON &SNAP1
-
END
*
* SYSTEM STARTS AT TRACK   00051 SECTOR   00010
*
* RTMGN FINISHED
```

INDEX

In this index, topics with multiple page references have the principle reference printed in **BOLDFACE** type.

In this index, topics with multiple page references have the principle reference printed in **BOLDFACE** type.

In this index, topics with multiple page references have the principle reference printed in **BOLDFACE** type.

S

In this index, topics with multiple page references have the principle reference printed in **BOLDFACE** type.

PART NO. 92064-90003
Printed in U.S.A. 3/77

HEWLETT hp PACKARD

Sales and service from 172 offices in 65 countries.
11000 Wolfe Road, Cupertino, California 95014